Evaluating the Performance of Various Convection Schemes on Free Surface Flows by Using Interfoam Solver

Vahid Emad

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the Degree of

Master of Science
in
Mechanical Engineering

Eastern Mediterranean University
September 2014
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

_____
Prof. Dr. Elvan Yılmaz
Director

I certify that this thesis satisfies the requirements as a thesis for the degree of Master of Science in Mechanical Engineering.

_____
Prof. Dr. Uğur Atikol
Chair, Department of Mechanical Engineering

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Master of Science in Mechanical Engineering.

_____
Prof. Dr. Ibrahim Sezai
Supervisor

Examining Committee
_____

1. Prof. Dr. Fuat Egelioğlu   _____

2. Prof. Dr. Ibrahim Sezai   _____

3. Assoc. Prof. Dr. Hasan Hacışevki _____

ABSTRACT

The main propose of this study is to analyze and compare the effects of different limiters on interface tracking within two phase flows. OpenFOAM source codes were employed as the base of analysis. In order to simulate the two phase incompressible flows in the OpenFOAM, the embedded InterFOAM solver has been used in this study. The solver is based on a modified volume of fluid (VOF) method. The VOF approach relies on a scalar indicator function between zero and unity to differentiate two different fluids. Since simulation results are affected by the applied convection schemes, selection of an appropriate scheme is significant. Discretization of the convection term is a controversial issue in the two phase flow simulations. In this thesis, investigation of the performance of the various limiters has been done on a variety of well-known validation test cases that include Dam-break problem, free bubble rise problem and advection of hollow shapes in an oblique velocity field problem. Results were obtained using the Upwind, Van-Leer, UMIST, QUICK and MULES convection schemes. With the comparison of these schemes, it has been observed that MULES is the most accurate scheme in tracking the interface.

Keywords: OpenFOAM, InterFOAM, VOF, limiters, convection schemes, free-surface flow

ÖZ

Bu tezin asil amacı iki fazlı akımlarda bulunan ara yüzeylerin takibi için farklı sınırlandırıcının incelenmesi ve bunların karşılaştırılmasıdır. Analiz işlemleri için OpenFOAM ana kodu kullanılmıştır. İki fazlı sıkışmayan akımların simülasyonu için, OpenFOAM içinde bulunan InterFOAM çözücüsü kullanılmıştır. Bu çözücünün temeli değiştirilmiş Volume of Fluid (VOF) yöntemidir. Sıfır ve bir arasında olan indikator fonksiyonuna dayanarak, VOF yöntemi iki akışkanın ayırdedilmesinde kullanılmaktadır. Simülasyon sonuçlarının seçilen konveksiyon yöntemine bağımlı olması nedeni ile uygun bir konveksiyon yöntemi seçilmesi çok önemlidir. İki fazlı akımların simülasyonu ile ilgili çalışmalarda konveksiyon teriminin ayrıştırılması en tartışılan konulardan biridir. Bu tezde çeşitli sınırlandırıcıların performansı bir kaç meşhur doğrulama testi ile incelenmiştir. Bu testler baraj yıkılma problemi, serbest baloncuk tırmanışı problemi ve içi boş şekillerin eğri hız alanı içinde adveksiyon problemidir. Tez Sonuçları Upwind, Van-Lear, UMIST, QUICK ve MULES konveksiyon yöntemleri kullanarak elde edilmiştir. Bu yöntemlerin karşılaştırılmasına göre MULES yöntemi arayüz takibi için en doğru yöntem olarak belirlenmiştir.

Anahtar kelimeler: OpenFOAM, InterFOAM, VOF, sınırlandırıcılar, konveksiyon yöntemleri, açik yüzeyli akişlar.

To my lovely family

ACKNOWLEDGMENT

First and above all, I praise God, the almighty for providing me this opportunity and granting me the capability to proceed successfully. This thesis appears in its current form due to the assistance and guidance of several people. I would therefore like to offer my sincere thanks to all of them.

I would like to express my sincere gratitude to my supervisor Prof. Dr. Ibrahim Sezai for his support, guidance and advice during this thesis, without his guidance, I would not be able to do the project.

I would also like to sincerely thank my family who encouraged me and prayed for me throughout the time of my research. Without their love, encouragement and support I would not have finished this project.

Last but not the least, I would like to remember and thank my friends Nima Agh, Navid Saffarizadeh, Alireza Kazemeini and Armin Javadi for providing support and friendship that I needed and Dr. Setare Satari for being supportive throughout my time.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

Co	Courant number
Eo	Eötvös number
F	Volumetric surface tension force
F	Mass flux
g	Gravity acceleration vector
Mo	Morton number
n	Normal vector of the interface
P	Pressure
p^*	Modified pressure
r	Displacement vector
Re	Reynolds number
S	Surface area vector
S	Source term
t	Time
u	Velocity
V	Volume vector
x	Position vector

Greek symbols:

ρ	density
τ	viscosity stress tensor
σ	surface tension coefficient
μ	dynamic viscosity

α	Indicator function
κ	curvature
Γ	diffusion coefficient
ϕ	A flow property
f_x	interpolation factor
ψ	flux limiter function

Subscripts:

P	Refer to the value at cell centroid
N	Refer to the value at upstream cell centroid
f	Refer to the value at the face
1	Denotes to fluid 1
2	Denotes to fluid 2

Superscripts:

*	Dimensionless value
0	The value at the previous time step
n	The value at the current time step

Chapter 1

INTRODUCTION

1.1 Background

The study of fluid dynamics has applications in various areas of science and engineering, and it helps to understand the environmental, chemical and biological flows. The two main methods for the study of complex flow problems are computer simulations and laboratory experiments. It is possible to analyze the complexity involved in these flows more clearly by computer simulations without time consuming, inaccurate and expensive experiments. This approach in simulation of fluid mechanics is known as Computational Fluid Dynamics (CFD).

 The objective of this thesis is to present an accurate, reliable and comprehensive Computational Fluid Dynamics (CFD) methodology that has the ability to predict the flow behaviour of immiscible fluids. The flow of immiscible fluids is regularly encountered in industrial processes and nature.

Writing one set of governing equations for the whole flow field, also commonly referred to as the single fluid formulation, has been considered as possible since the beginning of multiphase flows large-scale computational studies.

In multiphase flow simulations for immiscible flows, the exact positions and shapes of the interfaces separating the immiscible fluids contribute actively to the physics of

the problem. The fluids will roughly have different densities and viscosities, thus, density and viscosity discontinuities occur across interfaces. For multiphase flows, each interface is a border between two immiscible fluids. Moreover, surface tension forces act at each interface. The intensity depends on the interface shape. In solving free-surface flow problems, it is important to retain the sharpness of the interface in between the two fluids.

In the numerical simulations, it is crucial to have an updated representation of the interfaces. A method for locating and advancing the free surface, as well as, for treating the free surface or multiphase flow boundary conditions for velocity and pressure, must be included in any solution procedure for free surface fluid flow or multiphase flow problems. The behavior of the dispersed phase is expressed only in terms of its phase fraction and velocity in the multi-fluid model. Many two-phase flows occurring in practice are completely or partly separated (e.g. annular or stratified flows) and Interface deformation and structure are crucial factors (Hewitt and Vassilicos (2005)).

1.2 Computation of free surface

To predict the common boundary between multiphase structures and topographical distributions of the different phases appropriately, the location of the interface must be determined properly as part of the solution. Interface tracking methods are developed to represent and record moving and deforming surfaces or curves that include various types of internal boundaries, interfaces and fronts. The interfaces are defined by constantly updated discretizations. These discretizations can be Eulerian or Lagrangian in nature and it depends on the interface tracking method that is used. The interface tracking techniques described in this chapter were developed to beat

the deficiencies of the artificial dissipation method. The tracking methods have small or no artificial dissipation close to the interface since the singularity is directly calculated and dealt with explicitly as a discontinuity. These methods are harder to implement and perform precisely for a large class of problems (Hyman (1984)).

Existing approaches for the analysis of free surfaces and fluid interfaces between two immiscible fluids on a random Eulerian mesh can in general be categorized in two groups:

- Surface methods (interface tracking methods)
- Volume methods (interface capturing methods)

Figure 1.1 shows a schematic definition of the above methods. The interface is tracked explicitly by marking it with marker points or by attaching it to a mesh that follows the movement of the interface in the surface method. In the volume method, not only the interface, but the fluid in the entire computational domain is marked. The applicability of interface tracking methods is usually limited to simple flow configurations. Both methods are based on stabilized formulations. Stabilized formulations like Streamline-Upwind/Petrov-Galerkin (SUPG) and pressure-stabilizing/Petrov-Galerkin (PSPG) prevent numerical fluctuations and other instabilities in solving problems with high Mach and/or Reynolds numbers, shocks and strong boundary layers, as well as, by using equal-order interpolation functions for pressure and velocity, and other unknowns (Tezduyar (2006)).

Surface methods act with the free surface as a sharp interface of which the motion is followed. The interface can either be tracked explicitly by marking it with special

marker points (particles) or by hooking up the points to a mesh surface which is then forced to move with the interface. In this type of solution, boundary-fitted grids are used and advanced each time the free surface is displaced (Ferziger and Perić (2002)). The surface tracking approaches are the easiest to implement, till interactions happen that change the topology of the interface during the computation. By dividing the domain into a union of disjoint solution regions, volume methods overcome the variable topology problems. The interface is the boundary between these areas (Hyman (1984)). The regions (fluids) on either side of the interface are marked by either massless particles, or alternatively, an indicator function. Indicator function can be used to reconstruct an approximate interface location at any time; it means the interface is algebraically set without reconstruction.

The analysis by volume methods are performed on a fixed grid, which extends beyond the free surface. Unlike the surface tracking methods, very little sub-grid scale structure is retained during the calculation. In this type of method, interface was not defined as a sharp boundary (Ferziger and Perić (2002)). Aside from which method is used, the essential features to accurately model free surface and fluid interface have to include a scheme to describe the location and geometry of the surface, a method to evolve the shape and location with time, and application of free surface boundary conditions at the surface. Sometimes, the shape of the interface affects the vector field of advection. A more detailed description of the different methods is given in the subsequent part.
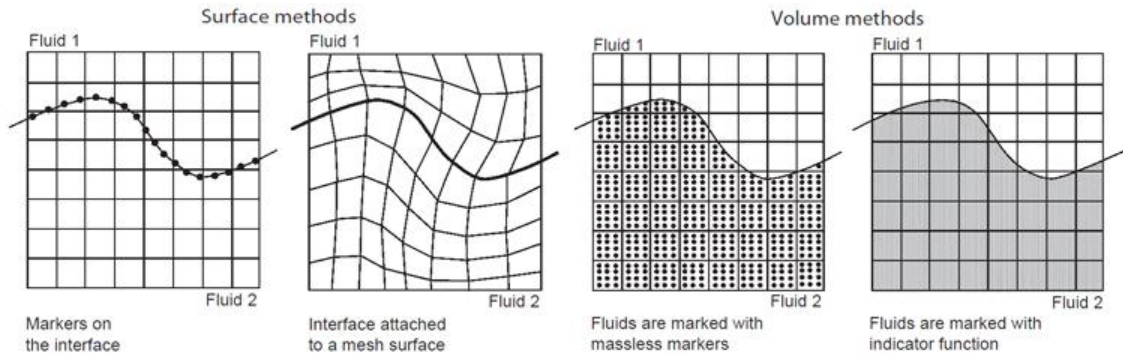
Figure 1.1. Different methods of representing the interface (Ubbink (1997))

1.2.1 Surface methods

As mentioned above in this class of methods, the interface is marked with special marker points located on the interface. Between the points, the position of interface is approximated by interpolation, usually piecewise polynomial. These time-dependent interfaces divide the problem domain into connected regions. The marker points might be represented either the distance from some reference surface or a parametric interpolation. It can be noted that, the distance function is easier to resolve, however the interface deformation is severely limited.

As the spatial domain occupied by the fluid changes in time, the mesh is updated. The accuracy of the surface tracking methods depends strongly on stability and accuracy of the interpolation method approximating the interface location between the marker points. The data structure and algorithms needed by surface tracking methods to account for interactions greatly increase the program complexity (Hyman (1984)).

Interface-tracking methods can follow the evolution of a simple interface very accurately. However, they encounter difficulties in dealing with changes in interface topology. When the flow conditions are such that there are no breakings or

overturning waves, the interface-tracking method is a good approach (Muzaferija and Peri´c (1997)).

For simulating two-fluid flows, in some cases the interface might be too complex to track while keeping the frequency of remeshing at an acceptable level. Due to the complexity of handling interactions and adaptively refining the interface in 3-D the only surface tracking codes that resolve multiple interactions are in 1- and 2-D. Because volume methods do not require mesh update, these methods are more flexible than the surface methods. However, for comparable levels of spatial discretization, volume methods yield less accurate representation of the interface.

The accuracy of the reconstructed interface plays a critical role in the performance of the advection scheme. The main drawback of these methods is the algorithmic complexity involved in reconstructing the interface in a continuous manner across the computational domain, with this difficulty being compounded in three-dimensional problems and these methods require too large computer resources on 3D geometries, so surface methods based on dynamic mesh do not appear of interest for industrial purpose (Mammoli and Brebbia (2005)). The biggest advantage of this approach is that the interface location is known. Three popular surface methods are explained in following sections.

1.2.1.1 Markers on interface

The original idea behind this approach is to precisely track an interface on a fixed mesh by using a set of connected marker particles with negligible mass to mark the interface. The local velocities are used to advect these massless particles in a Lagrangian manner (Ubbink (1997)). Instead of advecting the marker function directly, the boundary between the different fluids can be tracked using marker

points, and the marker function then reconstructed from the location of the interface. Methods using marker points are mostly referred to as front-tracking methods. The inclusion of surface tension is fundamentally different in front-tracking and marker-function methods (Prosperetti and Tryggvason (2009)).

One of the front-tracking methods worth mentioning is that of Unverdi and Tryggvason (1992) in which the Lagrangian interface, represented by a set of connected line segments. This methodology avoids the interfacial numerical diffusion problems associated with VOF. Figure 1.2 presented the interface as an unstructured grid inside a fixed Cartesian grid. Grid points are added to or subtracted from the interface grid as the interface changes shape. The density and viscosity change in a small thickness zone from the values for the gas phase to those for the liquid phase. In this case, the method is alike to the VOF approach, though numerical diffusion is avoided.
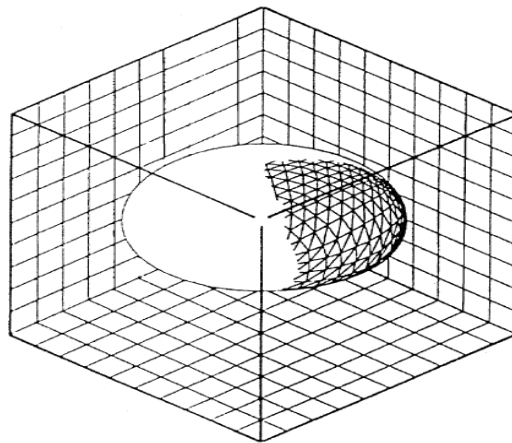

Figure 1.2. Grid system used for the embedded interface method
(Unverdi and Tryggvason (1992))

One of biggest disadvantages of markers on interface method is that the method is sensitive to the spacing between the markers particles (Yeoh and Tu(2009)).

Furthermore, in three dimensions, bookkeeping of the particle connectivity becomes a nearly impossible task (Ubbink (1997)). It is quite hard to implement a merging algorithm in the front-tracking method due to the unstructured positioning of Lagrangian markers.

1.2.1.2 Level set method

Level Set approaches are based on the application of a continuous function to represent the interface within two mediums. The value of the level set function at each point is defined as the shortest distance between that point and the interface (Sussman et al. (1994)), and therefore, the interface is described by the 0 value of level set function. A negative sign is assigned to the distance function for one of the fluids to recognize the two fluids on sides of the interface.

Level set function is updated in order to follow the position of the interfaces and an advection equation is written for this function that allows the position of the interface to be followed. Both for the front-tracking method and the level-set method, the Navier-Stokes equations were discretized using finite difference techniques (Tornberg (2000)). Level set approaches have become widely used for capturing interface evolution particularly when the interface bears extreme topological variations, e.g., merging or pinching off. In the level-set method, topological merging will always occur when interfaces defined by the same level-set function get close relative to the resolution of the grid. Therefore, explicit action must be taken in order to prevent merging (Tornberg (2000)). On the left of Figure 1.3 the two contours represent two distinct circular fluid blobs, but on the right the fluid blobs have merged into one.
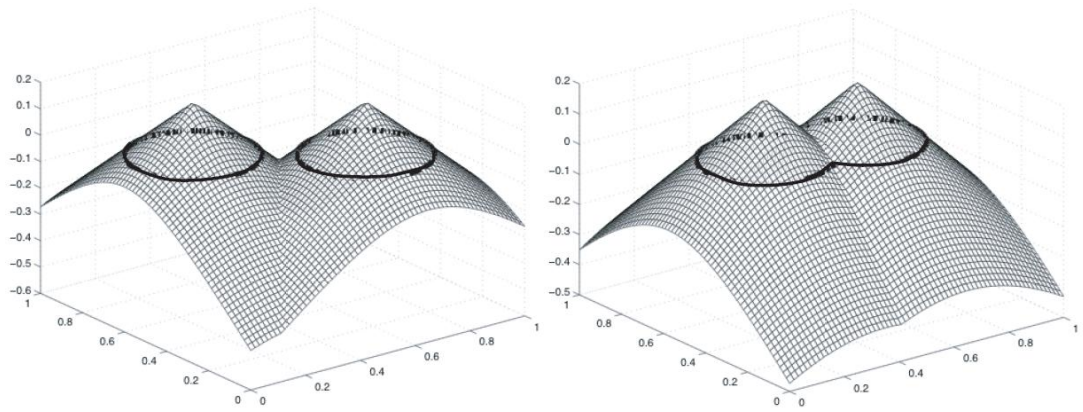
Figure 1.3. Level-set representation of an interface. The interface consists of two distinct circles on the left, but on the right the circles are closer and form one interface. (Prosperetti and Tryggvason (2009))

Level set approaches have been used to simulate multiphase immiscible incompressible flows. These methods are attractive because they admit a convenient description of topologically complex interfaces and are quite simple to implement. One advantage of the level set approach is it being capable of signify topological changes both in 2D or 3D geometry pretty naturally (Mammoli and Brebbia (2005)). The natural ability of the level-set method to handle topological changes in the interfaces is for many applications an important advantage of the method.

Some problems arise when the level set method is developed; namely high-velocity gradient can produce wide spreading and stretching level set, and continuous function will no longer remain a distance function (Mammoli and Brebbia (2005)). There are two reasons for this phenomenon. Firstly, the use of standard differencing schemes for solving the convection equation introduces numerical diffusion to the initial distance function. Secondly, the level set function keeps its initial minimum and maximum values; thus, the maximum value between two merging interfaces remains the same during the calculation, causing a steep gradient in the level set function. The latter is a significant drawback of the level set method (see Figure 1.4).

Since preserving the level-set function as a distance function (at least in the proximity of the interface) is essential for the coupling with the fluid equations in modelling of multiphase flows, it is necessary to reinitialize the level-set function. Without reinitialization, the magnitude of the gradient of the level-set function can become enormous or very small near the zero level-set of the continuous function.

The computational time is a drawback of the level-set technique (Tornberg (2000)).
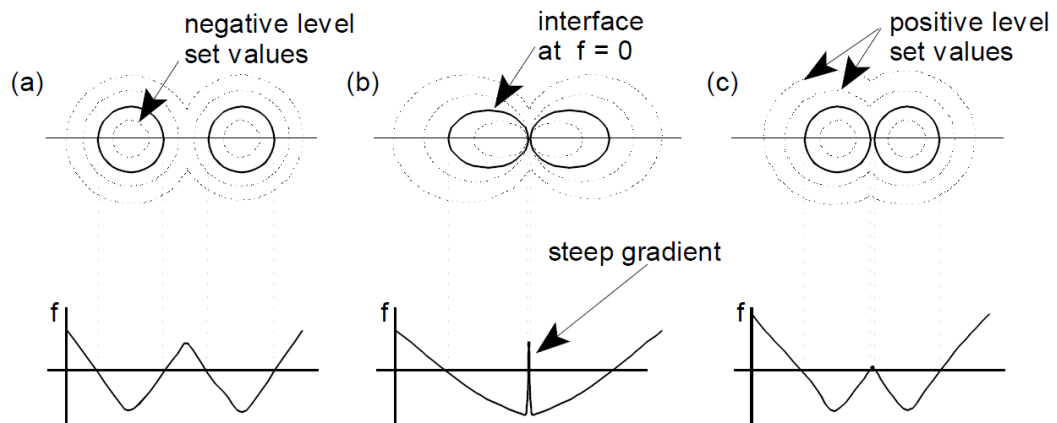

Figure 1.4. Contours of the level set function: (a) initial configuration, (b) just before merging with no corrections (c) just before merging with corrections (Ubbink (1997)

In this method mass loss in under-resolved regions can be generated. This is the main disadvantage of level set methods, but to develop mass conservation, two main extensions of the method can be enhanced, specifically, the particle level set (Enright et al. (2002)) and a coupling between Level Set and VOF (Sussman and Puckett (2000)).

The local density and viscosity are defined as a function of the level of set function. In order to predict the density of a partially filled cell, it is necessary to reconstruct the interface during each time step. This enables the calculation of the fluid proportions filling the cell. These proportions, also known as volume fractions, are

then used to predict the mean density of the fluids occupying a particular cell. For this reason, the level set method can also be categorized as an interface capturing technique, rather than a surface tracking method. (Noh & Woodward (1976))

1.2.1.3 Surface fitted method

The mesh on which the fluid equations are discretized and solved is updated continuously to fit the deforming fluid interfaces. The interfaces are treated as internal boundaries on which boundary conditions are prescribed. Disadvantages with this method include the cost of the computation and the interpolation errors introduced when remeshing. One example of an algorithm based on interface fitted meshes is the work of Hu and Joseph (1994).

Application of surface-fitted methods is mostly motivated by (i) a reduction in the computer storage needed for the interface markers, (ii) always ensuring a sharp interface and (iii) avoidance of partially filled cells (or empty cells in the case where free surface flow between a liquid and void is simulated). As the mesh and fluid are allowed to move together, the mesh automatically tracks the free surface. As the interface moves incrementally in time, the critical factors of this approach are the efficiency and stability of the numerical algorithm. Maintaining a well-defined mesh is a principal feature, an improper surface mesh usually degrades the numerical computation of the fluid flow. With large amplitude changing, it may also be imperative to rapidly regenerate the internal volume mesh encapsulating the fluid domain, which implement of this approach to simulate a multi-phase problem becomes more twisted. Numerical results on the collapse of a liquid column presented by Ramaswamy & Kawahara (1987) reveal that such large deformations may happen even in an early stage of the computation (see Figure 1.7). Another disadvantage of the mentioned approach is that it can only be applied if the interface

is not subjected to large deformation since it can head to a serious distortion of the internal volume mesh. Nevertheless, the greatest limitation is that this method cannot accommodate interface that breaks apart or intersects.
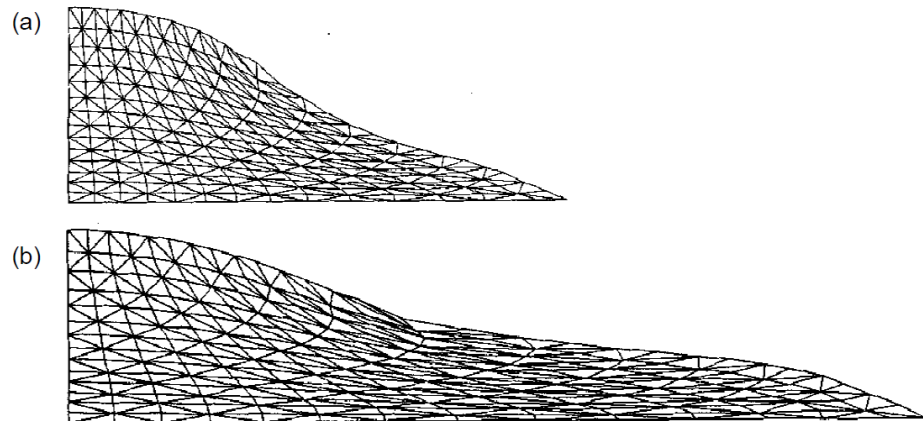

Figure 1.5. Collapse of a liquid column with an interface fitted method (a) at time t=3.0 and (b) at time t=4.0 (Ramaswamy & Kawahara (1987))

1.2.2 Volume methods

The Volume methods (interface-capturing methods) advanced primarily for free-surface, and two-fluid interface flows are expressed consistently over non-moving meshes, using an advection formula in addition to the flow formulations. The advection equation oversees the evolution of an interface function that marks the position of the interface.

In interface-capturing methods, a compressive scheme is used to avoid smearing of the interface. However, this has been understood to lead to stepping off the interface (i.e., loss of curvature) whenever the flow is not aligned with the computational grid.

The equations representing conservation of momentum, mass and energy, obviously hold for any fluid, even when density and viscosity change abruptly and the primary challenge in this method is to advect the phase boundary and to calculate terms

concentrated at the interface precisely, like surface tension (Prosperetti and Tryggvason (2009)).

1.2.2.1 Marker and cell

This approach was based on using marker particles distributed uniformly in each fluid to identify the various fluids. Massless marker particles are spread over the volume occupied by a fluid with a free surface in the marker and cell (MAC) method of Harlow &Welch (1965) (see Figure 1.6). A cell with no marker particles is considered to be empty. A cell with marker particles, lying adjacent to an empty cell, contains a segment of the interface. All other cells with marker particles are considered to be filled with fluid.
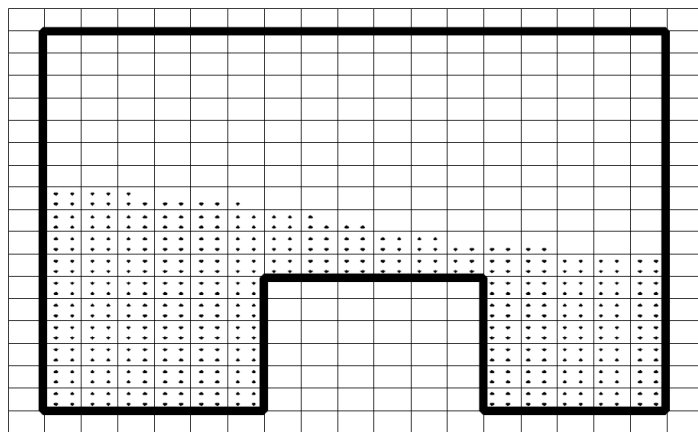

Figure 1.6. Schematic representation of a typical marker and cell mesh layout (Ubbink (1997)

The material properties were reconstructed from the marker particles, and sometimes separate surface markers were also introduced to facilitate the computation of the surface tension. While the historical importance of the marker and cell (MAC) method for multiphase flow simulations cannot be overstated, it is now obsolete. The fluids must be identified through a marker function that is advected by the current when the governing equations are figured out on a fixed grid. Several algorithms

have been advanced for that reason. The volume-of-fluid (VOF) approach is the oldest and, after many enhancements and innovations, keeps being widely used.

The MAC scheme is attractive because it can treat complex phenomena like wave breaking. However, the computing effort is large, especially in three dimensions because, in addition to solving the equations governing fluid flow, one has to follow the motion of a large number of marker particles (Ferziger and Perić (2002)).

1.2.2.2 The VOF method

In the VOF (volume of fluid) method (Harlow and Welch, 1965), the flow field in a two-phase flow is considered as a flow field of a single fluid which has physical properties that vary with a scalar that is transported by the flow. The VOF approach relies on a scalar indicator function between zero and unity to distinguish between two different fluids. A value of zero indicates the presence of one fluid, and value of unity indicates the second fluid. On a computational mesh, volume fraction rates between these two numbers indicate the presence of the interface and the value provides an indication of the relative proportions occupying the cell volume. Using volume fractions is, in general, more economical than markers as only one value must be assigned to each mesh cell.

For an air–water flow, for instance, the density and viscosity may be considered to change between the extremes of air and water over a small range of difference of the scalar as illustrated in Figure 1.7 and this allows identification of the interface state within the computational domain (Hewitt and Vassilicos (2005)).

Another advantage of using volume fractions is that only a scalar convective equation is needed to be solved to circulate the volume fractions into the

computational area. Earlier versions of the VOF method did not take particular account of surface tension, but, more recently, changes of pressure across the interface due to interface curvature and surface tension have been taken account of in the modeling. A detailed review of VOF methods is given by Scardovelli and Zaleski (1999).

In the VOF method, in addition to the conservation of momentum and mass equations, one solves an equation for the filled fraction of each control volume; it is important to ensure that the method does not generate overshoots or undershoots. Fortunately, it is likely to derive schemes that both keep the interface sharp and produce monotone profiles of across it; see Ubbink (1997) or Muzaferija and Perić (1997) for methods specifically developed for interface-capturing in free surface flows. This approach is more efficient than the MAC scheme and can be used for complex free surface shapes including breaking waves. However, the free surface profile is not clearly defined; it is usually smeared over one to three cells.

The VOF method offers an extremely useful tool for investigating interfacial behaviour in multi-phase flows. One of the challenges with the method is that the interface tends to diffuse. A wide variety of 'interface sharpening' methods have been devised to offset this tendency (Hewitt and Vassilicos (2005)).
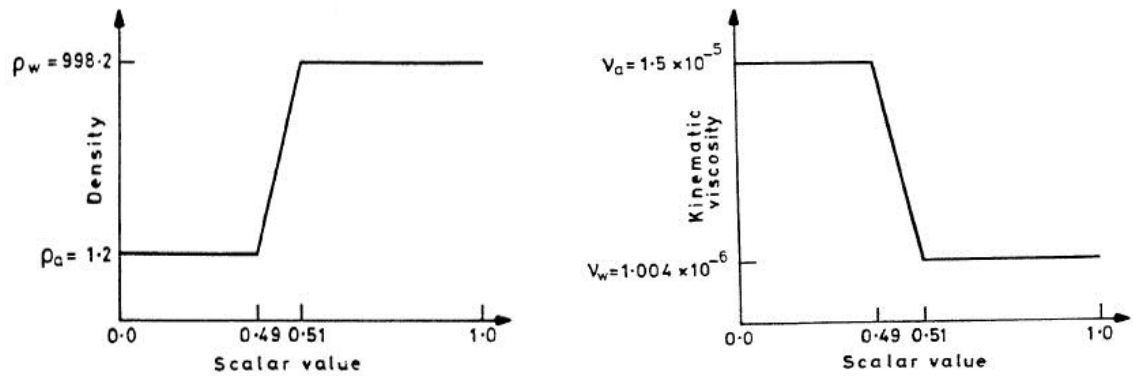
Figure 1.7. Variation of physical properties with scalar value in the VOF method
(Hewitt and Vassilicos (2005))

In volume-tracking or Volume of Fluid (VOF) methods, a fractional volume function is specified to indicate the volume fraction of a particular fluid in each grid cell. In these methods, no explicit representations of the interfaces are defined. Instead, they are reconstructed locally. In order to simulate surface tension effects, which are challenging in the VOF methods, the continuum surface force (CSF) model was introduced by Brackbill et al., (1991). More modern implementations of these approaches were made by Wu et al. (1998).

The renovated interface is not smooth or even continuous, decreasing the precision of the geometrical data (normals and curvature) at the interface discrediting the entire solution. Many scientists have studied to enhance the accuracy of the VOF geometrical data using convolution.

To preserving the conservation of mass through the advection of the fraction factor, accurate algorithms are required. This represents the disadvantage of the method of VOF since conventional differencing methods for the convection term, which ensure a volume fraction field serving the physical bounds of zero and unity, smear the step profile of the interface over many mesh cells because of numerical diffusion, like

16

upwind method. Various techniques have been introduced to account properly for a well-defined interface within the VOF framework. They fall into the groups of donor – acceptor formulation and line methods (geometrical reconstruction).

1.2.2.2.1 Donor-acceptor scheme

The donor – acceptor formulation includes utilizing the volume fraction value of the downstream (acceptor) cell to foretell the level of volume fraction transported to it through a time step. Nevertheless, application of the downstream value may, in general, make the volume fraction values to be unbounded; they may grow larger than unity or narrow down smaller than zero.

If fluid 1 is taken to be the grey-shaded fluid in Figure 1.8 the use of downwinding differencing (the acceptor cell value) in the fluid configuration of Figure 1.8 (a) will ultimately result in volume fraction values much larger than unity in the donor cell due to more fluid 2 being required by the acceptor cell than that available in the donor cell. On the other hand, downwinding differencing in the fluid configuration of Figure 1.8 (b) will ultimately result in negative volume fraction values. The reason is more of fluid 1 is needed by the acceptor cell than there is available in the donor cell. To ensure boundness, the availability of fluid in the donor cell (volume fraction value of the donor cell) must be resorted in order to set the level predicted by the acceptor cell, which brings forth the idea of controlled downwinding (Yeoh and Tu(2009)).
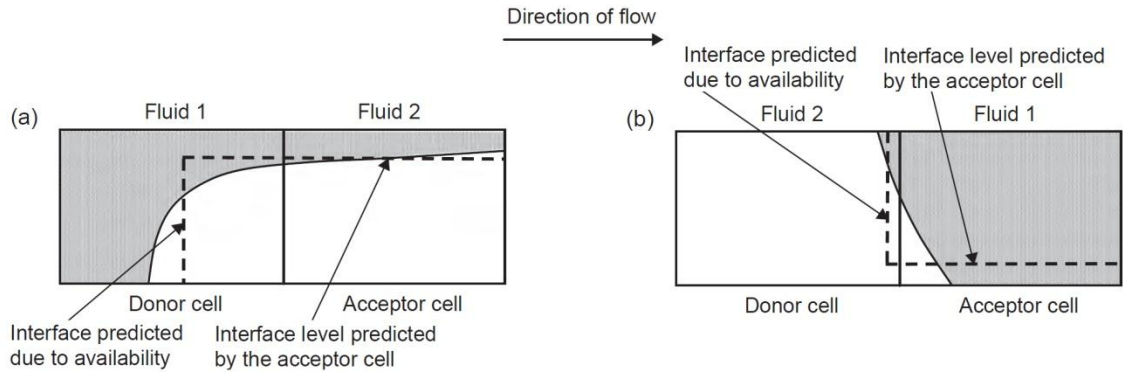
Figure 1.8. A schematic representation of donor–acceptor cell configurations.
(Ubbink (1997)

The boundness criteria for the volume fractions at their respective discrete areas are effectively the physical bounds of zero and unity.

1.2.2.2.2 Line techniques

The well-known Simple Line Interface Calculation (SLIC) by Noh and Woodward (1976) that has been proposed for multi-fluid flows gets place in this category. In this approach, the reconstructed interface is made up of a series of line segments aligned with the grid – the interface is rebuilt using a straight line parallel to one of the coordinate directions and assumes different fluid configurations in that cell for the horizontal and vertical movements, respectively.

A useful refinement to the SLIC method is to fit the interface through oblique lines or piecewise linear segments. First proposed by Youngs (1982), this more accurate line technique is commonly known as the Piecewise Linear Interface Construction (PLIC) method. One critical simplifying feature of this method is that the interface is not required to be reconstructed as a chain of joined segments (a continuous chain of segments) but as a discontinuous chain with however asymptotically small discontinuities.

The reconstructed interface through the PLIC method by Youngs (1982) of the original fluid distribution is illustrated in Figure 1.9. In the same Figure, Ashgriz and Poo (1991) developed a Flux Line-Segment Model for Advection and Interface Reconstruction (FLAIR) by constructing the line segments on the cell faces instead of line segments within the cells.
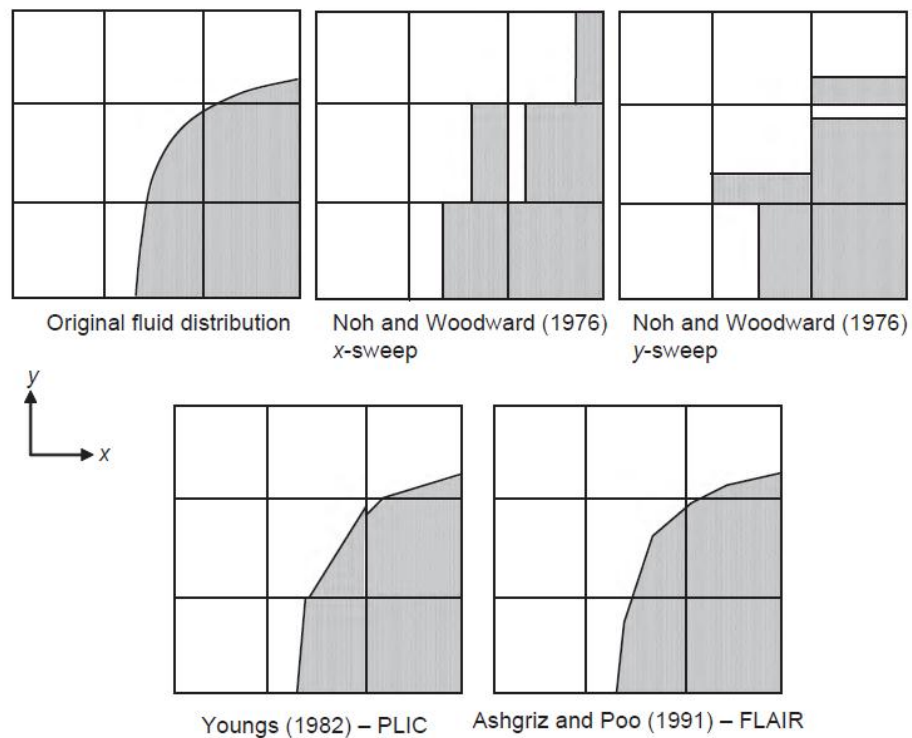

Figure 1.9. Comparison of different line techniques for the prediction of the fluid distribution (Yeoh and Tu(2009))

The local velocities are used to move the reconstructed fluid distributions in each cell in a Lagrangian manner. The new fluid distribution in each cell is then used to update the volume fraction value in each cell.

In all the above methods the rectangular shapes of the mesh cells are implicitly part of the reconstruction algorithm of the interface. The reconstruction of the fluid distribution for these cells in two dimensions is already very complex. The

complexity increases substantially when arbitrary shaped cells in three dimensions are used.

Chapter 2

LITERATURE REVIEW

The purpose of this study is to analyze and compare the effects of different limiters to track the interfaces in free-surface flows. The dynamics of majority of two-phase flows in engineering fields are effectively simulated by the Navier-Stokes equations developed by an equation of state and a Newtonian law of viscosity. Mass and Heat transfer, chemical reactions and also phase changes are not discussed in this study. Open Source Field Operation and Manipulation (OpenFOAM) source codes were employed as the base of analysis. In the source code library, InterFOAM solver has been chosen for extraction of required results using VOF method. Three well-known test cases have been chosen in order to illustrate the differences in various schemes applications: Dam-break, free bubble rise, and advection of hollow shapes in an oblique velocity field.

2.1 OpenFOAM

OpenFOAM (Open Field Operation and Manipulation) is basically a C++ toolbox developed for the customizing and extension of expanding solvers for continuum mechanics simulations. Computational fluid dynamics (CFD) simulations are included in these simulations as well. OpenFOAM includes an expanding collection of solvers that can be applied to a wide range of simulations and problems. This software is originally developed by UK company OpenCFD Ltd. and the source codes are freely available (open source) under the GPL. Initial development was started in order to find a stronger and more flexible simulation platform and triggered

21

in the late 1980s at Imperial College, London. Since then it has evolved by using the latest developments in C++ programing language and adequately rewriting many times over.

In case that the experiment is complex, the solution of the complete Navier-Stokes is essential and approximation methods to capture or track the interface are required. OpenFOAM is software developed to model engineering phenomena of interest in continuum mechanics, especially those relevant to heat transfer and fluid flows. OpenFOAM greatly depends on finite volume numerical method in order to solve a complex of partial differential equations (PDE). This programming software is licensed under the General Public License (GPL) and it is considered as an open source program. For modelling a multiphase flow, this provides a robust and very flexible development environment.

The present study uses the OpenFOAM that is open-source code software for computational fluid dynamics (CFD). OpenFOAM Ltd. releases the software, publicly accessible for Linux operating systems through internet. OpenFOAM can be operated at a regular laptop in a Linux-based operating system environment, and supports parallelization which means it provides the capability of processing multiple simulations over multiple processors simultaneously. In this study, simulations are executed in parallel.

OpenFOAM is constituted by a set of C++ modules and has limited graphical user interface. The built-in utility "paraFoam" provides capabilities of picturing the grid and the obtained results. However, it is helpful for the user to have some knowledge of C++ programming in order to apply this software efficiently. The software is open

source; therefore, the user may access the source codes in any library, utility and solver. OpenFOAM also provides for users numerous capabilities and accompanying demands to vary choices.

2.2 InterFOAM Solver

InterFOAM is a solver embedded in the OpenFOAM application suite that solves multiphase incompressible flows. The solver is based on a modified volume of fluid (VOF) method. This approach incorporates an interfacial compression flux term to reduce the effects of numerical smearing of the interface. It arranges a part of the utilities of OpenFOAM and C++ libraries and is getting more popular and viral in the multiphase flow researchers community.

Rusche (2002) used interFOAM to test various cases of rising air bubbles in water. The velocity of an air bubble rising in a quiescent fluid was agreeing well with experimental data. Paterson (2008) reported recent developments in interFOAM applications. In analysis, the interFOAM solver was coupled with a 3DoF motion solver, so that translational motions could be understood. Modellings of a floating 2D cylinder were carried out for different cylinder weights and the submergence of the center-of-gravity (COG) was compared with theoretical data. The results of the computations matched these values very well.

Recent studies on multiphase flows that have been performed by many researchers during last few years include the following.

Liu and Garcia (2008) compared the velocity profiles for a flow around a partially submerged cylinder with the experimental results. Berberovic *et al* in (2009) compared the process of crater formation and droplet impact with their experiments.

In 2009, Saha and Mitra (2009) regarding to dynamic and static contact angles, compared microfluidic capillary flows with their experiments. Saha *et al* (2009) introduced a comparison of a microfluidic flow in rectangular channels with an analytical solution of Zeng J. in 2007. Ishimoto *et al* (2010) modeled the cavitation assisted atomization in the gasoline injection system and pictorially with a modified version of InterFoam, compared the analyzed instantaneous spray profiles and breakup behavior with their experiments. Roisman *et al* (2010) modeled the droplet impact on a porous surface and measured up droplet spreading with their researches. Srinivasan *et al* (2011) modeled the modulated jets and compared the atomization properties with the experiments of Chaves, H. et al in 2000. Maiwald and Schwarze (2011) modeled the plane plunging jets and compared the obtained data for air entrainment for critical conditions with the experiments of Bolton et al. (1980). Raach *et al* (2011) executed an energy equation in the framework of InterFoam. The solver was not using surface tension simulation to model heat transfer in a film falling over turbulence wires. They evaluated their study by comparing phase velocities and wave peak heights with their experiments in 2008. Gopala *et al* (2011) applied a model for Bingham plastics. They evaluated their simulation against analytical solutions for velocity profile in channel flow. They validated the solver with a flowability test in a V-funnel. Trujillo *et al* (2011) used an energy equation in InterFoam to compute droplet impact heat transfer. Evaluation of this study has been performed by comparing of the experiments of Soriano et al. in 2010 and temperatures with their own trials. Deshpande *et al* (2012) compared the liquid fraction and velocity profiles in a plunging jet flow with their trials. Zhainakov (2012) performed a large scale 3D analyze on flows in complex topological relief areas. The Navier-Stokes Reynolds-averaged equations constitute the basis of the 3D

unsteady simulating by implementing the well-known Volume of Fluid (VOF) in the InterFOAM solver embedded in OpenFOAM 1.7.1 to track the free boundary location. The results have been evaluated by comparing to experimental studies.

2.3 Volume of Fluids (VOF) method

In Volume methods a boundary of volume is used for representing the free surface. The region completely is marked by massless particles or by an indicator function.

Volume fraction methods are of the most commonly used approaches that can be applied to the free surface. Three important volume fraction methods were introduced within 10 years between 70's and 80's: the DeBar's method at 1974, the SLIC method (Noh and Woodward, 1976) and the Hirt and Nichols'VOF method (Hirt and Nichols, 1981). Volume fraction function is used as a scalar indicator function in all above mentioned approaches. To distinguish the presence of phase fluid, this function's range varies from zero (no material) to one (completely filled with material).

In VOF method, the volume which is occupied by one fluid cannot be occupied by another fluid; this verifies the continuity law at all times. The flow properties (i.e. density and viscosity) are a proportional mixture of the properties of both phases. The main disadvantage of the VOF method is that in a numerical modeling with large grid magnitude, the formation of small bubbles or droplets, smaller than the minimum size of the grid is ignored, therefore leads to limiting the method.

In OpenFOAM, the VOF model was initially deployed by Ubbink (1997) within the InterFoam solver codes. Trujillo *et al.* (2007) particularized experimental and numerical studies of horizontal jets below a free surface. Lobosco *et al.* tested the

interFoam in self-aeration regions of stepped spillways in 2011. The researchers successfully reproduced the entrapped-air, however, they faced some problems with the air-entrainment modellings. Deshpande *et al.* in 2012 used InterFoam to analyze and model horizontal jets plunging into a pool and evaluated the results with experimental data. The obtained mean vertical velocity profile along the experimental data concludes satisfactory results. Regarding to the surface curvature, even in modest grid resolution, the solver proved to be accurate; excellent mass conservation; and acceptable advection errors.

The dispersed phases modelling benefits drag and virtual mass models, whereas the resolved phases use the surface-tension models and interface compression of the volume of fluid method (OpenFOAM, 2011).

 In the InterFoam solver, the regular VOF method introduced by Hirt and Nichols in 1981 is used. As mentioned in before, InterFOAM uses the volume fraction as an indicator function to distinguish which part of the cell is filled by the fluid.

The discretization of the scalar transport equation with high-order difference schemes could be performed by the introduction of high-resolution schemes. Various methods can be found in the literature such as Total Variation Diminishing (TVD).

2.4 Convection Discretisation Schemes

Discretisation of the convection term has been a subject of continual intense debate. In order to achieve stability, first-order accurate differencing schemes have been introduced (Courant, Isaacson and Rees (1952), Lax (1954), Gentry et al. (1966)). The numerical procedure can produce values of the dependent variable that are outside of its physically meaningful bounds. If one considers the transport of scalar

properties common in fluid flow problems, such as phase fraction, turbulent kinetic energy, progress variable etc., the importance of boundedness becomes clear. It is therefore essential to obtain bounded numerical solutions when solving transport equations for bounded properties.

A view on the issues of accuracy and boundedness can be based on the sufficient boundedness criterion for the system of algebraic equations. The only convection differencing scheme that guarantees boundedness is Upwind Differencing (UD), as all the coefficients in the system of algebraic equations will be positive even in the absence of physical diffusion (Patankar (1981)). Several researchers (Boris and Book (1973), Raithby (1974) and (1976), Leonard (1979)) shown that in cases of high streamline-to-grid skewness, this degradation of accuracy becomes unacceptable. Several possible solutions to these problems have been proposed, falling into one of the following categories:

Locally analytical schemes (LOADS by Raithby (1979), Power-Law scheme by Patankar (1981)) use the exact or approximate one-dimensional solution for the convection diffusion equation in order to determine the face value of the dependent variable. These schemes are bounded and somewhat less diffusive than UD.

Upwind-biased differencing schemes, including first-order upstream-weighted differencing by Raithby and Torrance (1974), Linear upwinding by Warming and Beam (1976) and Leonard's QUICK differencing scheme (1979). The interpolation that is used in these schemes depends on the direction of the flux. The amount of numerical diffusion is somewhat smaller than for UD, but boundedness is not preserved.

27

Switching schemes are another category to present better accuracy. In his Hybrid Differencing scheme, Spalding (1972) recognizes that the sufficient boundedness criterion holds even for Central Differencing if the cell Peclet number is smaller than two. Under such conditions, Hybrid Differencing prescribes the use of CD, while UD is used for higher Pe-numbers in order to guarantee boundedness.

Blended Differencing, introduced by Peric (1985). Peric proposes a blending approach, using a certain amount of upwinding combined with a higher-order scheme (CD or LUD) by applying a constant blending factor for the whole mesh until boundedness is achieved. Although this approach potentially improves the accuracy, it is not known in advance how much blending should be used.

The quest for bounded and accurate differencing schemes continues with the concept of flux-limiting. Boris and Book (1973) introduce a flux-limiter in their Flux Corrected Transport (FCT) differencing scheme, generalized for multi-dimensional problems by Zalesak (1979). After that Vanleer's researchs (1973) and (1974) established the Total Variation Diminishing (TVD) schemes. TVD schemes have been developed by Harten (1983) and (1984), Roe (1985), Chakravarthy and Osher (1983) and others. A general procedure for constructing a TVD differencing scheme has been described by Osher and Chakravarthy (1984). Sweby (1984) introduces a graphical interpretation of limiters (Sweby's diagram) and examines the accuracy of the method.

TVD schemes can be classified as a switching-blending methodology in which the discretisation practice depends on the local shape of the solution. If offers reasonably good accuracy and at the same time guarantees boundedness. It has been noted

28

(Hirsch (1991), Leonard (1991)) that limiters giving good step-resolution, such as Roe's SUPERBEE (1985) tend to distort smooth profiles. On the other hand, limiters such as MINMOD (Chakravarthy and Osher (1983)), although being suitable for smooth profiles are still too diffusive.

2.5 Dam-Break Problem

In 1995 Koshizuka performed an experimental study over dam break problem and obtained a comprehensive data, leading it to become a reliable source for evaluating numerical method data results. J.S Wang (2000) triggered an investigation over solving dam break problems using total variation diminishing (TVD) method. Applying different limiter schemes to 1D and 2D dam break, they achieved comparative data and illustrated the different performance of the numerical methods results. Larese (2008) performed analysis on the same problem using particle finite element method (PFEM) and the obtained results were in good agreement in comparison to Koshizuka's experimental results. Furthermore, many other studies have been performed by Koshizuka considering different initial conditions and geometries. Koh (2012) used dam break as a test case to apply consistent particle method (CPM) which solves Navier-Stokes equations. In this study, partial differential equations were approximated using Tylor series expansion and obtained a fairly acceptable results comparing to experimental data. F. Moukalled et al. (2012) performed a comprehensive simulation over dam break problem using various transient schemes including Euler, B-SOUE, and recently developed $TICS^{1.76}$ and $TICS^{2.5}$, solved using STACS and SMART convective scheme. By visualizing the results using interface contour, it was concluded that $TICS^{1.75}$ and $TICS^{2.5}$ present more accurate and less diffusive results and preserves interface boundedness and sharpness.

2.6 Free bubble rise problem

The experimental researches of free bubble rise in viscous liquids due to buoyancy have received extensive debates in the literature within last ten years by Grace (1973), Clift et al., 1978 and Raymond and Rosant (2000).

Early studies on the rise of a bubble in an inviscid or a viscous fluid were reported in the works of Hartunian and Sears (1957), Walters and Davidson (1962), Walters and Davidson (1963), Wegener and Parlange (1973) and Bhaga and Weber (1981). However, our understanding on bubble rise and deformation is still limited to a few flow regimes only, due to the difficulties in experiments. It is rather difficult to measure, without any interference, the flow pattern and pressure distribution within a bubble and its surrounding liquid while it is rising and deforming. As a result, approximate theoretical solutions have been derived in the limit of very small bubble deformations (low Bond number) for either high (Moore (1959)) or low (Taylor (1964)) Reynolds numbers, where the bubble shape is relatively stable. In the analysis work of Davies and Taylor (1950), the rising speed of a spherical-cap bubble was related to the radius/curvature of the bubble at the forward stagnation point. Hence, they took the overall spherical-cap as a priori shape rather than being determined as part of solution.

An enhanced numerical method for front tracking method is developed to model the rising of a bubble in quiescent viscous liquid by Jinsong Hua and Jing Lou (2007). In the new numerical approach that presented by them, volume correction is introduced to maintain the volume conservation of the bubble while the bubble's rising and deforming is being tracked and volume flux conservation based SIMPLE algorithm

using finite volume method is utilized to solve the Navier–Stokes equation for fluid flow. The modelling data results were compared with the experimental results in terms of terminal bubble shape and velocity. The experimental results clearly agreed with the predicted bubble shape and terminal velocity. Recently, A. Albadawi et al. (2013) modeled the bubble behavior and categorize its geometrical properties under different physical influences to understand the effects of the bubble and the wake generated behind it on mass and heat transfer problems.

Chapter 3

NUMERICAL METHOD

The solver of open-foam toolbox is an inter-foam multiphase solver and can simulate multiphase flows. In this part, the mathematical formulation and equation discretization of inter-Foam solver are explored to understand the source code better. This Section considers the works of Jasak (1996), Ubbink (1997) and Rusche (2002).

3.1 Conservation Equations

The movement of fluid is determined by a set of equations expressing the conservation of mass, energy and momentum. The mathematical statements of the conservation laws of physics are presented by the defined equations of fluid flow. The physical behavior of the fluid is completely determined and completely independent of the nature of the fluid. The governing equations of fluid continuum mechanics can be written in a 3D system in the differential form (Aris, 1989).

3.1.1 Conservation of Mass

Mass can neither be destroyed nor created. That means that mass must be conserved over time.

$$\frac{\partial \rho}{\partial t} + \nabla.(\rho \mathbf{u}) = 0 \tag{3.1}$$

where ρ denotes fluid density and $\mathbf{u}$ is the fluid velocity. This result is known as the continuity equation. For an incompressible fluid (when density is constant), this equation is shortened to:

$$\nabla. \mathbf{u} = 0 \tag{3.2}$$

3.1.2 Conservation of Momentum

As the Newton's second law says, the sum of the forces on a fluid particle equals the rate of change of momentum. Velocity profile would be obtained from this equality. Forces on a particle of fluid could be classified into two types; the first one is surface forces such as normal forces (pressure and stress) or viscous forces, and the second one is body forces such as gravity forces or Coriolis forces. They are usually incorporated as additional source terms into the momentum equations.

$$\frac{\partial(\rho\mathbf{u})}{\partial t} + \nabla \cdot (\rho\mathbf{u}\mathbf{u}) = -\nabla P + \nabla \cdot \tau + \rho g + \mathbf{F} \tag{3.3}$$

Where gravity acceleration vector and velocity vector represented by g and $\mathbf{u}$ respectively, P denotes the pressure, τ is the viscosity stress tensor, and F is related to the surface tension:

$$\mathbf{F} = \int_{S(t)} \sigma \kappa' n' \delta(\mathbf{x} - \mathbf{x}')dS \tag{3.4}$$

In this equation, σ denotes the surface tension coefficient, $\mathbf{x}$ is the position vector, $\mathbf{x}'$ denotes the position vector of interface, $\mathbf{n}$ are the normal vector of the interface, S is the surface area vector and κ represent the curvature that described in section 3.3. More information about surface tension is available. To obtain more efficiency the viscous stress term can be reformulated. The final form of this term is as below equation:

$$\nabla \cdot \tau = \nabla \cdot (\mu[\nabla\mathbf{u} + (\nabla\mathbf{u})^{\mathrm{T}}]) = \nabla \cdot (\mu\nabla\mathbf{u}) + (\nabla\mathbf{u}) \cdot \nabla\mu \tag{3.5}$$

where μ is a property called dynamic viscosity. The modified pressure p* (p_$\rho g h$ in OpenFOAM code) is adopted in interFoam by removing the hydrostatic pressure ($\rho g \cdot \mathrm{x}$) from the pressure P. This is an advantage for the specification of pressure at the boundaries of the space domain (Rusche, 2002). The gradient of the modified pressure is defined as:

$$\nabla p^* = \nabla P - \nabla(\rho g \cdot \mathrm{x}) = \nabla P - \rho g - g \cdot \mathrm{x} \nabla \rho \qquad (3.6)$$

3.2 Indicator Function (VOF method)

The conventional volume of fluid method presented by Hirt and Nichols (1981) is

used in the interFoam solver. In this method, the volume fraction is taken as an

indicator function (alpha in OpenFOAM code) to describe which segment of the cell

is occupied by the fluid, as mentioned in below equation:

$$\alpha = \begin{cases} 1 & \text{for a cell occupied by fluid 1} \\ 0 < \alpha < 1 & \text{for a cell involving the interface} \\ 0 & \text{for a cell occupied by fluid 2} \end{cases} \qquad (3.7)$$

The transport of α is calculated by solving scalar convection equations represented

as:

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{u}) = 0 \qquad (3.8)$$

The previewed equation proves that the conservation of mass is equivalent to the

conservation of volume in an incompressible fluid and based on that conservation of

the function α (C erne $et\ al.$, 2001) is observed. The local fluid properties (μ and ρ)

evaluated using the following mixture relations:

$$\rho = \alpha \rho_1 + (1-\alpha)\rho_2 \quad \text{and} \quad \mu = \alpha \mu_1 + (1-\alpha)\mu_2 \qquad (3.9)$$

The subscript 1 denotes to fluid 1 and subscript 2 represents fluid 2. In the case of

having more than two fluids. Equations (3.8) and (3.9) are reformulated and shown

below as equations (3.10) and (3.11), respectively.

$$\frac{\partial \alpha^{(k)}}{\partial t} + \nabla \cdot (\alpha^{(k)} \mathbf{u}) = 0 \qquad (3.10)$$

$$\rho = \sum_{k=1}^{n} \alpha^{(k)} \rho^{(k)} \quad \text{and} \quad \mu = \sum_{k=1}^{n} \alpha^{(k)} \mu^{(k)} \quad \text{n = number of fluids} \qquad (3.11)$$

where $\alpha^{(k)}$ denotes the volume fraction α of the k th fluid that it constrained by a conservation of volume equation as:

$$\sum_{k=1}^{n} \alpha^{(k)} = 1 \quad \text{for } k = 1, 2, ..., n \tag{3.12}$$

where n represents the number of fluids. In the case of high-density fluids, the conservation of the phase fraction is essential. That's where significant errors on the physical properties can be caused by small errors on the volume fraction. Function (3.8) is against the effect of high-density fluids (Rusche, 2002) and as a respond many researchers has presented alternative techniques to solve this problem (Ubbink, 1997; Ubbink and Issa, 1999). The best alternative was created by Weller (2002). He introduced an extra term in the phase fraction function − the artificial compression term.

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \overline{\mathbf{u}}) + \underbrace{\nabla \cdot [\mathbf{u}_r \alpha (1 - \alpha)]}_{\text{Artifical compression term}} = 0 \tag{3.13}$$

where $\overline{\mathbf{u}}$ represents the mean velocity and $\mathbf{u}_r = \mathbf{u}_1 - \mathbf{u}_2$ denotes compression velocity that is the vector of relative velocity between the two fluids (Berberović *et al.*, 2009).The mean velocity $\overline{\mathbf{u}}$ computed by a weighted average of the velocity between the two fluids:

$$\overline{\mathbf{u}} = \alpha \mathbf{u}_1 + (1 - \alpha) \mathbf{u}_2 \tag{3.14}$$

3.3 Surface Tension Force

The surface tension force acts on the interface between the two phases. The interface is not tracked explicitly in the interface-capturing methodology and its exact form and location are unknown (Rusche, 2002). In the momentum equation(3.3), the source term (F) relative to the surface tension, cannot be solved directly. Brackbill *et al.* (1991) improved the Continuum Surface Force (CSF) method that solves this

35

problem by converting the F term into a volume force function of the surface tension. In this CSF model, the surface curvature (κ) is formulated from local gradients in the surface normal (**n**) at the interface, that is a function of the phase fraction ($\mathbf{n} = \nabla \alpha$) (Tang and Wrobel, 2005):

$$\kappa = \nabla \cdot \hat{\mathbf{n}} = \nabla \cdot \frac{\mathbf{n}}{|\mathbf{n}|} = \nabla \cdot \left(\frac{\nabla \alpha}{|\nabla \alpha|} \right) \tag{3.15}$$

The volumetric surface tension force (**F**) is written in terms of the surface tension, and subsequently, to the jump pressure across the interface.

$$\mathbf{F} = \sigma \kappa \frac{\rho}{0.5(\rho_1 + \rho_2)} \approx \sigma \kappa \nabla \alpha \tag{3.16}$$

By considering the viscous stress term (3.5), the modified pressure (3.6) and the volumetric form of surface tension (3.16), the last form of the Navier-Stokes equation is as below:

$$\frac{\partial(\rho \mathbf{u})}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) - \nabla \cdot (\mu \nabla \mathbf{u}) = -\nabla p^* + (\nabla \mathbf{u}) \cdot \nabla \mu - g \cdot \mathbf{x} \nabla \rho + \sigma \kappa \nabla \alpha \tag{3.17}$$

The final form of the mathematical model by using VOF concept is constituted by the continuity equation (3.2), the modified indicator function (3.13) and the momentum equation (3.17). The constitutive relations for dynamic viscosity and density (equation(3.9)) can help to solve these equations.

3.4 Finite Volume Method (FVM)

This section defines the discretization of the governing equations using the FVM method. The FVM of solution is subdivided in two parts: time domains and space (Rusche, 2002).

3.4.1 Discretization of the General Transport Equation

The FVM (Finite Volume Method) discretization of space needs a subdivision of the simulation domain into a finite number of very small control volumes (CVs)

(Ferziger & Peric, 2002). Cells bounded by a set of flat faces are called control volumes. Each face is shared by two cells. The first one is control volume cell (or owner cell) and the other one is neighbouring cell.

In Figure 3.1 an example of an owner cell is shown. The point N denotes the centroid of the neighbouring cell and the centroid of the computational cell is represented by the point P. Those cells have the internal face f in common, having the normal vector shown by **A**. The vector **A** points always outwards from the computational cell, with magnitude equal to the area of the face f. The vector **d** connects the point P to N while the vector **D** is the vector with the same direction of **d** but magnitude able to satisfy the conditions proposed by Jasak (1996).

$$\mathbf{A} = \mathbf{D} + \mathbf{k} \tag{3.18}$$

$$\mathbf{D} = \frac{\mathbf{d}|\mathbf{A}|^2}{\mathbf{d} \cdot \mathbf{A}} \tag{3.19}$$

In orthogonal meshes, the angle between **A** and **d** is zero and the vectors **D** and **k** are omitted (Ubbink, 1997).
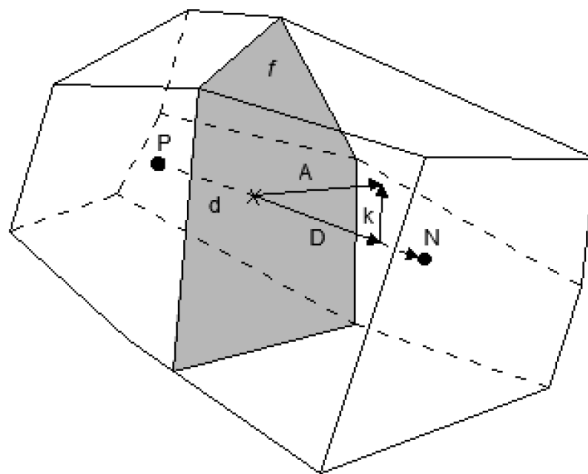
Figure 3.1. CV and parameters of the discretisation of the solution domain. P and N are the centroid of two neighbouring cells, **d** is the vector between P and N and **A** the vector normal to the face f common to both cells (addapted from Ubbink (1997))

The FVM discretization of the governing equations is done in a few levels. At the first step the equations are presented as volume integrals over each CV and later by using using Gauss"s theorem, converted to surface flux terms. Then, the surface integrals are calculated by a sum of fluxes over all CVs faces. Finally, to determine the fluxes, cell face values of variables are estimated by interpolation using cell centered values at neighbouring cells.

The description below describes and summarizes the finite volume discretization for a generic transport equation by applying a generic scalar ϕ. The generic transport equation represents any conservation law.

$$\underbrace{\frac{\partial(\rho\phi)}{\partial t}}_{\text{transient term}} + \underbrace{\nabla\cdot(\rho\phi\mathbf{u})}_{\text{convection term}} = \underbrace{\nabla\cdot(\Gamma_\phi\nabla\phi)}_{\text{diffusion term}} + \underbrace{S_\phi}_{\text{source term}} \qquad (3.20)$$

where the density is ρ, $\mathbf{u}$ denotes the velocity vector, transported variable ϕ represents a flow quantity (property), Γ and S_ϕ are diffusion coefficient and the source term, respectively. Discretizing (3.20) over a time interval t, $t+\Delta t$ and over the volume V_P (cell with the centroid point P), the volume integral form, results:

$$\int_t^{t+\Delta t}\left[\frac{\partial}{\partial t}\int_{V_P}\rho\phi dV + \int_{V_P}\nabla\cdot(\rho\phi\mathbf{u})dV - \int_{V_P}\nabla\cdot(\Gamma_\phi\nabla\phi)dV\right]dt = \int_t^{t+\Delta t}\left[\int_{V_P}S_\phi dV\right]dt \quad (3.21)$$

The parts below show the main levels for the spatial and temporal discretization of the transport equation.

3.4.1.1 Gradient Term

As stated before by using the Gauss theorem, the advection and diffusion terms needs to be simplified into surface integrals over the cell faces. The Gauss"s divergence theorem is represented:

$$\int_V(\nabla\cdot\phi)dV = \oint_{\partial V}\phi\cdot dS \qquad (3.22)$$

where ∂V represents closed surface bounding the volume V and dS denotes an extremely small surface element outward pointing normal on ∂V.

Since the variable ϕ are stored on the cell center, the respective value on the face needs to be obtained by interpolation. Considering the linear variation of ϕ in space x, the integral of this variable in the volume V_P is:

$$\int_V \phi(\mathrm{x})dV = \int_{V_P} \left[\phi_P + (\mathrm{x} - \mathrm{x}_P) \cdot (\nabla\phi)_P\right]dV = \phi_P V_P \qquad (3.23)$$

It is possible to transform equation (3.22) into a sum of integrals over the faces and a linear variation of ϕ, by using Gauss theorem it can be discretized as:

$$\int_{V_P} \nabla\phi \, dV = \oint_{\partial V_P} \phi \, dS = \sum_f \left(\int_f \phi \, dS\right) \approx \sum_f A_f \phi_f \qquad (3.24)$$

where ϕ_f is the face value of variable ϕ and A is the outward normal surface area vector of the faces in the control cell.

3.4.1.2 Face interpolation scheme

There are many schemes to interpolate the field ϕ. In following sections some of the face interpolation schemes that are used in this thesis are presented.

3.4.1.2.1 The Upwind Differencing (UD) scheme

This scheme assumes ϕ_f, determined according to the direction of the flow. It means that the face value ϕ_f is set equal to the cell-center value of ϕ in the upstream cell. In this scheme, the boundedness of solution is guaranteed, however the order of accuracy of the discretization is not guaranteed as second order and the solution can become distorted (Jasak, 1996).

$$\phi_f = \begin{cases} \phi_P & \text{for} \quad \text{flux} \geq 0 \\ \phi_N & \text{for} \quad \text{flux} < 0 \end{cases} \qquad (3.25)$$

3.4.1.2.2 The 2nd order Upwind Differencing (UD) scheme

When second-order accuracy is aimed, values at cell faces are computed using a multidimensional linear reconstruction approach. In this approach, higher-order accuracy is achieved at cell faces through a Taylor series expansion of the cell-centered solution about the cell centroid. Therefore the face value ϕ_f is computed by applying the next equation ((3.26)) when second-order upwinding is chosen:

$$\phi_f = \begin{cases} \phi_P + \nabla \phi_P \cdot r_{Pf} & \text{for} \quad \text{flux} \geq 0 \\ \phi_N + \nabla \phi_N \cdot r_{Nf} & \text{for} \quad \text{flux} < 0 \end{cases} \tag{3.26}$$

where r_{Nf} and r_{Pf} are the displacement vectors from the upstream cell centroid (point N) to the face centroid and from the cell centroid (point P) to the face centroid, respectively.

3.4.1.2.3 The Central Differencing (CD) scheme

CD scheme assumes a linear variation of ϕ between the upstream cell centroid N and the cell centroid (Point P). The face value ϕ_f is calculated based on the

$$\phi_f = f_x \phi_P + (1 - f_x) \phi_N \tag{3.27}$$

where the interpolator factor f_x is defined as the ratio of the distances $\mathbf{r}_{Nf}$ and $\mathbf{r}_{PN}$. Ferziger and Peric (2002) showed that this scheme is second order accurate, although it causes violates the boundedness of the solution and unphysical oscillations in the solution (Pantakar, 1980).

3.4.1.2.4 The TVD Scheme

For finding the face value ϕ_f by TVD scheme, the equation (3.28) is used as shown below:

$$\phi_f = \phi_N + \frac{1}{2}\psi(r_f)(\phi_P - \phi_N) \quad \text{if P is upstream node}$$

$$\phi_f = \phi_P + \frac{1}{2}\psi(r_f)(\phi_N - \phi_P) \quad \text{if N is upstream node}$$

(3.28)

Where $\psi(r_f)$ is a flux limiter function and r_f ratio becomes

$$r_f = \frac{(2\nabla\phi_N \cdot \mathbf{r}_{PN})}{\phi_P - \phi_N} - 1 \quad \text{if P is upstream node}$$

$$r_f = \frac{(2\nabla\phi_P \cdot \mathbf{r}_{PN})}{\phi_N - \phi_P} - 1 \quad \text{if N is upstream node}$$

(3.29)

Note that this r_f-formulation can also be applied for any higher order scheme. Some

of the most popular schemes which are used as a flux limiter are written as below

UD Scheme $\quad \psi(r) = 0$

QUICK Scheme $\quad \psi(r) = 0.25r + 0.75$

VAN LEER Scheme $\quad \psi(r) = \dfrac{r + |r|}{1 + |r|}$

UMIST Scheme $\quad \psi(r) = \max\left[0, \min\left(2r, (1+3r)/4, (3+r)/4, 2\right)\right]$

MINMOD Scheme $\quad \psi(r) = \max\left[0, \min\left(1, r\right)\right]$

SUPERBEE Scheme $\quad \psi(r) = \max\left[0, \min\left(2r, 1\right), \min\left(r, 2\right)\right]$

(3.30)

3.4.1.3 Transient Term

The transient term [first term of equation (3.21)] is usually discretized using a 2nd

order or a first order accurate scheme in time. The Euler implicit is an example of a

first order time differencing scheme that is used in this work:

$$\int_{V_P} \frac{\partial \rho\phi}{\partial t} dV \approx V_P \frac{\rho^n \phi^n - \rho^0 \phi^0}{\Delta t}$$

(3.31)

where ϕ^0 is the known value of ϕ from the previous time step and ϕ^n is the unknown

value of ϕ at the current time step.

3.4.1.4 Convection Term

Appling Equation (3.24) to the convection term of the momentum equation gives:

$$\int_{V_P} \nabla \cdot (\rho \mathbf{u} \phi) dV = \sum_f A_f \cdot (\rho \mathbf{u} \phi)_f = \sum_f A_f \cdot (\rho \mathbf{u})_f \phi_f = \sum_f F_f \phi_f \qquad (3.32)$$

where F_f represents mass flux through the face based on a known velocity field and

ϕ_f is determined through one of the interpolation schemes.

3.4.1.5 Diffusion Term

In a similar way to the convection term, the diffusion term is discretized as

$$\int_{V_P} \nabla \cdot (\Gamma \nabla \phi) dV = \sum_f \Gamma_f A_f \cdot (\nabla \phi)_f \qquad (3.33)$$

where Γ represented the diffusivity and Γ_f the diffusivity at face is computed by

applying the one interpolation scheme.

If the mesh is orthogonal, i.e. vectors $\mathbf{A}$ and $\mathbf{D}$ are parallel as shown in Figure 3.1.

By considering equation (3.19) it is possible to use the following expression:

$$A_f \cdot (\nabla \phi)_f = |\mathbf{D}_f| \frac{\phi_N - \phi_P}{|\mathbf{d}_f|} \qquad (3.34)$$

If the mesh is non-orthogonal then it is essential to define an additional explicit term

($\mathbf{k}_f$) to induce higher accuracy of the equation (3.34):

$$A_f \cdot (\nabla \phi)_f = \mathbf{D}_f \cdot (\nabla \phi)_f + \mathbf{k}_f \cdot (\nabla \phi)_f \qquad (3.35)$$

There are many possible decompositions to correct the orthogonality. In open foam, the orthogonal correction is made using an Over-relaxed approach (Jasak, 1996; Ubbink, 1997).

For finding $(\nabla \phi)_f$, Equation (3.36) is used as given below:

$$(\nabla \phi)_f = f_x (\nabla \phi)_P + (1 - f_x)(\nabla \phi)_N \qquad (3.36)$$

where f_x is an interpolation factor and is defined as

$$f_x = \frac{\left|\mathbf{x}_f - \mathbf{x}_N\right|}{\left|\mathbf{x}_f - \mathbf{x}_N\right| + \left|\mathbf{x}_f - \mathbf{x}_P\right|} \tag{3.37}$$

3.4.1.6 Source Term

The right side terms in equation (3.21), which is not possible to be written as diffusion, convection or transient terms, are treated as sources. A simple process of linearization follows the work of Pantakar (1980):

$$S_\phi(\phi) = S_u + S_p\phi \tag{3.38}$$

Using the assumption made in Equation (3.23) the volume integral of the source term is:

$$\int_{V_P} S_\phi(\phi)dV = S_u V_P + S_p \phi_P V_P \tag{3.39}$$

3.4.2 MULES Approach

The Flux Corrected Transport (FCT) is a technique introduced by Boris and Book (Boris and Book, 1973) and improved by Zalesak (Zalesak, 1979) as a way to guarantee boundedness in the solution of hyperbolic problems. OpenFOAM implementation of FCT theory is called MULES (Multidimensional Universal Limiter for Explicit Solution); it relies on similar concepts with respect to Zalesak's limiter (Zalesak, 1979) but the determination of λ's (the limiter in equation (3.45)) is iterative.

This approach is described first by rewriting equation (3.8) in integral form as

$$\int_t^{t+\Delta t}\left[\int_V \frac{\partial \alpha}{\partial t}dV\right]dt + \int_t^{t+\Delta t}\left[\int_V \nabla \cdot (\alpha \mathbf{u})dV\right]dt = 0 \tag{3.40}$$

After discretizing equation (3.40) by using equations (3.31) and (3.32), it becomes:

$$\frac{\left(\alpha_P\right)^n - \left(\alpha_P\right)^0}{\Delta t} = -\frac{1}{V_P}\sum_f (F_f \alpha_f)^0 \tag{3.41}$$

43

where the transient term is discretized using the forward Euler scheme, and the convection term appears as a summation over the cell faces. The values of the flux depend on many variables but particularly on the values of α at faces. Boundedness of the temporal solution can be achieved via face value limiting, such as in TVD schemes, or by limiting the face fluxes.

In this approach the values of F are obtained by a lower order and bounded method and a limited portion of the values obtained by a high order and possible unbounded method, so equation (3.41) can be written as

$$\frac{(\alpha_P)^n - (\alpha_P)^0}{\Delta t} = -\frac{1}{V_P} \sum_f (F_u + \lambda_M F_c)^0 \tag{3.42}$$

where λ_M is a limiter that is implemented in the MULES solver and is equal to one in the transition region (interface) and zero elsewhere. The advective fluxes F_u and F_c are defined as

$$\begin{aligned} F_u &= \gamma_f \alpha_{f,upwind} \quad \text{and} \\ F_c &= \gamma_f \alpha_f + \gamma_{rf} \alpha_{rf} (1-\alpha)_{rf} - F_u \end{aligned} \tag{3.43}$$

where γ_f (volume flux) is given by

$$\gamma_f = \mathbf{u}_f \cdot \mathbf{A}_f \tag{3.44}$$

The face values for volume fraction to find F_c are calculated using a blending of central and upwind schemes as follows:

$$\alpha_f = \alpha_P + \frac{(\alpha_N - \alpha_P)}{2} \left[1 - \varsigma(\gamma_f)(1 - \lambda_\gamma) \right] \tag{3.45}$$

where $\varsigma(\gamma_f)$ is a step function and the limiter λ_γ can be chosen from among several alternatives, such as Van-Leer, SuperBee, Minmod and QUICK. In all the

results presented here, the Van-Leer scheme was used to obtain the face centered

values for α. $\varsigma(\gamma_f)$ is given by

$$\varsigma(\gamma_f) = \begin{cases} 1 & \text{for} \quad \gamma_f \geq 0 \\ -1 & \text{for} \quad \gamma_f < 0 \end{cases} \tag{3.46}$$

Regarding the compressive flux, $\gamma_{rf}\alpha_{rf}(1-\alpha)_{rf}$ in equation (3.43) is given by

$$\gamma_{rf} = \min\left(C_\gamma \frac{|\alpha_f|}{|A_f|}, \max\left[\frac{|\alpha_f|}{|A_f|}\right]\right)\left(\mathbf{n}_f \cdot A_f\right) \tag{3.47}$$

where the max operation is performed over the entire domain, while the min

operation is done locally at each face. The constant C_γ is a user-specified value,

which serves as a parameter to restrict interface smearing.

Finally, the quantities α_{rf} similar to equation (3.45) is calculated as

$$\alpha_{rf} = \alpha_P + \frac{(\alpha_N - \alpha_P)}{2}\left[1 - \varsigma(\gamma_f)(1-\lambda_{\gamma r})\right] \tag{3.48}$$

where the limiter $\lambda_{\gamma r}$ is calculated as

$$\lambda_{\gamma r} = \min\left\{\max\left(1 - \max\left[\left(1 - \left(4\alpha_P\left(1-\alpha_P\right)\right)\right)^2, \left(1 - \left(4\alpha_N\left(1-\alpha_N\right)\right)\right)^2\right], 0\right), 1\right\} \tag{3.49}$$

3.4.3 Discretization of the Spatial Terms of Momentum Equation

Following the discretization process presented for the general transport equation, the

momentum equation (3.17) of the Navier-Stokes equations over the control volume

and the time step Δt can be presented as follow:

$$\int_t^{t+\Delta t}\left[\frac{\partial}{\partial t}\int_{V_P}\rho\mathbf{u}dV + \int_{V_P}\nabla\cdot(\rho\mathbf{u}\mathbf{u})dV - \int_{V_P}\nabla\cdot(\mu\nabla\mathbf{u})dV\right]dt$$
$$= \int_t^{t+\Delta t}\left[\int_{V_P}(-\nabla p^* + (\nabla\mathbf{u})\cdot\nabla\mu + g\cdot\mathbf{x}\nabla\rho + \sigma\kappa\nabla\alpha)dV\right]dt \tag{3.50}$$

45

The finalized form of the momentum equation after the terms discretization is as below:

$$\int_t^{t+\Delta t}\left[V_P\frac{\rho^n\phi^n-\rho^0\phi^0}{\Delta t}+F_f\mathbf{u}_f+\sum_f\mu_f A_f\cdot(\nabla\mathbf{u})_f\right]dt$$
$$=\int_t^{t+\Delta t}\left[S_uV_P+S_p\phi_PV_P\right]dt \tag{3.51}$$

3.4.4 Discretization of the Phase Fraction Transport Equation

The final form of the indicator function transport equation is defined in previous parts by the equation (3.13). The finite volume discretization over the volume control and the time step Δt assumes this form:

$$\int_t^{t+\Delta t}\left[\int_V\frac{\partial\alpha}{\partial t}dV\right]dt+\int_t^{t+\Delta t}\left[\int_V\nabla\cdot(\alpha\mathbf{u})dV\right]dt+\int_t^{t+\Delta t}\left[\int_V\nabla\cdot[\mathbf{u}_r\alpha(1-\alpha)]dV\right]dt$$
$$=0 \tag{3.52}$$

The first term of the equation by assuming the linear variation of indicator function (α), can reduced to:

$$\int_V\frac{\partial\alpha}{\partial t}dV=\frac{\partial\alpha_P}{\partial t}V_P \tag{3.53}$$

The second and third terms of the equation (3.52) are discretized applying the Gauss theorem.

Following the description of the open foam manual (openfoam, 2014), the transient PDE presented in (3.21) can be simplified as the following equation by considering the spatial terms $\Upsilon\phi$ where Υ is a spatial operator:

rewriting equ (3.21) $\rightarrow$

$$\int_t^{t+\Delta t}\left[\frac{\partial}{\partial t}\int_{V_P}\rho\phi dV+\underbrace{\int_{V_P}\nabla\cdot(\rho\phi\mathbf{u})dV-\int_{V_P}\nabla\cdot(\Gamma_\phi\nabla\phi)dV-\int_{V_P}S_\phi dV}_{\int_{V_P}\Upsilon\phi dV}\right]dt=0 \tag{3.54}$$
$$\int_t^{t+\Delta t}\left[\frac{\partial}{\partial t}\int_{V_P}\rho\phi dV+\int_{V_P}\Upsilon\phi dV\right]dt=0$$

By applying the Euler implicit method (Equation (3.31)), the first term of the Equation (3.54) returns:

$$\int_t^{t+\Delta t}\left[\frac{\partial}{\partial t}\int_{V_P}\rho\phi dV\right]dt=\int_t^{t+\Delta t}\left[\frac{\left(\rho_P\phi_P V_P\right)^n-\left(\rho_P\phi_P V_P\right)^0}{\Delta t}\right]dt$$
$$=\rho_P\frac{\left(\phi_P\right)^n-\left(\phi_P\right)^0}{\Delta t}V_P \tag{3.55}$$

The second term of the equation (3.54) can be discretized in open foam by three ways: Euler implicit, Euler explicit and Crank Nicholson.

The Euler implicit guarantees boundedness, it is unconditionally stable and is a 1st order scheme accurate in time (Hirsch 1988). This scheme uses the current values of ϕ, thereby the solution needs to be achieved using a matrix.

The Euler explicit is a first order scheme accurate in time and is unstable for Courant numbers greater than unity. This scheme uses only the old values of ϕ. The Courant number is defined as:

$$Co=\mathbf{u}\,\frac{\overset{\text{time step}}{\Delta t}}{\underset{\text{length of a cell}}{\Delta x}} \tag{3.56}$$

The Crank Nicholson is a second order scheme accurate in time. This scheme does not guarantee the boundedness of the solution (Jasak, 1996). It uses the trapezoidal rule to discretize the spatial terms, thereby taking the mean value of the current values and old values of ϕ.

$$\int_t^{t+\Delta t}\Upsilon\phi dt=\frac{1}{2}\Upsilon(\phi^n+\phi^0) \tag{3.57}$$

Assuming now that the viscosity (μ) and density (ρ) do not change in time and by applying the Crank Nicolson scheme (3.57), the momentum equation returns:

47

$$V_P \frac{\rho \mathbf{u}^n}{\Delta t} - V_P \frac{\rho \mathbf{u}^0}{\Delta t} + \frac{1}{2} \left(\sum_f F_f \mathbf{u}_f^n + \sum_f F_f \mathbf{u}_f^0 \right)$$

$$- \frac{1}{2} \left(\sum_f \mu_f \mathbf{A}_f \cdot (\nabla \mathbf{u})_f^n + \sum_f \mu_f \mathbf{A}_f \cdot (\nabla \mathbf{u})_f^0 \right) - \frac{1}{2} \left(S_p V_P \phi_P^n + S_p V_P \phi_P^0 \right) + S_u V_P = 0$$

rewrite it to following form $\rightarrow$ (3.58)

$$V_P \frac{\rho \mathbf{u}^n}{\Delta t} + \frac{1}{2} \left[\sum_f F_f \mathbf{u}_f^n - \sum_f \mu_f \mathbf{A}_f \cdot (\nabla \mathbf{u})_f^n - S_p V_P \phi_P^n \right]$$

$$= V_P \frac{\rho \mathbf{u}^0}{\Delta t} - \frac{1}{2} \left[\sum_f F_f \mathbf{u}_f^0 - \sum_f \mu_f \mathbf{A}_f \cdot (\nabla \mathbf{u})_f^0 - S_p V_P \phi_P^0 \right] + S_u V_P$$

The discretisation and linearization procedure outlined produces a linear algebraic equation for each control volume. Since $\mathbf{u}_f$, $(\nabla \mathbf{u})_f$ and other terms depend on the values on the neighbouring cells, the new values of $\mathbf{u}_P$ can be implicitly achieved from the generic equation:

$$a_P \mathbf{u}_P^n + \sum_N a_N \mathbf{u}_N^n = S_\phi \qquad (3.59)$$

Chapter 4

RESULTS

4.1 Dam-Break Problem

In Figure 4.1 schematic of the model is showed. In the mentioned model, a column of water with width of $a = 0.146$m and height $2a$ is located on the left side of a square tank with sides of size $4a$ filled with air. Water is considered viscous, with a constant density ρ_w and viscosity μ_w of values 998.2 kg/m^3 and 0.993×10^{-3} $Pa \cdot s$, respectively. Air is also considered to be of constant density and viscosity $(\rho_a = 1.164 \ kg/m^3, \ \mu_a = 1.824 \times 10^{-5} \ Pa \cdot s)$. The water column, which is initially at rest, starts collapsing under its own weight at time $t = 0 \ s$.

The above mentioned problem is different from the previously showed problems that the velocity field is not known and it is achieved as a part of the solution. Furthermore, the continuity and Navier-Stokes equations have to be solved in addition to the volume fraction α equation. The resulting flow field at time $t > 0$ is modeled as laminar, the gravitational acceleration g is assigned the value of $9.81 \ m/s^2$, and surface tension effects are neglected. The boundary conditions are all set as no-slip wall conditions for solving momentum equation and for simulating volume fraction equation zero gradient condition applied for the boundary conditions. The physical domain is subdivided into 80×80 control-volumes, and the problem is solved for three different time intervals, with values of 0.0026 s, 0.0052 s

and 0.0078 s. With the maximum theoretical velocity not exceeding (

$$u_{max} = \sqrt{2 \times g \times (2a)} = 2.392 \ m/s \text{),} \qquad \text{these} \qquad \text{time} \qquad \text{steps} \qquad \text{correspond} \qquad \text{to}$$

$$Co_{max} = u_{max} \frac{\Delta t}{\Delta x} = 0.852, \ 1.704, \text{ and } 2.556 \text{ ,} \quad \text{respectively.} \quad \text{In the Navier-Stokes}$$

equations, the convective terms are discretized using the UMIST scheme, while the transient term is discretized via a first-order Euler scheme. In the volume fraction equation, however, the convective terms are discretized using various schemes introduced earlier, while the transient term is discretized by applying the same unsteady scheme that is used for solving transient term in the momentum equation.
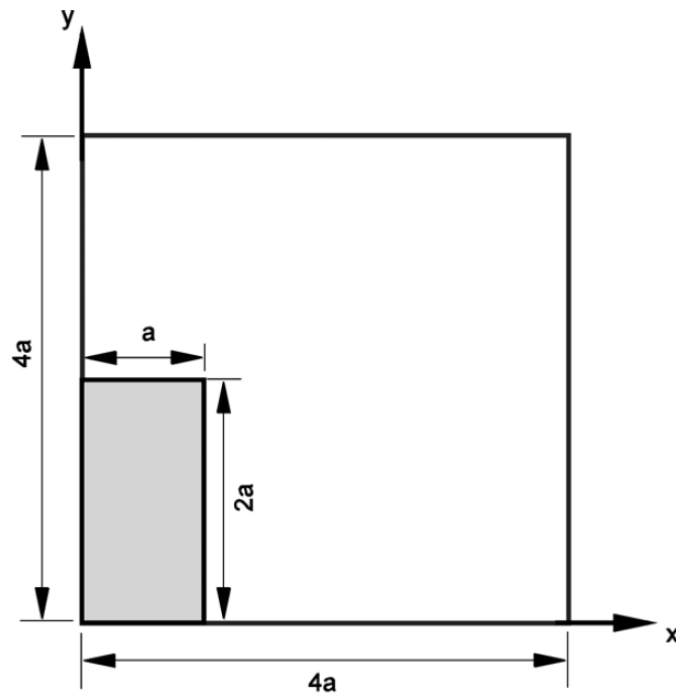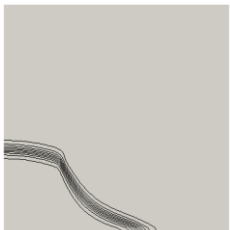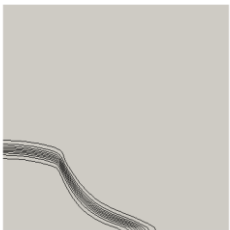
Figure 4.1. A schematic representation of dam-break problem

Contour plots of the α (indicator function) simulated by various convection schemes are illustrated in Figure 4.2, Figure 4.3 and Figure 4.4 in three different time steps.

	$Co_{max} = 0.852$	$Co_{max} = 1.704$	$Co_{max} = 2.556$
Upwind			
Vanleer			
UMIST			
QUICK			
MINMOD			
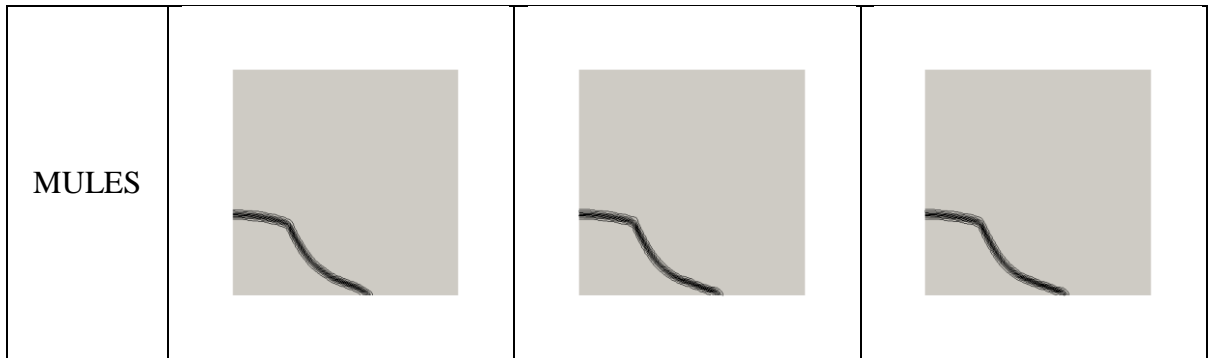
$Co_{max} = 0.852$	$Co_{max} = 1.704$	$Co_{max} = 2.556$

MULES			

Figure 4.2. α-contour plots for the dam-break problem over an 80×80 grid at time $t^*=1.278$.

	$Co_{max} = 0.852$	$Co_{max} = 1.704$	$Co_{max} = 2.556$
Upwind			
Vanleer			
UMIST			

QUICK			
MINMO D			
MULES			

Figure 4.3. α -contour plots for the dam-break problem over an 80×80 grid at time t^{*}=2.54.

	$Co_{max} = 0.852$	$Co_{max} = 1.704$	$Co_{max} = 2.556$
Upwind			

Vanleer			
UMIST			
QUICK			
MINMO D			
MULES			

Figure 4.4. α-contour plots for the dam-break problem over an 80×80 grid at time t^*=6s.

54

As can be seen in Figure 4.2 until Figure 4.4, solutions are generated for the various convection schemes. Results at any time step are assumed to be converged in all the computations when the maximum absolute value of the residual of all variables falls below a small numbers set at 10^{-6}. In Figure 4.2, Figure 4.3 and Figure 4.4, solutions achieved using the Upwind, Van-Leer, UMIST, QUICK and MULES convection schemes at three values of the Co_{max} are shown. Figure 4.2 show interfaces at the dimensionless time $t^* = t\sqrt{g/a} = 1.278$, Figure 4.3 and Figure 4.4 at $t^* = 2.54$ and $t^* = 6$, respectively. By comparing Courant number for $Co_{max} = 0.852$, $Co_{max} = 1.704$ and $Co_{max} = 2.556$ shows little effect on the interface anticipated results by all of the convection schemes. For $Co_{max} = 2.556$ it is possible to see small wiggles and the results become slightly more diffusive when Co_{max} increases, however, the results shows that the interface is almost independent with Courant number for the schemes that have been used. For smaller Co numbers the results are slightly better. The Upwind scheme profiles are greatly diffusive at all Co considered, and it's the most diffusive at all Co_{max} and t^*. Results by MINMOD are much better than UPWIND's but the results are still diffusive. Interfaces predicted by the Van-Leer and UMIST schemes are somewhat sharper than those achieved with the QUICK, but the results are slightly more diffusive. Results for Van-Leer and UMIST are nearly the same, but the prediction of interface by Van-Leer is a bit sharper and less diffusive. Profiles predicted by MULES are much sharper and it's the lowest diffusive results in comparison with the other schemes that are used. As seen in the plots, all schemes are capable of predicting the collapse of the water column on the left wall of the domain $t^* = 1.278$, the rise of water on the opposite

side $t^* = 2.54$, and then its descent $t^* = 6$ but the results by Upwind is so diffusive, it means that it's not logical to use this scheme for predicting interfaces.

4.2 Advection of Hollow Shapes in an Oblique Velocity Field

In this problem in comparison with the previous one, the momentum equation is not solved. In an oblique velocity field presented by $\mathbf{u}[2,1]\, m/s$, the convection of three different hollow shapes are considered.

(a)

(b)

(c)
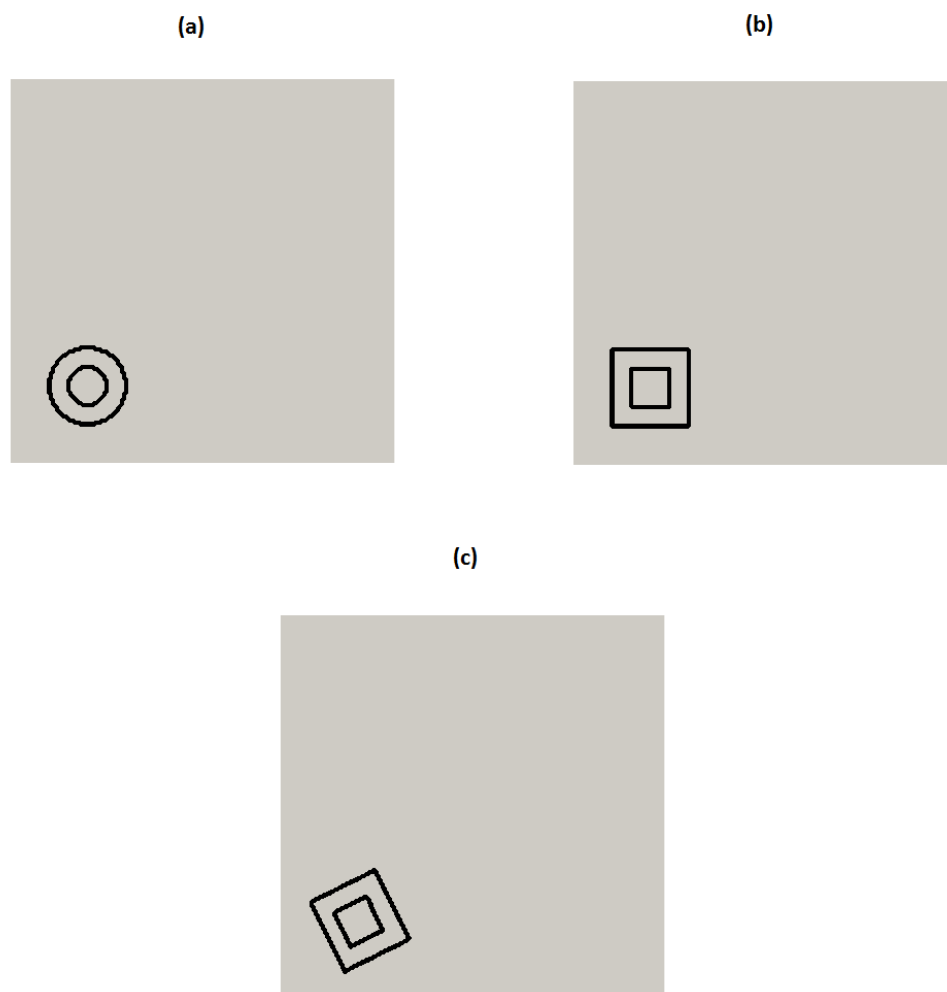
Figure 4.5. schematic of initial condition advection of (a) hollow circle (b) hollow square (c) hollow rotated square
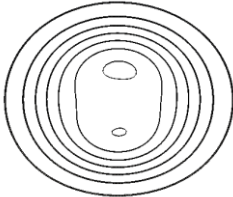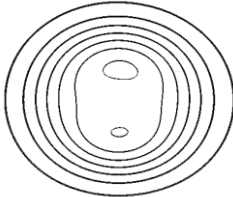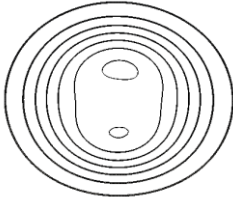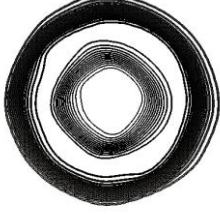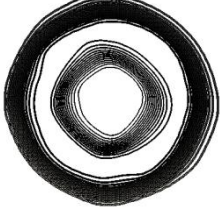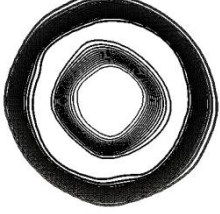
The following three shapes, depicted in Figure 4.6 to Figure 4.11, are taken to the account: A hollow square (Figure 4.7 and Figure 4.10) aligned with the coordinate

axes 0.1m and 0.2 m of an inner side length and outer side length that are divided into 20 and 40 cells for the regular rectangular mesh used, respectively; with respect to the x-axis a hollow square rotated through an angle of 26.57° (Figure 4.8 and Figure 4.11), of same dimensions as shown in above hollow square; A hollow circle (Figure 4.6 and Figure 4.9) with 0.2m and 0.1m as an outer diameter and inner diameter that subdivided 40 and 20 structured cells, respectively.

Initially, all of these shapes are centered at (0.2, 0.2) meter and after 0.3 seconds their exact positions centered at (0.8, 0.5) meter. For discretizing the transient scheme, the second-order Crank-Nicholson scheme is used. Computations are performed over the 200×200 structured grid system for three different time steps which are Δt= 0.0004168 seconds yields Courant number values of 0.25, Δt= 0.0008332 , and 0.0012499 seconds, yielding, Co=0.5, and Co=0.75, respectively. These Co values are indicated by low courant number, medium courant number, and high courant number on the following simulation results. For the different convection schemes and shapes, Contour plot results of the α fields at different courant number values after the lapse of 0.2 s and 0.3 s are displayed in Figure 4.6 to Figure 4.8 and Figure 4.9 to Figure 4.11, respectively.

At all Courant numbers considered, the UPWIND scheme profiles are greatly diffusive. The MINMOD scheme creates better simulation results in comparison with the UPWIND scheme results, but the interfaces by using the MINMOD scheme is not resolved sharply and it could not track the shape of the interface correctly. The convected shapes seem to be little effected by the Courant numbers that are predicted by all of the convection schemes. By using MULES the best profiles are achieved and are almost independent of Co and preserving the sharpness of the interfaces. By

the comparison of the contours it is possible to see that the results by QUICK have the lowest diffusivity except MULES but this scheme cannot follow the interface sharply. The simulation results of UMIST are close to results of Van-Leer scheme where the predicted interface by Van-Leer is more accurate and less diffusive.

	Low Co	Medium Co	High Co
Upwind			
Vanleer			
UMIST			
QUICK			
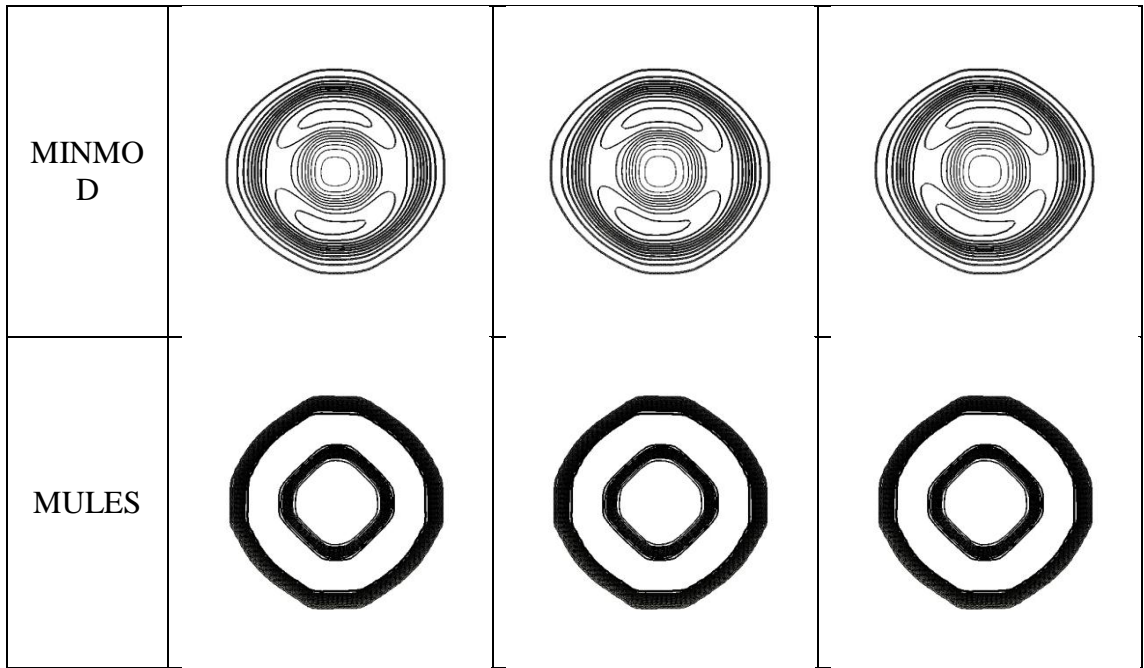
Figure 4.6. advection contour plots of α (indicator function) over 200×200 grid after 0.3 seconds of a hollow circle.

	Low Co	Medium Co	High Co
Upwind			
Vanleer			
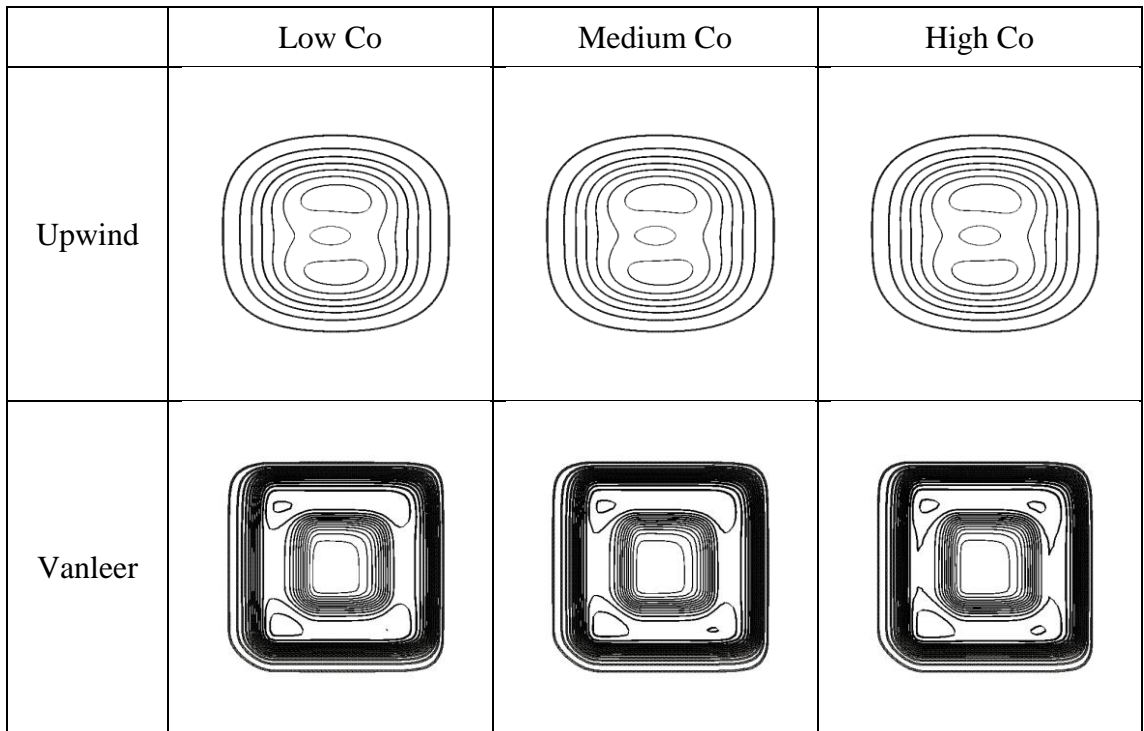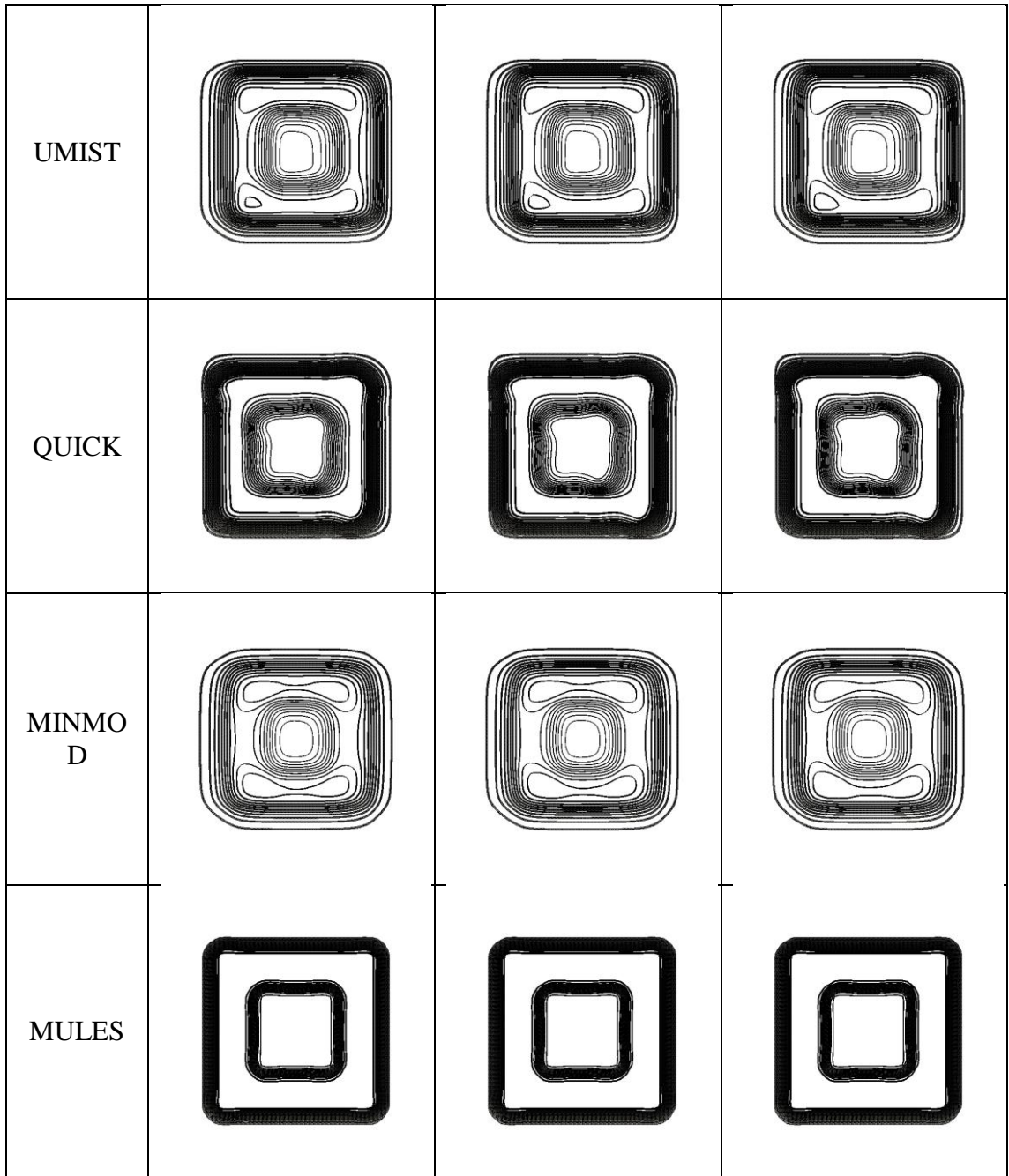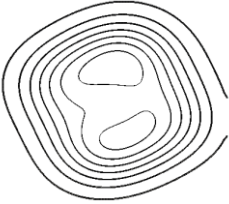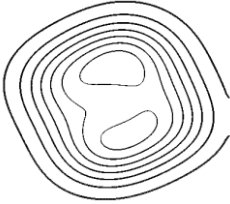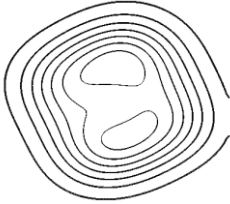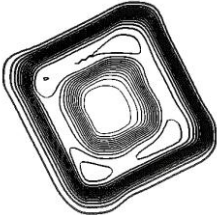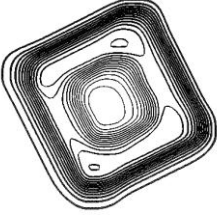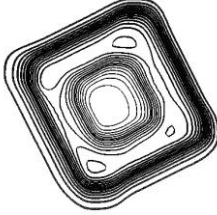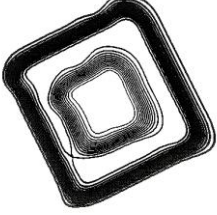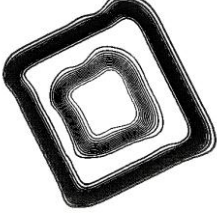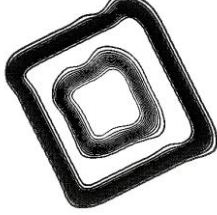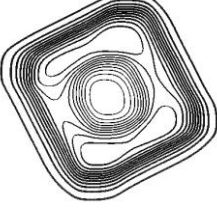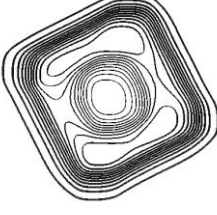
Figure 4.7. advection contour plots of α (indicator function) over 200×200 grid after 0.3 seconds of a hollow square.

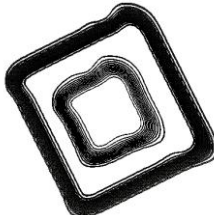|	Low Co	Medium Co	High Co
Upwind			
Vanleer			
UMIST			
QUICK			
MINMO D			

	MULES		

Figure 4.8. advection contour plots of α (indicator function) over 200×200 grid after 0.3 seconds of a rotated hollow square.

	Low Co	Medium Co	High Co
Upwind			
Vanleer			
UMIST			
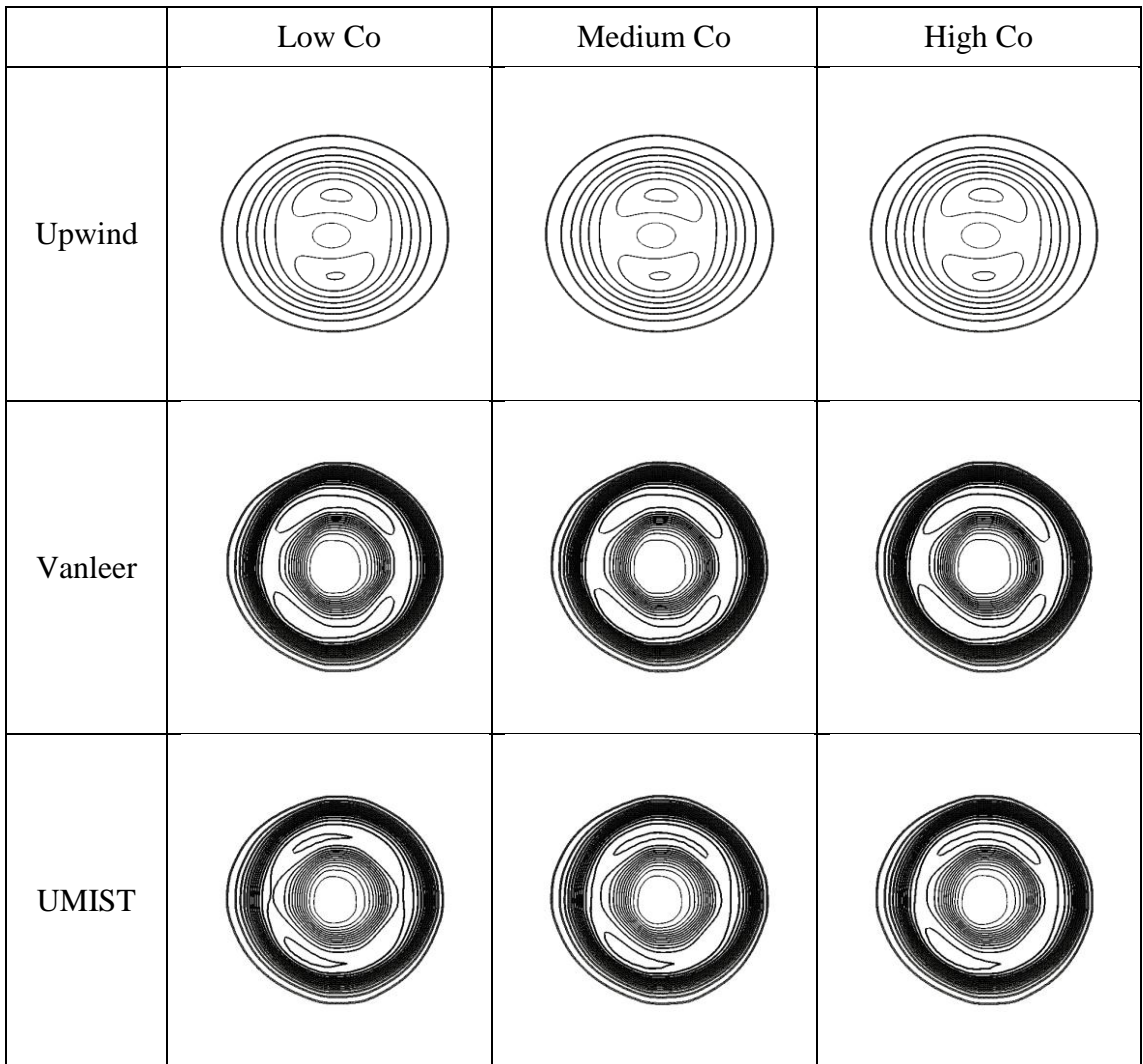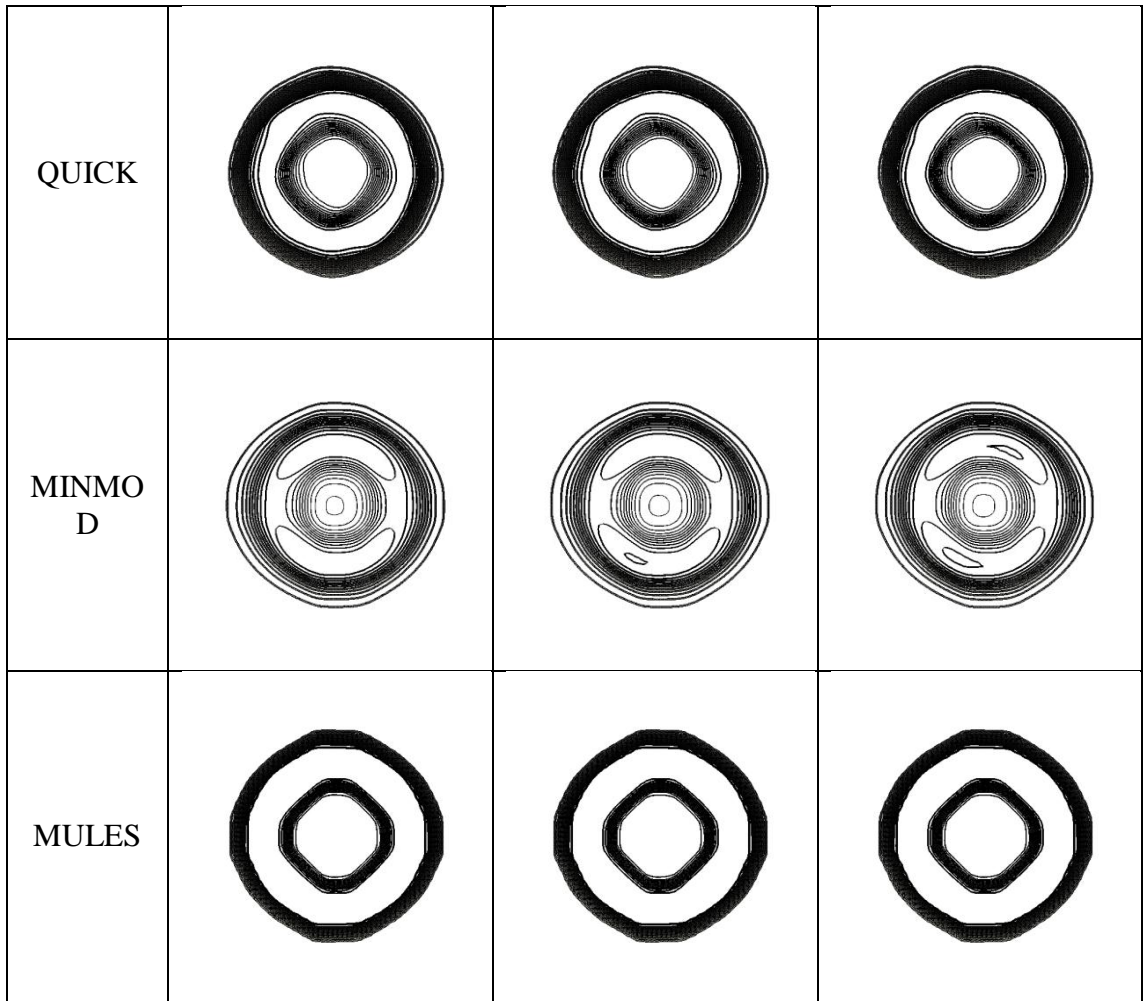
Figure 4.9. advection contour plots of α (indicator function) over 200×200 grid after 0.2 seconds of a rotated hollow circle.

Vanleer			
UMIST			
QUICK			
MINMO D			
MULES			
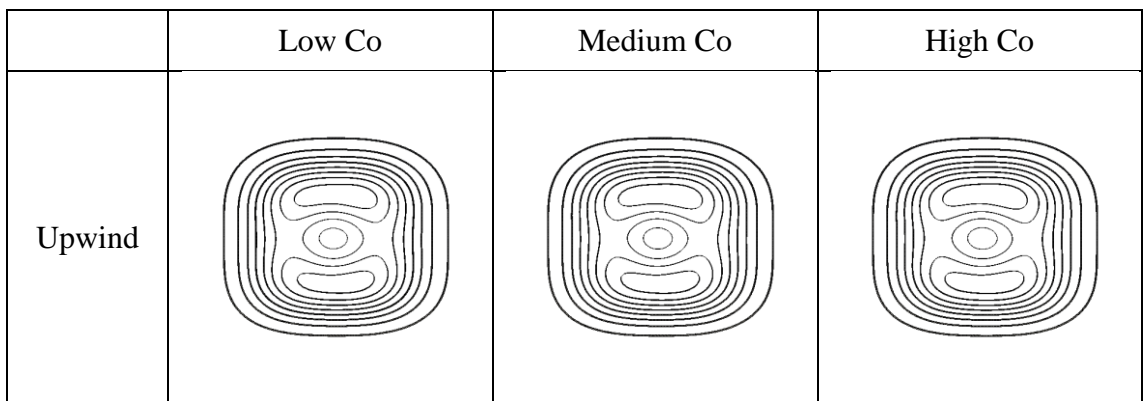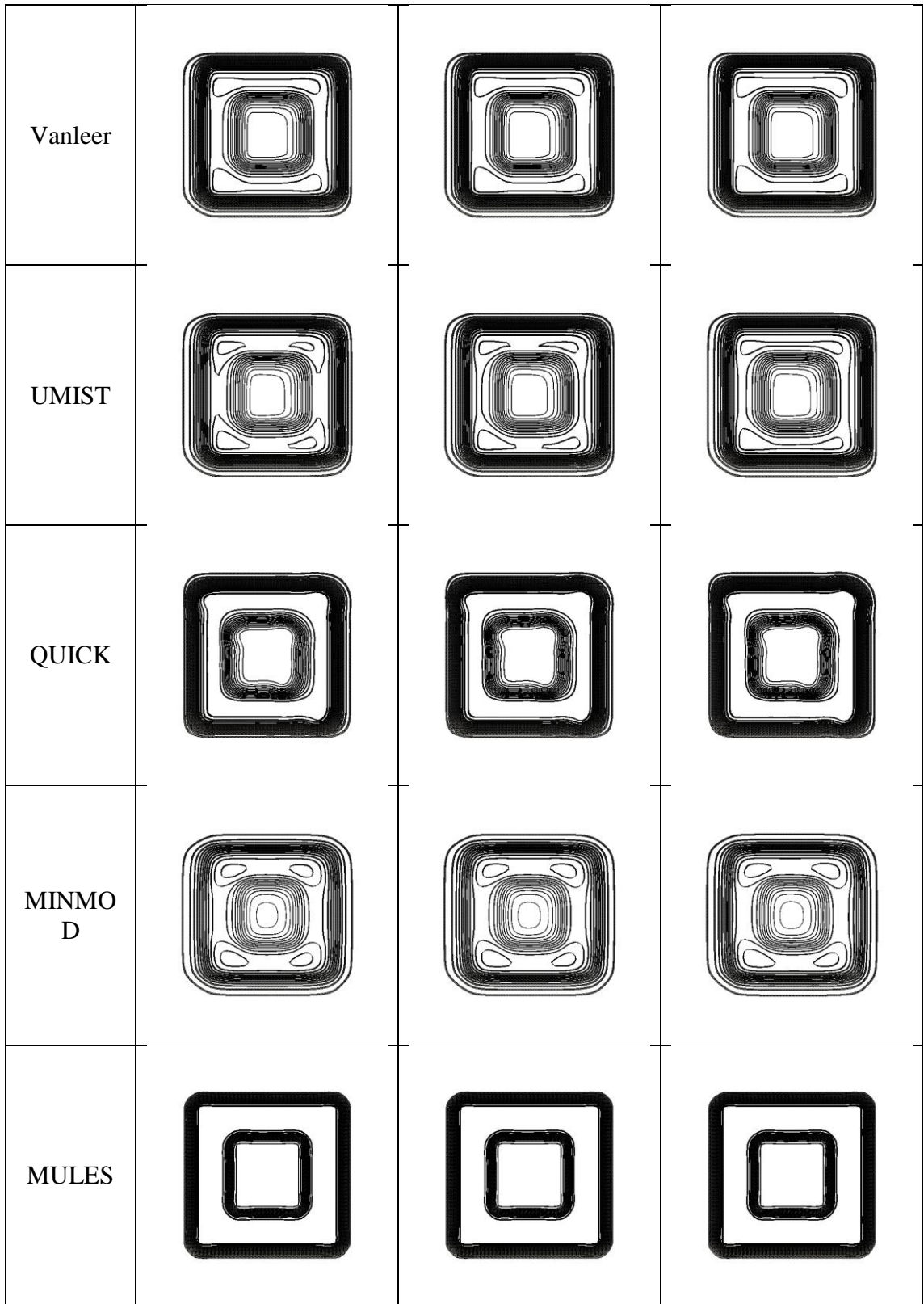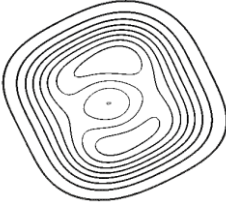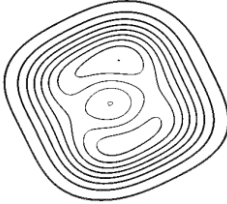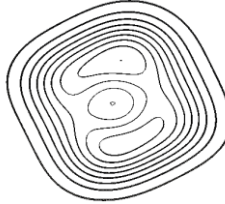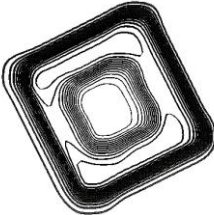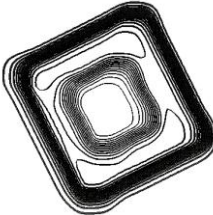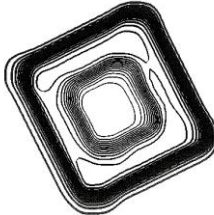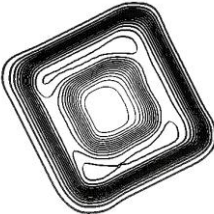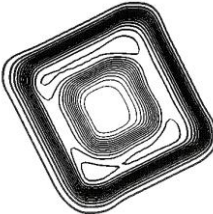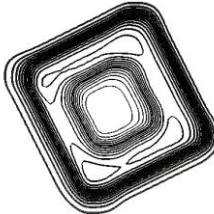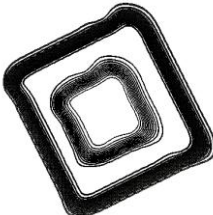
Figure 4.10. advection contour plots of α (indicator function) over 200×200 grid after 0.2 seconds of a hollow square.

64

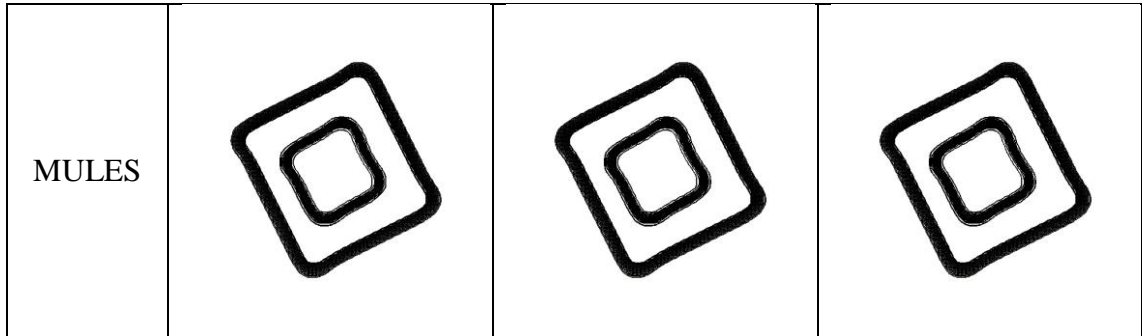|	Low Co	Medium Co	High Co
Upwind			
Vanleer			
UMIST			
QUICK			
MINMO D			

Figure 4.11. advection contour plots of α (indicator function) over 200×200 grid after 0.2 seconds of a rotated hollow square.

Profiles for the hollow square and rotated hollow square problems (Figure 4.7, Figure 4.10, Figure 4.8 and Figure 4.11) anticipated by the QUICK scheme have wiggles along interfaces that is the square lines do not remain straight, but Van-Leer and UMIST predicted the interfaces sharper. By comparing α-contours achieved with the various schemes (Figure 4.6 to Figure 4.11), the performance of MULES is preferable to all other schemes, considered in this work and Results demonstrated that interfaces obtained with MULES are sharper and bounded.

4.3 Free bubble rise

In a liquid under the influence of buoyancy, free rising of a gas bubble is considered. For solving the problem by using different schemes to solve the convection term of volume fraction equation, the single bubble rising in a motionless liquid is simulated same as done in the previous problem, but the surface tension affect is applied in this model. The analysis is done for two series of physical properties as shown in Table 4.1 and for two different equivalent diameters; 3mm and 5mm.

Table 4.1. physical properties for simulation free rise bubble problem

Series	μ_l (Pa s)	ρ_l (kg/m^3)	σ (N/m)	Mo	$\mu^*(\mu_b/\mu_l)$	$\rho^*(\rho_b/\rho_l)$
S3	0.242	1230	0.063	0.11	7.61×10^{-5}	9.57×10^{-4}
S5	0.0733	1205	0.064	9.0×10^{-4}	2.51×10^{-4}	9.89×10^{-4}

The size of the numerical domain is (0.04 m, 0.075 m, 0.04 m) and (0.04 m, 0.075 m, 0.04 m) that correlates with ($8D_{eq}$, $15D_{eq}$, $8D_{eq}$) (D_{eq} is the equivalent diameter of bubble) for diameter of bubble 5 mm and 3mm, respectively. Hua and Lou (2007) have demonstrated that no liquid influence from the boundary walls on the bubble motion for the same geometry and physical properties.

It has been found that the bubble can deform to different shapes depending on three dimensionless numbers: the Morton number (Mo), the Eötvös number (Eo) (also called as the Bond number) and the Reynolds number (Re) (Grace(1973) and Clift et al. (1978)), where these numbers are defined as

$$Mo = \frac{g\mu_l^4\Delta\rho}{\rho_l^2\sigma^3} \tag{4.1}$$

$$Eo = \frac{g\Delta\rho D_{eq}^2}{\sigma} \tag{4.2}$$

$$Re = \frac{\rho_l V_\infty D_{eq}}{\mu_l} \tag{4.3}$$

where g is gravity acceleration, μ_l and ρ_l denotes viscosity and density of liquid around the bubble respectively and D_{eq} represents the equivalent diameter of bubble.

The solution domain size in the radial direction should be large enough so that the boundary effects on bubble rising can be ignored in the simulation, and the bubble rising can be reasonably assumed as in an infinite quiescent liquid. To discretize the fluid domain, a structured mesh is used. In this mesh system along the diameter of bubble, 12 cells are distributed. The convergence of the simulation results is guaranteed by this size of mesh and it is sufficient as well. (Van Sint et al. (2005))

used 12 cells per diameter so a 96×180 rectangular grid is used in this work. The boundary conditions are all set as no-slip wall conditions.

For solving free bubble rise problem, the same schemes for discretizing the convection and transient terms for momentum and volume fraction equations are used as dam-break problem. Alpha-contours are shown in Figure 4.12 to Figure 4.15 at three different times.The non-dimensional Reynolds number and Eötvös number are given in Table 4.2.

Table 4.2. The simulation parameters for free bubble rise problem in different fluids

Bubble diameter (m)	Re^* for S3	Eo^* for S3	Re^* for S5	Eo^* for S5
0.003	2.610	1.722	8.445	1.660
0.005	5.616	4.783	18.172	4.612

where the non-dimensional Reynolds number and Eötvös number are defined by Equations (4.4) and (4.5), respectively.

$$\mathrm{Re}^* = \frac{\rho_l g^{1/2} D_{eq}^{3/2}}{\mu_l} \tag{4.4}$$
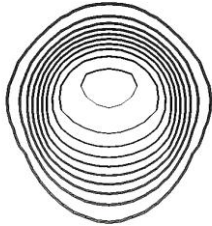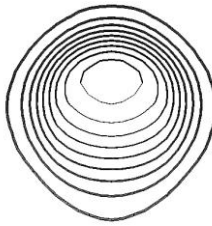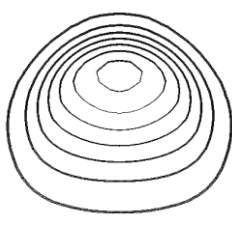
$$Eo^* = \frac{g \rho_l D_{eq}^2}{\sigma} \tag{4.5}$$

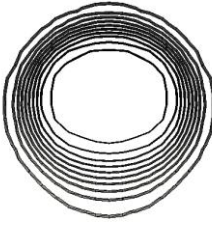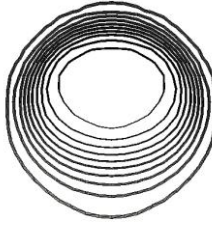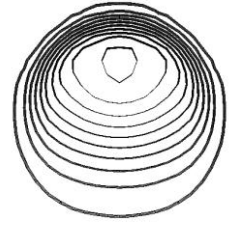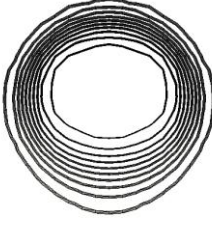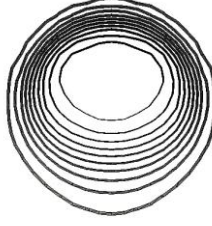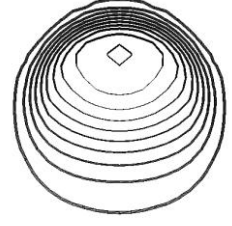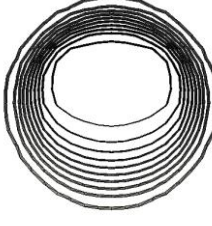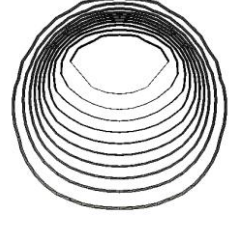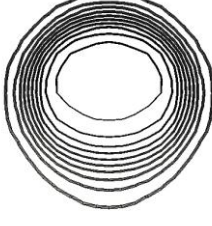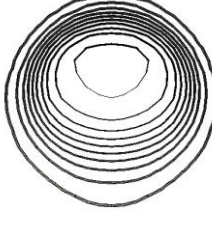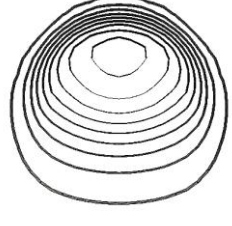The terminal bubble shape for the different bubble diameters with the S3 fluid is illustrated in Figure 4.12 and Figure 4.14 and with the S5 fluid is displayed in Figure 4.13 and Figure 4.15 where the initial spherical bubble is shown to deform from spherical to ellipsoidal shape as the initial diameter increases.

The shape of bubbles, with intermediate Reynolds and Bond numbers ($1 < Re < 100$ and $1 < Eo < 100$), are affected significantly by the flow conditions. Various bubble

shapes (oblate ellipsoid, disk-like, oblate ellipsoidal cap, skirt bubble, and spherical-cap) have been found in various flow regimes by the experimental investigations. In spite of the difference in shapes, the bubbles rise steadily in the liquid along a straight path (Hua and Lou (2006)).

By comparing results for same initial bubble diameter (Figure 4.12 and Figure 4.14 for 3mm D_{eq}; Figure 4.13 and Figure 4.15 for 5mm D_{eq}) shows that the shape of bubble is affected by changing the density ratio and viscosity ratio, bubble shape distort more by increasing these ratios.

The UPWIND scheme profiles are greatly diffusive at all of the time intervals and it gets worse throughout the time and results show that this scheme cannot follow the bubble shape properly specially for the higher initial bubble diameter. Interfaces predicted by MINMOD are less diffusive and sharper than UPWIND but the results by this scheme are still diffusive. QUICK scheme could not track the shape of the interface correctly in comparison with Van-Leer and UMIST results especially it gets worse by increasing the bubble diameter (Figure 4.13 and Figure 4.15) but the result by Quick is less diffusive. Results for Van-Leer and UMIST are nearly the same, but the prediction of interface by Van-Leer is a bit less diffusive. Results demonstrated that interfaces obtained with MULES are sharper and MULES generated the lowest results in comparison with the other schemes that are used in this work.

	t=0.1s	t=0.15s	t=0.3s
Upwind			
Vanleer			
UMIST			
QUICK			
MINMO D			

Figure 4.12. α-contour plots for the free bubble rise problem over an 96×180 grid with S3 physical properties for bubble diameter 3 mm in three different time steps 0.1s, 0.15s and 0.3s respectively.

	t=0.1s	t=0.15s	t=0.3s
Upwind			
Vanleer			
UMIST			

QUICK			
MINMO D			
MULES			

Figure 4.13. α-contour plots for the free bubble rise problem over an 96×180 grid with S3 physical properties for bubble diameter 5 mm in three different time steps 0.1s, 0.15s and 0.3s respectively.

	t=0.1s	t=0.15s	t=0.3s
Upwind			

Figure 4.14. α-contour plots for the free bubble rise problem over an 96×180 grid with S5 physical properties for bubble diameter 3 mm in three different time steps 0.1s, 0.15s and 0.3s respectively.

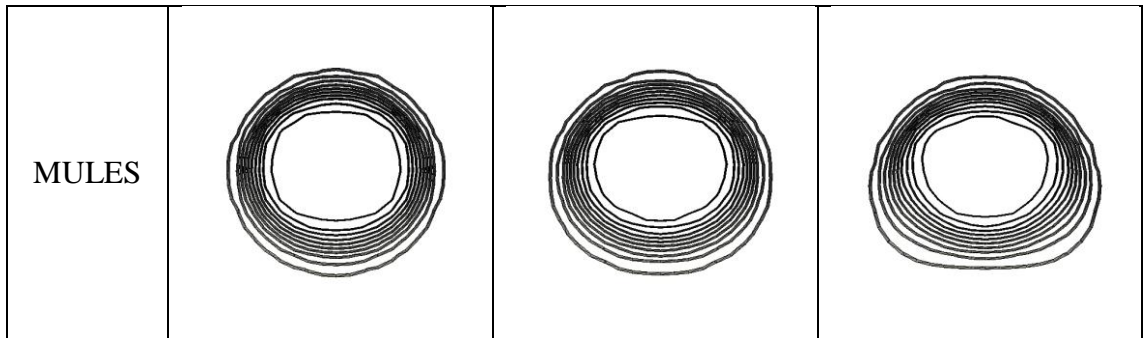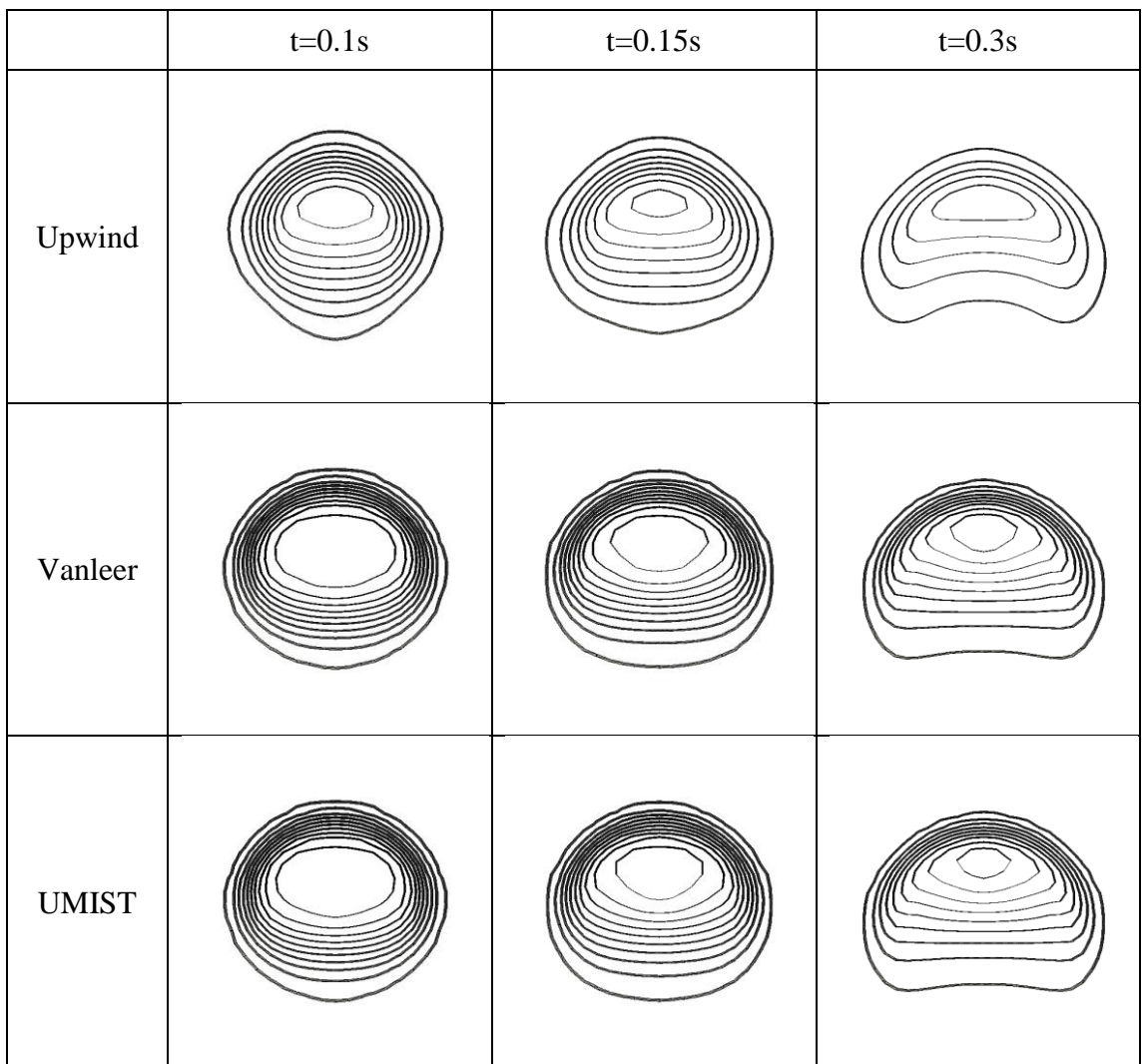|	t=0.1s	t=0.15s	t=0.3s
Upwind			
Vanleer			
UMIST			
QUICK			
MINMOD			

Figure 4.15. α-contour plots for the free bubble rise problem over an 96×180 grid with S5 physical properties for bubble diameter 5 mm in three different time steps 0.1s, 0.15s and 0.3s respectively.
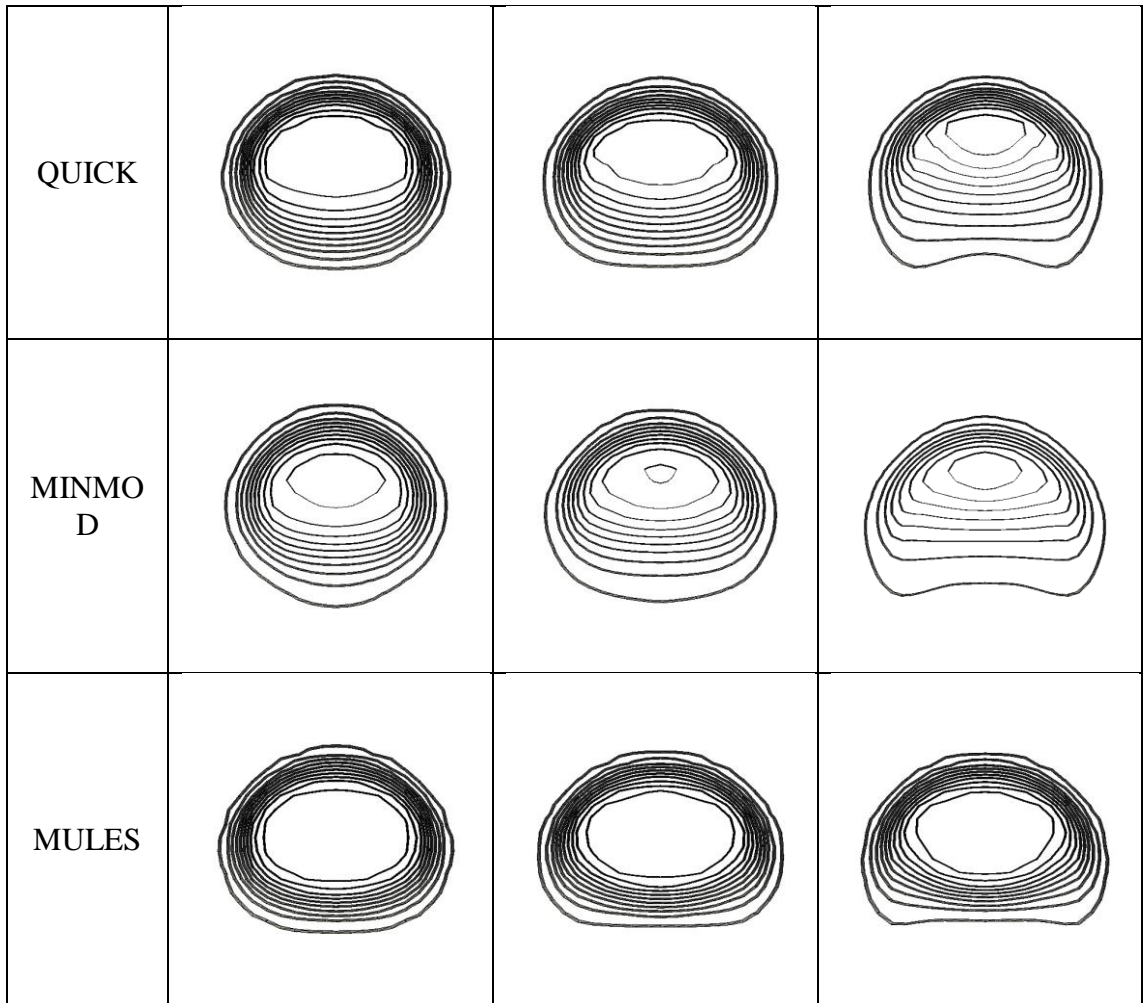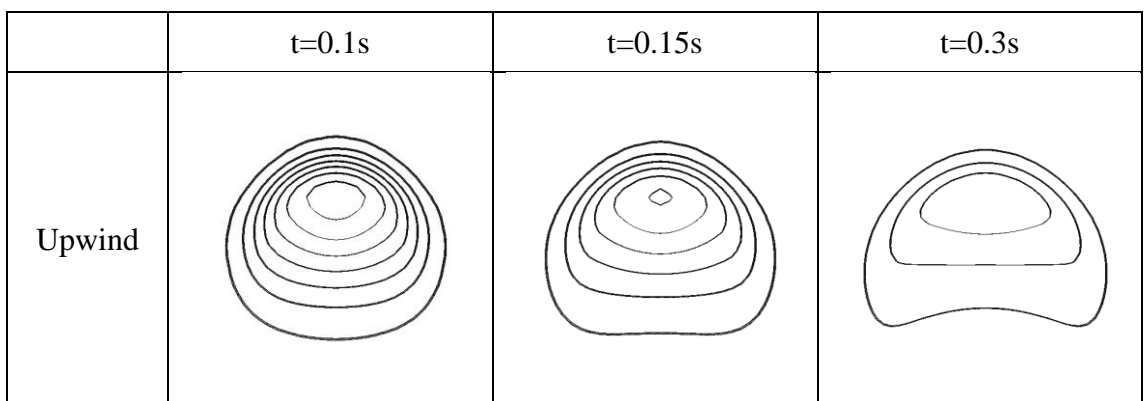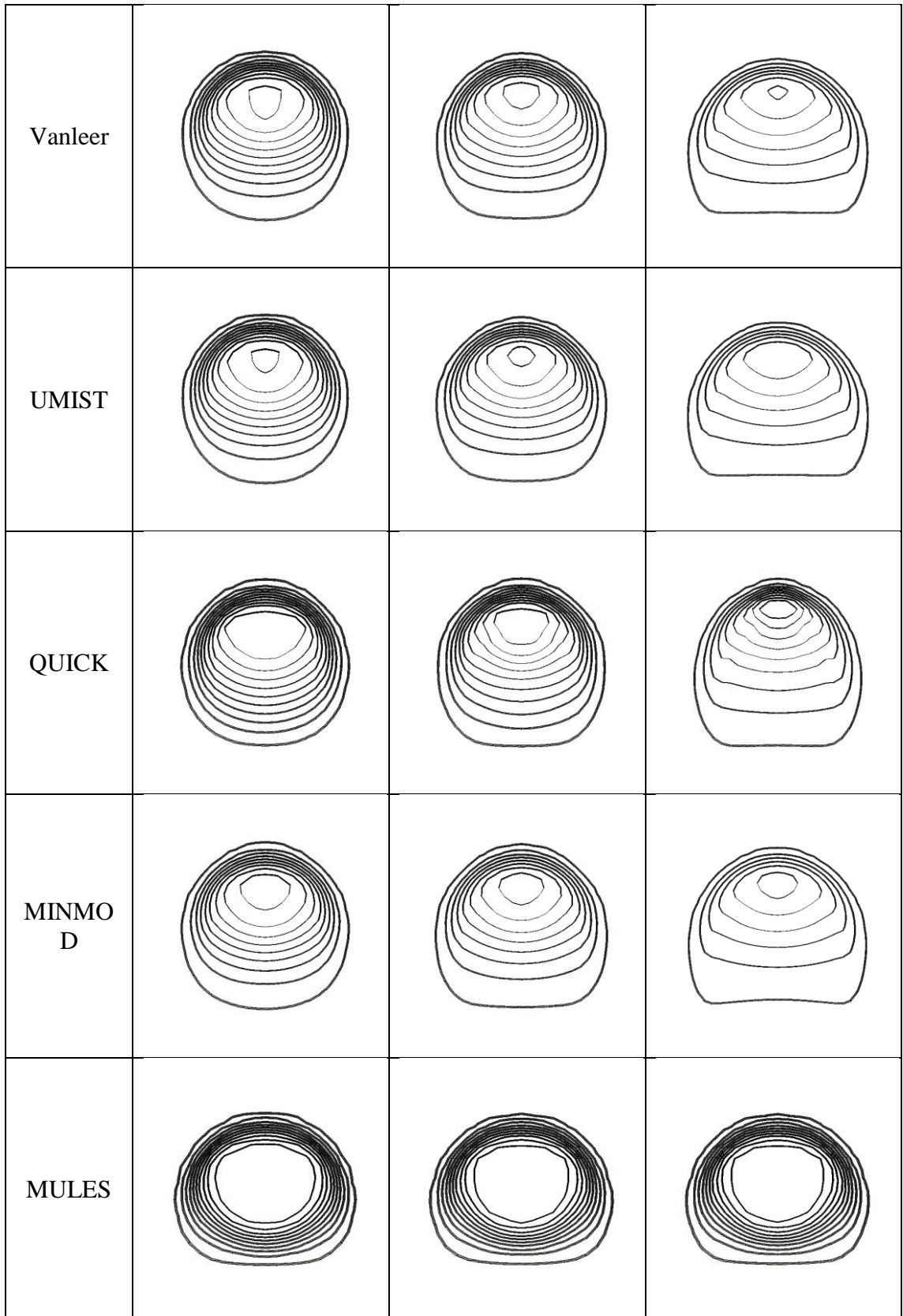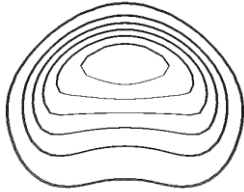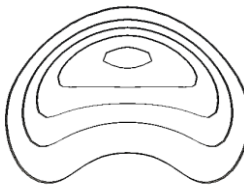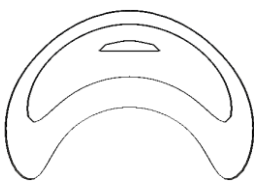
Chapter 5

CONCLUSION

Discretization of the convection term is a controversial issue in the two phase flow simulations. This research work started with the study of effect of different convection schemes on solving volume fraction equation to track the interfaces in free surface flows. Three well-known test cases (dam-break, free bubble rise and advection of hollow shapes in an oblique velocity field) have been chosen in order to illustrate the differences in various schemes applications.

The OpenFOAM is a free open-source CFD toolbox that is largely used in multiphase flows. OpenFOAM includes an expanding collection of solvers that can be applied to a wide range of simulations and problems. The interFoam is one of these solvers, which employs the VOF method to predict the free-surface that is used in this work. The VOF method was found to be more accurate in capturing the sharp interface between two fluids than other methods available in literature. The simulation in this research has been carried out for 2D problems.

To see the performance of different convection schemes, the UPWIND scheme as a first order accurate scheme, QUICK as a higher order accurate scheme, MINMOD, Van-Leer and UMIST as a flux limiter function in newly TVD scheme and MULES as a flux limiter are chosen.

As given results in previous chapter (Figure 4.2 to Figure 4.4 for dam-break problem, Figure 4.6 to Figure 4.11 for advection of hollow shapes problem and Figure 4.12 to Figure 4.15 for free bubble rise problem), the results by UPWIND scheme profiles are greatly diffusive due to the fact that it's a first order scheme, this scheme is the most diffusive scheme in comparison to the other schemes that are used, also it is not possible to track interface accurately by using UPWIND scheme. The MINMOD scheme creates better simulation results in comparison with the UPWIND but the results by this scheme are still diffusive. The predicted interfaces by using the QUICK scheme is not resolved sharply but in comparison with results of Van-Leer and UMIST schemes is less diffusive. The simulation results of UMIST are close to results of Van-Leer scheme where the predicted interface by Van-Leer is more accurate and less diffusive. The most accurate and lowest diffusive results are given by MULES.

New methods such as CICSAM or STACS based on the blending strategy are developed for solving the diffusivity problem. These methods can be applied and developed in future works by changing the blending schemes and applying new blending factor to achieve more accurate, sharp and less diffusive results.

REFERENCES

Albadawi, A., Donoghue, D. B., Robinson, A. J., Murray, D. B., & Delauré, Y. M. C. (2013). Influence of surface tension implementation in Volume of Fluid and coupled Volume of Fluid with Level Set methods for bubble growth and detachment. *International Journal of Multiphase Flow*, *53*, 11-28.

Aris, R., & Vectors, T. the Basic Equations of Fluid Mechanics, 1989.

Ashgriz, N., & Poo, J. Y. (1991). FLAIR: Flux line-segment model for advection and interface construction. *Journal of Computational Physics*, *93*, 449–468.

Berberović, E., van Hinsberg, N. P., Jakirlić, S., Roisman, I. V., & Tropea, C. (2009). Drop impact onto a liquid layer of finite thickness: Dynamics of the cavity evolution. *Physical Review E*, *79*(3), 036306.

Bhaga, D., & Weber, M. E. (1981). Bubbles in viscous liquids: shapes, wakes and velocities. *Journal of Fluid Mechanics*, *105*, 61-85.

Bolton, B., & Middleman, S. (1980). Air entrainment in a roll coating system.*Chemical Engineering Science*, *35*(3), 597-601.

Boris, J. P., & Book, D. L. (1973). Flux-corrected transport. I. SHASTA, A fluid transport algorithm that works. *Journal of computational physics*, *11*(1), 38-69.

Brackbill, J. U., Kothe, D. B., & Zemach, C. (1992). A continuum method for modeling surface tension. *Journal of computational physics*, *100*(2), 335-354.

Cerne, G., Petelin, S., & Tiselj, I. (2001). Coupling of the interface tracking and the two-fluid models for the simulation of incompressible two-phase flow.*Journal of computational physics*, *171*(2), 776-804.

Chakravarthy, S.R., & Osher, S. (1983). High resolution application of the OSHER upwind scheme for the Euler equation, *AIAA Paper* 83-1943.

Chaves, H., Obermeier, F., Seidel, T., & Weise, V. (2000). Fundamental investigation of disintegration of a sinusoidally forced liquid jet. *8th Int. Conf. on Liquid Atomization and Spray Systems (Pasadena, USA),* 1018–25.

Clift, R., Grace, J. R., & Weber, M. E. Bubbles, Drops and Particles, Academic Press, New York, (1978). *Vol. 5. Nos. 1-4, 1988 Modelling of Three Phase Sparged Catalytic Reactors*.

Courant, R., Isaacson, E., & Rees, M. (1952). On the solution of nonlinear hyperbolic differential equations by finite differences. *Communications on Pure and Applied Mathematics*, *5*(3), 243-255.

Darwish, M., & Moukalled, F. (2006). Convective schemes for capturing interfaces of free-surface flows on unstructured grids.*Numerical Heat Transfer, Part B: Fundamentals*, *49*(1), 19-42.

Davies, R.M., Taylor, F.I. (1950). The mechanism of large bubbles rising through extended liquids and through liquids in tubes. *Proc. Royal Soc. Lond,* A 200, 375–390.

DeBar, R.B. (1974). Fundamentals of the KRAKEN code. [Eulerian hydrodynamics code for compressible nonviscous flow of several fluids in two-dimensional (axially symmetric) region], *Technical Report UCID-17366*

Deshpande, S. S., Trujillo, M. F., Wu, X., & Chahine, G. (2012). Computational and experimental characterization of a liquid jet plunging into a quiescent pool at shallow inclination. *International Journal of Heat and Fluid Flow*, *34*, 1-14.

Enright, D., Fedkiw, R., Ferziger, J., & Mitchell, I. (2002). A hybrid particle level set method for improved interface capturing. *Journal of Computational Physics*,*183*(1), 83-116.

Ferziger, J. H., & Perić, M. (2002). *Computational methods for fluid dynamics*(Vol. 3). Berlin: Springer.

Gentry, R. A., Martin, R. E., & Daly, B. J. (1966). An Eulerian differencing method for unsteady compressible flow problems. *Journal of Computational Physics*, *1*(1), 87-118.

Gopala, V. R., Lycklama à Nijeholt, J. A., Bakker, P., & Haverkate, B. (2011). Development and validation of a CFD model predicting the backfill process of a nuclear waste gallery. *Nuclear Engineering and Design*, *241*(7), 2508-2518.

Grace, J. R. (1973). Shapes and velocities of bubbles rising in infinite liquids.*Trans. Inst. Chem. Eng.*, *51*(2), 116-120.

Harlow, F. H., & Welch, J. E. (1965). Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Physics of fluids*, *8*(12), 2182.

Harten, A. (1983). High resolution schemes for hyperbolic conservation laws.*Journal of computational physics*, *49*(3), 357-393.

Harten, A. (1984). On a class of high resolution total-variation-stable finite-difference schemes. *SIAM Journal on Numerical Analysis*, *21*(1), 1-23.

Hartunian, R. A., & Sears, W. R. (1957). On the instability of small gas bubbles moving uniformly in various liquids. *Journal of Fluid Mechanics*, *3*(01), 27-47.

Hewitt, G., & Vassilicos, C. (Eds.). (2005). *Prediction of turbulent flows*. Cambridge University Press.

Hirsch, C. (1988). *Numerical Computation of Internal and External Flows Vol.1 - Fundamentals of Numerical Discretization*, JohnWiley & Sons, Inc.

Hirsch, C. (1991). *Numerical Computation of Internal and External Flows Vol.2 - Computational Methods for Inviscid and Viscous Flows*, JohnWiley & Sons, Inc.

Hirt, C. W., & Nichols, B. D. (1981). Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of computational physics*, *39*(1), 201-225.

Hu, H. H., & Joseph, D. D. (1994). Evolution of a liquid drop in a spinning drop tensiometer. *Journal of colloid and interface science*, *162*(2), 331-339.

Hua, J., & Lou, J. (2007). Numerical simulation of bubble rising in viscous liquid.*Journal of Computational Physics*, *222*(2), 769-795.

Hyman, J. M. (1984). Numerical methods for tracking interfaces. *Physica D: Nonlinear Phenomena*, *12*(1), 396-407.

Hyman, J. M., & Naughton, M. (1983). Adaptive static rezoning methods. In*proceedings AMS-SIAM Conference on Large Scale computations in Fluid Mechanics*.

Ishimoto, J., Sato, F., & Sato, G. (2010). Computational prediction of the effect of microcavitation on an atomization mechanism in a gasoline injector nozzle.*Journal of Engineering for Gas Turbines and Power*, *132*(8), 082801.

Jasak, H. (1996). *Error analysis and estimation for the finite volume method with applications to fluid flows* (Doctoral dissertation, Imperial College London (University of London)).

Koh, C. G., Gao, M., & Luo, C. (2012). A new particle method for simulation of incompressible free surface flow problems. *International Journal for Numerical Methods in Engineering*, *89*(12), 1582-1604.

Koshizuka, S. (1995). A particle method for incompressible viscous flow with fluid fragmentation. *Comput. Fluid Dynamics J.*, *4*, 29-46.

Larese, A., Rossi, R., Oñate, E., & Idelsohn, S. R. (2008). Validation of the particle finite element method (PFEM) for simulation of free surface flows.*Engineering Computations*, *25*(4), 385-425.

Lax, P. D. (1954). Weak solutions of nonlinear hyperbolic equations and their numerical computation. *Communications on pure and applied mathematics*,*7*(1), 159-193.

Leonard, B. P. (1979). A stable and accurate convective modelling procedure based on quadratic upstream interpolation. *Computer methods in applied mechanics and engineering*, *19*(1), 59-98.

Leonard, B. P. (1991). The ULTIMATE conservative difference scheme applied to unsteady one-dimensional advection. *Computer methods in applied mechanics and engineering*, *88*(1), 17-74.

Liu, X., & García, M. H. (2008). Three-dimensional numerical model with free water surface and mesh deformation for local sediment scour. *Journal of waterway, port, coastal, and ocean engineering*, *134*(4), 203-217.

Lobosco, R. J., Schulz, H. E., & Simões, A. L. A. Analysis of Two Phase Flows on Stepped Spillways. *Hydrodynamics - Optimizing Methods and Tools*

Maiwald, A., & Schwarze, R. (2011). Numerical analysis of flow-induced gas entrainment in roll coating. *Applied Mathematical Modelling*, *35*(7), 3516-3526.

Mammoli, A. A., & Brebbia, C. A. (Eds.). (2005) *Computational Methods in Multiphase Flow III* (Vol. 50), Wit Press.

Moore, D. W. (1959). The rise of a gas bubble in a viscous liquid. *Journal of Fluid Mechanics*, *6*(01), 113-130.

Moukalled, F., & Darwish, M. (2012). Transient Schemes for Capturing Interfaces of Free-Surface Flows. *Numerical Heat Transfer, Part B: Fundamentals*, *61*(3), 171-203.

Muzaferija, S., & Peri´c, M. (1997). Computation of free-surface flows using the finite-volume method and moving grids. *Numerical Heat Transfer*, *32*(4), 369-384.

Noh, W. F., & Woodward, P. (1976). January SLIC (simple line interface calculation). *Proceedings of the Fifth International Conference on Numerical Methods in Fluid Dynamics June 28–July 2, 1976 Twente University, Enschede* (pp. 330-340), Springer Berlin Heidelberg.

OpenFOAM, (2011). OpenFOAM v2.1.0: Multiphase Modelling [online], Available from: http://www.openfoam.org/version2.1.0/multiphase.php [Accessed 3 Sep 2014].

Osher, S., & Chakravarthy, S. (1984). High resolution schemes and the entropy condition. *SIAM Journal on Numerical Analysis*, *21*(5), 955-984.

Pantakar, S. V., (1980) *Numerical heat transfer and fluid flow: Computational methods in mechanics and thermal science*, Taylor & Francis

Paterson, E. (2008, July). Multiphase and Free-Surface Flow Simulations. *3rd OpenFOAM Workshop*.

Peric, M. (1985). *A finite volume method for the prediction of three-dimensional fluid flow in complex ducts* (Doctoral dissertation, Imperial College London (University of London)).

Prosperetti, A., & Tryggvason, G. (Eds.). (2009). *Computational methods for multiphase flow*. Cambridge University Press.

Raach, H., & Somasundaram, S. (2008). Numerical investigations on heat transfer in falling films around turbulence wires. In *Proc. Int. Conf.'Eurotherm 2008'(Eindhoven, the Netherlands)*.

Raach, H., Somasundaram, S., & Mitrovic, J. (2011). Optimisation of turbulence wire spacing in falling films performed with OpenFOAM. *Desalination*, *267*(1), 118-119.

Raithby, G. D. (1976). Skew upstream differencing schemes for problems involving fluid flow. *Computer Methods in applied mechanics and engineering*, *9*(2), 153-164.

Ramaswamy, B., & Kawahara, M. (1987). Lagrangian finite element analysis applied to viscous free surface fluid flow. *International Journal for Numerical Methods in Fluids*, *7*(9), 953-984.

Raymond, F., & Rosant, J. M. (2000). A numerical and experimental study of the terminal velocity and shape of bubbles in viscous liquids. *Chemical Engineering Science*, *55*(5), 943-955.

Roe, P. L. (1985). Large scale computations in fluid mechanics. *Lectures in applied maths*, *22*, 163-193. Springer Verlag.

Roisman, I. V., Weickgenannt, C. M., Lembach, A. N., & Tropea, C. (2010, September). Drop impact close to a pore: experimental and numerical investigations. In *ILASS—Europe, 23rd annual conference on liquid atomization and spray systems, Brno, Czech Republic*.

Rusche, H. (2002). *Computational Fluid Dynamics of Dispersed Two-Phase Flows at High Phase Fractions* (Doctoral dissertation, University of London).

Saha, A. A., & Mitra, S. K. (2009). Effect of dynamic contact angle in a volume of fluid (VOF) model for a microfluidic capillary flow. *Journal of colloid and interface science*, *339*(2), 461-480.

Saha, A. A., Mitra, S. K., Tweedie, M., Roy, S., & McLaughlin, J. (2009). Experimental and numerical investigation of capillary flow in SU8 and PDMS microchannels with integrated pillars. *Microfluidics and nanofluidics*, *7*(4), 451-465.

Scardovelli, R., & Zaleski, S. (1999). Direct numerical simulation of free-surface and interfacial flow. *Annual Review of Fluid Mechanics*, *31*(1), 567-603.

Soriano, G., Alvarado, J. L., & Lin, Y. P. (2010, January). Experimental characterization of single and multiple droplet impingement on surfaces subject to constant heat flux conditions. In *2010 14th International Heat Transfer Conference* (pp. 707-715). American Society of Mechanical Engineers.

Spalding, D. B. (1972). A novel finite difference formulation for differential expressions involving both first and second derivatives. *International Journal for Numerical Methods in Engineering*, *4*(4), 551-559.

Srinivasan, V., Salazar, A. J., & Saito, K. (2011). Modeling the disintegration of modulated liquid jets using volume-of-fluid (VOF) methodology. *Applied Mathematical Modelling*, *35*(8), 3710-3730.

Sussman, M., & Puckett, E. G. (2000). A coupled level set and volume-of-fluid method for computing 3D and axisymmetric incompressible two-phase flows.*Journal of Computational Physics*, *162*(2), 301-337.

Sussman, M., Smereka, P., & Osher, S. (1994). A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational physics*, *114*(1), 146-159.

Sweby, P. K. (1984). High resolution schemes using flux limiters for hyperbolic conservation laws. *SIAM journal on numerical analysis*, *21*(5), 995-1011.

Tang, H., & Wrobel, L. C. (2005). Modelling the interfacial flow of two immiscible liquids in mixing processes. *International journal of engineering science*, *43*(15), 1234-1256.

Taylor, T. D., & Acrivos, A. (1964). On the deformation and drag of a falling viscous drop at low Reynolds number. *Journal of Fluid Mechanics*, *18*(03), 466-476.

Tezduyar, T. E. (2006). Interface-tracking and interface-capturing techniques for finite element computation of moving boundaries and interfaces. *Computer methods in applied mechanics and engineering*, *195*(23), 2983-3000.

Tornberg, A. K. (2000). *Interface tracking methods with application to multiphase flows* (Doctoral dissertation, Royal Institute of Technology, Stockholm).

Trujillo, M. F., Alvarado, J., Gehring, E., & Soriano, G. S. (2011). Numerical simulations and experimental characterization of heat transfer from a periodic impingement of droplets. *Journal of Heat Transfer*, *133*(12), 1-10.

Trujillo, M. F., Hsiao, C., Choi, J., Paterson, E. G., Chahine, G. L., & Peltier, L. J. (2007). Numerical and experimental study of a horizontal jet below a free surface. In *9th International conference on numerical ship hydrodynamics, Ann Arbor, MI*.

Ubbink, O. (1997). *Numerical prediction of two fluid systems with sharp interfaces* (Doctoral dissertation, University of London).

Ubbink, O., & Issa, R. I. (1999). A method for capturing sharp fluid interfaces on arbitrary meshes. *Journal of Computational Physics*, *153*(1), 26-50.

Unverdi, S. O., & Tryggvason, G. (1992). A front-tracking method for viscous, incompressible, multi-fluid flows. *Journal of computational physics*, *100*(1), 25-37.

Van Leer, B. (1973). Towards the ultimate conservative differencing scheme, In Cabannes, H. and Temem, R., editors, *Proceedings of the Third International Conference on Numerical Methods in Fluid Mechanics*, volume 1, 163-168. Springer.

Van Leer, B. (1974). Towards the ultimate conservative difference scheme. II. Monotonicity and conservation combined in a second-order scheme. *Journal of computational physics*, *14*(4), 361-370.

Walters, J. K., & Davidson, J. F. (1962). The initial motion of a gas bubble formed in an inviscid liquid Part 1. The two-dimensional bubble. *Journal of Fluid Mechanics*, *12*(03), 408-416.

Walters, J. K., & Davidson, J. F. (1963). The initial motion of a gas bubble formed in an inviscid liquid. *Journal of Fluid Mechanics*, *17*(03), 321-336.

Wang, J. S., Ni, H. G., & He, Y. S. (2000). Finite-difference TVD scheme for computation of dam-break problems. *Journal of Hydraulic Engineering*, *126*(4), 253-262.

Warming, R. F., & Beam, R. M. (1976). Upwind second-order difference schemes and applications in aerodynamic flows. *AIAA Journal*, *14*(9), 1241-1249.

Wegener, P. P., & Parlange, J. Y. (1973). Spherical-cap bubbles. *Annual Review of Fluid Mechanics*, *5*(1), 79-100.

Wong, H. H., & Raithby, G. D. (1979). Improved finite-difference methods based on a critical evaluation of the approximation errors. *Numerical Heat Transfer, Part A: Applications*, *2*(2), 139-163.

Wu, J., Yu, S. T., & Jiang, B. N. (1998). Simulation of two-fluid flows by the least-squares finite element method using a continuum surface tension model.*International journal for numerical methods in engineering*, *42*(4), 583-600.

Yeoh, G. H., & Tu, J. (2009). *Computational techniques for multiphase flows*. Elsevier.

Youngs, D. L. (1982). Time-dependent multi-material flow with large fluid distortion. *Numerical methods for fluid dynamics*, *24*, 273-285.

Zalesak, S. T. (1979). Fully multidimensional flux-corrected transport algorithms for fluids. *Journal of computational physics*, *31*(3), 335-362.