

Numerical prediction of two fluid systems with sharp interfaces

by

Onno Ubbink

Thesis submitted for the
Degree of Doctor of Philosophy of the University of London
and
Diploma of Imperial College

Department of Mechanical Engineering
Imperial College of Science, Technology & Medicine

January 1997



To my wife Marike and my children

Anja, Carla and Johan

Acknowledgements

I would like to express my deepest gratitude to my supervisors, Prof. A.D. Gosman and Dr R.I. Issa, for the guidance provided throughout the course of my research, their constructive criticism, many valuable suggestions and continuous support.

I would like to thank all the members and previous members of the Fluid Sections with whom I had numerous fruitful discussions, in particular Mr H. Jasak, Dr G. Tabor, Mr G. Bennett, Mr D. Hill and Mr D. Brennen. I also owe many thanks to Mr H. Weller for all the helpful discussions and the use of his research code FOAM. My thanks also go to Prof. S. Koshizuka, visiting professor from the University of Tokyo, who kindly made available the photographs of the experiments on the collapse of a liquid column, carried out by them.

I am grateful to the small, but active, CFD community in South Africa for all their efforts to keep the research field as up to date as possible. A special word of thank to Mrs L. van Eeden, Dr P.S. du Toit, Dr I.M. Snyman, Mr T.G.N. Kotze and late Dr G.P. van Zyl (previous director of Compuflow Consultants C.C.) for numerous discussions, the financial support and arrangement of many administrative matters. Thank you for the opportunity to study abroad.

My thanks also go to Mrs N. Scott Knight who arranged many of the administrative matters here at Imperial College.

Finally, I take the opportunity to thank my family and friends for their support and interest throughout the research. I am also much indebted to my wife Marike and my children Anja, Carla and Johan for all their love, support and patience during the three years abroad. I am quite sure, I could not have brought this work to this degree of completion without their sustained encouragement.

Abstract

There is a need for a computational fluid dynamic (CFD) methodology, capable of predicting the topology and effects of fluid/fluid interfaces, which can be used as a design tool. This methodology should be robust and general enough to be applicable over a wide spectrum of engineering applications.

This study contributes to the field by presenting a method capable of capturing a moving interface between two immiscible fluids on an arbitrary Eulerian mesh. The two fluids are modelled as a single continuum with a fluid property jump at the interface. A volume fraction indicator function is used to mark the individual fluids and the interface is defined as the transitional region between the fluids. A transient scalar convection equation is used to propagate it through the computational domain. In conjunction, the conservation equations for mass and momentum are solved for the dynamics of the fluid flow. Surface tension effects at the interface are included.

A finite volume discretisation is applied to the transport equations. The volume fraction convection equation is discretised with a new compressive high resolution differencing scheme which satisfies local boundedness while keeping the transitional area between the two fluids restricted to one cell width. The transient discretisation in multiple dimensions of the convection equation is achieved using a new implicit predictor-corrector method. The consistent discretisation of the momentum and continuity equations with respect to the discretisation of the volume fraction equation is used to ensure a solution algorithm which satisfies the conservation of the flow properties at all times.

This enables the modelling of interfaces which are exposed to large deformations. The merging or breakup of the interface is also handled in a natural way. Several test cases are presented to show the ease and accuracy with which this method can be used to predict the flow behaviour of immiscible fluids. Comparison with exact solutions and/or experimental data is made for several of these cases.

Table of contents

Acknowledgements	3
Abstract	4
Table of contents	5
List of figures	8
List of tables	13
Nomenclature	14
Abbreviations	18
1 Introduction	19
1.1 Background	19
1.2 Previous related studies	21
1.2.1 Surface methods	22
1.2.2 Volume methods	27
1.3 Current status and shortcomings	36
1.4 Present contribution	38
1.5 Outline of thesis	39
2 The mathematical model	41
2.1 Introduction	41
2.2 General transport equation	42
2.3 Governing equations	43

2.4	Surface tension	48
2.5	Initial and boundary conditions	52
2.6	Closure	55
3	Numerical model	56
3.1	Introduction	56
3.2	Spatial discretisation	57
3.3	Discrete equations	58
3.3.1	Indicator equation	61
3.3.2	Momentum equation	64
3.3.2.1	Convection term	65
3.3.2.2	Diffusion term	68
3.3.2.3	Curvature	69
3.3.2.4	Temporal discretisation	73
3.3.3	Pressure equation	74
3.4	Boundary conditions	77
3.4.1	Fixed value boundary conditions	78
3.4.2	Fixed gradient boundary conditions	80
3.4.3	Wall adhesion	81
3.5	Solution algorithm	82
3.5.1	Pressure-velocity coupling with PISO	83
3.5.2	Solution procedure for the two-fluid system	84
3.6	Conclusions	84
4	Compressive differencing scheme	86
4.1	Introduction	86
4.3	Donor-acceptor flux approximation	88
4.4	Normalised Variable Diagram	92
4.4.1	The Normalised Variable for arbitrary meshes	96
4.5	Comparison of compressive differencing schemes	98
4.6	CICSAM differencing scheme	100

4.7	Multiple dimensions	104
4.7.1	Boundedness	108
4.7.2	Predictor-corrector solution procedure	110
4.8	Closure	116
5	Case studies	117
5.1	Introduction	117
5.2	Post processing	118
5.3	Uniform density flow	120
5.3.1	Circle in an oblique velocity field	121
5.3.2	A square 'bubble' in an oblique velocity field	127
5.3.3	A wave in an oblique velocity field	130
5.3.4	Rotation of a rectangle	130
5.3.5	Surface tension	138
5.4	Sloshing	143
5.5	Taylor bubble	149
5.6	Collapse of a liquid column	162
5.6.1	Collapse with a return wave	162
5.6.2	Collapse with an obstacle	168
5.7	Hydraulic bore	173
5.8	Flow through a sluice-gate	177
5.9	Splash of drop	179
5.10	Closure	182
6	Summary and conclusions	183
6.1	Summary	183
6.2	Future research	185
6.3	Conclusions	187
References		188

List of figures

Figure 1.1:	Different methods of representing the interface	22
Figure 1.2:	Schematic representation of height functions for (a) open interfaces (b) closed interfaces	24
Figure 1.3:	Contours of the level set function: (a) initial configuration, (b) just before merging with no corrections (c) just before merging with corrections	25
Figure 1.4:	Collapse of a liquid column with an interface fitted method (a) at time $t=3.0$ and (b) at time $t=4.0$ (from: Ramaswamy & Kawahara, 1987, Int. J. Numer. Methods Fluids, Vol. 7, Fig. 5 p. 965 and Fig. 6 p. 966)	27
Figure 1.5:	Schematic representation of a typical marker and cell mesh layout.	28
Figure 1.6:	Volume fractions on a discrete mesh	30
Figure 1.7:	Comparison of different line techniques for the prediction of the fluid distribution in a cell	31
Figure 1.8:	Interface approximations for the centre cell with the SLIC method	32
Figure 1.9:	Schematic representation of a donor-acceptor cell configuration .	33
Figure 1.10:	Deformation of interface: (a) square in oblique velocity field and (b) rectangle in circular velocity field (Lafaurie <i>et al.</i> , 1994, J. Comput. Phys., Vol. 113, Fig. 4 p. 138 and Fig. 7 p. 139.)	35
Figure 2.1:	General form of the conservation law	42
Figure 2.2:	Forces on a curved interface	44
Figure 2.3:	Continuity of the velocity field and discontinuity of the momentum	47
Figure 2.4:	Schematic representation of the transitional area	50
Figure 2.5:	Fluid arrangement and the sign of the curvature	51
Figure 3.1:	Control Volume	57

Figure 3.2:	The smoothness of the curvature (from: Brackbill <i>et al.</i> , 1992, J. Comput. Phys., Vol. 100, p. 344, Fig. 3.)	71
Figure 3.3:	Smoothing techniques applied by (a) Brackbill <i>et al.</i> (1992) and (b) Lafaurie <i>et al.</i> (1994)	72
Figure 3.4:	Plane view of a control volume with a boundary face	77
Figure 4.1:	One-dimensional control volume	87
Figure 4.2:	The donor-acceptor interface approximation	89
Figure 4.3:	Fluid configuration in the donor cell	90
Figure 4.4:	Original and normalised variables	92
Figure 4.5:	Normalised Variable Diagram (NVD) with the convection boundedness criteria (CBC)	93
Figure 4.6:	The convection boundedness criteria for explicit flow calculations	94
Figure 4.7:	The prediction of the upwind value for an arbitrary cell arrangement	97
Figure 4.8:	Comparison between the donor-acceptor formulation and the upper bound of the CBC	99
Figure 4.9:	Effect of the critical angle on propagation (from: Lafaurie <i>et al.</i> , 1994, J. Comput. Phys., Vol. 113, Fig. 6, p. 138)	101
Figure 4.10:	Building blocks of the CICSAM differencing scheme: (a) The upper bound of the CBC (b) ULTIMATE-QUICKEST	103
Figure 4.11:	NVD for CICSAM differencing scheme	104
Figure 4.12:	Prediction of the face value in multiple dimensions	106
Figure 4.13:	Consequence of controlled downwinding in multi-dimensional flow calculations	109
Figure 4.14:	The CBC for multi-dimensional flow calculations	110
Figure 5.1:	Different methods of presenting the interfacial structures	119
Figure 5.2:	A circle in an oblique velocity field (a) a 200x100 uniformly spaced mesh (b) enlargement of the initial shape (c) enlargement of the final	

shape predicted with the VOF method (Hyper-C (Leonard (1991)))	123
Figure 5.3: Enlargements of the final shapes predicted with different spatial differencing schemes	124
Figure 5.4: Enlargements of the final shapes predicted by Hyper-C with different temporal differencing schemes ($c_D=0.334$)	125
Figure 5.5: Enlargements of the final shapes for CICSAM using CN temporal discretisation with different Courant numbers	126
Figure 5.6: A volume fraction square convected with an oblique velocity field on a uniform mesh	128
Figure 5.7: A volume fraction square convected with an oblique velocity field on a non-uniform mesh	129
Figure 5.8: A volume fraction wave convected with a uniform oblique velocity field on a (70x30) mesh with cyclic boundary conditions	131
Figure 5.9: A volume fraction wave convected with a uniform oblique velocity field on a (140x60) mesh with cyclic boundary conditions	131
Figure 5.10: Schematic representation of the interface on (a) hexagons and (b) triangles	132
Figure 5.11: Volume fraction rectangle in a circular velocity field	133
Figure 5.12: Results obtained with (a) Hyper-C and (b) CICSAM	133
Figure 5.13: The rotation of a rectangle captured on a mesh consisting of hexagons	134
Figure 5.14: The rotation of a rectangle captured on a mesh consisting of triangles	135
Figure 5.15: The half rotation of a rectangle obtained with different values of k_y	136
Figure 5.16: Time evolution of the shape change of a square subjected to surface tension forces	140
Figure 5.17: Pressure jump induced by surface tension: (a) the volume fraction contours of a circle with radius 1mm, (b) the curvature predicted with	

the CSF model, (c) pressure contours and (d) the pressure distribution on the diagonal	141
Figure 5.18: Parasite currents at the surface of a circle with radius 1mm for different surface tension coefficients and viscosity values	142
Figure 5.19: Initial geometry of the wave	143
Figure 5.20: Plots of the wave position and velocity vectors for the first period of the sloshing of an inviscid liquid under the influence of gravity	147
Figure 5.21: Position of the interface at the left boundary plotted against the time	148
Figure 5.22: Wave positions for the first six periods	148
Figure 5.23: Correlation of the rise velocity of cylindrical bubbles in liquids in vertical tubes (from: White & Beardmore 1962, Chemical Engineering Science, Vol. 17, Fig. 5. p. 356.)	150
Figure 5.24: Cross plot showing the regions in which the effect of some variables become unimportant (from: White & Beardmore, 1962, Chemical Engineering Science, Vol. 17, Fig. 6 p. 357.)	150
Figure 5.25: Schematic representation of the computational setup and the boundary conditions	152
Figure 5.26: Time evolution of a Taylor bubble ($E_o=17.59$, $M=0.0015$)	155
Figure 5.27: Time evolution of a Taylor bubble ($E_o=39.76$, $M=0.015$)	156
Figure 5.28: The shape of a Taylor bubble for $M=0.0015$ and $E_o=17.59$	157
Figure 5.29: Rise velocity of a Taylor bubble for $M=0.0015$ and $E_o=17.59$...	157
Figure 5.30: Shape of a Taylor bubble for $M=0.015$ and $E_o=39.76$	157
Figure 5.31: Rise velocity of a Taylor bubble for $M=0.015$ and $E_o=39.76$	157
Figure 5.32: Predicted velocity vectors and bubble shapes for different Eötvös numbers with $M=0.0015$	158
Figure 5.33: Predicted velocity vectors and bubble shapes for different Morton numbers with $E_o=21.21$	159

Figure 5.34: (a) Plots of Froude number against Eötvös number for M=0.015 and (b) comparison between experimental and calculated Froud numbers for $E_o=21.21$	161
Figure 5.35: Initial geometry of the water column	162
Figure 5.36: Experimental results of a collapsing water column (Koshizuka (1995))	164
Figure 5.37: Numerical results of the collapsing water column	165
Figure 5.38: The height of the collapsing water column versus time	167
Figure 5.39: The position of the leading edge versus time	167
Figure 5.40: Geometry of collapsing water experiment with an obstacle	168
Figure 5.41: Experimental results of a collapsing water column hitting an obstacle (Koshizuka (1995))	171
Figure 5.42: Numerical results of a collapsing water column (Koshizuka (1995))	172
Figure 5.43: Schematic representation of the computational domain for the bore	173
Figure 5.44: Velocity vectors and fluid configuration for the bore calculated with $g=1$ at times 3.50s, 4.90s, 7.94s, 9.88s	175
Figure 5.45: Velocity vectors and fluid configuration for the bore calculated with $g=0.4548$ at times 0.97s, 7.10s, 8.42s, 15.89s	176
Figure 5.46: Schematic representation of the computational domain for the sluice-gate	177
Figure 5.47: Velocity vectors and fluid distribution of the flow through a sluice-gate at different times	178
Figure 5.48: The computational mesh used for the prediction of the splash of a short jet	180
Figure 5.49: Three-dimensional splash of a short jet at different times	181

List of tables

Table 5.1:	Comparison of temporal and spatial errors	146
Table 5.2:	Conditions for the comparison of bubble shapes and rise velocities	153
Table 5.3:	Conditions for the comparison of Taylor bubbles for different Eötvös numbers	160
Table 5.4:	Conditions for the comparison of Taylor bubbles for different Morton numbers	160
Table 5.5:	Comparison of the height of a bore	174

Nomenclature

A	- outward-pointing face area vector
a	- wave amplitude, general variable
a_p	- central coefficient
a_{nb}	- matrix coefficient of neighbour
b	- general variable
c	- Courant number
D	- tube diameter
D	- orthogonal part of the face vector
d	- vector between the computational point P and its neighbour N
d	- general variable
dS	- general surface area vector
$\mathcal{E}^{(l)}$	- B-spline of degree l
E_o	- Eötvös number
E^-	- magnitude of the unbounded volume fraction less than zero
E^+	- magnitude of the unbounded volume fraction greater than unity
e	- skewness error
$\mathcal{F}(\alpha)$	- Laplacian filter
F_C	- boundary flux due to convection
F_D	- boundary flux due to diffusion
F	- volumetric flux
F_r	- Froude number
f_σ	- surface tension force
f	- face, point in the centre of the face
g	- gravitational acceleration
$H(u)$	- transport part
h	- mesh size, average fluid depth
I	- unit tensor

$K(x; \epsilon)$	- smooth integration kernel
K	- dimensionless number
k	- non-orthogonal correction vector
k	- prescribed constant for the gamma differencing scheme, wave number
k_γ	- prescribed constant for the CICSAM differencing scheme
k_s	- small constant
\mathcal{L}_P	- interpolation factor
M	- number of cells in the horizontal direction, Morton number
m	- times to apply Laplacian filter, cell counter
N	- point in the centre of the neighbouring control volume
n	- normal vector to the interface
\hat{n}	- normal to the interface at the wall
n_w	- unit normal to the wall pointing towards the wall
n_t	- unit normal tangential to the wall pointing towards the liquid
n	- amount of faces of a control volume
P	- pressure, point in the centre of the control volume, period of sloshing
P_i	- pressure on the concave side of the curved interface
P_t	- pressure in the transitional area
Q_v	- volume source
Q_s	- surface source
R_1, R_2	- principal radii of curvature of the surface
S_{α_p}	- source term of discretised α equation
S_{u_p}	- source term of discretised momentum equations
T	- stress tensor for a Newtonian fluid
t	- time
t_s	- simulation time
t_t	- theoretical time
u	- velocity
u	- a component of u
V	- volume
x	- arbitrary point in the flow domain

y_s	- predicted interface height
y_t	- theoretical interface height
α	- indicator function, volume fraction
β	- weighting factor for the prediction of the CICSAM face value
Γ	- diffusion coefficient
γ_f	- weighting factor for the CICSAM differencing scheme
Δ	- difference operator
δ	- small finite thickness
δt	- time step
ε	- small variable
η_f	- linear weighting of face values between time steps
θ_{eq}	- static contact (wetting) angle of interface with wall
θ_f	- angle between the normal to interface and orthogonal component of the face area vector
κ	- interface curvature
λ	- blending factor
μ	- dynamic viscosity
μ_1	- dynamic viscosity fluid 1
μ_2	- dynamic viscosity fluid 2
ρ	- density
ρ_1	- density fluid 1
ρ_2	- density fluid 2
σ	- surface tension coefficient
ϕ	- general variable
∂V	- surface area of the control volume

Superscripts

ϕ^T	- transpose of a vector
$\bar{\phi}$	- smoothed values
$\tilde{\phi}$	- normalised values

ϕ^* - guessed or predicted values

ϕ^{**} - first correction

ϕ^{***} - second correction

Subscripts

ϕ_A - acceptor cell

ϕ_D - donor cell

ϕ_U - upwind cell of donor cell

ϕ_b - value on the boundary face

ϕ_{nb} - nearest neighbours of a control volume

ϕ_f - value on the face

ϕ_{f_n} - value on the north face (structured mesh)

ϕ_{f_e} - value on the east face (structured mesh)

ϕ_{f_s} - value on the south face (structured mesh)

ϕ_{f_w} - value on the west face (structured mesh)

Abbreviations

CFD	Computational Fluid Dynamics
CBC	Convection Boundedness Criteria
CD	Central Differencing
CICSAM	Compressive Interface Capturing Scheme for Arbitrary Meshes
CN	Crank-Nicolson
CSF	Continuum Surface Force model
FLAIR	Flux Line-segment model for Advection and Interface Reconstruction
HR	High Resolution
ISLIC	Improved Simple Line Interface Calculation
MAC	Marker And Cell
NVD	Normalized Variable Diagram
PIC	Particle In Cell
PISO	Pressure Implicit with Splitting of Operator
RMS	Root Mean Square
SLIC	Simple Line Interface Calculation
STOIC	Second- and Third-Order Interpolation for Convection
UD	Upwind Differencing
UQ	ULTIMATE-QUICKEST
VOF	Volume Of Fluid

1 Introduction

1.1 Background

The objective of this study is the development of an accurate, general and robust Computational Fluid Dynamics (CFD) methodology capable of predicting the flow behaviour of immiscible fluids, separated by a well defined interface.

Immiscibility of two fluids is a result of strong cohesion forces between their molecules and depends on the nature of the fluids. The ease with which the fluids can be mixed is expressed with an experimentally determined coefficient known as surface tension. The larger the value of this coefficient, the stronger the resistance to be mixed. A negative value indicates no resistance to mixing (Batchelor (1967)).

The flow of immiscible fluids is commonly encountered in nature and industrial processes. In some circumstances it results from a chemical reaction or phase change due to high temperatures, but in all cases it influences the flow behaviour. One of the frequently encountered regimes of such flows is slug flow in pipes. This is characterised by the pseudo-periodic occurrence of large bullet-shaped gas bubbles which occupy most of the cross-sectional area of the pipe. This pattern is encountered in a wide variety of industrial applications, including oil and gas wells, steam generating boilers, process

vaporisers and in certain stages of core cooling of a nuclear power plant during emergency operation (Mao & Dukler (1989)). In other industrial processes, a gas may be added to a liquid to enhance a certain chemical reaction or to create a certain flow pattern. Such processes are for example the ladle refining of steel or the oxygenation of water in sewage systems (Fujibayashi *et al.* (1995)). Other examples of industrial importance are the continuous casting of a metal or plastic under the influence of a free surface, and the free surface of a stirred tank.

Free surface flows feature most prominently in the marine environment and are characterised by wind-water interactions and unsteady waves. This combined interaction exerts unsteady loads on fixed and floating structures and has an important influence on the design of harbours and ships (Stoker (1958)). Another area of interest is the formation of cavitation bubbles which, for example, form on the low pressure surface of a marine propeller. The sudden collapse of these bubbles in higher pressure regions erodes metallic surfaces such as the hulls of boats and can destroy them eventually (White (1994)).

The flow of immiscible fluids can be classified into three groups based on the interfacial structures and topographical distributions of the phases, namely segregated flows, transitional or mixed flows and dispersed flows (Ishii (1975)). The three classes can be explained by considering a closed container partially filled with a liquid, with the other part occupied by a gas. The first class occurs when the container is oscillating very gently with a low amplitude and frequency and the two phases remain separated with a single well defined interface. Mixed or transitional flow occurs when the frequency and amplitude are increased to the extent that the waves become unstable and break. Part of the interface breaks up and small bubbles are trapped in the liquid. Dispersed flows occur when the container is shaken violently and the gas is suspended as small bubbles within the liquid.

This study deals mainly with the first group, namely segregated flows and the proposed CFD methodology should be able to cope with interfacial structures exposed to large deformations, merging or breakup. Dispersed flow, as defined above, is not necessarily

excluded, because on a scale where the mesh size is much smaller than the dispersed interfacial structures, individual dispersed phase globules can still be treated as segregated flow.

Segregated flow is complicated by the presence of a well defined interface, whose position must be determined as part of the solution algorithm (Ferziger & Perić (1996)). The requirements for a good numerical single-phase prediction method, such as conservation, generality, high accuracy, minimal computer storage and calculation time are also applicable to the prediction of multi-phase flow. The additional requirements, such as:

- the representation of the interface on a discrete grid,
- the movement of the interface with time,
- the treatment of partially filled cells and
- the coupling of the interface conditions with the equations of motion,

are addressed in this research.

References to CFD methodology capable of predicting interfacial flow date as far back as the early sixties. Since then there have been numerous developments in the field, but there are still limitations attached to each of them. The next section gives an overview of these developments, and their advantages and disadvantages. This is followed by a section which outlines the present contribution. The chapter is closed with a presentation of the structure of the thesis.

1.2 Previous related studies

Existing methods for the computation of free surfaces and fluid interfaces can be classified into two groups (Ferziger & Perić (1996)), namely:

- surface methods (surface fitting), and

- volume methods (surface capturing).

Figure 1.1 gives a schematic representation of these methods. In the first of these categories the interface is represented and tracked explicitly either by marking it with special marker points, or by attaching it to a mesh surface which is forced to move with the interface. In the second category of methods the fluids on either side of the interface are marked by either massless particles or an indicator function. Detailed descriptions of the two categories follow next.

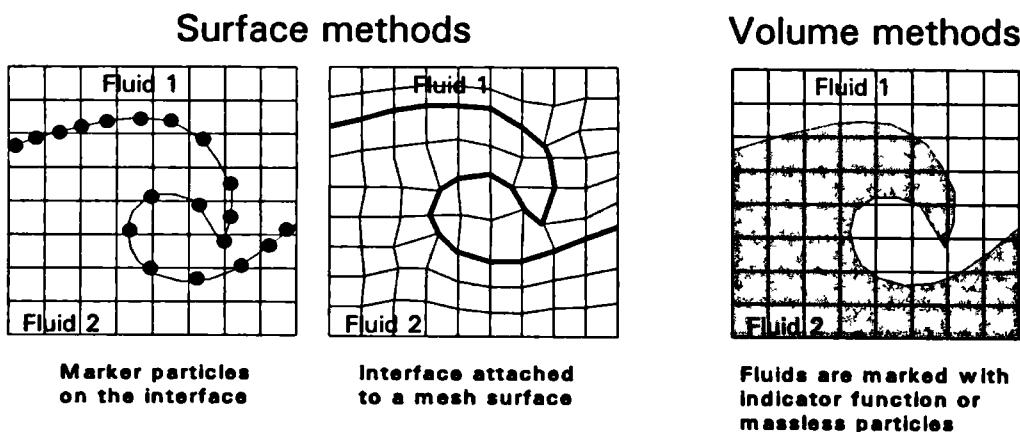


Figure 1.1: Different methods of representing the interface

1.2.1 Surface methods

As mentioned above, in this class of method the interface is marked with special marker points located on the interface. Between these points, its position is approximated by interpolation, usually piecewise polynomial (Hyman (1984)).

The advantage of this approach is that the interface position is known throughout the calculation and it remains sharp as it is convected across the mesh. This also eases the computational effort needed for the calculation of the interface curvature and its subsequent implementation for the inclusion of the surface tension force.

There are several ways of marking the interface, of which three will be explained here.

(a) Particles on interface: Daly (1969) presents a method to track an interface explicitly on a fixed grid by marking the interface with a set of connected massless marker particles. The local velocities are used to advect these massless particles in a Lagrangian manner.

The method is sensitive to the spacing between the marker particles. When the particles are far apart, the interface is not well resolved. If they are too close, then local fluctuations in the new positions of the particles can give rise to a very high interface curvature resulting in strong surface tension forces. The interface particles however, do not retain their spacing throughout the calculation as the surface is moved and it is therefore necessary to add or delete marker particles dynamically. For the calculation of the interface curvature, the order of the particles needs to be sequential along the interface, thus adding and deleting interface particles necessitates a continuous renumbering of the particle positions to keep them in sequence.

The fixed particle sequence mentioned above is a major disadvantage because it puts a restriction on the prediction of merging or rupturing interfaces. Furthermore, in three dimensions, bookkeeping of the particle connectivity becomes an almost impossible task. In this case the two-dimensional line-segments become three-dimensional surfaces which need to be attached to each other in a specific sequence (Takizawa *et al.* (1992)).

(b) Height functions: Nichols & Hirt (1973) extend the idea of interface marker particles by relating the reference points on the interface to points on a certain reference plane. The interface location is then defined by its height or distance from the reference plane. Figure 1.2 (a) gives a schematic representation of its implementation for open interfaces. For closed interfaces, such as a droplet or bubble, it is possible to define a single reference point inside the object. In this case the radius at different angular positions is the distance function (see Figure 1.2 (b)).

A major limitation of this method is that every reference coordinate can represent only one interface value. Therefore, it is not possible to predict situations where the reference coordinate can be multi-valued as is the case for example with droplet breakup or

overturning and breaking of waves as illustrated in Figure 1.2. However, this approach is extremely efficient in terms of computer storage and is most suitable for the modelling of non-complex free surfaces in a three-dimensional space.

Nichols & Hirt (1973) used height functions for three-dimensional wave prediction near obstacles on a fixed rectangular grid. Soulis (1992) and Lai & Yen (1993) implemented height functions in a two-dimensional implicit interface fitted framework and respectively presented results for the initial stages of a collapsing liquid column and channel flow with small disturbances at the free surface. Farmer *et al.* (1994) and Muzaferija *et al.* (1995) applied this method successfully to a three-dimensional interface fitted calculation of small amplitude waves.

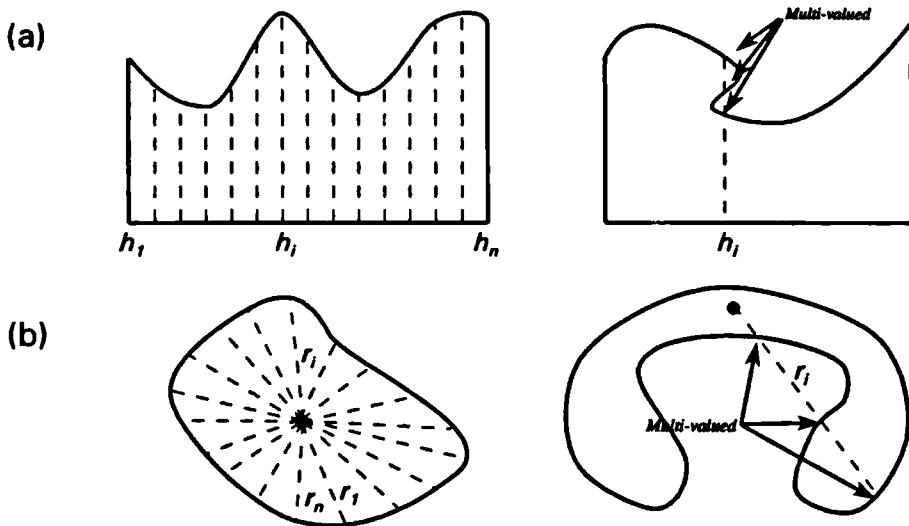


Figure 1.2: Schematic representation of height functions for (a) open interfaces (b) closed interfaces

(c) Level set method: A continuous function, known as a level set function (Osher & Sethian (1988) and Sethian (1996)), is introduced over the whole computational domain. 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)). Thus, the interface is defined to be where the function has a value of zero. In order to distinguish between the

two fluids on either side of the interface a negative sign is attached to the distance function for one of the fluids.

The level set function is a scalar property associated with the fluids and is propagated with the fluids by solving a scalar convection equation for it. Osher & Sethian (1988) prove that the interface will always remain at the zero level, but the initial distance function does not remain a distance function. There are two causes 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 throughout the calculation, causing a steep gradient in the level set function. The latter is a significant drawback of the level set method and will be discussed in more detail with the aid of Figure 1.3.

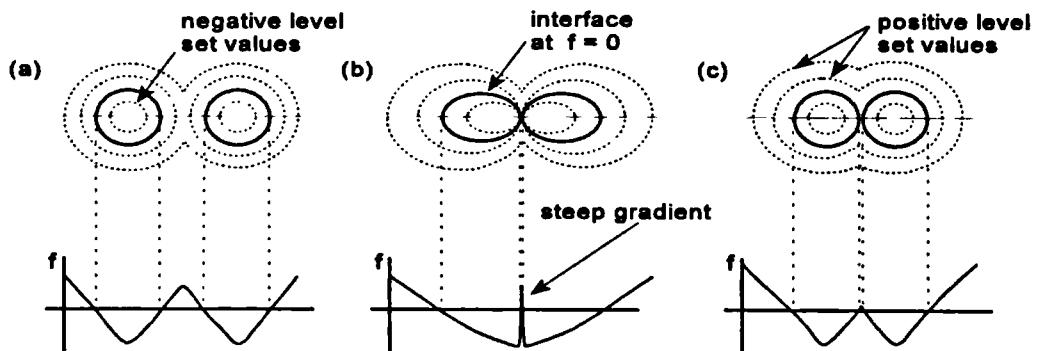


Figure 1.3: Contours of the level set function: (a) initial configuration, (b) just before merging with no corrections (c) just before merging with corrections

Figure 1.3 (a) shows the typical level set contours for two droplets moving towards each other. Figure 1.3 (b) shows the level set contours just before merging. The initial maximum level set value between the two interfaces is still present and forms an impenetrable sheet between the two interfaces, preventing the natural merging of the two

droplets. It is therefore necessary to reinitialize the distance function after each time step (*Sussman et al. (1994)*).

The level set approach has initially been used for the modelling of flame advection (*Osher & Sethian (1988)*). *Sussman et al. (1994)* use it for the modelling of bubble and droplet motions. *Li (1993)* and *Zhou (1995)* present results for Rayleigh-Taylor instability.

Additional to the above mentioned difficulties, the interface often cuts through cells, dividing them into two or more parts. These partially filled cells need special treatment. 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 occupying the cell. These proportions, also known as volume fractions (*Noh & Woodward (1976)*), are then used to predict the mean density of the fluids occupying a particular cell. For this reason the level set technique can also be classified as an interface capturing technique, rather than a surface tracking method.

(d) Surface fitted methods: Various techniques to attach the interface to a mesh surface have been developed during the past years (*Dervieux & Thomasset (1979)*, *Glimm et al. (1986 & 1988)*), *Takizawa et al. (1992)*, *Li & Zhan (1993)* and *Clarke & Issa (1995)*). A detailed description of the implementation of these methods is given elsewhere (*Ferziger & Perić (1996)*) and will not be repeated here.

These methods are implemented mainly for three reasons:

- to reduce computer storage needed for the interface markers,
- to ensure a sharp interface and/or
- to avoid partially filled cells (or empty cells in the case where free surface flow between a liquid and void is modelled).

The problem of partially filled cells, which has been addressed in the previous section, is the driving force in the development of most of the interface fitted methods and also applies to the last four references cited in the section on height methods. As in the case

of body-fitted coordinates, this approach is attractive because it allows for the accurate prescription of the boundary conditions of the interface.

These methods are limited to interfaces which are not subjected to large deformations, because these lead to significant distortion of the mesh. Numerical results on the collapse of a liquid column presented by Ramaswamy & Kawahara (1987) show that such large deformations may occur even in an early stage of the computation (see Figure 1.4). To maintain a well defined mesh it would therefore be necessary to re-mesh continuously, which brings with it yet more complexities.

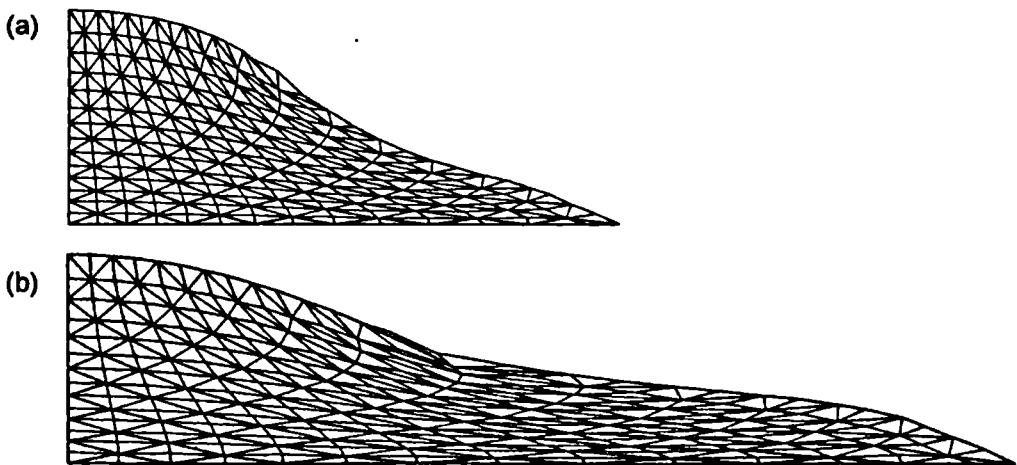


Figure 1.4: Collapse of a liquid column with an interface fitted method (a) at time $t=3.0$ and (b) at time $t=4.0$ (from: Ramaswamy & Kawahara, 1987, Int. J. Numer. Methods Fluids, Vol. 7, Fig. 5 p. 965 and Fig. 6 p. 966)

1.2.2 Volume methods

As noted before, volume methods mark the fluids on either side of the interface. This gives rise to the main drawback of this technique, as the exact position of the interface is not known explicitly and special techniques need to be applied to capture a well defined interface as part of the solution algorithm. Such techniques and the methods to mark the fluids are discussed in the remainder of this section.

The calculation of the surface tension forces in volume methods has long been a problem but has been partially overcome by the continuum surface force (CSF) model of Brackbill *et al.* (1992). Their model has been implemented successfully in various volume based methods (Kothe & Mjolsness (1992), Lafaurie *et al.* (1994)).

(a) Particles in fluid: In the marker and cell (MAC) method of Harlow & Welch (1965), massless marker particles are spread over the volume occupied by a fluid with a free surface (see Figure 1.5). 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.

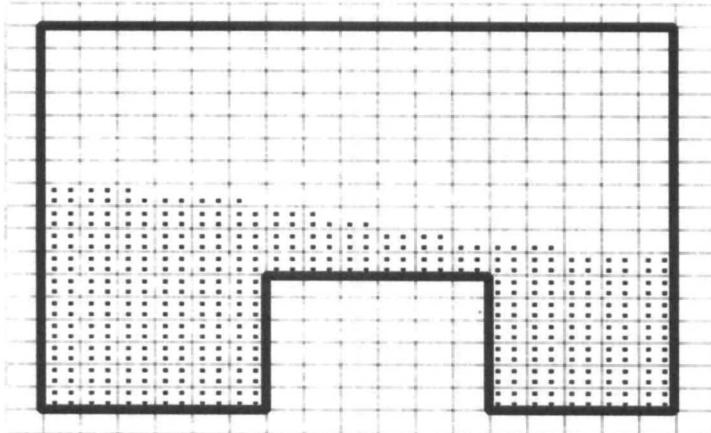


Figure 1.5: Schematic representation of a typical marker and cell mesh layout.

The marker particles are used to distinguish between the fluids and do not participate directly in the calculation as in the particle in cell (PIC) method of Harlow *et al.* (1976) and the grid-less method of Koshizuka *et al.* (1995). The marker particles serve as a flow identification aid whereby fluid element trajectories can be tracked. During each time step the particles are moved according to the velocity components in their vicinities.

Daly (1967) extended the MAC method of Harlow & Welch (1965) to deal with two-fluid calculations. Initially the marker particles are classified according to the fluid they

represent and this association remains for the duration of the calculation. A mesh cell containing marker particles from both fluids contains a segment of the interface. In Daly's method (1967), the marker particles are not only used to distinguish between the two fluids but also to calculate the density and viscosity of the fluid mixture in each cell. Further extensions and refinements on this approach can be found in Hirt & Shannon (1968) and Harlow *et al.* (1976).

The method of marker particles is attractive because it can treat complex phenomena like wave breaking. It is also readily extendable to three-dimensional computations. However, for the three-dimensional case it has the disadvantage of a significant increase in computing effort, because in addition to solving the equations governing the fluid flow, the motion of a large number of particles needs to be followed. The requirement for computer storage also increases significantly because the coordinates for each particle need to be stored.

(b) Volume fractions: A scalar indicator function between zero and one, known as the volume fraction, is used to distinguish between two different fluids. A value of zero indicates the presence of one fluid and a value of unity indicates the second fluid. On a computational mesh, volume fraction values between these two limits indicate the presence of the interface and the value itself gives an indication of the relative proportions occupying the cell volume (see Figure 1.6).

Using volume fractions is more economical than marker particles, as only one value (the volume fraction) needs to be accorded to each mesh cell. Another benefit of using volume fractions is that only a scalar convective equation, like other transport equations, needs to be solved to propagate the volume fractions through the computational domain. However this also has the drawback that most convective differencing schemes which guarantee a volume fraction field which obeys the physical bounds of zero and unity, such as upwinding, smear the step profile of the interface over several mesh cells.

During the past years different researchers proposed various techniques in order to maintain a well defined interface within the volume fraction framework. These fall into

the categories of *line techniques*, the *donor-acceptor formulation* and *higher order differencing schemes*. These approaches are reviewed next.

0.09	0.22	0.00	0.00
0.96	1.00	0.64	0.68
1.00	1.00	1.00	1.00

Figure 1.6: Volume fractions on a discrete mesh

- **Line techniques:** The well known SLIC method (Simple Line Interface Calculation) of Noh & Woodward (1976) falls in this category and has been developed to deal with multi-fluid flow. The fluid distribution in a cell containing part of the interface is obtained by using the volume fraction distribution of the neighbouring cells. Different methods for the determination of this distribution can be found in the literature and these are depicted schematically in Figure 1.7.

The original SLIC method of Noh & Woodward (1976) approximates the interface in each mesh cell as a line parallel to one of the coordinate axes and assumes different fluid configurations in that cell for the horizontal and vertical movements respectively. For the x-sweep the volume fraction values on the left and the right of a cell are used for the approximation of the fluid distribution in that cell and for the y-sweep the values above and underneath the cell are used. Four types of interface cells are allowed for with the original SLIC method (Noh & Woodward (1976)) and can be explained with the aid of Figure 1.8 as an example for the x-sweep:

- each neighbour is filled with a different fluid
- one of the neighbours contains a part of the interface
- both neighbours contain parts of the interface
- both neighbours contain the same fluid

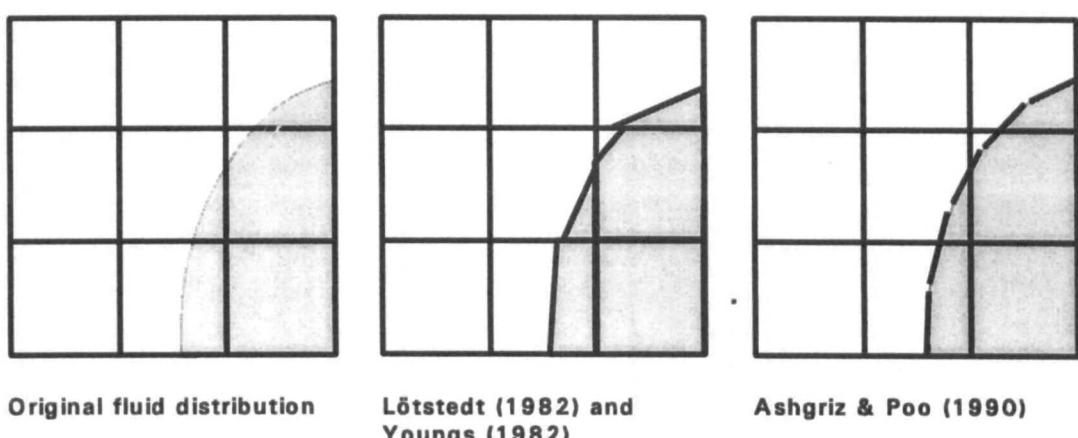
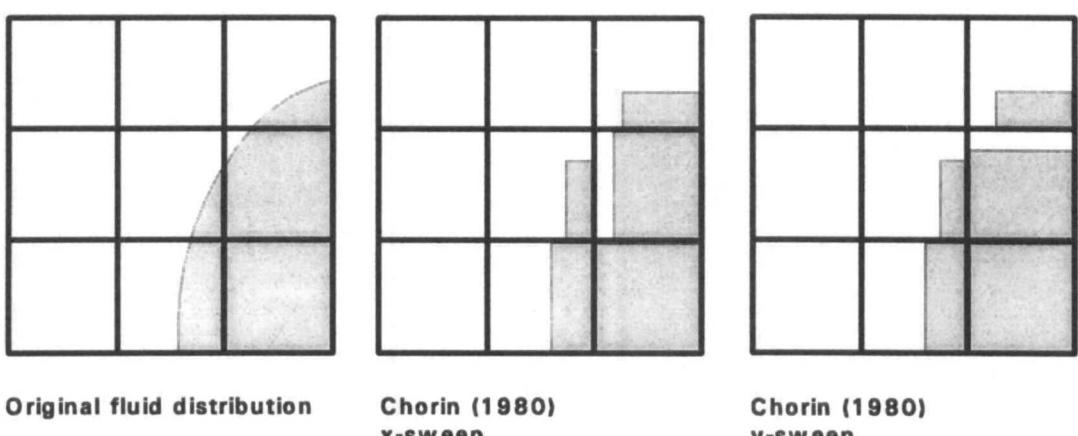
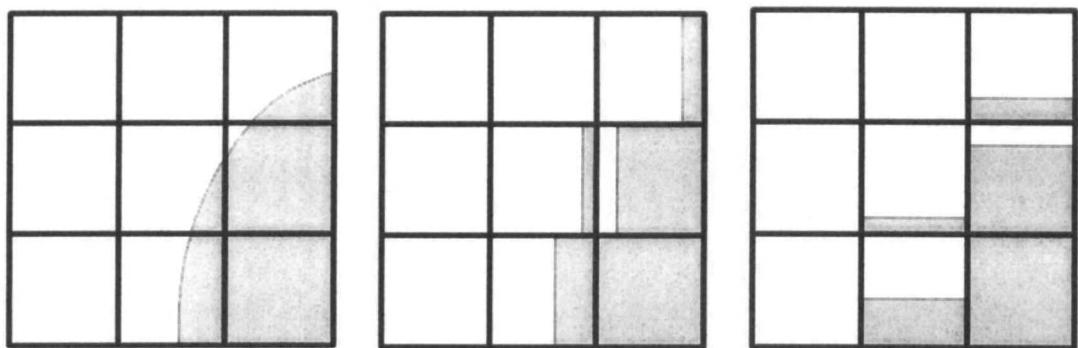


Figure 1.7: Comparison of different line techniques for the prediction of the fluid distribution in a cell

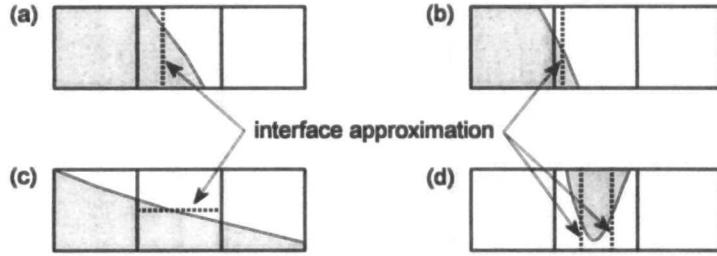


Figure 1.8: Interface approximations for the centre cell with the SLIC method

A similar set can be constructed for the y-sweep and together with those above they form the building blocks with which the SLIC interface are reconstructed as shown in Figure 1.7.

Chorin (1980) improves on the above method by using all the direct neighbours for the approximation of the fluid distribution in an interface cell and by introducing a fifth type of building block, namely a corner element. This approach also takes the direction of motion into account and gives different fluid distributions in a cell for the different sweeps as shown in Figure 1.7.

Youngs (1982) and Lötstedt (1982) give a useful refinement to the SLIC method with the use of oblique lines to approximate the interface in a cell. The main difference between the methods of Youngs (1982) and Lötstedt (1982) is the way in which the neighbouring cells are used to approximate the slope of the interface in each cell. Ashgriz & Poo (1991) improve on the SLIC method with their Flux Line-segment Model for Advection and Interface Reconstruction (FLAIR), by constructing the line-segments on the cell faces (see Figure 1.7).

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.

To implement these methods in higher dimensions it is necessary to use multiple one-dimensional sweeps, also known as *XY splitting* or *split operator technique*, to update the volume fractions during each time step (Youngs (1982)). The sequence of the sweeps normally alternates - the first sweep is in the x-coordinate direction followed by a sweep in the y-coordinate direction; for the next time step the operation is effected in the reverse order. Such methods are bound up with structured meshes and are therefore restricted to such mesh topologies.

- **Donor-acceptor scheme:** The basic idea of the donor-acceptor formulation is that the volume fraction value of the downwind cell (acceptor cell) of a cell face is used to predict the level of volume fraction transported through it during a time step (see Figure 1.9). The problem associated with the use of the downwind value is that the volume fraction values may become non-physical or unbounded, that is volume fraction values becoming larger than unity or smaller than zero.

Consider Figure 1.9 and let fluid 1 be the grey shaded fluid, represented by a volume fraction value of one. For the fluid configuration on the left, the use of downwind differencing (the acceptor cell value) will eventually cause volume fraction values larger than unity in the donor cell because more fluid 2 is required by the acceptor cell than

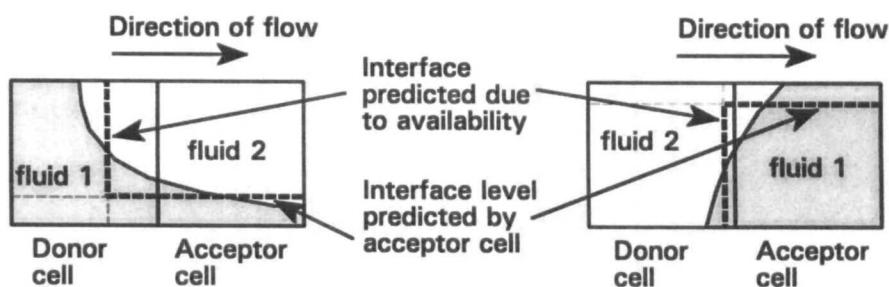


Figure 1.9: Schematic representation of a donor-acceptor cell configuration

available in the donor cell. Similarly, for the fluid configuration on the right of Figure 1.9 downwind differencing will eventually cause negative volume fraction values because more fluid 1 is required by the acceptor cell than available in the donor cell.

To ensure boundedness, the availability of fluid in the donor cell (volume fraction value of the donor cell) is used to adjust the level predicted by the acceptor cell (Ramshaw & Trapp (1976)). The derivation of the donor-acceptor formulation is given in Section 4.3.

A problem associated with the donor-acceptor formulation as presented by Ramshaw & Trapp (1976) is that it changes any finite gradient into a step, causing an incorrect steepening on interfaces which are aligned with the flow direction. Hirt & Nichols (1981) improve on the donor-acceptor formulation with their Volume of Fluid (VOF) method, by including some information on the slope of the interface into the fluxing algorithm. A donor-acceptor equation is presented which automatically switches between downwinding and upwinding depending on the orientation of the interface. The VOF method has also been extended to track interfaces in three dimensions (see Torrey *et al* (1987)). Lafaurie *et al.* (1994) presented an interesting version of this method, called SURFER, which can deal with merging and fragmenting of interfaces in multi-phase flows.

VOF is perhaps the most widely used method - results for the free surface flows of a collapsing liquid column, breaking bore and Rayleigh-Taylor instability are presented by Hirt & Nichols (1981), landslide-generated water-waves are presented by Heinrich (1991) and bubble motion by Tomiyama *et al.* (1993(a), 1993(b) and 1994). It is robust and therefore features in many commercial CFD codes. However it has two problems associated with it, namely:

- As in the case of the line-segments approach, VOF is derived for one-dimensional flow and its implementation in higher dimensions has until now been done with the operator splitting approach described earlier. It is therefore limited to structured meshes.

- Numerical results presented by Ashgriz & Poo (1991) and Lafaurie *et al.* (1994) show that the donor-acceptor formulation of VOF can deform the shape of the interface. Two figures which have been presented by Lafaurie *et al.* (1994), demonstrating this phenomenon, are shown in Figure 1.10. Panel (a) shows the deformation of a volume fraction square in an oblique velocity field at four different times. Figure 1.10 (b) shows the rotation of a volume fraction square in a circular velocity field. In both cases the final shape is significantly distorted.

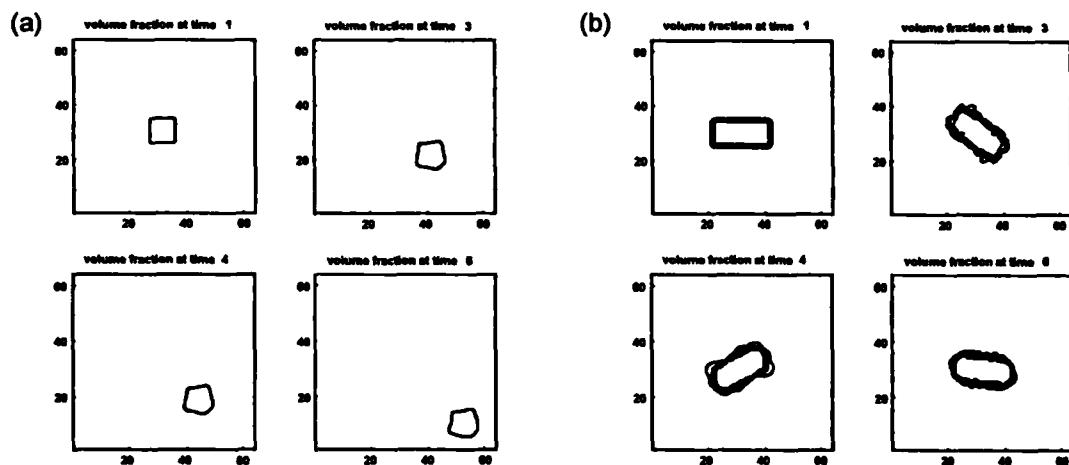


Figure 1.10: Deformation of interface: (a) square in oblique velocity field and (b) rectangle in circular velocity field (Lafaurie *et al.*, 1994, J. Comput. Phys., Vol. 113, Fig. 4 p. 138 and Fig. 7 p. 139.)

- Higher order differencing schemes:** Another approach to capture interfaces is to discretise the convective scalar transport equation using a higher order or blended differencing scheme. Davis (1994) discusses different higher order schemes for interface tracking and outlines the difference between less/non-diffusive and compressive schemes. A less/non-diffusive scheme prevents the smearing of a given profile. A compressive scheme not only prevents smearing, but also removes any diffusion that may exist in the vicinity of the interface. The latter appears to be exactly what is needed to maintain a sharp well defined interface.

Ghobadian (1991) applies a higher order scheme which has been published by Van Leer (1977). However, the scheme does not entirely eradicate the problem of interface

smearing. Special treatment is devised for the density field in order to reduce the effects due to numerical diffusion. The same scheme is also applied by Pericleous & Chan (1994) and results presented by them also indicate the problem of smearing.

A scheme called STOIC (second- and third-order interpolation for convection) is proposed by Darwish (1993) as a bounded non-diffusive scheme suitable for interface capturing. The scheme is based on the Normalised Variable Diagram (NVD) of Leonard (1991). The propagation of a step profile in different velocity fields has been used to compare the scheme with five other higher order schemes. However results presented indicate that it fails to keep the step within one cell. Furthermore, it has not been implemented for the modelling of two-fluid systems.

The above methods utilised (or proposed) to capture the interfaces are fairly recent, but initial indications are that unacceptable levels of diffusion of the interface often persist, as an outcome of the amount of numerical dissipation that is added to the fields to ensure boundedness. Moreover, most if not all such schemes have been developed for one-dimensional flow, with extensions to multi-dimensions being made by splitting the operations in the coordinate directions as mentioned before. Thus, their generalisation for use in conjunction with unstructured meshes is still required.

This completes the description of the methods for tracking/capturing fluid interfaces. Next a summary of the current status and shortcomings in the CFD methodology for the modelling of segregated fluid systems will be given. This will be followed by a section on the present contribution to this field.

1.3 Current status and shortcomings

The literature study presented in the previous section reveals that:

- Two approaches can be followed in the modelling of segregated flows, namely:
(i) surface methods (surface fitting), and (ii) volume methods (surface capturing).

- Various surface fitting techniques have been presented over the years and the single most important limitation attached to all of them is the inability to cope naturally with folding or rupturing interfaces, a feature which is necessary for a general method capable of predicting the flow of two fluids separated by a well defined interface. Only at a cost of complexity can these methods be extended to cope with these phenomena.
- It seems that methods dealing with fluid regions rather than the interface itself, are more capable of predicting flows where the interface interacts with itself (or folds over).
- The use of volume fractions to mark the fluid regions is preferred to the use of marker particles because of its economy in computer storage and the natural way in which the interface can be treated as an internal moving boundary. The volume fractions are convected through the domain by solving a scalar convection equation.
- Normal differencing schemes applied to the volume fraction convection equation introduce too much numerical diffusion and smear the step profile of the interface over several cells. To avoid this smearing, special care needs to be taken to minimise the numerical diffusion. Three approaches are used to accomplish this, namely:
 - the simple line techniques such as SLIC,
 - the donor-acceptor approach such as VOF and
 - high resolution differencing schemes.
- It is undesirable to use a Lagrangian approach, such as SLIC or FLAIR, due to its restriction to rectangular mesh cells and the difficulty posed in extending it to three-dimensional calculations.
- VOF is a powerful approach, although it is not known to have been implemented on unstructured meshes, also instances of non-physical deformation of the interface shape have been reported.

- The use of high resolution schemes is promising, but existing schemes are still too diffusive for the capturing of sharp interfaces. The implementation in multiple dimensions is also restricted to the technique of operator splitting. A generalisation for implementation on unstructured meshes is still required.

In the light of the above it was decided to use the volume fractions approach in the present study. A shortcoming which needs to be overcome in order to use volume fractions on arbitrary meshes with either the VOF method or high resolution schemes is a generalisation of the split operator technique. Furthermore, either the non-physical deformations caused by VOF or the numerical diffusion introduced by existing high resolution schemes needs to be addressed.

1.4 Present contribution

The aim of this research is to develop a scheme to predict the flow between two immiscible fluids, separated by a well defined interface. It should be robust, accurate and easy to be implemented on an arbitrary unstructured mesh, especially in three dimensions. To meet these requirements and to overcome the difficulties inherent in previous techniques the present contribution comprises the following:

- The mechanism of the donor-acceptor formulation used in VOF has been studied and a reformulation of it into a differencing scheme suitable for unstructured meshes is presented.
- The reported non-physical deformations to the interface caused by the above approach, have been investigated and their origin has been found. It has been established that a cure lies within the realm of high resolution schemes.
- A link between VOF and the upper bound of the convection boundedness criteria (Gaskell & Lau (1988)) used in high resolution schemes has been found and is presented.

- A new bounded discretisation scheme for the treatment of the advection terms in the transport equation of fluid fractions is presented. This scheme is based on the normalised variable diagram (NVD) but unlike other high resolution schemes, it is specifically tailored to deal with contact discontinuities such as fluid interfaces.
- The mechanism of the explicit operator splitting technique for multi-dimensional flows has been analysed and a new implicit formulation suitable for arbitrary meshes which dispenses with the need for operator splitting is presented. This enables the prediction of multi-dimensional flows free from directional dependence.
- Previously, in volume fraction based methodologies, only the non-conservative form of the momentum equations has been employed because of the difficulty in predicting the densities at cell faces. The present study overcomes these difficulties and solves the conservative form of these equations.
- The technique is implemented into a finite volume procedure utilising unstructured meshes. A solution procedure for the coupled set of transport equations is presented.
- The methodology developed here has been tested extensively and validations against analytical solutions and experimental data are presented herein. This includes the case of a collapsing water column with an obstacle in its path.

1.5 Outline of thesis

The remainder of the thesis is structured as follows:

Chapter 2: In this chapter, the concept of conservation laws which are used to construct the governing equations is introduced. The governing equations for the two-fluid system are given in a conservative coordinate free form and the expressions needed for modelling the surface tension force are also included in the momentum equations.

These equations, together with the necessary initial and boundary conditions, describe the flow of the two-fluid problem completely.

Chapter 3: A conservative finite volume discretisation scheme is used for the discretisation of the governing equations for the two-fluid system. This involves both spatial and temporal discretisation.

Chapter 4: The new bounded differencing scheme for the volume fraction equation is derived in this chapter. Special attention is given to both the spatial and temporal discretisation. The implementation for multi-dimensional flow is also presented in the form of a predictor-corrector solution algorithm.

Chapter 5: The two-fluid method developed is applied to three groups of problems. The first, comprising of uniform density flow calculations, is used to evaluate the behaviour of the new differencing scheme for pure scalar advection. The second, consisting of sloshing and slug flow, is used to compare the behaviour of the new method against analytical solutions and experimental data. The last group, consisting of the collapsing of a liquid column, breaking bore and splash of a droplet, is used to illustrate the versatility of the current scheme and the ease with which it can predict merging and rupturing interfaces.

Chapter 6: The thesis is summarised, and the main conclusions and suggestions for future research are given.

2 The mathematical model

2.1 Introduction

As mentioned in the first chapter, the aim of this study is the development of a CFD methodology capable of predicting the flow of two immiscible fluids, separated by a well defined interface. A mathematical model, describing the flow of the two fluids and the movements of the free boundary or interface separating them, is the subject of this chapter.

A standard mathematical model, which uses the continuum mechanics approach is used as a basis for this study (see for instance Ramshaw & Trapp (1976)). The fluids are modelled as a continuum with a jump in the fluid properties at the interface. The fluids are each marked with a different value of an indicator function and the interface is implicitly defined as the region where this function undergoes a step change.

This approach is general and applies equally to both laminar and turbulent flows, where in the case of the latter, turbulence needs to be modelled in some way, as has been successfully carried out by Hagiwara & Madarame (1992) and Lemos (1992). As the purpose of the current work is to develop and test a methodology for interface capturing and since the test cases involve flows dominated by inertial and pressure forces rather

than by turbulence effects, the issue of turbulence modelling will not be considered in any depth.

2.2 General transport equation

The fluid flow is mathematically described by three conservation laws, namely, the conservation of mass, momentum and energy. These laws completely determine the physical behaviour of the fluid and are totally independent of the nature of the fluid, which is defined by additional properties such as viscosity, heat conductivity, surface tension and compressibility. The general form of the conservation equation for a flow quantity ϕ with reference to the control volume shown in Figure 2.1, is (see for instance Hirsch (1988)):

$$\frac{\partial}{\partial t} \int_V \phi dV + \oint_{\partial V} F_C \cdot dS - \oint_{\partial V} F_D \cdot dS = \int_V Q_V dV + \oint_{\partial V} Q_S \cdot dS \quad (2.1)$$

where t is time, $F_C = \phi u$ the flux over the boundary due to convection (or motion of the fluid), u the fluid velocity, F_D the flux over the boundary due to diffusion, Q_V the internal source, Q_S the source at the boundary, V the volume of the control volume, ∂V its boundary and dS the outward-pointing surface element vector.

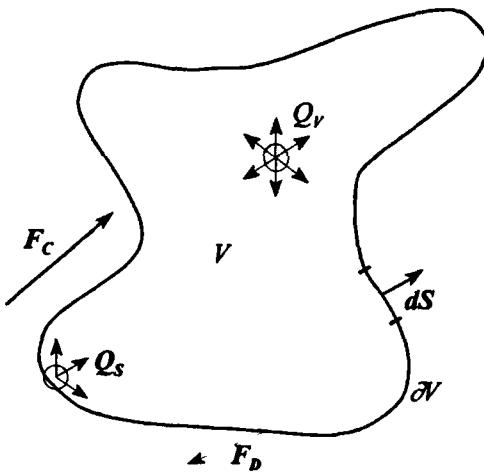


Figure 2.1: General form of the conservation law

For continuous fluxes and surface sources, Gauss's theorem can be applied to eq. (2.1):

$$\frac{\partial}{\partial t} \int_V \phi dV + \int_V \nabla \cdot \mathbf{F}_C dV - \int_V \nabla \cdot \mathbf{F}_D dV = \int_V Q_v dV + \int_V \nabla \cdot \mathbf{Q}_S dV \quad (2.2)$$

Equation (2.2) holds for an arbitrary volume. If the volume is contracted to a single point, then eq. (2.2) reduces to the general conservative differential form:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot \mathbf{F}_C - \nabla \cdot \mathbf{F}_D = Q_v + \nabla \cdot \mathbf{Q}_S \quad (2.3)$$

This general transport equation for a flow quantity, which can be a scalar, vector or tensor field is used in the next section for the derivation of the full set of governing equations for a two-fluid system. From eq. (2.1) it is clear that the conservation of a flow quantity depends in some flows on the conditions at the boundaries of the domain. These boundary conditions are addressed in Section 2.5.

2.3 Governing equations

The transport equation for the conservation of mass is derived by substituting $\phi = \rho$, the mass per unit volume, in eq. (2.3). With the assumption of no sources such as chemical reactions or phase changes in the system, this equation reduces to:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \mathbf{u} = 0 \quad (2.4)$$

This relationship, also known as the continuity condition, states that the mass of a fluid in a closed domain can only be changed by flow across the boundaries, for the incompressible case.

The transport equation for the conservation of momentum is derived by substituting ϕ in eq. (2.3) by $\rho \mathbf{u}$, the momentum per unit volume. It is further assumed that no diffusion of momentum can take place when the fluid is in rest, thus $\mathbf{F}_D = 0$. The sources

are defined by adding all the external forces per unit volume to the sum of all the internal forces. The only external force acting on the fluid volume considered here is ρg , the force due to gravity, where g is the gravitational acceleration. The internal forces cancel out in pairs at every point inside the fluid volume and are manifested as stresses at the fluid boundary, where no counter forces are available.

The stress tensor T for a Newtonian fluid in local thermodynamic equilibrium, which is not exposed to very high temperatures or pressure ranges, is defined as:

$$T = -\left(P + \frac{2}{3}\mu\nabla\cdot u\right)I + \mu(\nabla\otimes u + (\nabla\otimes u)^T) \quad (2.5)$$

where P is the pressure, μ the dynamic viscosity of the fluid and I the unit tensor.

An internal force not yet addressed is f_σ , the force due to surface tension, see for instance Shepherd (1965). Surface tension is a tensile force tangential to the interface separating the two fluids which tries to keep the fluid molecules at the free boundary in contact with the rest of the fluid. Its magnitude depends mainly on the nature of the two fluids. For a curved interface the surface tension force also has a component normal to the interface (see Figure 2.2). If the fluids are in equilibrium this normal component f_σ is mechanically balanced by a pressure jump across the interface - otherwise the interface will accelerate. Clearly this pressure jump depends on the surface tension coefficient σ and the curvature of the interface. More details about f_σ and the calculation of the interface curvature are given in Section 2.4.

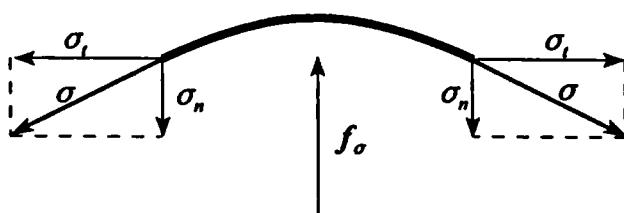


Figure 2.2: Forces on a curved interface

With the above assumptions and definitions substituted in eq. (2.3), the equation reduces to:

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u} - \mathbf{T}) = \rho g + \mathbf{f}_o \quad (2.6)$$

The equations of motion are closed with the constitutive relations for the density and dynamic viscosity:

$$\rho = \alpha \rho_1 + (1 - \alpha) \rho_2 \quad (2.7)$$

and

$$\mu = \alpha \mu_1 + (1 - \alpha) \mu_2 \quad (2.8)$$

where the subscripts 1 and 2 denote the different fluids. The indicator function α is defined as:

$$\alpha(x,t) = \begin{cases} 1 & \text{for the point } (x,t) \text{ inside fluid 1} \\ 0 & \text{for the point } (x,t) \text{ inside fluid 2} \end{cases} \quad (2.9)$$

where $\alpha(x,0)$ is given by the initial distribution of the fluids.

The above definition of α implies that it is a step function and a consequence of this is that the density defined by eq. (2.7) is piecewise continuous. In order to model the two fluids as a continuum by using equations (2.4) and (2.6) the density ρ should be continuous and differentiable over the whole flow domain (Unverdi & Tryggvason (1992), Sussman *et al.* (1994)). For the calculation of the interface curvature the requirement on the smoothness of α is even more stringent, needing to be twice differentiable (Brackbill *et al.* (1992), Lafaurie *et al.* (1994)). A possible way to achieve this is to give the transitional area between the two fluids, which is in reality a discontinuous step, a small finite thickness δ :

$$\alpha(x,t) = \begin{cases} 1 & \text{for the point } (x,t) \text{ inside fluid 1} \\ 0 & \text{for the point } (x,t) \text{ inside fluid 2} \\ 0 < \alpha_\delta < 1 & \text{for the point } (x,t) \text{ inside the transitional area} \end{cases} \quad (2.10)$$

As already noted, particular values of α are associated with each fluid. They are therefore propagated with the fluids as a Lagrangian invariant and thus have a zero material derivative (Hirt & Nichols (1981)):

$$\frac{D\alpha}{Dt} = \frac{\partial \alpha}{\partial t} + \mathbf{u} \cdot \nabla \alpha = 0 \quad (2.11)$$

This equation completes the mathematical description of the two-fluid system for laminar flows. In the case of turbulent flows a possible approach would be to separate the local value of a variable into the mean and the fluctuation around the mean. This averaging is then applied to the Navier-Stokes equations ((2.4) and (2.6)), which results in a set of averaged equations containing the so called Reynolds stress tensor. The $k-\epsilon$ turbulent model (Launder & Spalding (1974)), used by Hagiwara & Madarame (1992) and Lemos (1992) for interfacial flows, closes the set of equations with a two equation model by introducing k the turbulent kinetic energy and ϵ the dissipation rate. Thus additional to the transport equations for the laminar case the transport equations for k and ϵ need to be solved for. As noted before, this is not dealt with in the current study.

The equations derived above are general and completely describe the motion of the two fluids and the interface between them. Analytical solutions which simultaneously satisfy this set of partial differential equations subject to initial and boundary conditions are only available for simple circumstances. Therefore it is usually necessary to solve them numerically. However, as written above they are not in a form suitable for a numerical solution due to the discontinuity of $\rho \mathbf{u}$ at the interface, and need to be reformulated (Spalding (1974) and Maxwell (1977)), as described below.

The continuity equation (2.4) can be reformulated in the manner used by Spalding (1974):

$$\begin{aligned} \frac{\partial \rho}{\partial t} + \mathbf{u} \cdot \nabla \rho + \rho \nabla \cdot \mathbf{u} &= 0 \\ \Rightarrow \nabla \cdot \mathbf{u} &= -\frac{1}{\rho} \left(\frac{\partial \rho}{\partial t} + \mathbf{u} \cdot \nabla \rho \right) = -\frac{1}{\rho} \frac{D\rho}{Dt} = -\frac{D(\ln \rho)}{Dt} \end{aligned} \quad (2.12)$$

Equation (2.12) is in the so-called non-conservative form of the continuity equation. For two-fluid systems with high density ratios it is much more suitable for numerical solution, because \mathbf{u} is by definition continuous at the interface, as explained by Richardson (1989). Figure 2.3 shows a closed domain containing two immiscible fluids with different densities. Fluid 1 with the higher density enters from the left and displaces the same amount of fluid 2 on the right. The velocity \mathbf{u} of the fluid entering and leaving the domain is the same, but the momentum $\rho\mathbf{u}$ of the fluid entering the domain differs from the momentum leaving it.

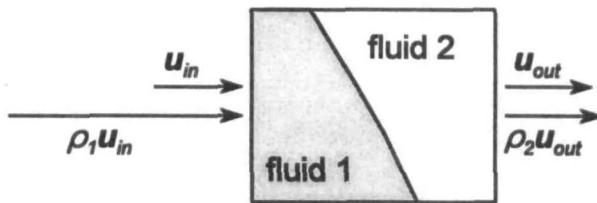


Figure 2.3: Continuity of the velocity field and discontinuity of the momentum

In this study it is assumed that both fluids are incompressible, in which case the r.h.s. of eq. (2.12) reduces to zero. This can be shown by substituting eq. (2.7) into eq. (2.12) and then applying eq. (2.11) to the result, thus:

$$\begin{aligned}\nabla \cdot \mathbf{u} &= \frac{-1}{\rho} \frac{D}{Dt} (\alpha(\rho_1 - \rho_2) + \rho_2) \\ &= \frac{-(\rho_1 - \rho_2)}{\rho} \left(\frac{D\alpha}{Dt} \right) = 0\end{aligned}\tag{2.13}$$

The incompressibility condition eq. (2.13) can be used to rearrange the transport equation for α , eq. (2.11), into a conservative form, suitable for finite volume discretisation by recognising that $\nabla \cdot \alpha \mathbf{u} = \alpha \nabla \cdot \mathbf{u} + \mathbf{u} \cdot \nabla \alpha$. This results into:

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot \alpha \mathbf{u} = 0\tag{2.14}$$

The incompressibility condition eq. (2.13) can also be used to reduce some of the terms in the stress tensor eq. (2.5), thus simplifying the momentum equations to:

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) - \nabla \cdot (\mu \nabla \otimes \mathbf{u}) = -\nabla P + \rho \mathbf{g} + \mathbf{f}_\sigma + (\nabla \otimes \mathbf{u}) \cdot (\nabla \mu) \quad (2.15)$$

To summarise, the final forms of the transport equations that need to be solved simultaneously are the continuity equation for incompressible flow eq. (2.13), the momentum equations (2.15) and the α transport equation (2.14), together with the constitutive relations for the density and dynamic viscosity respectively given by equations (2.7) and (2.8). The surface tension source term f_σ in the momentum equation is discussed in the next section.

2.4 Surface tension

In the general framework of thermodynamics, surface tension is dealt with on a macroscopic level based on the statistical behaviour of interfaces (Adamson (1976)). However, to understand the mechanism of this phenomenon, it should be considered at a microscopic level (Sabersky & Acosta (1964)). A liquid molecule is surrounded by many other molecules and away from the interface the forces which act on it are uniform in all directions. However, a molecule on the interface experiences only half of these forces, producing a resultant force pulling inward. This force opposes any increase in the area of the interface. In order to produce such an increase, a certain amount of work, proportional to the area increase, needs to be done.

The surface tension coefficient σ is defined as the amount of work necessary to create a unit area of free surface, see for instance Yuan (1970), Massey (1979). It always exists for any pair of fluids, and its magnitude is determined by the nature of the fluids. For immiscible fluids, the value is always positive and for miscible fluids such as water and alcohol, it is negative (Batchelor (1967)).

An important aspect of surface tension is that it creates a pressure jump ΔP across a curved surface. This produces the equilibrium which has been referred to during the derivation of the momentum equation. Laplace & Young (1805) (see Adamson (1976)) show that the magnitude of this pressure jump is a function of the mean interface curvature:

$$\Delta P = P_i - P_o = \sigma \left(\frac{1}{R_1} + \frac{1}{R_2} \right) = \sigma \kappa \quad (2.16)$$

where R_1 and R_2 are the principal radii of curvature of the surface, κ the mean curvature, σ the surface tension coefficient and P the pressure - the larger pressure P_i is on the concave side of the curved surface.

To include this pressure jump ΔP into the momentum equations it is necessary to express it as a proper gradient $\nabla P = f_\sigma$. Two immediate difficulties arise and they need to be addressed:

- The pressure jump manifests itself only at the interface. The momentum equations are defined for a continuum and hold for the whole flow domain. The two fluids are modelled as a continuum and the task now is to create a $f_\sigma \neq 0$ at the transitional area and zero everywhere else in the flow domain.
- The pressure gradient $\nabla P = f_\sigma$ needs to be calculated, but the jump in the pressure at the interface is discontinuous.

Brackbill *et al.* (1992) overcame the above difficulties with their continuum surface force (CSF) model. The novelty of the CSF model lies in the smooth α field used to define a normal vector to the interface, which is zero everywhere in the flow domain except in the δ transitional area between the fluids. Secondly, the pressure is assumed to rise within the δ transitional area with the same smoothness as the indicator function α . A short description of their approach follows.

The indicator function α has been defined as a continuous twice differentiable function which implies that it changes smoothly from zero to one in the transitional area. Figure 2.4 (a) gives an enlargement of this transitional area. According to the definition of α it follows that $\alpha(x,y,z) = c$, where $0 < c < 1$, defines different layers of the interface in the transitional area - these are also shown in Figure 2.4 (a). The line ab can be at any place along the interface and is defined to be normal to the different layers of the interface. Figure 2.4 (b) shows the smooth change of the indicator function (on the left axis) and the pressure (on the right axis), within the δ transitional area.

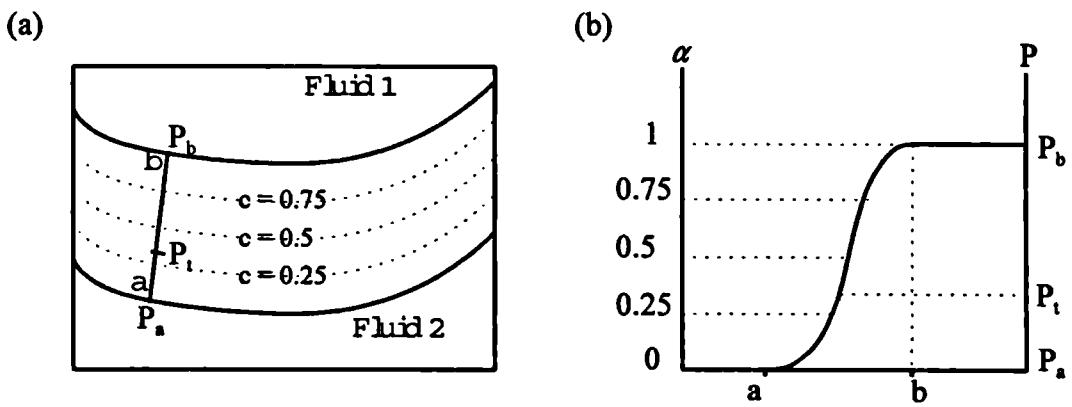


Figure 2.4: Schematic representation of the transitional area

The gradient of α gives the vector normal to each of these layers, which always points from fluid 2 towards fluid 1:

$$\mathbf{n} = \nabla\alpha \quad (2.17)$$

Thus, $\nabla\alpha$ is a continuous function which is zero everywhere in the flow domain except at the δ transitional area of the interface.

The curvature of the interface may therefore be expressed in terms of the divergence of the unit normal vector to the interface, as follows:

$$\kappa = -\nabla \cdot \left(\frac{\nabla\alpha}{|\nabla\alpha|} \right) \quad (2.18)$$

The definition of the above curvature is such that for $\kappa > 0$ fluid 1 lies on the concave side of the interface and for $\kappa < 0$ it is fluid 2 that lies on the concave side. The vector $\kappa \nabla \alpha$, however, always points towards the fluid lying on the concave side of the interface. A schematic representation of the relations between the different vectors, the positions of the fluids and the curvature of the interface is given in Figure 2.5.

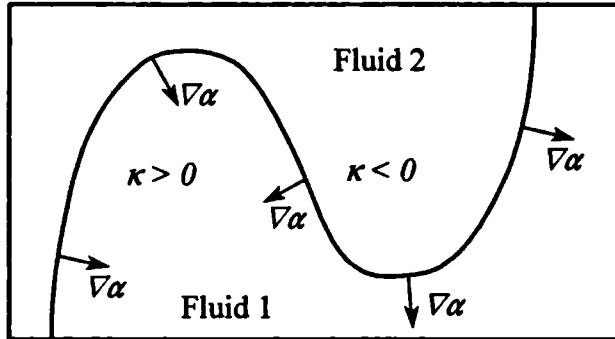


Figure 2.5: Fluid arrangement and the sign of the curvature

Equation (2.18) gives the curvature value for each of the interface layers in the transitional area and it is assumed that variation in κ , along the line ab shown in Figure 2.4, is negligible. The pressure P , due to surface tension at any point in the transitional area along the line ab shown in Figure 2.4 is:

$$P_t = P_a + \sigma \kappa (\alpha_t - \alpha_a) \quad (2.19)$$

where the curvature κ is defined in eq. (2.18). The above definition of the induced pressure satisfies eq. (2.16) for the pressure jump over the whole transitional area.

Due to the localised nature of the interface curvature it is possible to extend eq. (2.19) to the whole flow domain to give the CSF formulation for f_σ :

$$f_\sigma = \nabla P = -\sigma \left(\nabla \cdot \left(\frac{\nabla \alpha}{|\nabla \alpha|} \right) \right) (\nabla \alpha) \quad (2.20)$$

Equation (2.20) is applicable only when the surface tension σ is a constant. However, in principle it is possible to extend this methodology to a variable surface tension.

2.5 Initial and boundary conditions

In order to completely specify the mathematical model it is necessary to define the initial conditions as well as the conditions at the boundaries of the flow domain. Mathematically there are two types of boundary conditions, namely Dirichlet (fixed value) conditions or Von Neuman (fixed gradient) conditions. However, in a complicated mixture of elliptic, parabolic and hyperbolic behaviours this oversimplified mathematical classification is of little use. It is more appropriate to define the conditions for the different types of boundaries on physical arguments (Versteeg & Malalasekera (1995)).

- **Inlet:** An inlet boundary is a boundary where the fluid has a specified velocity distribution. The pressure at the inlet is unknown and a boundary value is extrapolated from the interior of the flow domain. However, if the gradient of the pressure at the inlet is small, it is sufficient to apply a zero gradient boundary condition.

A study of the nature of boundary conditions for flow with moving interfaces has been carried out by Renardy & Renardy (1991). Their main finding is that the position of the interface at an inlet needs to be defined. For this study it implies that the inlet condition of the indicator function needs to be fixed.

- **Outlet or open boundary:** Commonly, outflow boundaries are positioned whenever possible at locations where there are small variations in the flow. The boundary condition itself should be specified in such a way that overall mass continuity for the computational domain is satisfied. The two commonly used approaches in finite volume methods to satisfy this condition are:
 - Extrapolation of all the flow quantities: The velocity distribution of the first row of cells next to the boundary is used for the construction of a velocity distribution. The boundary velocities are then scaled according to this velocity profile such that overall continuity is satisfied.

- Fixed pressure boundary condition: The pressure on the boundary is fixed and a zero gradient condition is applied to the velocities (Versteeg & Malalasekera (1995)). This definition also allows fluid to enter the flow domain, an essential feature when interfacial flow in open containers is studied.

The position of the interface at an open boundary, is normally unknown a priori (Renardy & Renardy (1991)). Thus, taking into account that the boundary is placed where little variation in the flow occurs, it is sufficient to apply a zero gradient condition to the indicator function.

- Rigid boundaries (Walls): A good description of the conditions at walls is given by Richardson (1989) and the most important aspects will be summarised here. For the velocity the so called zero slip condition applies which means that the fluid at the wall takes the velocity of the wall.

The values of the other flow properties, such as the indicator function and the pressure, are unknown at a wall. In the case of the indicator function it is sufficient to apply a zero gradient boundary condition (Brackbill *et al.* (1992) and Kothe & Mjolsness (1992)). If the contact angle is not normal, the implementation of wall adhesion (described below) will compensate for this approximation.

For the approximation of the pressure boundary condition however, it is necessary to make sure that it is consistent with the velocity at the boundary. Furthermore, the pressure boundary condition should always hold, even for a system at rest. The momentum equations (2.15) for a fluid under these conditions reduce to:

$$\nabla P = \rho g \quad (2.21)$$

Equation (2.21) prescribes an imposed gradient boundary condition relation. Alternatively, extrapolation can be used where the gradient at the wall is approximated by the pressure gradient nearby in the flow domain. In the present study eq. (2.21) is applied.

Wall adhesion is a phenomenon which occurs when an interface meets with a wall and needs to be accounted for (Brackbill *et al.* (1992) and Kothe & Mjolsness (1992)). The forces between the molecules of a liquid and the molecules of a solid give rise to adhesion between them (Massey (1979)). If the forces of adhesion are larger than the cohesion forces among the liquid molecules themselves, the liquid molecules crowd towards the wall. The contact area between them tends to increase and the fluid wets the solid. The opposite is also true, if the adhesion forces are smaller than the cohesion forces then the area of contact will decrease. The angle between the interface and the wall, also known as the wetting angle, is measured experimentally. The wetting angle depends on the nature of the mediums in contact, as well as the roughness of the solid surface.

The vector \mathbf{n} normal to the interface (eq. (2.17)) which is used for the calculation of the interface curvature should be specified at the wall to account for wall adhesion. Brackbill *et al.* (1992) and Kothe & Mjolsness (1992) define the normal $\hat{\mathbf{n}}$ to the interface at the wall as follows:

$$\hat{\mathbf{n}} = \mathbf{n}_w \cos\theta_{eq} + \mathbf{n}_t \sin\theta_{eq} \quad (2.22)$$

where θ_{eq} is the static contact (wetting) angle, \mathbf{n}_w the unit normal vector to the wall pointing towards the wall and \mathbf{n}_t , the unit vector tangential to the wall pointing towards the liquid. A value of $\theta_{eq}=90^\circ$ means the interface is normal to the wall and $\theta_{eq}<90^\circ$ indicates the fluid wets the wall.

- **Symmetry plane:**

At a symmetry plane it is assumed that there is a mirror image of the fluid flow on the other side of the plane. For the velocity it implies that the normal component to the boundary should be zero and the gradient of the tangential component should be zero. For the other scalar properties it implies that their gradients should be fixed at zero.

- **Cyclic boundary:** These boundaries are

introduced to take advantage of a repetition in the flow configuration. The pairs of

cyclic boundaries are mapped on each other and values of all variables prevailing at them are equated to each other and treated as unknowns.

- **Initial conditions:** For transient calculations all the initial variables need to be consistent with each other. This can be achieved by specifying the velocity field as well as the indicator function.

The initial pressure field is irrelevant to the subsequent evolution of the flow field in incompressible flows; hence any guess of this field would suffice in theory. However, because the solution algorithm utilises the pressure at the previous time step, the initial pressure field does in practice influence the computational work involved in the first time step. Thus it is advantageous to initialise the pressure field to a distribution consistent with the velocity and density field in order to reduce this computational effort.

2.6 Closure

A mathematical model for the prediction of interfacial flow has been presented in this chapter. This model simulates the time dependent, incompressible, turbulence free, viscous flow of two-fluid systems under the influence of gravitational and surface forces. The analytical solution of the set of differential equations is only possible for very simple cases and therefore it is appropriate to use numerical techniques to obtain solutions for the more general flow scenarios. All the differential equations are presented in a conservative form, ready to be discretised with a finite volume technique, which is the subject of the next chapter.

3 Numerical model

3.1 Introduction

The construction of a numerical method for the solution of the transport equations derived in the previous chapter requires the flow domain to be represented as a mesh of discrete points (nodes) at which each term of the partial differential equations is evaluated. This results in a system of coupled algebraic equations that requires simultaneous solution by means of a suitable solution algorithm.

The discretisation method used in the present study is the well established finite volume method (Patankar (1980) and Versteeg & Malalasekera (1995)). This method subdivides the flow domain into a finite number of smaller non-overlapping control volumes. The transport equations are then integrated over each of these control volumes by approximating the variation of the flow properties between mesh points with piecewise profiles. This integration ensures that the conservation laws used in the derivation of the transport equations are satisfied locally at each of the control volumes. The overall conservation of the flow properties are therefore automatically preserved. These piecewise approximations, also known as differencing schemes, are constructed to support physical flow behaviour such as convection or diffusion.

This chapter presents the discretised equations, their treatment at the boundaries and the solution algorithm for solving the resulting coupled algebraic equations. Standard differencing schemes are used for the momentum equations and a new special compressive differencing scheme is used for the fluid indicator equation. The derivation of this new compressive differencing scheme will be treated separately in Chapter 4.

3.2 Spatial discretisation

An implementation of the finite volume method which uses a face addressing technique is used as a basis for this study. The objective of the face addressing technique is to allow the use of arbitrary shaped cells whilst preserving the efficiency of the computational algorithm. This approach uses the co-located variable arrangement (Perić (1985)) where all the flow properties are defined at the centres of the control volumes. This arrangement simplifies the computer programming and minimises the amount of geometrical information required about the computational grid. The notation used by Jasak (1996) is also used in the current study.

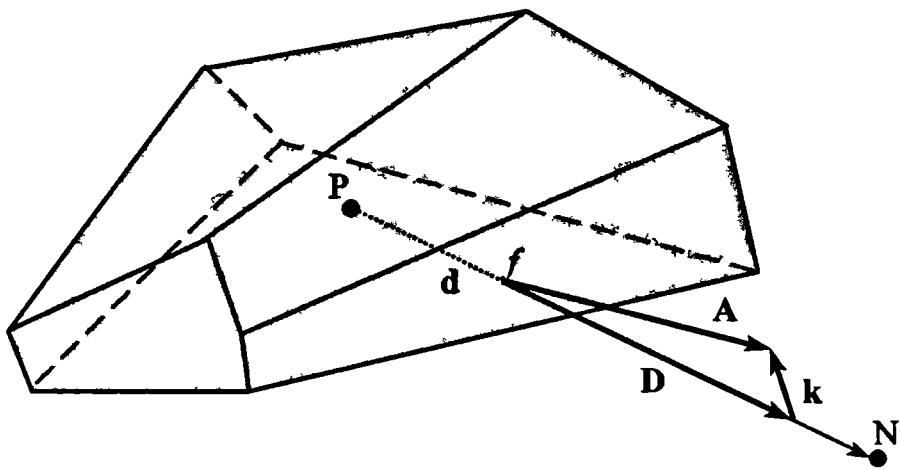


Figure 3.1: Control Volume

Figure 3.1 represents a typical control volume with an arbitrary number of faces and will be relevant throughout this chapter. The computational point P is in the centre of the

control volume. The vector $\mathbf{d} = \underline{P}\underline{N}$ connects the computational point P with its neighbour N and \mathbf{A} is the outward-pointing face area vector normal to the face.

The mesh is defined as being non-orthogonal if the angle between the vectors \mathbf{A} and \mathbf{d} is not zero. The vectors \mathbf{D} and \mathbf{k} are then introduced to account for the orthogonal and non-orthogonal contributions to gradients at cell faces respectively. Jasak (1996) describes three different approaches for the splitting of the non-orthogonal contribution \mathbf{k} from the orthogonal component \mathbf{D} . The so called over relaxed approach has shown to be the superior of the three approaches and will be used in this study for the discretisation of the momentum and pressure equations. The vectors \mathbf{D} and \mathbf{k} for the over relaxed approach (Jasak (1996)) are defined as:

$$\mathbf{D} = \frac{\mathbf{d}}{\mathbf{d} \cdot \mathbf{A}} |\mathbf{A}|^2 \quad (3.1)$$

$$\mathbf{k} = \mathbf{A} - \mathbf{D} \quad (3.2)$$

This concludes the discretisation of the space and it is now possible to construct the discretised finite volume equations over this discrete space.

3.3 Discrete equations

In the discretisation of the transport equations the same terms and algebraic expressions show up frequently. It is therefore useful to define them beforehand. One of them is the generalised form of Gauss' theorem:

$$\int_V \nabla \cdot \Phi dV = \oint_{\partial V} dS \cdot \Phi \quad (3.3)$$

$$\int_V \nabla \phi dV = \oint_{\partial V} dS \phi \quad (3.4)$$

$$\int_V \nabla \otimes \Phi dV = \oint_{\partial V} dS \otimes \Phi \quad (3.5)$$

where ϕ is an arbitrary scalar function, Φ an arbitrary vector function, dS the surface area vector and ∂V the surface area of the control volume (see Figure 2.1).

The control volume is bounded by a series of flat faces, and therefore eq. (3.3) can be transformed into a sum of integrals over the faces:

$$\begin{aligned} \int_V \nabla \cdot \Phi dV &= \oint_{\partial V} dS \cdot \Phi = \sum_{f=1}^n \left(\int_f dS \cdot \Phi \right) \\ &\approx \sum_{f=1}^n A_f \cdot \Phi_f \end{aligned} \quad (3.6)$$

where f is the centre of the cell face, A_f the face area vector and n is the number of faces of a control volume. For simplicity of notation, the faces of a control volume are also denoted with f . The summation in eq. (3.6) is done in two steps. Every face is shared by two neighbouring cells of which one owns the face and the other cell is the neighbour. The face area vector A_f always points from the owner cell into the neighbour cell of the face. This can be accounted for by splitting the sum over the faces into sums over the owner faces and neighbour faces:

$$\sum_{f=1}^n A_f \cdot \Phi_f = \sum_{\text{owner faces}} A_f \cdot \Phi_f - \sum_{\text{neighbour faces}} A_f \cdot \Phi_f \quad (3.7)$$

This is true for every summation over the faces and will automatically be assumed for the remainder of this study.

Unless stated otherwise the face values are predicted with linear interpolation:

$$\Phi_f = \mathcal{L}_P \Phi_P + (1 - \mathcal{L}_P) \Phi_N \quad (3.8)$$

where the interpolation of a vector means the interpolation of each of its components and \mathcal{L}_P is the interpolation factor defined as ratio of the distances $|fN|$ and $|PN|$:

$$\mathcal{L}_P = \frac{|fN|}{|PN|} \quad (3.9)$$

Similar relations to eq (3.6) can be derived for equations (3.4) and (3.5):

$$\int_V \nabla \phi dV \approx \sum_{f=1}^n A_f \phi_f \quad (3.10)$$

and

$$\int_V \nabla \otimes \Phi dV \approx \sum_{f=1}^n A_f \otimes \Phi_f \quad (3.11)$$

Equations (3.6), (3.10) and (3.11) can also be used to get a second order accurate discretised prediction of the gradient of a flow property over a cell:

$$(\nabla \cdot \Phi)_P \approx \frac{1}{V_P} \sum_{f=1}^n A_f \cdot \Phi_f \quad (3.12)$$

$$(\nabla \phi)_P \approx \frac{1}{V_P} \sum_{f=1}^n A_f \phi_f \quad (3.13)$$

$$(\nabla \otimes \Phi)_P \approx \frac{1}{V_P} \sum_{f=1}^n A_f \otimes \Phi_f \quad (3.14)$$

The above equations will be used in the derivation of the discretised transport equations given in the following sections.

3.3.1 Indicator equation

Before discretising the indicator equation, it is necessary to define α on a discrete mesh. Several methods have been described in the first chapter. The so called volume fraction method will be used here (see for instance Ramshaw & Trapp (1976), Noh & Woodward (1976), Chorin (1980) and Hirt & Nichols (1981)). With the volume fraction method, α is defined as the volume of fluid 1 in a control volume divided by the volume of the cell:

$$\alpha_P = \frac{\text{Volume of fluid 1}}{\text{Total volume of the control volume}} \quad (3.15)$$

The finite volume discretisation of the volume fraction convection equation is based on the integral form of eq. (2.14) over the control volume and the time step δt :

$$\int_t^{t+\delta t} \left(\int_V \frac{\partial \alpha}{\partial t} dV \right) dt + \int_t^{t+\delta t} \left(\int_V \nabla \cdot \alpha \mathbf{u} dV \right) dt = 0 \quad (3.16)$$

If P is the centroid of the constant-volume cell, then the first term of eq. (3.16) reduces to:

$$\begin{aligned} \int_t^{t+\delta t} \left(\int_V \frac{\partial \alpha}{\partial t} dV \right) dt &= \int_t^{t+\delta t} \left(\frac{\partial \alpha_P}{\partial t} V_P \right) dt \\ &= (\alpha_P^{t+\delta t} - \alpha_P^t) V_P \end{aligned} \quad (3.17)$$

where V_P is the volume of the cell.

Gauss' theorem (eq. (3.6)) can be applied to the volume integral of the convection term (the second term in eq. (3.16)) to give:

$$\begin{aligned} \oint_{\partial V} \alpha dS \cdot \mathbf{u} &\approx \sum_{f=1}^n \alpha_f A_f \cdot \mathbf{u}_f \\ &= \sum_{f=1}^n \alpha_f F_f \end{aligned} \quad (3.18)$$

where F_f is the volumetric flux at the face:

$$F_f = A_f \cdot u_f \quad (3.19)$$

The calculation of the volumetric flux F is described in Section 3.3.3.

In order to approximate the time integral of the convection term an assumption about the variation of ϕ and u with time needs to be made (Versteeg & Malalasekera (1995)). It is possible to use any value of ϕ and u between t and $t+\delta t$. A general approach is to introduce a weighting parameter η having a value between zero and one (Versteeg & Malalasekera (1995)) to obtain:

$$\int_t^{t+\delta t} \left(\sum_{f=1}^n \alpha_f F_f \right) dt = \sum_{f=1}^n \left((1-\eta)[\alpha_f F_f]^t + \eta[\alpha_f F_f]^{t+\delta t} \right) \delta t \quad (3.20)$$

The most popular choices of η are $\eta=0$ for the explicit, $\eta=1$ for the Euler implicit or $\eta=1/2$ for the Crank-Nicolson schemes.

Numerical results presented in Section 5.3 show that the implicit scheme applied to the convection equation introduces numerical diffusion in the direction of the flow and the explicit scheme introduces numerical diffusion normal to the flow direction. The Crank-Nicolson scheme which uses a linear variation in the face values over time is second order accurate in time (Fletcher (1991)) and introduces no numerical diffusion.

The Crank-Nicolson scheme as defined above is more expensive in terms of computer storage because it needs both the old and new time level's values for the volumetric flux F at the faces. However this can be overcome, because for a small enough time step the variation of F is negligible in comparison with the larger variation of α and therefore it is reasonable to use only the most recent value of F .

Substitution of eq. (3.17) and eq. (3.20) together with the above assumptions on the variation of the volumetric flux F back into eq. (3.16) yields:

$$\begin{aligned}
(\alpha_P^{t+\delta t} - \alpha_P^t) V_P &= - \sum_{f=1}^n \frac{1}{2} (\alpha_f^t + \alpha_f^{t+\delta t}) F_f \delta t \\
&= - \sum_{f=1}^n \alpha_f^* F_f \delta t
\end{aligned} \tag{3.21}$$

where

$$\alpha_f^* = \frac{1}{2} (\alpha_f^t + \alpha_f^{t+\delta t}) \tag{3.22}$$

Equation (3.21) can be arranged by separating the α values of the old and new time levels:

$$\alpha_P^{t+\delta t} \frac{V_P}{\delta t} + \sum_{f=1}^n \frac{1}{2} \alpha_f^{t+\delta t} F_f = S_{\alpha_P} \tag{3.23}$$

where S_{α_P} is the source term defined as:

$$S_{\alpha_P} = \alpha_P^t \frac{V_P}{\delta t} - \sum_{f=1}^n \frac{1}{2} \alpha_f^t F_f \tag{3.24}$$

Equation (3.23) uses the value for α in the centre of the control volume as well as at the faces of the control volume. For a co-located arrangement, the cell centre values are used to interpolate the values on the faces. Linear interpolation (eq. (3.8)) which is equivalent to central differencing is not suitable because a bounded solution of eq. (3.21) is not guaranteed (Hirsh (1990)). Other available differencing schemes which guarantee boundedness are known to be too diffusive (Davis (1994)).

A new compressive differencing scheme which guarantees a bounded solution whilst maintaining the sharpness of the interface has been developed in this work and is presented in the next chapter. This compressive differencing scheme supplies a weighting factor β which expresses the α face value in terms of a cell and its nearest neighbours:

$$\alpha_f = \beta \alpha_p + (1 - \beta) \alpha_N \quad (3.25)$$

This enables eq. (3.23) to be reformulated in terms of a cell and its nearest neighbours:

$$a_p \alpha_p^{t+\delta t} = \sum_{nb=1}^n a_{nb} \alpha_{nb}^{t+\delta t} + S_{\alpha_p} \quad (3.26)$$

where the subscript *nb* denotes the nearest neighbours.

Equation (3.26) represents the finite volume discretisation of the indicator function in a form suitable for a direct solver (see for instance Patankar (1980) and Versteeg & Malalasekera (1995)).

3.3.2 Momentum equation

The finite volume discretisation is based on the integral form of eq. (2.15) over the control volume and the time step δt :

$$\begin{aligned} & \int_t^{t+\delta t} \left(\int_V \frac{\partial \rho \mathbf{u}}{\partial t} dV \right) dt + \int_t^{t+\delta t} \left(\int_V \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) dV \right) dt - \int_t^{t+\delta t} \left(\int_V \nabla \cdot (\mu \nabla \otimes \mathbf{u}) dV \right) dt \\ &= - \int_t^{t+\delta t} \left(\int_V \nabla P dV \right) dt + \int_t^{t+\delta t} \left(\int_V \rho g dV \right) dt \quad (3.27) \\ &+ \int_t^{t+\delta t} \left(\int_V \sigma \kappa \nabla \alpha dV \right) dt + \int_t^{t+\delta t} \left(\int_V (\nabla \otimes \mathbf{u}) \cdot (\nabla \mu) dV \right) dt \end{aligned}$$

Equation (3.27) represents the conservation of a vector quantity, which can be broken into the conservation of each of its components, thus in a three-dimensional space three separate conservation equations are obtained, one for each of the coordinate directions.

The discretisation of the momentum equations is similar to the discretisation of the α equation given in Section 3.3.1 and is presented in several other works, see for instance Patankar (1980), Perić (1985) and Versteeg & Malalasekera (1995). The partially discretised form of the momentum equations in terms of a cell and its face values will therefore be given without further proof. The important issues that will be addressed here are: (i) the prediction of the face values in terms of cell centre values prevailing at the two cells sharing the same face, (ii) the treatment of mesh non-orthogonality and (iii) consistent discretisation.

The partially discretised form of the momentum equations is:

$$\int_t^{t+\delta t} \left(\left(\frac{\partial \rho \mathbf{u}}{\partial t} \right)_P V_P + \sum_{f=1}^n \rho_f F_f \mathbf{u}_f - \sum_{f=1}^n \mu_f A_f \cdot (\nabla \otimes \mathbf{u})_f \right) dt = \int_t^{t+\delta t} \left(S_{\mathbf{u}_P} V_P \right) dt \quad (3.28)$$

where $S_{\mathbf{u}_P}$ is the source term defined as:

$$S_{\mathbf{u}_P} = -(\nabla P)_P + g \rho_P + \sigma \kappa_P (\nabla \alpha)_P + (\nabla \otimes \mathbf{u})_P \cdot (\nabla \mu)_P \quad (3.29)$$

The terms of interest in the above equations are the convection, the diffusion and the κ curvature terms. These terms are discussed in the following sections. The gradients over a cell of the different flow properties can be calculated according to equations (3.14) and (3.13). The implementation of the other terms is straightforward and needs no further discussion.

3.3.2.1 Convection term

The finite volume discretisation of the convection term reduces to:

$$\int_V \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) dV \approx \sum_{f=1}^n \rho_f F_f \mathbf{u}_f \quad (3.30)$$

where F_f is the volumetric flux defined in eq. (3.19).

Here consistent discretisation plays an important role. The product $\rho_f F_f$ represent the mass flux at the face. The approximation of the densities at the faces should be done in such a way that mass is also conserved. The continuity equation (2.4) for variable density flow deals with the conservation of mass, but has been substituted with the α equation (2.14) and the continuity equation for incompressible flow (2.13), which has already been used for the calculation of the conservative volumetric fluxes. The discretised α equation (3.21) together with the constitutive relation for the density (eq. (2.7)) are then used for the approximation of the density at the faces:

$$\rho_f = \alpha_f^* \rho_1 + (1 - \alpha_f^*) \rho_2 \quad (3.31)$$

This definition of the face densities guarantees conservative mass fluxes at the faces.

To approximate the velocity at the face it is necessary to use an accurate and bounded differencing scheme. The widely used upwind differencing scheme,

$$u_f = \begin{cases} u_P & \text{if } F_f \geq 0 \\ u_N & \text{if } F_f < 0 \end{cases} \quad (3.32)$$

complies with the boundedness criteria. However, it introduces a large amount of numerical diffusion on the momentum. Therefore a more accurate prediction of the velocity at the faces is needed.

The use of central differencing is more accurate, but introduces unboundedness on the momentum, which manifests itself in a spurious velocity field. Perić (1985) introduces a blended differencing scheme which attempts to preserve both boundedness and the accuracy of the solution. This is achieved by introducing a linear weighting between upwind differencing (UD) eq. (3.32) and central differencing (CD) eq. (3.8):

$$u_f = \begin{cases} (1-\lambda)u_P + \lambda(\mathcal{L}_P u_P + (1-\mathcal{L}_P)u_N) & \text{if } F_f \geq 0 \\ (1-\lambda)u_N + \lambda(\mathcal{L}_P u_P + (1-\mathcal{L}_P)u_N) & \text{if } F_f < 0 \end{cases} \quad (3.33)$$

where $0 \leq \lambda \leq 1$ is the blending introduced to reduce the numerical diffusion. Perić (1985) proposes a constant λ for all the faces.

A more recent technique used here, which calculates a blending factor for each face is the centered differencing scheme proposed by Jasak *et al.* (1996). It is a High Resolution (HR) convection-diffusion differencing scheme based on the Normalised Variable Diagram (NVD) (Leonard (1991)), specially developed for unstructured meshes and will be used in this study. For completeness the equations needed for the calculation of the velocity face values are given without further proof.

Let u be the magnitude of the velocity vector \mathbf{u} , then the face value for this component is defined as:

$$u_f = \begin{cases} u_D & \text{for } \tilde{u}_D \leq 0 \text{ or } \tilde{u}_D \geq 1 \\ \frac{1}{2}(u_D + u_A) & \text{for } k \leq \tilde{u}_D < 1 \\ \left(1 - \frac{\tilde{u}_D}{2k}\right)u_D + \frac{\tilde{u}_D}{2k}u_A & \text{for } 0 < \tilde{u}_D < k \end{cases} \quad (3.34)$$

where k is a prescribed constant with a value between 0 and 0.5. The recommended value is $k=0.1$. The subscripts D and A are determined by the direction of the flow:

$$F_f \geq 0 \rightarrow D=P \text{ & } A=N \text{ or } F_f < 0 \rightarrow D=N \text{ & } A=P \quad (3.35)$$

The decision factor \tilde{u}_D is defined as:

$$\tilde{u}_D = 1 - \frac{u_A - u_D}{2(\nabla u)_D \cdot \mathbf{d}} \quad (3.36)$$

where $(\nabla u)_D$ is calculated according to eq. (3.13):

$$(\nabla u)_D \approx \frac{1}{V_D} \sum_{f=1}^n A_f u_f \quad (3.37)$$

3.3.2.2 Diffusion term

The finite volume discretisation of the diffusion term reduces to:

$$\int_V \nabla \cdot (\mu \nabla \otimes u) dV \approx \sum_{f=1}^n \mu_f A_f \cdot (\nabla \otimes u)_f \quad (3.38)$$

where the dynamic viscosity at the face μ_f is calculated according to eq. (3.8):

$$\mu_f = \mathcal{Q}_P \mu_P + (1 - \mathcal{Q}_P) \mu_N \quad (3.39)$$

The calculation of $(\nabla \otimes u)_f$ can also be achieved by applying eq. (3.8):

$$(\nabla \otimes u)_f = \mathcal{Q}_P (\nabla \otimes u)_P + (1 - \mathcal{Q}_P) (\nabla \otimes u)_N \quad (3.40)$$

where eq. (3.14) is used to derive an expression for the gradient over the cell:

$$(\nabla \otimes u)_P \approx \frac{1}{V_P} \sum_{f=1}^n A_f \otimes u_f \quad (3.41)$$

The computational molecule resulting from this approach is no longer the compact molecule consisting of a cell and its nearest neighbours, as the skew and far neighbours are now used to calculate the gradient over the neighbouring cell. The use of a larger computational molecule presents complications as it is difficult to treat the contributions of all these extra neighbours implicitly. Therefore, if the mesh is orthogonal, it is more appropriate to calculate the gradient at the face directly, using the nodal values straddling the face. According to eq. (3.1), $A_f = D_f$ for an orthogonal mesh and therefore it is

possible to use the following expression for the dot product between the face area vector and the velocity gradient at the face:

$$\mathbf{A}_f \cdot (\nabla \otimes \mathbf{u})_f = \mathbf{D}_f \cdot (\nabla \otimes \mathbf{u})_f = |\mathbf{D}_f| \frac{\mathbf{u}_N - \mathbf{u}_P}{|\mathbf{d}_f|} \quad (3.42)$$

However, if the mesh is non-orthogonal it is necessary to compensate for the non-orthogonal contribution. This is done by using $\mathbf{A}_f = \mathbf{D}_f + \mathbf{k}_f$, the over relaxed non-orthogonal decomposition of the face area vector, see Figure 3.1:

$$\mathbf{A}_f \cdot (\nabla \otimes \mathbf{u})_f = \mathbf{D}_f \cdot (\nabla \otimes \mathbf{u})_f + \mathbf{k}_f \cdot (\nabla \otimes \mathbf{u})_f \quad (3.43)$$

Orthogonal contribution *non-orthogonal correction*

Equation (3.42) is used for the prediction of the main (or orthogonal) contribution and eq. (3.40) is used to calculate the gradient at the face for the non-orthogonal correction. In this case the inclusion of the skew and far neighbours is inevitable. This non-orthogonal correction will be treated explicitly.

3.3.2.3 Curvature

The term containing the surface tension in the momentum equations is:

$$\int_V \sigma \kappa \nabla \alpha \, dV = \sigma \kappa_P (\nabla \alpha)_P V_P \quad (3.44)$$

One consequence of the CSF model, outlined in paragraph 2.4, is the demand for a smooth κ_P throughout the whole transitional area, that is everywhere where $(\nabla \alpha)_P \neq 0$ (Brackbill *et al.* (1992)).

The gradient $(\nabla \alpha)_P$ follows directly from eq. (3.14):

$$(\nabla \alpha)_P \approx \frac{1}{V_P} \sum_{f=1}^n A_f \alpha_f \quad (3.45)$$

where central differencing (eq. (3.8)) is used to derive an expression for α_f :

$$\alpha_f = \mathcal{Q}_P \alpha_P + (1 - \mathcal{Q}_P) \alpha_N \quad (3.46)$$

The use of equations (3.45) and (3.46) gives rise to non-zero values for $(\nabla \alpha)_P$ in the interface cell and its nearest neighbours. Thus, the CSF model demands a smooth curvature value for at least the interface cell and its nearest neighbours.

Equation (2.18) defines the curvature as divergence of the unit normal to the interface which is calculated directly from the gradient of the α indicator function:

$$\kappa_P = -\left(\nabla \cdot \left(\frac{\nabla \alpha}{|\nabla \alpha|} \right) \right)_P = \frac{-1}{V_P} \sum_{f=1}^n A_f \cdot \left(\frac{(\nabla \alpha)_f}{|\nabla \alpha|_f} \right) \quad (3.47)$$

The gradient over the face $(\nabla \alpha)_f$ is obtained by means of eq. (3.8):

$$(\nabla \alpha)_f = \mathcal{Q}_P (\nabla \alpha)_P + (1 - \mathcal{Q}_P) (\nabla \alpha)_N \quad (3.48)$$

where $(\nabla \alpha)_P$ is calculated with eq. (3.45).

The volume fraction which represents the indicator function on a discretised mesh is not at all a smooth function and needs to be smoothed for the calculation of the curvature (Unverdi & Tryggvason (1992), Brackbill *et al.* (1992), Lafaurie *et al.* (1994), Sussman *et al.* (1994)). The need therefore is illustrated with the aid of Figure 3.2. Panel (A) shows the original indicator field of a circle with radius R and panel (C) shows the curvature times R calculated from this original field. Figure 3.2 (B) shows the indicator field after smoothing has been applied and panel (D) shows the curvature calculated from this field times the drop radius. The errors made with the prediction of the droplet radius using the original non-smooth field is as high as 100%.

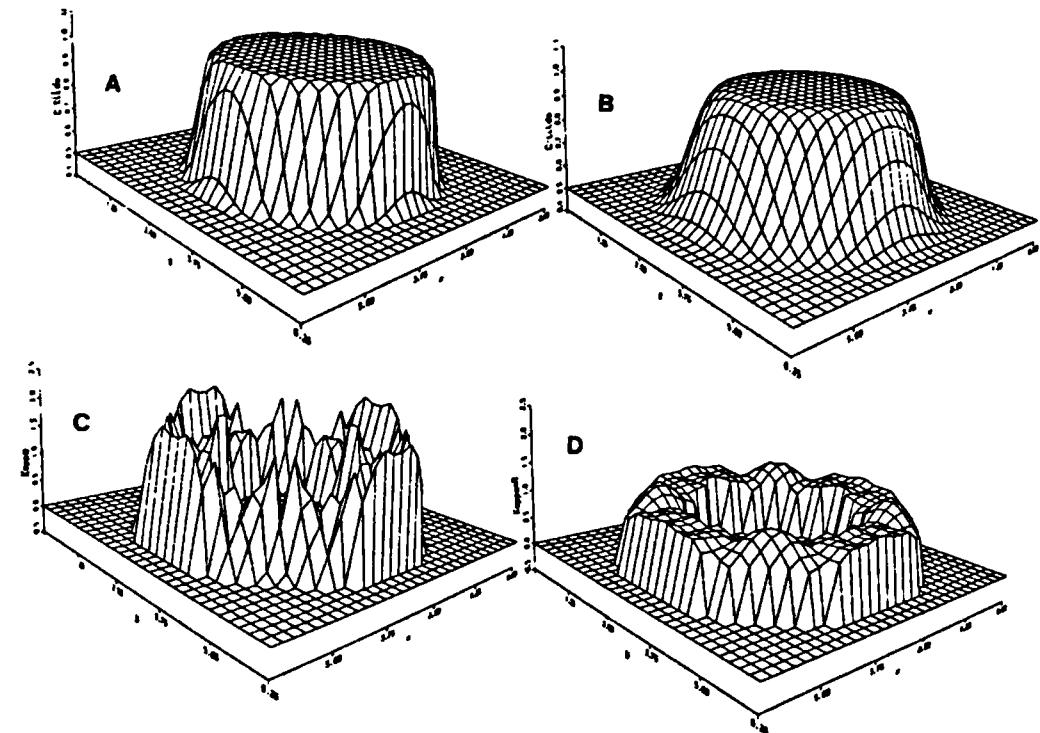


Figure 3.2: The smoothness of the curvature (from: Brackbill *et al.*, 1992, J. Comput. Phys., Vol. 100, p. 344, Fig. 3.)

A smoothly varying indicator function $\bar{\alpha}$ (Brackbill *et al.* (1992), Lafaurie *et al.* (1994)) can be constructed from:

$$\bar{\alpha}(x) = \int_V \alpha(x') K(x-x';\varepsilon) dx' \quad (3.49)$$

where $K(x;\varepsilon)$ is a smooth integration kernel and ε is a function of the distance between mesh points.

Brackbill *et al.* (1992) use $\mathcal{E}^{(l)}(|x'-x|;h)$, a B-spline of degree l , where $\mathcal{E}^{(l)} \neq 0$ only for $|x'-x| \leq 3/2 h$ and h is of the order of the distance between two mesh points. This implies that α is smoothed by taking into account the values of the direct and skew neighbours of a cell. A schematic representation of this approach is given in Figure 3.3 (a).

Lafaurie *et al.* (1994) use a regular mesh and define a Laplacian filter \mathcal{F} that transforms α into $\mathcal{F}(\alpha)$. On a discrete mesh it is defined as:

$$\mathcal{F}(\alpha)_P = \sum_{f=1}^n \frac{\alpha_f}{n} \quad (3.50)$$

The application of the filter is repeated for m times yielding the smoothed volume fraction field $\bar{\alpha} = \mathcal{F}^{(m)}(\alpha)$. The recommended value for the amount of filtering is $m = 2$. This implies that the direct as well as skew and far neighbours of a cell are used for the smoothing. A schematic representation of this approach is given in Figure 3.3 (b).

With the current face addressing implementation the only information available for a cell is a reference list of its direct neighbours. Therefore the smoothing technique of Brackbill *et al.* (1992) cannot be implemented easily. The technique used by Lafaurie *et al.* (1994) however, can easily be adapted for the current study by defining the Laplacian filter \mathcal{F} as:

$$\mathcal{F}(\alpha)_P = \sum_{f=1}^n \alpha_f |A_f| \Bigg/ \sum_{f=1}^n |A_f| \quad (3.51)$$

where central differencing (eq. (3.8)) is used to derive an expression for α_f :

$$\alpha_f = \mathcal{L}_P \alpha_P + (1 - \mathcal{L}_P) \alpha_N \quad (3.52)$$

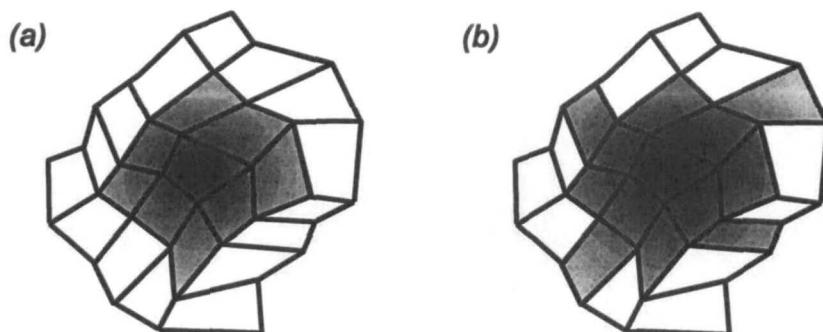


Figure 3.3: Smoothing techniques applied by (a) Brackbill *et al.* (1992) and (b) Lafaurie *et al.* (1994)

3.3.2.4 Temporal discretisation

The temporal discretisation of the discretised form of the momentum equations (3.28) is similar to the temporal discretisation of the α equation and can also be found in Versteeg & Malalasekera (1995). In order to be consistent with the discretisation of the α equation, the Crank-Nicolson differencing scheme should also be utilised for the temporal discretisation of the momentum equations. As noted before, the variation of the velocity is small in comparison to the variation of the density. Therefore it is sufficient to apply a first order differencing scheme. The Euler implicit method is used in the present study for its stability features.

The final form of the momentum equations discretised with the Euler implicit method for a non-orthogonal mesh reduces to:

$$\frac{(\rho \mathbf{u})_P^{t+\delta t}}{\delta t} + \frac{1}{V_P} \sum_{f=1}^n \rho_f F_f \mathbf{u}_f^{t+\delta t} + \frac{1}{V_P} \sum_{f=1}^n \mu_f \mathbf{D}_f \cdot (\nabla \otimes \mathbf{u})_f^{t+\delta t} = S_{\mathbf{u}_P} - (\nabla P)_P \quad (3.53)$$

where $S_{\mathbf{u}_P}$ is the source term defined as:

$$S_{\mathbf{u}_P} = \frac{(\rho \mathbf{u})_P^t}{\delta t} + g \rho_P + \sigma \kappa_P (\nabla \alpha)_P + (\nabla \otimes \mathbf{u})_P \cdot (\nabla \mu)_P + \frac{1}{V_P} \sum_{f=1}^n \mu_f k_f (\nabla \otimes \mathbf{u})_f \quad (3.54)$$

With this formulation the contribution of the non-orthogonal correction and the gradient of the viscosity at the interface are lagged in time. If necessary, an iteration over the explicit non-orthogonal terms of eq. (3.53) can be done. This iteration forms part of the overall solution algorithm and is addressed in Section 3.5.1.

All the face velocities on the left hand side of eq. (3.53) are in terms of a cell and its nearest neighbours and it is possible to reformulate the above equations in terms of these values:

$$a_P \mathbf{u}_P^{t+\delta t} = \sum_{nb=1}^n a_{nb} \mathbf{u}_{nb}^{t+\delta t} + S_{\mathbf{u}_P} - (\nabla P)_P \quad (3.55)$$

where the subscript nb denotes the nearest neighbours.

It is useful to present the discretised momentum equations (3.55) as:

$$\mathbf{u}_P^{t+\delta t} = \frac{\mathbf{H}(\mathbf{u})_P}{a_{P_P}} - \frac{1}{a_{P_P}} (\nabla P)_P \quad (3.56)$$

where $a_{P_P} \equiv a_P$ and:

$$\mathbf{H}(\mathbf{u})_P = \sum_{nb=1}^n a_{nb} \mathbf{u}_{nb}^{t+\delta t} + S_{\mathbf{u}_P} \quad (3.57)$$

This formulation of the discretised momentum equations will be used for the derivation of the pressure equation, which follows in the next section.

3.3.3 Pressure equation

The pressure equation is derived from the incompressibility condition eq. (2.13) and the discretised momentum equation (3.56). The new time level velocity is used for the derivation of the pressure equation and for the simplicity of the notation it is appropriate to drop the superscript $(t+\delta t)$ in eq. (3.56) for the time being.

The finite volume discretisation of the incompressibility condition is based on the integral form of eq. (2.13) over the control volume:

$$\oint_V \nabla \cdot \mathbf{u} dV = 0 \quad (3.58)$$

Gauss' theorem (eq. (3.6)) can be applied to eq. (3.58) to give:

$$\sum_{f=1}^n \mathbf{A}_f \cdot \mathbf{u}_f = 0 \quad (3.59)$$

In the co-located variable arrangement, which is used in the present study, special care needs to be taken in the prediction of \mathbf{u}_f in order to avoid the decoupling of the velocity and pressure (Prakash (1981)).

The face interpolation proposed by Rhie and Chow (1983) is adopted in the present study. The discretised momentum equation (3.56) is used to predict the face value of the velocity. This is done by isolating the contribution of the pressure from eq. (3.56) when interpolating it to the face. The contribution of the pressure gradient at the face is then added explicitly to \mathbf{u}_f by calculating it directly from the pressure values at the nodes sharing the face. The face value of the velocity is therefore defined as:

$$\mathbf{u}_f = \left(\frac{\mathbf{H}(\mathbf{u})}{a_p} \right)_f - \left(\frac{1}{a_p} \right)_f (\nabla P)_f \quad (3.60)$$

where the face values other than the pressure gradient are calculated by using linear interpolation (eq. (3.8)):

$$\left(\frac{\mathbf{H}(\mathbf{u})}{a_p} \right)_f = \mathcal{Q}_P \left(\frac{\mathbf{H}(\mathbf{u})}{a_p} \right)_P + (1 - \mathcal{Q}_P) \left(\frac{\mathbf{H}(\mathbf{u})}{a_p} \right)_N \quad (3.61)$$

$$\left(\frac{1}{a_p} \right)_f = \mathcal{Q}_P \left(\frac{1}{a_p} \right)_P + (1 - \mathcal{Q}_P) \left(\frac{1}{a_p} \right)_N \quad (3.62)$$

Substitution of \mathbf{u}_f from eq. (3.60) into eq. (3.59) gives:

$$\sum_{f=1}^n \left(\frac{1}{a_p} \right)_f \mathbf{A}_f \cdot (\nabla P)_f = \sum_{f=1}^n \mathbf{A}_f \cdot \left(\frac{\mathbf{H}(\mathbf{u})}{a_p} \right)_f \quad (3.63)$$

The term containing the pressure gradient over the faces is calculated similarly to the diffusion term of the momentum equation (Section 3.3.2.2) by taking into account the

orthogonal and non-orthogonal contributions of the neighbours. This is done by using the over relaxed non-orthogonal decomposition of the face area vector, see Figure 3.1:

$$\mathbf{A}_f \cdot (\nabla P)_f = \mathbf{D}_f \cdot (\nabla P)_f + \mathbf{k}_f \cdot (\nabla P)_f \quad (3.64)$$

Orthogonal contribution *non-orthogonal correction*

The orthogonal contribution is given by:

$$\mathbf{D}_f \cdot (\nabla P)_f = |\mathbf{D}_f| \frac{\mathbf{P}_N - \mathbf{P}_P}{|\mathbf{d}_f|} \quad (3.65)$$

and the non-orthogonal correction is given by:

$$\mathbf{k}_f \cdot (\nabla P)_f = \mathbf{k}_f \cdot \left(\mathcal{Q}_P \left(\frac{1}{V_P} \sum_{f=1}^n \mathbf{A}_f \mathbf{P}_f \right)_P + (1 - \mathcal{Q}_P) \left(\frac{1}{V_N} \sum_{f=1}^n \mathbf{A}_f \mathbf{P}_f \right)_N \right) \quad (3.66)$$

This non-orthogonal correction is treated explicitly by adding it to the source term to give the required pressure equation:

$$\sum_{f=1}^n \left(\frac{1}{a_p} \right)_f \mathbf{D}_f \cdot (\nabla P)_f = S_p \quad (3.67)$$

where

$$S_p = \sum_{f=1}^n \mathbf{A}_f \cdot \left(\frac{\mathbf{H}(\mathbf{u})}{a_p} \right)_f - \sum_{f=1}^n \left(\frac{1}{a_p} \right)_f \mathbf{k}_f \cdot (\nabla P)_f \quad (3.68)$$

The pressures in eq. (3.67) are now defined in terms of a cell and its nearest neighbours and it is possible to reformulate the pressure equation in terms of these values as follows:

$$a_p P_p = \sum_{nb=1}^n a_{nb} P_{nb} + S_p \quad (3.69)$$

where the subscript nb denotes the nearest neighbours.

Further, the velocity at the face given by eq. (3.60) is also used for the calculation of the conservative volumetric fluxes, first defined in eq. (3.19):

$$F_f = \mathbf{A}_f \cdot \left(\left(\frac{H(\mathbf{u})}{a_p} \right)_f - \left(\frac{1}{a_p} \right)_f (\nabla P)_f \right) \quad (3.70)$$

This concludes the discretisation of the governing equations for the interior of the flow domain. The discretisation of the pressure equation, the α equation and the convection and diffusion terms of the momentum equations at boundaries is given in the next Section. The numerical treatment of wall adhesion is also treated in the same section. The solution algorithm for the discretised equations is then presented in Section 3.5.

3.4 Boundary conditions

This section presents the discrete implementation of the boundary conditions defined in Section 2.5. Figure 3.4 gives a planar view of an arbitrary control volume with a

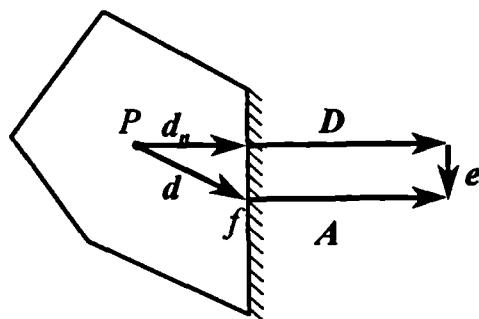


Figure 3.4: Plane view of a control volume with a boundary face

boundary face. The outward-pointing face area vector A is at the centre of the face and d is the vector from the cell centre to the centre of the boundary face. The vector d_n is the

vector from the centre of the cell normal to the boundary face and reduces with a simple algebraic manipulation to:

$$\mathbf{d}_n = \frac{\mathbf{d} \cdot \mathbf{A}}{|\mathbf{A}|^2} \mathbf{A} \quad (3.71)$$

It is assumed that the specified boundary condition is valid along the whole face and therefore it is reasonable to assume that $\mathbf{A} = \mathbf{D}$.

3.4.1 Fixed value boundary conditions

The fixed value boundary condition prescribes the value of a flow property ϕ at the boundary face to be ϕ_b . This needs to be taken into account with the discretisation of the equations where the value at the face is used directly in the implicit implementation.

- **Pressure equation:** According to eq. (3.67)

$$\left(\frac{1}{a_p} \right)_f \mathbf{D}_f \cdot (\nabla P)_f \quad (3.72)$$

needs to be calculated at the boundary face.

A neighbouring value a_{p_N} does not exist and therefore the value at the face will be approximated with the value of the boundary cell. The fixed boundary value P_b is used for the approximation of the face gradient $(\nabla P)_f$. Similar to eq. (3.65) the face gradient can be approximated by:

$$\mathbf{D}_f \cdot (\nabla P)_f = |\mathbf{D}_f| \frac{(P_b - P_p)}{|\mathbf{d}_n|} \quad (3.73)$$

With these assumptions the approximation of (3.72) at the boundary face reduces to:

$$\begin{aligned}
\left(\frac{1}{a_p} \right)_f \mathbf{D}_f \cdot (\nabla P)_f &= \frac{1}{a_p} |\mathbf{D}_f| \frac{(P_b - P_p)}{|\mathbf{d}_n|} \\
&= \frac{|\mathbf{D}_f|}{a_{P_p} |\mathbf{d}_n|} P_b - \frac{|\mathbf{D}_f|}{a_{P_p} |\mathbf{d}_n|} P_p
\end{aligned} \tag{3.74}$$

The part containing P_p goes into the central coefficient and the part containing P_b goes into the source term.

- **Diffusion term:** The term that needs to be evaluated for the diffusion term at the boundary face follows from equations (3.38) and (3.42):

$$\mu_f \mathbf{D}_f \cdot (\nabla \otimes \mathbf{u})_f \tag{3.75}$$

Similar to the pressure equation described above, the approximation of (3.75) reduces to:

$$\mu_f \mathbf{D}_f \cdot (\nabla \otimes \mathbf{u})_f = \frac{\mu_f |\mathbf{D}_f|}{|\mathbf{d}_n|} \mathbf{u}_b - \frac{\mu_f |\mathbf{D}_f|}{|\mathbf{d}_n|} \mathbf{u}_p \tag{3.76}$$

where μ_f is calculated from the indicator function value at the face. Once again, the part containing \mathbf{u}_p goes into the central coefficient and the part containing \mathbf{u}_b goes into the source term.

- **Convection term:** Both the α equation and the momentum equations contain a convection term. The derivation will therefore be given for a general flow quantity ϕ which can be either a vector or a scalar. The finite volume discretisation of the convection term reduces to:

$$\sum_{f=1}^n b_f F_f \phi_f \tag{3.77}$$

where $b_f = 1$ for the α equation and $b_f = \rho_f$ for the momentum equations. The value of ϕ at the boundary face is given as ϕ_b . The value of the whole term at the boundary face, given by:

$$b_f F_f \phi_b \quad (3.78)$$

is constant and therefore forms part of the source term.

3.4.2 Fixed gradient boundary conditions

In the case of a fixed gradient boundary condition, the gradient of the flow property over the face is specified, thus

$$\frac{\mathbf{d}_n}{|\mathbf{d}_n|} \cdot (\nabla\phi)_f = (\nabla\phi)_b \quad (3.79)$$

where $(\nabla\phi)_b$ is the specified gradient. For a zero gradient boundary condition $(\nabla\phi)_b = 0$.

Similar to the fixed value boundary condition, the fixed gradient boundary condition needs to be taken into account where the value at the face is used in the implicit implementation.

- **Diffusion term:** The gradient at the face is specified and \mathbf{D}_f is normal to the face. Therefore the term containing the gradient at the boundary face in (3.75) reduces to:

$$\mu_f \mathbf{D}_f \cdot (\nabla \otimes \mathbf{u})_f = \mu_f \mathbf{D}_f \cdot (\nabla \otimes \mathbf{u})_b \quad (3.80)$$

where $(\nabla \otimes \mathbf{u})_b$ is the specified value at the boundary. This term forms part of the source term.

- **Convection term:** In the same manner as the fixed value boundary conditions, the fixed gradient boundary condition will only be given for a general flow quantity ϕ which can be either a vector or a scalar.

The specified gradient $(\nabla\phi)_b$ at the boundary is used to get an approximation for the face value:

$$\phi_f = \phi_P + d_n \cdot (\nabla\phi)_b \quad (3.81)$$

The convection term at the boundary reduces to:

$$b_f F_f \phi_f = b_f F_f \phi_P + b_f F_f d_n \cdot (\nabla\phi)_b \quad (3.82)$$

The term containing ϕ_P forms part of the central coefficient while the remainder is treated as part of the source term.

3.4.3 Wall adhesion

Wall adhesion has been described in paragraph 2.5 and the task is now to determine the normal vector to the interface in a boundary cell. Equation (3.45) defines the normal to the interface as:

$$(\nabla\alpha)_P \approx \frac{1}{V_P} \sum_{f=1}^n A_f \alpha_f \quad (3.83)$$

In order to calculate the above gradient Brackbill *et al.* (1992) propose a zero gradient boundary condition on α . Thus, according to eq. (3.81), the indicator function at the boundary face reduces to the indicator function value of the cell centre value, $\alpha_f = \alpha_P$.

Equation (3.83) is now used to calculate a vector normal to the interface in the boundary cell which also takes into account wall adhesion. This normal vector is defined by Brackbill *et al.* (1992) and Kothe & Mjolsness (1992):

$$\hat{\mathbf{n}} = \frac{\mathbf{d}_n}{|\mathbf{d}_n|} \cos\theta_{eq} + \frac{(\nabla\alpha)_P}{|(\nabla\alpha)_P|} \sin\theta_{eq} \quad (3.84)$$

where θ_{eq} is the static contact angle. The default value in the numerical model is taken as $\theta_{eq} = 90^\circ$. This means the interface is normal to the wall.

This concludes the treatment of the boundary conditions for the relevant equations. The solution algorithm for the two-fluid system will now be presented in the next section.

3.5 Solution algorithm

The equations of motion for the two-fluid system are closely coupled with each other and the ideal way is to solve them simultaneously for the whole flow domain (Caretto *et al.* (1972) and Vanka (1986)). However, this approach takes up a huge amount of computer resources, because the resulting matrix to be solved is several times larger than the number of computational points.

Another approach to satisfy the equations of motion simultaneously is the so called segregated approach, where the different equations are solved sequentially by iterating over them. The velocity and pressure have a strongly linear coupling which has been a research topic for several years (Patankar & Spalding (1972), Raithby & Schneider (1979), Issa (1986)). The PISO (Pressure Implicit with Splitting of Operators) algorithm of Issa (1986) is a pressure-velocity algorithm specially developed for the non-iterative computation of unsteady compressible flows and is used in the present study. The PISO algorithm is widely in use and good descriptions of its implementation can also be found in Versteeg & Malalasekera (1995) and Ferziger & Perić (1996).

The implementation of the PISO algorithm is described first, followed by a presentation of the total solution algorithm for the two-fluid system.

3.5.1 Pressure-velocity coupling with PISO

The PISO algorithm can be described as follows:

- **Momentum prediction:** The momentum equations (3.55) are solved firstly with a guessed pressure field P^* , normally the pressure field of the previous time step. The solution of the momentum equations gives a new velocity field \mathbf{u}^* which does not satisfy the continuity condition.
- **Pressure solution:** These predicted velocities \mathbf{u}^* are used to assemble $\mathbf{H}(\mathbf{u}^*)$ (eq. (3.57)) which is needed for the pressure equation (3.69). The solution of the pressure equation gives rise to a new pressure field P^{**} . If the mesh is non-orthogonal then the source term eq. (3.68) is updated and a more accurate solution of the pressure equation is obtained. If wanted, an iteration over the explicit non-orthogonal terms can be repeated until a pre-defined tolerance is met.
- **Explicit velocity correction:** Equation (3.70) gives a new set of conservative volumetric fluxes consistent with the new pressure field. The new pressure field is used in eq. (3.56) to do an explicit correction on velocity. The new velocity \mathbf{u}^{**} is now consistent with the new pressure field.

The velocity in a cell is given by eq. (3.56) which means that the velocity not only depends on the pressure gradient but also on $\mathbf{H}(\mathbf{u})$, which includes contribution from the neighbouring cells. The PISO algorithm uses \mathbf{u}^{**} to calculate $\mathbf{H}(\mathbf{u}^{**})$. This gives rise to P^{***} which in turn is used to calculate \mathbf{u}^{***} . This iteration over the last two steps continues until a pre-defined tolerance is met.

3.5.2 Solution procedure for the two-fluid system

All the equations have been discretised and the coupling of the velocity and pressure has been addressed. It is now possible to describe the solution sequence of the two-fluid system.

For the prediction of the momentum, eq. (3.53) the new densities, together with the face densities eq. (3.31), satisfying mass continuity is needed. These are obtained by solving the α equation (3.23) first, using the old volumetric fluxes. The time step enforced by the compressive differencing scheme (see next chapter) is small enough to allow neglect of the change in the density field due to the new volumetric fluxes generated by the PISO algorithm, as is the case with other variable density flows.

The solution sequence is:

1. Initialise all the variables.
2. Calculate the Courant number and adjust the time step if necessary.
3. Solve the α equation by using the old time level's volumetric fluxes.
4. Use the new α values together with the constitutive relations to obtain an estimate for the new viscosity, density and the face densities.
5. Use the above values to do a momentum prediction and continue with the PISO algorithm.
6. If the final time has not yet been reached advance to the next time level and return to step 2.

3.6 Conclusions

The finite volume discretisation of the governing equations for two-fluid systems has been presented in this chapter. It has been decided to discretise the flow domain with control volumes of arbitrary topology. The indicator function defined in the previous chapter has been approximated with a volume fraction field. This volume fraction field

gives the transitional area between the two fluids a finite thickness, but it is not at all a smooth field. The calculation of the interface curvature requires a smooth transitional area, hence the introduction of a smoothing technique, which is involved only for the purpose of computing this curvature.

A compressive differencing scheme which can capture this transitional area over one control volume is still needed for the indicator function. Such a differencing scheme, also applicable to arbitrary meshes, has been developed as part of the present study and will be presented in the next chapter.

4 Compressive differencing scheme

4.1 Introduction

This chapter deals with the derivation of a new compressive differencing scheme for the discretisation of the volume fraction equation which enables the capturing of sharp fluid/fluid interfaces. Compressive differencing schemes are suitable for this purpose, because they prevent the smearing of a step profile by removing numerical diffusion that may occur.

As outlined in the first chapter, several problems and limitations exist with current differencing schemes applied to interface capturing. They are either bounded but too diffusive, or they create a non-physical deformation of the interface (examples are shown in Section 1.2.2, page 35). Furthermore, they are limited to rectangular meshes with structured cell arrangements. The new scheme developed here, named CICSAM (Compressive Interface Capturing Scheme for Arbitrary Meshes), is a High Resolution differencing scheme (HR scheme) based on the Normalised Variable Diagram (NVD) (Leonard (1991)) and has been derived to address the above problems and restrictions.

The derivation of HR schemes, including CICSAM, is usually first done for explicit one-dimensional flow; such derivation is presented in the first part of the chapter.

Generalisation of CICSAM to multiple dimensions on arbitrary meshes is given in the second part of this chapter.

First the variable and control volume arrangements are defined. This is followed by a section on the so called donor-acceptor flux approximation (Ramshaw & Trapp (1976) and Hirt & Nichols (1981)), which forms the basis of compressive differencing schemes and is also used for the derivation of the new differencing scheme.

4.2 Variable and control volume arrangement

The interpolation of the face values needed for the finite volume discretisation of the indicator equation (Section 3.3.1) is the same for all faces and it is therefore sufficient to effect the derivation for a single face only. Furthermore, as mentioned before, the derivation is done initially for one-dimensional flow. Its subsequent implementation in multiple dimensions is given in Section 4.7.

A schematic representation of such a one-dimensional control volume and its neighbours is given in Figure 4.1. The centre cell (donor cell), referred to with a subscript D , has two neighbours known as the acceptor cell, referred to by subscript A , and the upwind cell, referred to by subscript U . The flow direction is used to determine the location of

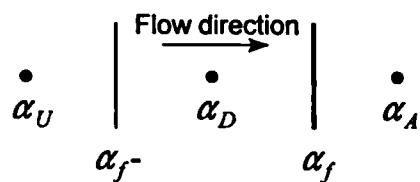


Figure 4.1: One-dimensional control volume

the neighbours. The cell receiving fluid from the centre cell is the acceptor cell and the other cell is the upwind cell. The face between the donor and acceptor cells, referred to with a subscript f , is the face under consideration.

4.3 Donor-acceptor flux approximation

In the earlier review of the related studies (Section 1.2.2, page 33) the concept of the donor-acceptor flux approximation was introduced and will be expanded upon here. It forms the basis of compressive differencing schemes and an understanding of it and the terminology introduced in this section is important for the remainder of this chapter.

The key to a successful compressive differencing scheme is the inclusion of downwind information in its formulation (Lafaurie *et al.* (1994)). The reason for this will become clear during this chapter. Downwinding however, does not comply with the boundedness criteria (Versteeg & Malalasekera (1995)) and may give rise to non-physical volume fraction values larger than unity or smaller than zero when it is applied to the α equation. Controlled downwinding, based on the availability of the fluids in a donor cell, is therefore introduced to avoid these non-physical volume fraction values (Ramshaw & Trapp (1976)).

Controlled downwinding can be explained by considering two neighbouring cells, a donor and an acceptor cell, as shown in Figure 4.2. Say, for instance, the donor cell contains part of the interface, and the acceptor cell is completely filled with one of the fluids (Figure 4.2 (a)). Downwinding implies that the donor cell has to donate the same fluid as presently contained in the acceptor cell, thus it ignores the presence of the other fluid in the donor cell. The donor cell however, contains both fluids and cannot donate more of this fluid than available inside it. Controlled downwinding means the donor cell will firstly donate all the available fluid required by the acceptor cell and then start to donate the other fluid. Similar arguments hold when both cells contain part of the interface (Figure 4.2 (b)). Downwinding then implies that the acceptor cell demands to receive the same fluid proportion as currently contained in it. Controlled downwinding implies that the donor cell will firstly comply with the demand, but if it runs out of that fluid it will start to donate a higher proportion of the other fluid.

The above mentioned boundedness criteria, the availability criteria and its implementation to control the downwind discretisation are described next. These are

then used later on in this chapter for the derivation of the new compressive differencing scheme.

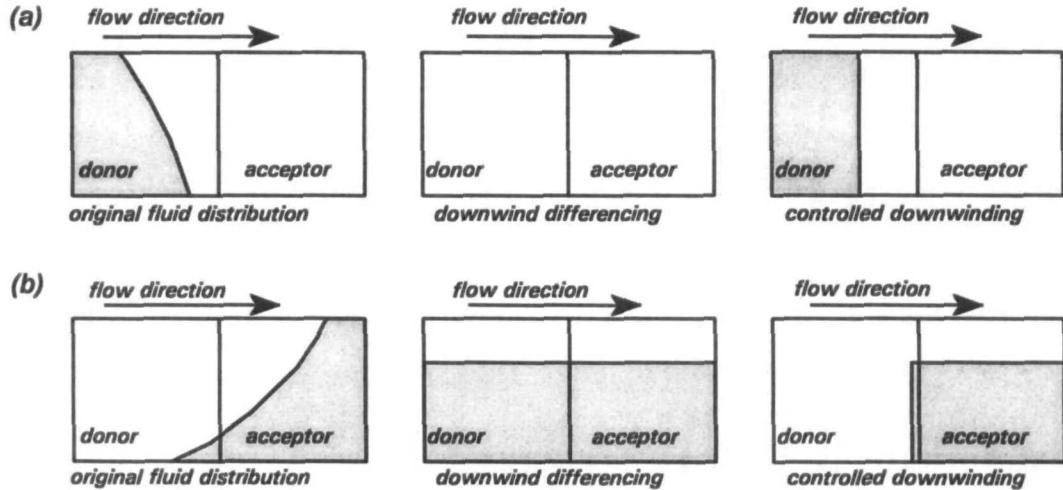


Figure 4.2: The donor-acceptor interface approximation

- **The boundedness criteria:** In the general CFD framework it is normally demanded that in the absence of sources, the value of a flow property in the flow domain cannot take values higher or lower than prescribed on the boundaries of the flow domain (Versteeg & Malalasekera (1995)). For the purpose of the volume fractions the boundedness criteria are taken as the physical bounds zero and one. Local boundedness, which is used later on in this chapter for the derivation of CICSAM, bounds a volume fraction value locally with its nearest neighbours.
- **The availability criteria:** Figure 4.3 gives a schematic representation of a donor cell containing part of the interface. If V_D is the volume of the donor cell, then $\alpha_D V_D$ is the amount of fluid 1 in the donor cell and $(1 - \alpha_D)V_D$ the corresponding amount of fluid 2. Let $c_f = |\mathbf{u}_f \delta t| / \delta x$ be the Courant number at the face and α_f the volume fraction of the volume which will be convected over the face in one time step, then $\alpha_f c_f V_D$ is the amount of fluid 1 convected over the cell face during a time step and $(1 - \alpha_f)c_f V_D$ is the corresponding amount of fluid 2.

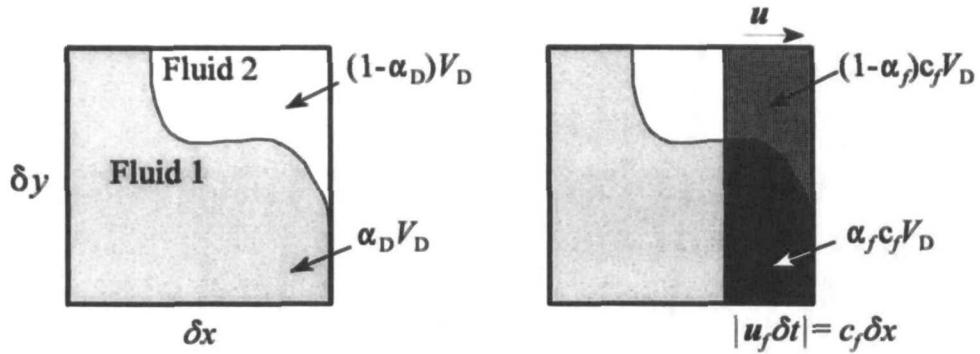


Figure 4.3: Fluid configuration in the donor cell

The availability criteria on fluid 1 dictates that the amount of it convected over a face during a time step should always be less than or equal to the amount available in the donor cell:

$$\alpha_f c_f V_D \leq \alpha_D V_D \quad (4.1)$$

$$\Rightarrow \alpha_f \leq \frac{\alpha_D}{c_f} \quad (4.2)$$

Similarly, the amount of fluid 2 convected over a face during a time step should always be less than or equal to the amount available in the donor cell:

$$(1-\alpha_f) c_f V_D \leq (1-\alpha_D) V_D \quad (4.3)$$

$$\Rightarrow \alpha_f \geq \frac{\alpha_D}{c_f} - \frac{1-c_f}{c_f}. \quad (4.4)$$

A combination of the two bounds (eq. (4.2) and eq. (4.4)) gives:

$$\frac{\alpha_D}{c_f} - \frac{1-c_f}{c_f} \leq \alpha_f \leq \frac{\alpha_D}{c_f}. \quad (4.5)$$

- **Controlled downwinding:** The requirement of maximum downwinding, that is $\alpha_f = \alpha_A$, limited to the amount of fluid available in the donor cell (eq. (4.5)) results in:

$$\begin{aligned}\alpha_f &= \min \left\{ \max \left\{ \frac{\alpha_D}{c_f} - \frac{1-c_f}{c_f}, \alpha_A \right\}, \frac{\alpha_D}{c_f} \right\} \\ &= \min \left\{ \alpha_A + \max \left\{ (1-\alpha_A) - \frac{(1-\alpha_D)}{c_f}, 0 \right\}, \frac{\alpha_D}{c_f} \right\}\end{aligned}\quad (4.6)$$

Equation (4.6) is the donor-acceptor face value approximation and it guarantees overall boundedness if it is applied to one-dimensional flow. It has also been used as a basis for the derivation of the interface method of Ramshaw & Trapp (1976), the VOF method (Hirt & Nichols (1981)), SURFER (Lafaurie *et al.* (1994)) and is applied in a rather special way for the derivation of CICSAM, the proposed compressive differencing scheme.

Ashgriz & Poo (1991) and Lafaurie *et al.* (1994) present numerical results for uniform density flows which indicate that all the methods mentioned in the previous paragraph give the interface a non-realistic deformation. During the early stages of the present study it became clear that these non-realistic deformations could be reduced by including α_U in the donor-acceptor formulation (eq. (4.6)), as will be shown later. Several attempts to take into account different gradients and values at the upwind cell were made, but for one reason or another none was successful. However when examined in the light of the High Resolution schemes based on the Normalised Variable diagram (Leonard (1991)) it became apparent that eq. (4.6) fails under certain circumstances because it does not comply with the local boundedness criteria.

As explained in the next section, the local boundedness criteria bound the value in the donor cell with the value in the upwind cell and this also provides guidance as to how to include the upwind value in a natural way.

The Normalised Variable Diagram, which forms a useful basis for formulating high resolution differencing schemes, is described next.

4.4 Normalised Variable Diagram

The normalised variable (Leonard (1991)) forms the basis of the high resolution schemes and is defined as:

$$\tilde{\alpha} = \frac{\alpha - \alpha_U}{\alpha_A - \alpha_U} \quad (4.7)$$

The normalised variable can be used to give expressions for $\tilde{\alpha}_D$ and $\tilde{\alpha}_f$:

$$\tilde{\alpha}_D = \frac{\alpha_D - \alpha_U}{\alpha_A - \alpha_U} \quad (4.8)$$

$$\tilde{\alpha}_f = \frac{\alpha_f - \alpha_U}{\alpha_A - \alpha_U} \quad (4.9)$$

With this normalization $\tilde{\alpha}_U=0$ and $\tilde{\alpha}_A=1$. Figure 4.4 illustrates the absolute and normalised variables for the case $\alpha_U < \alpha_D < \alpha_A$.

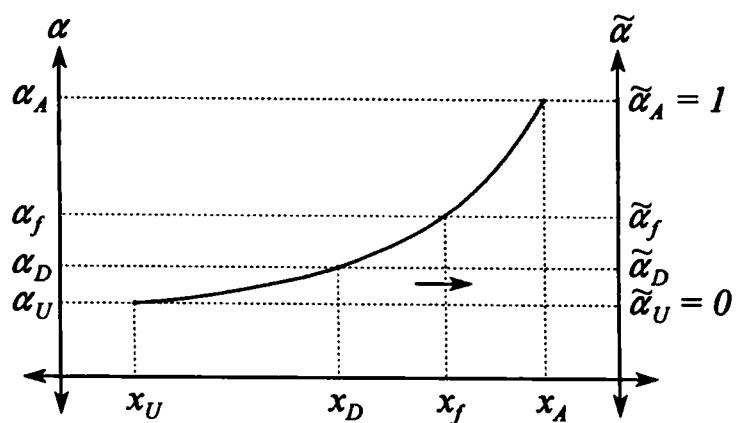


Figure 4.4: Original and normalised variables

Gaskell & Lau (1988) present a convection boundedness criteria (CBC) for one-dimensional implicit flow calculations. The CBC uses the normalised variable and presents bounds on $\tilde{\alpha}_f$ for which an implicit differencing scheme always preserves the local boundedness criteria:

$$\begin{aligned}\tilde{\alpha}_f &= \tilde{\alpha}_D && \text{for } \tilde{\alpha}_D < 0 \text{ or } \tilde{\alpha}_D > 1 \\ \tilde{\alpha}_D \leq \tilde{\alpha}_f \leq 1 & && \text{for } 0 \leq \tilde{\alpha}_D \leq 1\end{aligned}\tag{4.10}$$

Figure 4.5 shows the Normalised Variable Diagram (NVD) which plots the normalised face value as a function of the normalised donor cell value. The downwind, upwind, second order upwind and central differencing schemes are also represented on the diagram. The CBC, defined by eq. (4.10), defines the shaded area together with the line representing the upwind differencing scheme.

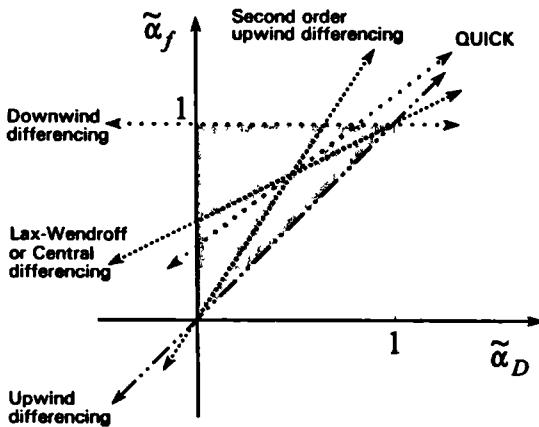


Figure 4.5: Normalised Variable Diagram (NVD) with the convection boundedness criteria (CBC)

Leonard (1991) shows that the various differencing schemes and the CBC can easily be adjusted for explicit flow calculations. This is done by introducing a linear weighting based on the Courant number c_f :

$$\tilde{\alpha}_f = (1 - c_f)\tilde{\alpha}_f^* + c_f\tilde{\alpha}_D\tag{4.11}$$

where $\tilde{\alpha}_f^*$ is the normalised face value for the implicit implementation.

With this linearisation $\tilde{\alpha}_f \rightarrow \tilde{\alpha}_f^*$ if $c_f \rightarrow 0$ and $\tilde{\alpha}_f \rightarrow \tilde{\alpha}_D$ if $c_f \rightarrow 1$, thus a point to point transfer of the upwind nodal value occurs if $c_f=1$. The CBC for explicit flow calculations reduces to:

$$\begin{aligned} \tilde{\alpha}_f &= \tilde{\alpha}_D && \text{for } \tilde{\alpha}_D < 0 \text{ or } \tilde{\alpha}_D > 1 \\ \tilde{\alpha}_D \leq \tilde{\alpha}_f \leq \min\left\{1, \frac{\tilde{\alpha}_D}{c_f}\right\} & && \text{for } 0 \leq \tilde{\alpha}_D \leq 1 \end{aligned} \quad (4.12)$$

Figure 4.6 shows the CBC for the explicit implementation with an arbitrary Courant number c_f .

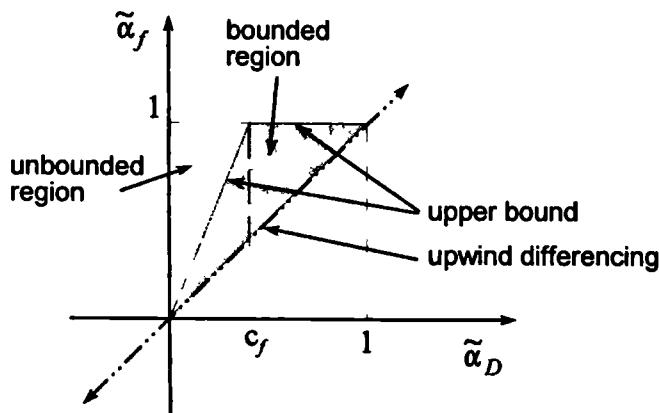


Figure 4.6: The convection boundedness criteria for explicit flow calculations

A differencing scheme which follows the upper bound of the CBC for explicit flow calculations is named Hyper-C (Leonard (1991)). Results presented by Leonard (1991) show that it is very compressive because it turns every finite gradient in a scalar field into a step profile, which is precisely what is needed for the present purpose.

The normalised variable diagram shown in Figure 4.6 can be used to evaluate high resolution schemes regarding boundedness and diffusivity, without actually

implementing them. A scheme falling within the upper left triangle will be unbounded. The nearer a scheme lies to the upper bound, the more compressive it is and the nearer it lies to the line representing upwind differencing, the more diffusive it is. Another interesting point to note is that the width of the bounded region diminishes as the Courant number increases. With a Courant number higher than unity, all schemes will be unbounded with an explicit implementation.

The CBC defined above, together with the knowledge gained from VOF (Hirt & Nichols (1981)) and SURFER (Lafaurie *et al.* (1994)) will be used in Section 4.6 for the construction of CICSAM, the proposed compressive differencing scheme tailored for sharp and accurate interface capturing.

This section is now closed with an algebraic manipulation of the equations defined earlier in this section in order to represent the rich information contained within the NVD in a form suitable for implementation on arbitrary meshes.

The NVD presents a normalised face value $\tilde{\alpha}_f$ for each normalised donor cell value $\tilde{\alpha}_D$ and equation (4.9) can be used for the calculation of the actual value at the face:

$$\alpha_f = \tilde{\alpha}_f \alpha_A + (1 - \tilde{\alpha}_f) \alpha_U \quad (4.13)$$

Equation (4.13) contains the upwind value, which can be eliminated with the aid of eq. (4.8). A reordering of eq. (4.8) yields:

$$\alpha_U = \frac{\alpha_D - \tilde{\alpha}_D \alpha_A}{1 - \tilde{\alpha}_D} \quad (4.14)$$

The expression for the upwind value, eq. (4.14), is substituted into eq. (4.13) and the final expression for the face value reduces to:

$$\alpha_f = (1 - \beta_f) \alpha_D + \beta_f \alpha_A \quad (4.15)$$

where

$$\beta_f = \frac{\tilde{\alpha}_f - \tilde{\alpha}_D}{1 - \tilde{\alpha}_D}. \quad (4.16)$$

With the above algebraic manipulation we have managed to express the face value in terms of the cell values bracketing the face. The family of HR schemes which is defined over a five point computational molecule has been reformulated to a blended up- and downwind scheme defined over a three point computational molecule. The weighting factor β_f , which contains the upwind value α_U , gives an indication of the gradient α at the face. The above formulation of the face value is important, because it enables the implementation of HR schemes on arbitrary meshes.

For the calculation of $\tilde{\alpha}_D$ (eq. (4.8)) the upwind value is needed but on arbitrary meshes this is not necessarily available. The determination of the upwind value for these meshes is given next.

4.4.1 The Normalised Variable for arbitrary meshes

As noted above, the upwind value is required for the calculation of the normalised variable defined by eq. (4.8, 4.17):

$$\tilde{\alpha}_D = \frac{\alpha_D - \alpha_U}{\alpha_A - \alpha_U} \quad (4.17)$$

On arbitrary meshes the upwind value α_U is not necessarily readily available and therefore high resolution schemes are restricted to rather simple mesh structures. Jasak *et al.* (1996) overcome this problem with a new definition of $\tilde{\alpha}_D$ which does not contain α_U directly:

$$\tilde{\alpha}_D = 1 - \frac{(\nabla \alpha)_f \cdot \mathbf{d}}{2(\nabla \alpha)_D \cdot \mathbf{d}} \quad (4.18)$$

where \mathbf{d} is the vector between the cell centres of the donor and acceptor cells, pointing from the donor cell towards the acceptor cell (see Figure 4.7). The gradient over the cell $(\nabla \alpha)_D$ is calculated with Gauss's theorem (eq. (3.4)). For uniform meshes equations

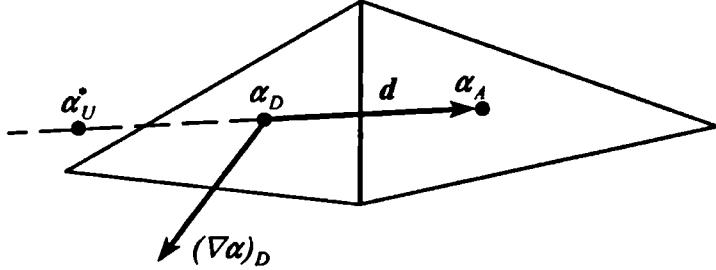


Figure 4.7: The prediction of the upwind value for an arbitrary cell arrangement

(4.17) and (4.18) are equivalent. Equation (4.18) is attractive because it does not need any information about the topological structure of the mesh and calculates $\tilde{\alpha}_D$ with no additional computational effort. However, numerical results on the convection of step profiles on arbitrary meshes have shown that this formulation does not necessarily guarantee a bounded solution.

To ensure boundedness, a new method for the calculation of $\tilde{\alpha}_D$ is presented. This method also uses the gradient over the cell, but it is used to obtain a bounded approximation α_U^* for the upwind value, instead. This approximated upwind value is then used in eq. (4.17) for the calculation of $\tilde{\alpha}_D$.

Figure 4.7 gives a schematic representation of two cells with an arbitrary arrangement. The gradient $(\nabla \alpha)_D$ is a vector quantity defined over the donor cell and points in the direction in which α changes the most. The dot product of this gradient with the unit vector tangential to \mathbf{d} ,

$$(\nabla \alpha)_{D_s} = (\nabla \alpha)_D \cdot \frac{\mathbf{d}}{|\mathbf{d}|} \quad (4.19)$$

gives the gradient of α over the centre cell in the direction of \mathbf{d} . The same gradient can also be approximated with central differencing:

$$(\nabla \alpha)_{D_s} = \frac{\alpha_A - \alpha_U^*}{2|d|} \quad (4.20)$$

Equations (4.19) and (4.20) are used to get an expression for α_U^* :

$$\alpha_U^* = \alpha_A - 2(\nabla \alpha)_{D_s} \cdot d \quad (4.21)$$

The above approximation does not guarantee a bounded α_U^* and therefore it is necessary to bound it with known bounds of α . The bounds can be either the maximum and minimum value of the whole flow or local values derived from the cell's nearest neighbours:

$$\alpha_U^* = \min \left\{ \max \{ \alpha_U^*, \alpha_{\min} \}, \alpha_{\max} \right\} \quad (4.22)$$

In the present study the physical bounds of the volume fractions (zero and one) is used for the lower and upper bounds respectively. This bounded approximation of the upwind value can now be used in eq. (4.17) for the calculation of $\tilde{\alpha}_D$.

4.5 Comparison of compressive differencing schemes

In Section 4.3 it has been mentioned that the donor-acceptor formulation, used in the derivation of VOF, deforms the interface shape because it does not comply with the local boundedness criteria. As noted in Section 4.4, the scheme following the upper bound of the CBC, which will be used in the present study, complies with the local boundedness criteria, is highly compressive and changes any smooth gradient into a step function.

First, the similarities between these two schemes will be addressed. Figure 4.8 (a) shows an arbitrary fluid configuration in three neighbouring cells. The cell under consideration is the donor cell, the centre cell of the three. Figure 4.8 (b) shows the fluid

distribution predicted with the donor-acceptor formulation. It shows that the level of the fluid in the upwind cell is ignored in the prediction of the fluid distribution in the donor cell. This results in an incorrect steepening of the interface gradient and eventually the non-physical deformation of the interface. Figure 4.8 (c) shows the fluid distribution predicted with Hyper-C, the scheme which follows the upper bound of the CBC. The level of the fluid in the upwind cell is taken into account, producing a more realistic fluid distribution in the donor cell.

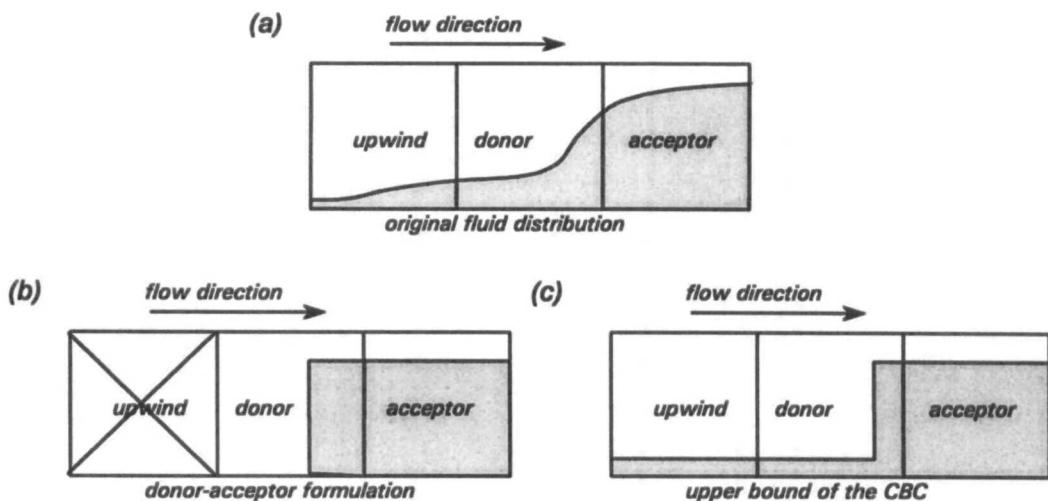


Figure 4.8: Comparison between the donor-acceptor formulation and the upper bound of the CBC

It can be shown that if the upwind value, predicted with eq. (4.22) is set to the physical bounds, zero or unity, then Hyper-C reduces to the donor-acceptor formulation. An alternative will be shown here - the donor-acceptor formulation will be applied to the normalised variables, resulting in the upper bound of the CBC. Thus the Hyper-C face value is the high resolution scheme equivalent of the donor-acceptor approximation.

The donor-acceptor formulation (eq. (4.6)) is general and can be applied to compress any nodal function which is bounded between zero and one. The normalised variable of the donor cell complies locally if $0 \leq \tilde{\alpha}_D \leq 1$ and therefore the donor-accepter formulation (eq. (4.6)) can be used to calculate the normalised face value of α . A substitution of the normalised variables into eq. (4.6) results in:

$$\begin{aligned}
\tilde{\alpha}_f &= \min \left\{ \tilde{\alpha}_A + \max \left\{ (1-\tilde{\alpha}_A) - \frac{(1-\tilde{\alpha}_D)}{c_f}, 0 \right\}, \frac{\tilde{\alpha}_D}{c_f} \right\} \\
&= \min \left\{ 1 + \max \left\{ -\frac{(1-\tilde{\alpha}_D)}{c_f}, 0 \right\}, \frac{\tilde{\alpha}_D}{c_f} \right\} \\
&= \min \left\{ 1, \frac{\tilde{\alpha}_D}{c_f} \right\}
\end{aligned} \tag{4.23}$$

If $\tilde{\alpha}_D$ is not between zero and one ($\tilde{\alpha}_D < 0$ or $\tilde{\alpha}_D > 1$) then the CBC demands $\tilde{\alpha}_f = \tilde{\alpha}_D$. This, together with eq. (4.23), define the normalised face value of the Hyper-C differencing scheme:

$$\tilde{\alpha}_{f_{cbc}} = \begin{cases} \min \left\{ 1, \frac{\tilde{\alpha}_D}{c_f} \right\} & \text{when } 0 \leq \tilde{\alpha}_D \leq 1 \\ \tilde{\alpha}_D & \text{when } \tilde{\alpha}_D < 0, \tilde{\alpha}_D > 1 \end{cases} \tag{4.24}$$

where $\tilde{\alpha}_D$ is calculated according to eq. (4.8).

4.6 CICSAM differencing scheme

In the previous section it was shown that Hyper-C is the most compressive differencing scheme which complies with the local boundedness criteria. Unfortunately, on its own it is not suitable for the modelling of interfacial flow, because it tends to wrinkle the interface (Hirt & Nichols (1981) and Lafaurie *et al.* (1994)). Numerical results demonstrating this behaviour are shown in Section 5.3.4. This is because downwinding tends to compress any gradient into a step profile, even if the orientation of the interface is almost tangential to the flow direction. This incorrect or artificial steepening of the volume fraction gradients is shown by Leonard (1991), who presents the result of the

advection of a one-dimensional semi-ellipse profile which becomes disfigured into a step profile by the scheme.

This problem is addressed in several works and all of them under certain conditions switch from the controlled downwind formulation to upwind differencing (Ramshaw & Trapp (1976), Hirt & Nichols (1981) and Lafaurie *et al.* (1994)). The most elementary procedure is to switch to upwinding when $|\alpha_D - \alpha_A| < k_s$, where k_s is a small prescribed constant (Ramshaw & Trapp (1976)).

The VOF method (Hirt & Nichols (1981)) determines the slope of the interface and switches to upwind differencing if the smallest angle between the interface and the face of the control volume is more than 45° . An extensive study on when to switch to upwind differencing has been carried out by Lafaurie *et al.* (1994). They present numerical results for a square volume fraction profile which is convected in a uniform oblique velocity field (see Figures 1.10 (a) and 4.9). The results for different critical angles show that the square can end up with any shape, from a badly smeared sphere to a rotated square with a tail of flotsam, or a bullet shape aligned with the flow direction.

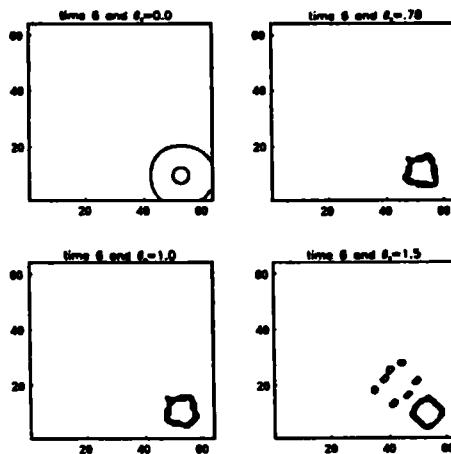


Figure 4.9: Effect of the critical angle on propagation (from: Lafaurie *et al.*, 1994, J. Comput. Phys., Vol. 113, Fig. 6, p. 138)

The results presented by Lafaurie *et al.* (1994) on when to switch to upwind differencing have confirmed the difficulties experienced in the present study. A re-examination of all the results has led to the realisation that the question shouldn't be one of *when* to switch, but rather one of *how* to switch to upwinding. Secondly, it has also been realised that upwind differencing is the worst scheme to switch to because it does not actually preserve the shape of an interface which lies almost tangentially to the flow direction. Thus, it is necessary to switch to another HR scheme which will preserve the interface shape better. In the present study it can be done with the aid of NVD and without much additional computational effort, because $\tilde{\alpha}_D$ is already available.

Leonard (1991) compares several higher order differencing schemes with each other on the convection of a step profile, sine function and semi-ellipse in a uniform flow field. The scheme which has performed the best regarding the convection of the sine function and semi-ellipse is ULTIMATE-QUICKEST (UQ), the one-dimensional explicit bounded version of QUICK (Leonard (1979)). Therefore, it has been decided to switch to the UQ differencing scheme instead (see Figure 4.10). The mathematical formulation of UQ used in the current study is obtained by applying eq. (4.11) to QUICK:

$$\tilde{\alpha}_{f_{UQ}} = \begin{cases} \min\left\{\frac{8c_f\tilde{\alpha}_D + (1-c_f)(6\tilde{\alpha}_D + 3)}{8}, \tilde{\alpha}_{f_{CBC}}\right\} & \text{when } 0 \leq \tilde{\alpha}_D \leq 1 \\ \tilde{\alpha}_D & \text{when } \tilde{\alpha}_D < 0, \tilde{\alpha}_D > 1 \end{cases} \quad (4.25)$$

A weighting factor $0 \leq \gamma_f \leq 1$, based on the angle between the interface and the direction of motion is introduced for the prediction of the normalised face value of CICSAM. This weighting factor is used to switch linearly between the controlled downwind differencing scheme eq. (4.24) and the less compressive differencing scheme eq. (4.25):

$$\tilde{\alpha}_f = \gamma_f \tilde{\alpha}_{f_{CBC}} + (1 - \gamma_f) \tilde{\alpha}_{f_{UQ}} \quad (4.26)$$

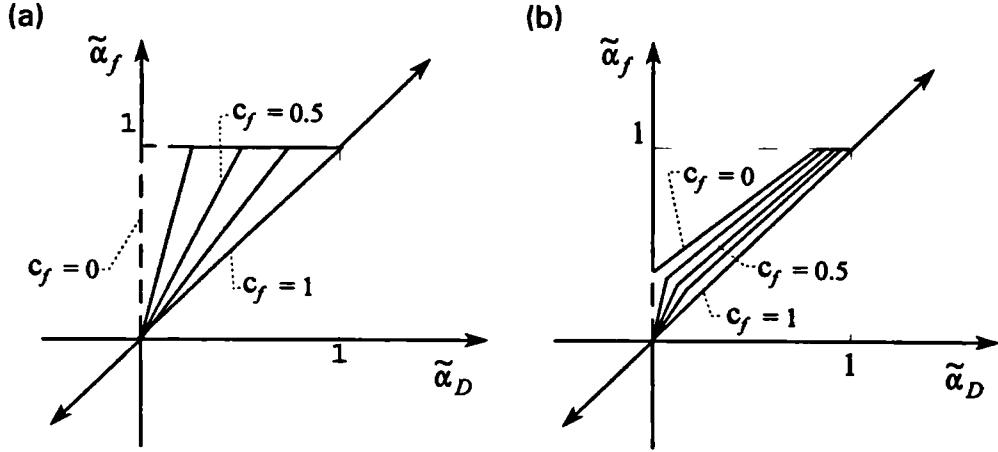


Figure 4.10: Building blocks of the CICSAM differencing scheme: (a) The upper bound of the CBC
(b) ULTIMATE-QUICKEST.

The NVD of CICSAM is shown in Figure 4.11. The value $\gamma_f=1$ is used when the interface orientation is normal to the direction of motion and $\gamma_f=0$ when the interface is tangential to it. This implies that UQ operates where Hyper-C fails to preserve the gradient in the interface and that Hyper-C operates where UQ fails to maintain the sharpness of the interface.

The derivation of CICSAM is completed with the definition of the weighting factor γ_f , which is based on the cosine of the angle θ_f between the vector normal to the interface $(\nabla \alpha)_D$ and the vector d_f , which connects the centres of the donor and acceptor cells. Thus:

$$\theta_f = \arccos \left| \frac{(\nabla \alpha)_D \cdot d_f}{|(\nabla \alpha)_D| |d_f|} \right| \quad (4.27)$$

$$\gamma_f = \min \left\{ k_\gamma \frac{\cos(2\theta_f) + 1}{2}, 1 \right\} \quad (4.28)$$

where $k_\gamma \geq 0$ is a constant introduced to control the dominance of the different schemes. The recommended value is $k_\gamma = 1$. For $k_\gamma = 0$ eq. (4.26) reduces to UQ and for a large k_γ it reduces to Hyper-C.

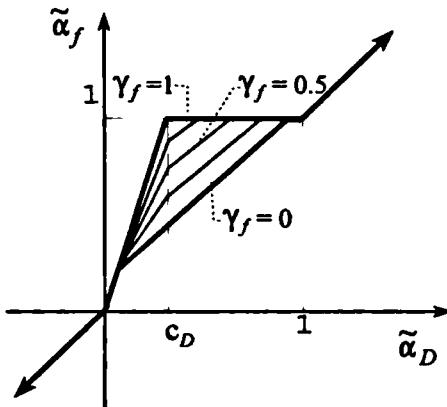


Figure 4.11: NVD for CICSAM differencing scheme

This completes the derivation of the normalised face value of the CICSAM differencing scheme. To summarise, the normalised face value predicted with the CICSAM differencing scheme for explicit one-dimensional uniform flow is defined by eq. (4.26). It can be reformulated in the form needed for the finite volume discretisation in Section 3.3.1 by using equations (4.15) and (4.16). The weighting factors β_f now carry all the information regarding the fluid distribution in the donor, acceptor and upwind cells as well as the interface orientation relative to the direction of motion. The next section describes the multi-dimensional implementation on arbitrary meshes.

4.7 Multiple dimensions

CICSAM has been derived to be as compressive as possible while preserving local boundedness for one-dimensional flow and an implementation in multi-dimensions may not necessarily maintain these characteristics.

Previously, to maintain these characteristics in multi-dimensional situations, use of an explicit split operator technique (Youngs (1982)) was resorted to. The basic idea of operator splitting is to apply the one-dimensional equation in separate steps for each of the coordinate directions. This limits the implementation to structured meshes in which

the faces of the control volume are aligned with the coordinate axis. In the literature no reference to compressive schemes implemented in ways other than the explicit split operator technique could be found. A possible reason for this is because the donor-acceptor equation, used in earlier methods, consists of a complicated *min/max* principle which makes it almost impossible to separate it in a linear weighting between the two cells sharing the same face.

This study deals with arbitrary meshes, and therefore it is necessary to find an alternative for the split operator technique. An analysis of the mechanism of the split operator technique has been made and is used for the construction of the new alternative technique to be presented herein.

As mentioned before, the operator splitting technique calculates the new volume fraction values in different sweeps, one for each coordinate direction. For symmetric differencing schemes, the new volume fraction values are independent of the order in which the sweeps are performed. The amount of volume fraction convected over a particular cell face however, varies with the sequence in which the sweeps are performed.

Figure 4.12 shows a donor cell with some of its neighbours and is used to support the derivation of the new implicit implementation. The right face of the donor cell is under consideration here. The donor cell contains two fluids. Figure 4.12 (a) shows the original and new positions of the fluids contained in the four cells after an x-sweep followed by a y-sweep. Figure 4.12 (b) shows it for a y-sweep followed by an x-sweep. The black shaded area shows the amount of fluid 1 which has crossed the right cell face of the donor cell. Clearly the new fluid distributions are exactly the same, but the amount of fluid 1 convected over the cell face during this time step differs. Figure 4.12 (c) shows the new fluid distribution after a single step and it is clear that the amount of fluid 1 which has crossed the right cell face is the average of the amounts shown in Figure 4.12 (a) and (b).

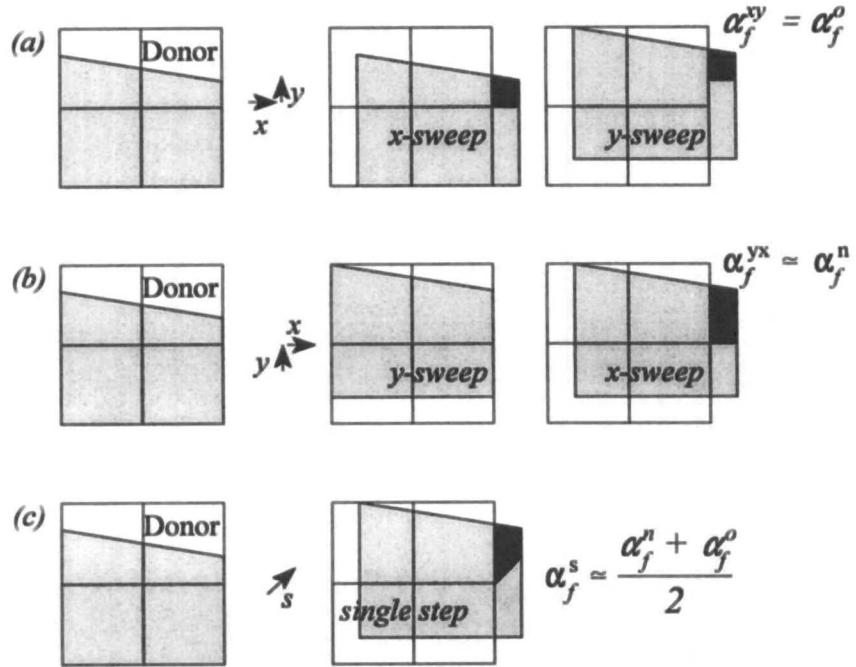


Figure 4.12: Prediction of the face value in multiple dimensions

The explicit split operator discretisation for the xy-sweep shown in Figure 4.12 (a) is:

$$\alpha_D^{t+\frac{1}{2}\delta t} = \alpha_D^t + c_{f_w} \alpha_{f_w}^t - c_{f_e} \alpha_{f_e}^t \quad (4.29)$$

$$\alpha_D^{t+\delta t} = \alpha_D^{t+\frac{1}{2}\delta t} + c_{f_s} \alpha_{f_s}^{t+\frac{1}{2}\delta t} - c_{f_n} \alpha_{f_n}^{t+\frac{1}{2}\delta t} \quad (4.30)$$

where the subscripts *n*, *e*, *s*, *w* indicates the north, east, south and west cell faces respectively. The superscript $t+\frac{1}{2}\delta t$ does not represent the real time in seconds, it only indicates an intermediate stage after a sweep in the one direction has been completed.

A combination of the two sweeps reduces to:

$$\alpha_D^{t+\delta t} = \alpha_D^t + c_{f_w} \alpha_{f_w}^t - c_{f_e} \alpha_{f_e}^t + c_{f_s} \alpha_{f_s}^{t+\frac{1}{2}\delta t} - c_{f_n} \alpha_{f_n}^{t+\frac{1}{2}\delta t} \quad (4.31)$$

Similar discretisation applies for the yx-sweep shown in Figure 4.12 (b):

$$\alpha_D^{t+\delta t} = \alpha_D^t + c_{f_w} \alpha_{f_w}^{t+\frac{1}{2}\delta t} - c_{f_e} \alpha_{f_e}^{t+\frac{1}{2}\delta t} + c_{f_s} \alpha_{f_s}^t - c_{f_n} \alpha_{f_n}^t \quad (4.32)$$

In order to remove the dependence on the sequence in which the sweeps are performed, the average of the xy-sweep and yx-sweep (eq. (4.31) and (4.32)) is taken and this results in:

$$\begin{aligned} \alpha_D^{t+\delta t} &= \alpha_D^t + \frac{1}{2} c_{f_w} (\alpha_{f_w}^t + \alpha_{f_w}^{t+\frac{1}{2}\delta t}) - \frac{1}{2} c_{f_e} (\alpha_{f_e}^t + \alpha_{f_e}^{t+\frac{1}{2}\delta t}) \\ &\quad + \frac{1}{2} c_{f_s} (\alpha_{f_s}^t + \alpha_{f_s}^{t+\frac{1}{2}\delta t}) - \frac{1}{2} c_{f_n} (\alpha_{f_n}^t + \alpha_{f_n}^{t+\frac{1}{2}\delta t}) \end{aligned} \quad (4.33)$$

In order to solve eq. (4.33) in a single step it is necessary to make an assumption about the face value of the volume fraction at the intermediate step denoted with a superscript $t+\frac{1}{2}\delta t$. From Figure 4.12 (b) it follows that if the time step is small enough, the face value of the intermediate step can be approximated by the face value of the new time. This assumption is not made on graphical considerations only, but is supported by the Crank-Nicolson scheme presented in Section 3.3.1. Thus, with these assumptions the split operator technique seems to be equivalent to a special case of the Crank-Nicolson scheme by defining the face value as:

$$\alpha_f^* = \frac{1}{2} (\alpha_f^t + \alpha_f^{t+\delta t}) \quad (4.34)$$

With this definition of the face values another problem arises. The face values are calculated according to eq. (4.15) and β_f , the weighting factors for the new time, are not available. Earlier it has been mentioned that the weighting factors represent the slope and orientation of the interface. If the time step is small enough, the interface slope and orientation will have a small variation and it is therefore reasonable to assume that the old weighting factors can be used. Thus, the single step face value is defined as:

$$\alpha_f^* = (1 - \beta_f) \frac{\alpha_D^t + \alpha_D^{t+\delta t}}{2} + \beta_f \frac{\alpha_A^t + \alpha_A^{t+\delta t}}{2} \quad (4.35)$$

Preliminary numerical tests however, have shown that neither the explicit split operator technique nor the new implicit implementation guarantees a new solution which satisfy the overall boundedness criteria. The reason and treatment for this are given in the next section.

4.7.1 Boundedness

Non-physical values of the volume fraction field in a multi-dimensional implementation occur because the face values are predicted in isolation of each other. Controlled downwinding at the different faces of a control volume may imply that the weighting factors of more than one face try to remove the interface from the donor cell, resulting in a non-physical volume fraction value. Thus, the available fluid to be convected over a face should be shared between the other faces.

Figure 4.13 illustrates this phenomenon where panel (a) shows an x-sweep being carried out first, which removes the one fluid completely from the donor cell by fluxing it across the right cell face. Figure 4.13 (b) shows the y-sweep being made first, which removes the same fluid completely from the donor cell by convecting it over the top face. In reality this fluid is convected over both of the faces. To achieve this in the controlled downwinding environment, the availability of the fluids to be convected over a face should be weighted against the activities at the other faces. Trying to achieve this with an implicit implementation on arbitrary meshes would be a daunting task, unless an assumption about the availability of the fluids in the donor cell is made.

The present study applies such an assumption. This is done by weighting the available fluid in the donor cell with a weighting factor based on the face Courant number c_f and the cell Courant number c_D , defined as c_f/c_D . The Courant number of the donor cell is defined as the total amount of fluid convected through the donor cell divided by the volume of the donor cell. The volumetric fluxes leaving the donor cell are used for its calculation:

$$c_D = \sum_{f=1}^n \max\left\{ \frac{-F_f \delta t}{V_D}, 0 \right\} \quad (4.36)$$

With this weighting, the CBC for multi-dimensional flow calculations reduces to:

$$\begin{aligned} \tilde{\alpha}_f &= \tilde{\alpha}_D && \text{for } \tilde{\alpha}_D < 0 \text{ or } \tilde{\alpha}_D > 1 \\ \tilde{\alpha}_D \leq \tilde{\alpha}_f \leq \min\left\{1, \frac{\tilde{\alpha}_D}{c_D}\right\} && \text{for } 0 \leq \tilde{\alpha}_D \leq 1 \end{aligned} \quad (4.37)$$

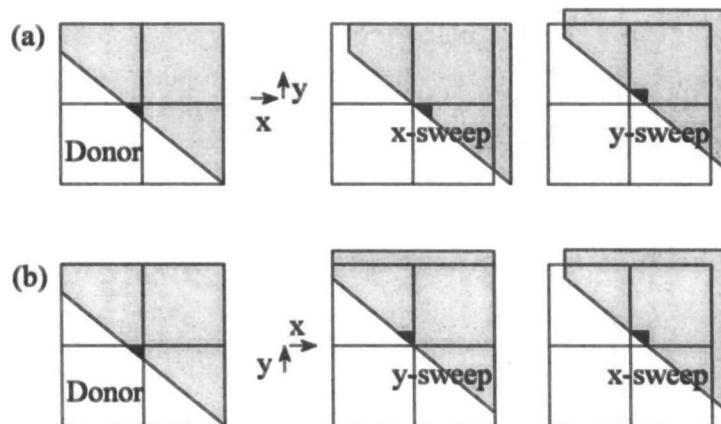


Figure 4.13: Consequence of controlled downwinding in multi-dimensional flow calculations

A schematic representation of the new CBC is shown in Figure 4.14, illustrating how the calculation of the normalised face value (eq. (4.24) and eq. (4.25)) which follows the upper bound of the CBC is modified using the Courant number of the donor cell centre instead. The final CICSAM face values is presented in Section 4.7.2.

Unfortunately this does still not guarantee a new volume fraction field with physically bounded values. Occasionally there may be volume fraction values slightly less than zero or slightly greater than unity, especially when the mesh consists of very badly

shaped cells. A predictor-corrector solution procedure which removes all the non-physical values is given in the final part of this chapter.

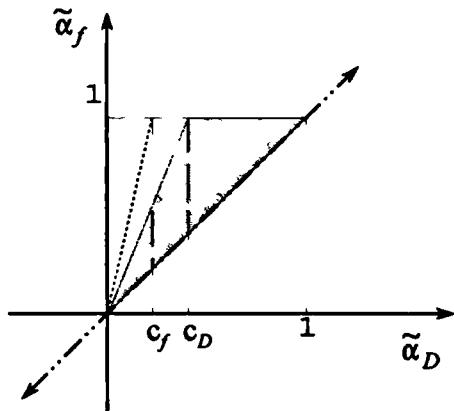


Figure 4.14: The CBC for multi-dimensional flow calculations

All the equations for the multi-dimensional implementation of CICSAM have now been formulated. The predictor-corrector solution algorithm for the α equation, including the final expression for the CICSAM face values, is given in the next section.

4.7.2 Predictor-corrector solution procedure

Compressive differencing schemes are derived to operate at the upper limit of the boundedness criteria and non-physical volume fraction values occur from time to time. Hirt and Nichols (1981) overcome these occurrences by resetting the non-physical values back to zero and one. They keep track of the accumulated changes of the fluid volumes and claim it can be neglected because the overall change of the fluid volumes is less than a fraction of a percent. With the conservative formulation of the momentum equations used in the present study, the resetting of the non-physical volume fractions values introduces a conservation error which manifests itself in a spurious velocity field.

It should be further emphasised that with the introduction of the relaxed CBC for multi-dimensional flow calculations in Section 4.7.1, the occurrences of non-physical values are rare. However, they do appear from time to time and therefore need to be corrected. The procedure to achieve this is to solve the α equation in two steps, namely a predictor and corrector sequence. The latter is only carried out when non-physical volume fraction values occur.

The predictor step consists of the solution of the discretised α equation with the CICSAM weighting factors, defined in this chapter. The various steps made are summarized as follows.

- **Predictor step**

1. Calculate the gradient $(\nabla\alpha)_D$ over the cell by using Gauss's theorem (eq. (3.4) and (3.13)):

$$(\nabla\alpha)_D \approx \frac{1}{V_D} \sum_{f=1}^n A_f \alpha_f \quad (4.38)$$

2. Calculate the Courant number for all the cells, using eq (4.36):

$$c_D = \sum_{f=1}^n \max\left\{\frac{-F_f \delta t}{V_D}, 0\right\} \quad (4.39)$$

3. Determine the donor and acceptor cell as explained in Section 4.2.
4. Predict the upwind value α_U^* according to eq. (4.21) and (4.22) and bound it with the physical bounds zero and unity:

$$\alpha_U^* = \min\left\{\max\{(\alpha_A - 2(\nabla\alpha)_D \cdot d), 0\}, 1\right\} \quad (4.40)$$

5. Use the predicted upwind value to calculate the normalised variable for the donor cell $\tilde{\alpha}_D$ (eq. (4.8)):

$$\tilde{\alpha}_D = \frac{\alpha_D - \alpha_U^*}{\alpha_A - \alpha_U^*} \quad (4.41)$$

6. Use the upper bound of the CBC for multi-dimensional flow calculations, defined in Section 4.7.1 for the calculation of the normalised face value:

$$\tilde{\alpha}_{f_{cbc}} = \begin{cases} \min\left\{1, \frac{\tilde{\alpha}_D}{c_D}\right\} & \text{when } 0 \leq \tilde{\alpha}_D \leq 1 \\ \tilde{\alpha}_D & \text{when } \tilde{\alpha}_D < 0, \tilde{\alpha}_D > 1 \end{cases} \quad (4.42)$$

7. Equation (4.25) is used for the calculation of the normalised face value for UQ:

$$\tilde{\alpha}_{f_{uq}} = \begin{cases} \min\left\{\frac{8c_D\tilde{\alpha}_D + (1-c_D)(6\tilde{\alpha}_D + 3)}{8}, \tilde{\alpha}_{f_{cbc}}\right\} & \text{when } 0 \leq \tilde{\alpha}_D \leq 1 \\ \tilde{\alpha}_D & \text{when } \tilde{\alpha}_D < 0, \tilde{\alpha}_D > 1 \end{cases} \quad (4.43)$$

8. Determine the weighting factor γ_f (eq. (4.28)) which takes into account the interface orientation and the direction of motion:

$$\gamma_f = \min\left\{k_\gamma \frac{\cos(2\theta_f) + 1}{2}, 1\right\} \quad (4.44)$$

where

$$\theta_f = \arccos \left| \frac{(\nabla \alpha)_D \cdot \mathbf{d}_f}{|(\nabla \alpha)_D| |\mathbf{d}_f|} \right| \quad (4.45)$$

and $k_\gamma = 1$ is the recommended value.

9. Calculate the normalised face value for the CICSAM differencing scheme:

$$\tilde{\alpha}_f = \gamma_f \tilde{\alpha}_{f_{cbc}} + (1 - \gamma_f) \tilde{\alpha}_{f_{uq}} \quad (4.46)$$

10. Equation (4.16) is used for the calculation of β_f , the CICSAM weighting factors:

$$\beta_f = \frac{\tilde{\alpha}_f - \tilde{\alpha}_D}{1 - \tilde{\alpha}_D} \quad (4.47)$$

11. Solve eq. (3.21), the discretised α equation, for the new volume fraction distribution and use eq. (4.35) to explicitly calculate the face values:

$$\alpha_f^* = (1 - \beta_f) \frac{\alpha_D^t + \alpha_D^{t+\delta t}}{2} + \beta_f \frac{\alpha_A^t + \alpha_A^{t+\delta t}}{2} \quad (4.48)$$

These face values are used for the prediction of the face densities, needed for the discretised momentum equations.

This concludes the predictor step for the new volume fraction values. If any of the volume fraction values have non-physical values, set $\beta_f^* = \beta_f$ and proceed with the correction procedure described below.

- **Corrector step**

For an explicit split operator calculation the face values of the donor cells with unbounded values are adjusted correspondingly with the degree of unboundedness in the donor cell. For the implicit scheme however, the face values cannot be corrected directly, because they are implicitly defined by the weighting factors and the new volume fraction values (eq. (4.47) and eq. (4.34) respectively). Thus it is the weighting factors which need to be corrected instead.

Furthermore, with the explicit split operator calculation the non-physical values remain localised. With the implicit implementation however, the localised non-physical values are convected throughout the domain and it is difficult to trace their origin. When correcting the weighting factors, it is therefore necessary to distinguish between non-physical values caused by too much downwinding or those being caused by convection into a cell during the implicit calculation. The correction procedure described here automatically distinguishes between the two.

The correction procedure consists of a visit to all cell faces. If the donor cell contains a non-physical volume fraction value, a test is performed to evaluate whether too much downwinding at the face under consideration has caused the unboundedness. If so, the weighting β_f^* is reduced with β_f , an amount proportional to the degree of unboundedness, as described below.

Firstly the negative volume fractions caused by too much downwinding will be discussed. A negative volume fraction caused by the above mentioned reason implies that more of fluid 1 has left the donor cell than is available in it. The new amount of fluid 1 to be convected over the face is determined by subtracting the unboundedness error from the original amount of fluid convected over the face, reducing to:

$$\alpha_f^{**} c_f V_D = \alpha_f^* c_f V_D - E^- V_D \quad (4.49)$$

$$\rightarrow \alpha_f^{**} = \alpha_f^* - \frac{E^-}{c_f} \quad (4.50)$$

where α_f^{**} is the new face value and E^- the magnitude of the unbounded volume fraction value, defined as:

$$E^- = \max\{-\alpha_D^{t+\delta t}, 0\} \quad (4.51)$$

The new face value α_f^{**} is substituted in the face value equation (4.48) to get a new approximation for the β weighting factor:

$$\alpha_f^{**} = (1 - \beta_f^{**}) \frac{\alpha_D^t + (\alpha_D^{t+\delta t} + E^-)}{2} + \beta_f^{**} \frac{\alpha_A^t + (\alpha_A^{t+\delta t} - E^-)}{2} \quad (4.52)$$

where

$$\beta_f^{**} = \beta_f^* - \beta_f' \quad (4.53)$$

According to eq (4.52) the corrected weighting factor β_f^{**} should always be less or equal to the previous weighting factor. If not, the contribution of the downwind cell starts to increase and so does the degree of unboundedness. The lower limit on β_f^{**} remains zero. This bound is applied to eq. (4.53) to obtain bounds for β_f' :

$$0 \leq \beta_f' \leq \beta_f^* \quad (4.54)$$

Equations (4.52) and (4.53) are substituted into eq. (4.50) to obtain:

$$(1 - (\beta_f^* - \beta_f)) \frac{\alpha_D^t + \alpha_D^{t+\delta t} + E^-}{2} + (\beta_f^* - \beta_f) \frac{\alpha_A^t + \alpha_A^{t+\delta t} - E^-}{2} = \alpha_f^* - \frac{E^-}{c_f} \quad (4.55)$$

$$\rightarrow \alpha_f^* + \beta_f^* \left(\frac{\alpha_D^t + \alpha_D^{t+\delta t}}{2} - \frac{\alpha_A^t + \alpha_A^{t+\delta t}}{2} \right) + \frac{E^-}{2} - \beta_f^* E^- + \beta_f E^- = \alpha_f^* - \frac{E^-}{c_f} \quad (4.56)$$

$$\rightarrow \beta_f^* = \frac{E^- (2 + c_f - 2c_f \beta_f^*)}{2c_f (\Delta \alpha^* - E^-)} \quad (4.57)$$

where

$$\Delta \alpha^* = \frac{\alpha_A^t + \alpha_A^{t+\delta t}}{2} - \frac{\alpha_D^t + \alpha_D^{t+\delta t}}{2} \quad (4.58)$$

Equation (4.57) needs to be bounded with the bounds defined in eq. (4.54):

$$\beta_f^* = \begin{cases} \min\left\{\frac{E^- (2 + c_f - 2c_f \beta_f^*)}{2c_f (\Delta \alpha^* - E^-)}, \beta_f^*\right\} & \text{when } \Delta \alpha^* > E^- \\ 0 & \text{when } \Delta \alpha^* \leq E^- \end{cases} \quad (4.59)$$

A similar corrector for the case where the volume fraction value exceeds unity can be derived:

$$\beta_f^* = \begin{cases} \min\left\{\frac{E^+ (2 + c_f - 2c_f \beta_f^*)}{2c_f (-\Delta \alpha^* - E^+)}, \beta_f^*\right\} & \text{when } \Delta \alpha^* < -E^+ \\ 0 & \text{when } \Delta \alpha^* \geq -E^+ \end{cases} \quad (4.60)$$

where

$$E^+ = \max\{\alpha_D^{t+\delta t} - 1, 0\} \quad (4.61)$$

Equations (4.59) and (4.60) are now used to update the weighting factors β_f^{**} (eq. (4.53)) and these are then used to solve eq. (3.21), the discretised α equation, for the new corrected volume fraction distribution. If non-physical values still exist, the corrector procedure is repeated. On an orthogonal mesh the corrector sequence is hardly used but on an arbitrary mesh with badly shaped control volumes it may be necessary to apply it more than once.

4.8 Closure

A new compressive differencing scheme for the capturing of sharp fluid/fluid interfaces on arbitrary meshes has been presented in this chapter. The scheme consists of a blend between upwind differencing and a differencing scheme which follows the upper bound of the convection boundedness criteria (CBC) for high resolution schemes. The similarity between the split operator technique and the Crank-Nicolson scheme has been outlined and a new CBC for the implicit Crank-Nicolson type of scheme in multi-dimensional flows has been derived. A predictor-corrector solution procedure which guarantees a bounded volume fraction field for the discretised α equation has been presented.

This completes the derivation of the discrete numerical model for two-fluid systems with sharp interfaces. The next chapter is used to evaluate the proposed method against exact solutions and/or experimental data.

5 Case studies

5.1 Introduction

A CFD methodology for the capturing of interfaces between immiscible fluids and the associated prediction of such flows was presented in the previous chapters. In this chapter the methodology will be tested against analytical solutions and experimental data.

The test cases selected are well established for the evaluation of interface capturing schemes. They can be classified into theoretical ones, where comparisons are made against analytical solutions, and real cases, where the comparisons are made against experimental data. The latter can be subdivided into experiments where the interfacial shapes are determined or where some other features are measured or demonstration cases where little other than flow visualisation data is available.

The test cases selected, which include all the above classes, are:

- uniform density flow calculations free from gravitational forces, consisting of:
 - different volume fraction shapes in a uniform oblique velocity field,
 - a rectangular volume fraction distribution in a circular velocity field,
 - surface tension forces applied to a square volume fraction field

- inviscid, low amplitude sloshing,
- the rise of a single Taylor bubble in a stagnant liquid,
- a collapsing water column,
- a bore / moving hydraulic jump,
- the flow underneath a sluice-gate and
- the splash of a liquid drop into a stagnant liquid.

The first set of cases, namely uniform density flows, validates the interface capturing methodology proposed in this thesis against other available methods. Its performance on arbitrary meshes is also evaluated.

The calculations for sloshing and Taylor bubbles validate the two-fluid methodology by comparisons against analytical and experimentally-determined shapes. Furthermore, some inferred data, such as the period of sloshing and the rise velocity of the Taylor bubble are also available for comparison.

Exact shapes of the interface of the cases of the collapsing water column and bore are not available, however some data relating to the wave speed of the collapsing column and the height difference between the front and back of the bore is available. Moreover, these cases also show the capability of the methodology presented here to predict rupturing and merging interfaces.

The last two cases, the flow underneath a sluice gate and the three-dimensional splash of a liquid drop are included to illustrate the variety of interfacial flow problems to which this methodology can be applied.

5.2 Post processing

The fluids are marked with a volume fraction field defined at discrete nodal points and the question arises of how to represent the interface using these values. Figure 5.1 shows different methods of presenting the interfacial structures. Panel (a) shows the

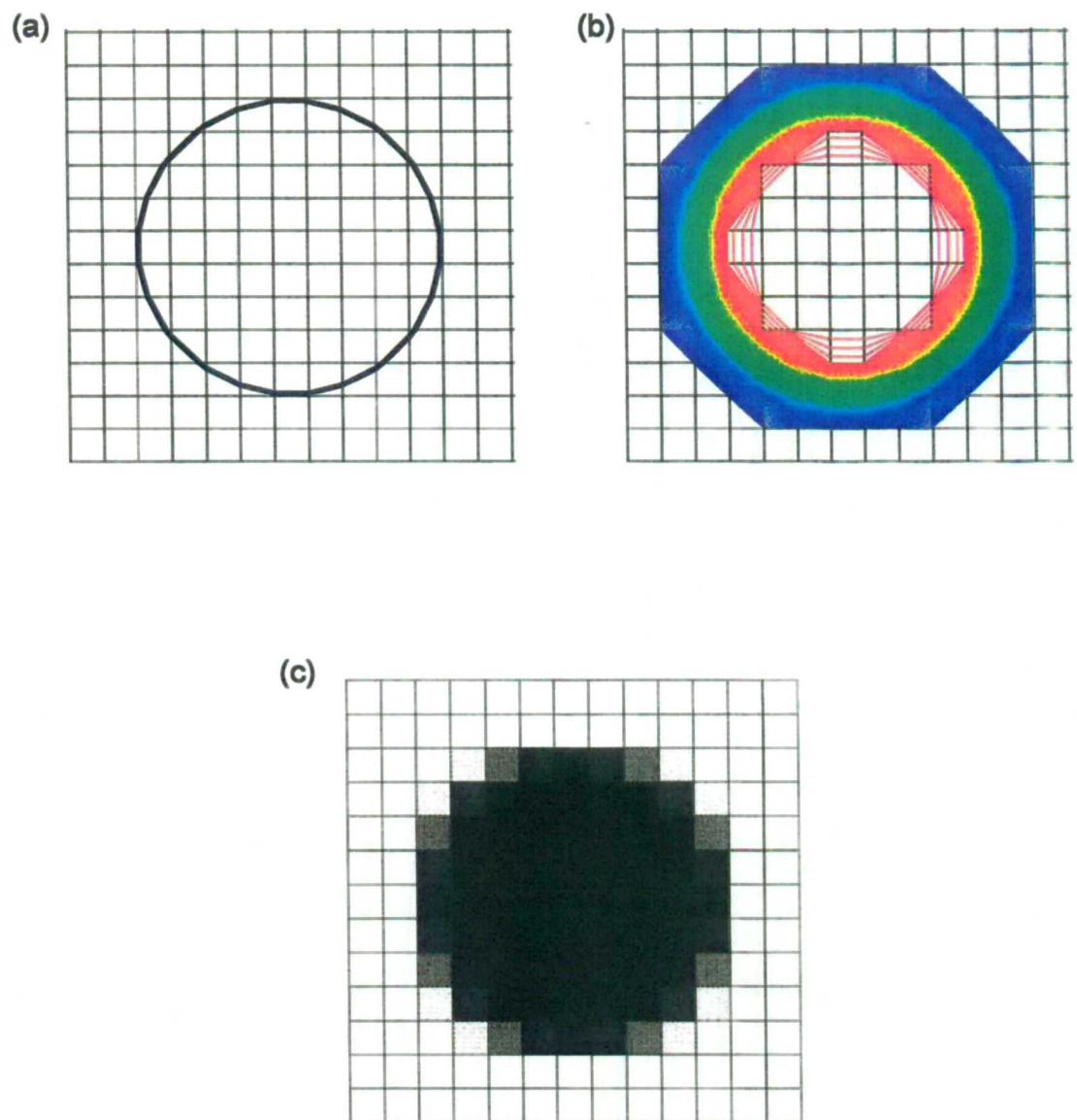


Figure 5.1: Different methods of presenting the interfacial structures

interface used to construct the volume fraction values for each mesh cell, whilst panel (b) shows one hundred contour lines, each representing a fixed volume fraction value - red indicating the contour $\alpha = 1$ and blue the contour $\alpha = 0$. The green contour $\alpha = 0.5$ corresponds to the original interface. The benefit of using all the contours between zero and unity is that the smearing of the interface is shown, whilst the use of just the middle contour ignores the existence of a diffused interface.

The drawback of using all the contours is that quantitative comparisons between different predictions cannot be easily made. Furthermore, the impression may be given that the interface is spread over a wide area. Panel (c) uses a grey scale and presents the volume fraction values in each cell on the basis of black representing a value of unity and white the value zero. This method points out the smearing of the interface very efficiently, but the actual interface shape is not very clear. Another problem associated with this technique is that most toner based printers and copiers cannot cope with large areas of black and introduce an unwanted grey scale.

In the present study all three methods will be used where appropriate. The contour $\alpha = 0.5$ will be used when accurate comparisons are to be made and not to hide the diffusion of the volume fraction values. Occurrences of numerical diffusion will always be revealed and discussed throughout this chapter.

5.3 Uniform density flow

The validation of interface-capturing schemes is commonly done by placing arbitrary-shaped fluid interfaces or structures in a known velocity field (Chorin (1980), Ashgriz & Poo (1991) and Lafaurie *et al.* (1994)). These methods validate the accuracy of both the convection procedure and the capturing methodology. Here these interfaces will first be placed in a uniform velocity field oblique to the mesh, in order to compare the capabilities of the methodology presented here with previous ones, also for the purpose of validation on different meshes and Courant numbers. Next, the behaviour of a

rotating rectangular volume fraction field in a non-uniform velocity field, computed on meshes consisting of arbitrary shaped control volumes, will be assessed. This section is then closed with an example of the surface tension methodology presented earlier applied to a rectangle, eventually deforming it to a circle with a uniform pressure jump across the interface.

5.3.1 Circle in an oblique velocity field

This test case consists of a fluid 'bubble' in the shape of a circle placed in another fluid of the same density, which has a uniform velocity oblique to the mesh. Surface tension effects are ignored. Aschry and Poo (1991) compare the VOF method (Hirt & Nichols (1981)) with the analytical solution for this case, which is simply that the bubble should be convected with unchanging shape at the fluid velocity. After 20 time steps the shape of the circle is compared with the initial shape. They report a deformation of the initial circular shape, although any reasonably good scheme should not show significant problems during the first 20 time steps.

In the present study, this test case is driven to its limits by placing a circle with a nine cell diameter in the lower left corner of a uniform (200x100) rectangular mesh (see Figure 5.2 (a)). The circle is convected to the upper right corner and close-up views of the initial and final volume fraction distributions predicted with the VOF method (Hirt & Nichols (1981)), are presented in panels (b) and (c) respectively. The position of the enlarged area is such that the final circle should be in its centre. This calculation confirms the non-physical deformation introduced by the VOF method (Hirt & Nichols (1981)), first reported by Aschry and Poo (1991).

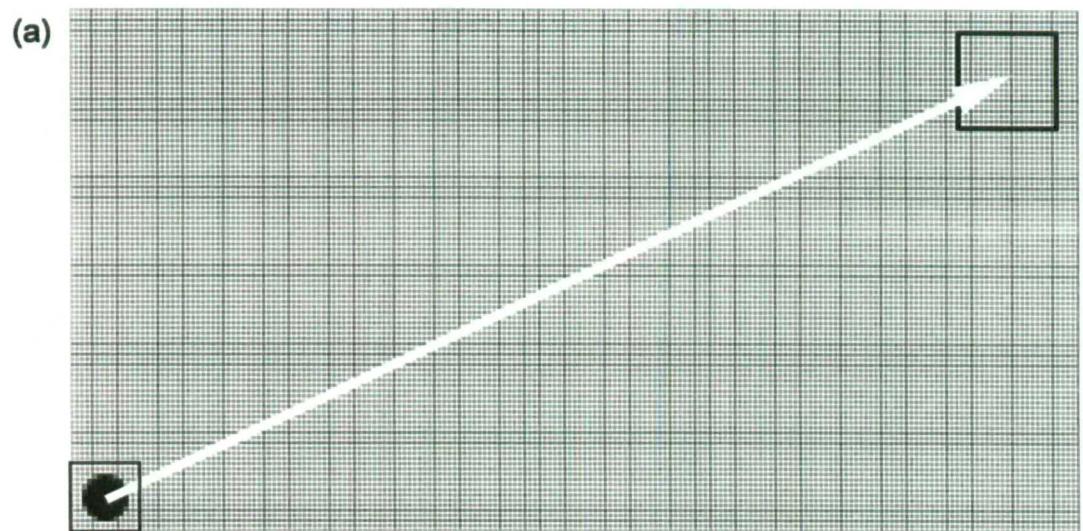
To stress the importance of a compressive scheme, results of four different spatial differencing schemes are presented in Figure 5.3. The schemes used vary from upwind differencing, the most diffusive scheme, to Hyper-C (Leonard (1991)), the most compressive differencing scheme, respectively shown in panels (a) and (d). Panel (b)

shows the results obtained with Van Leer's scheme (Van Leer (1977)), previously employed for interfacial flow calculations (Ghobadian (1991) and Pericleous & Chan (1994)). The results obtained with STOIC (Darwish (1993)), a scheme also recommended for the capturing of sharp interfaces, is shown in panel (c). Hyper-C (Leonard (1991)) performs well for this specific case, however as will be shown later in this section it cannot be employed at all times because it introduces a waviness when the interface orientation is almost tangential to the direction of motion.

As noted in Sections 3.3.1 and 4.7, the implementation of schemes capturing sharp interfaces has previously been done with explicit split operator techniques (Youngs (1982)). The results presented in Figures 5.2 and 5.3 have also been obtained with this technique.

Figure 5.4 presents results obtained with Hyper-C for different temporal discretisation techniques. Panel (a) shows the final shape predicted with operator splitting. Panel (b) shows the numerical diffusion of the circle normal to the direction of flow with the explicit scheme and panel (c) shows this diffusion to occur in the direction of motion for the implicit scheme when no operator splitting is implemented. The results obtained with the Crank-Nicolson (CN) type of scheme with a Courant number of $c_D = 0.334$ is shown in panel (d). Taking into account the large distance over which the circle has been convected without deforming its initial shape, it is reasonable to accept the small amount of numerical diffusion introduced by the implicit implementation.

The numerical diffusion introduced by the CN implementation is a function of the Courant number. This effect is illustrated with the aid of Figure 5.5, presenting results obtained with CICSAM for four different Courant numbers. Panel (a) presenting the results obtained with $c_D = 0.15$, indicates that almost no numerical diffusion is introduced, whilst panel (b) presenting results obtained with $c_D = 0.30$ indicates reasonable levels of numerical diffusion. Panels (c) and (d) presenting results obtained with $c_D = 0.45$ and $c_D = 0.60$ respectively, indicate unacceptable levels of deformation



Initial configuration

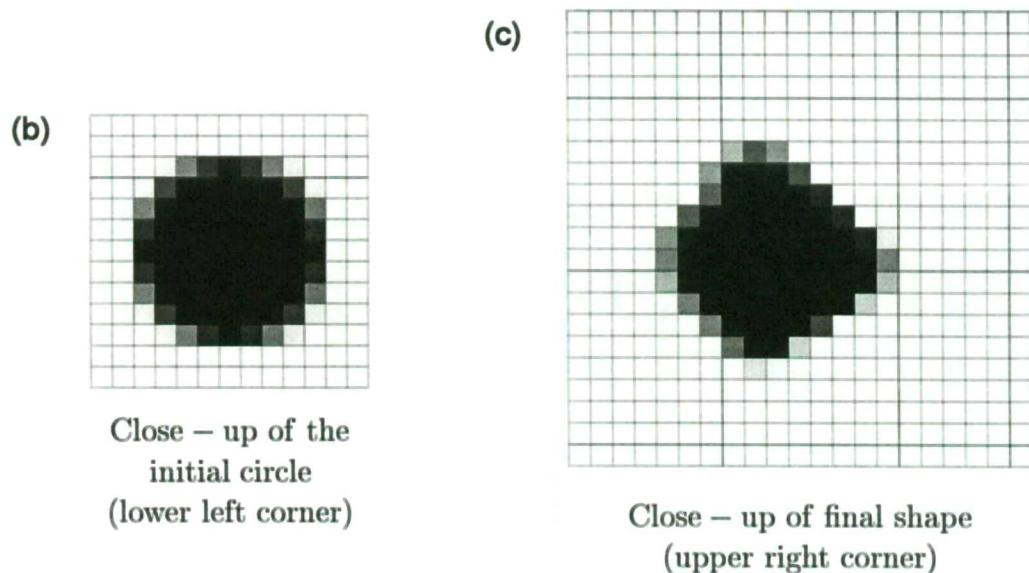


Figure 5.2: A circle in an oblique velocity field (a) a 200x100 uniformly spaced mesh (b) enlargement of the initial shape (c) enlargement of the final shape predicted with the VOF method

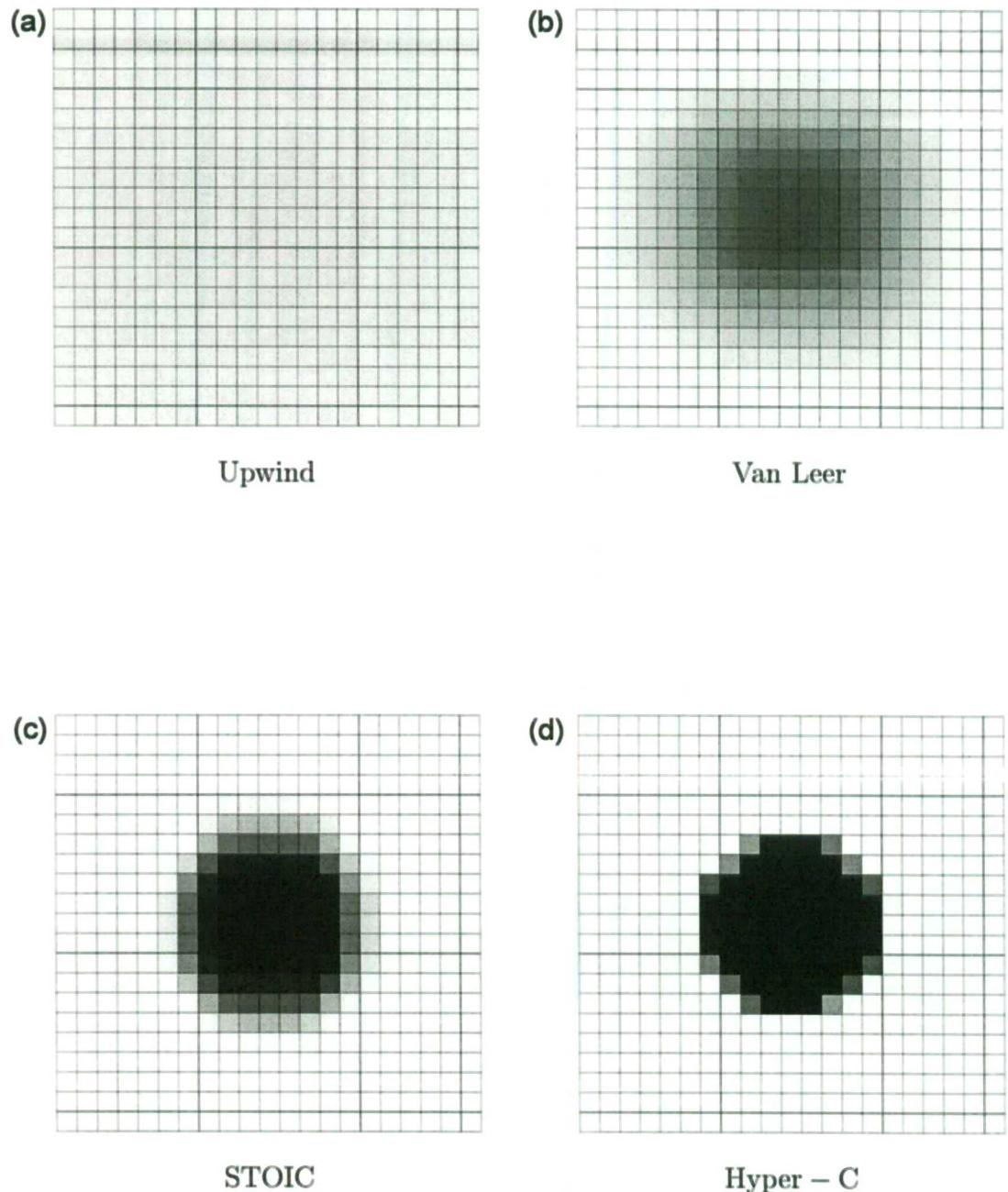
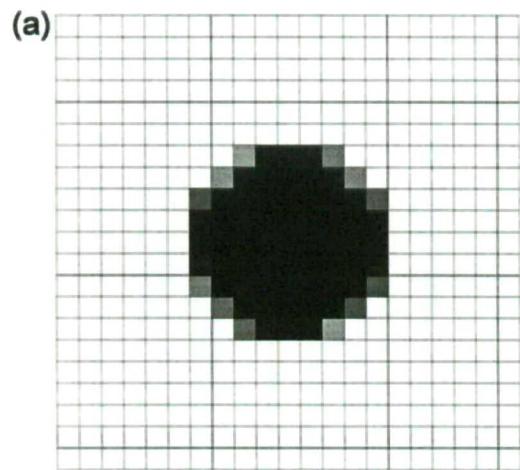
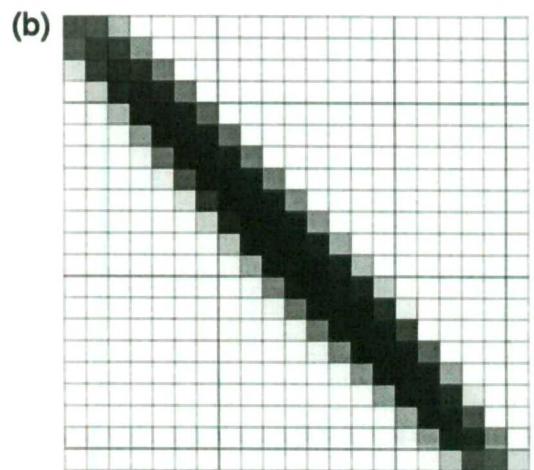


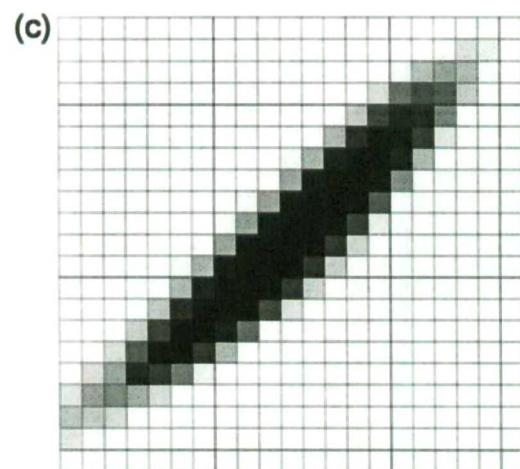
Figure 5.3: Enlargements of the final shapes predicted with different spatial differencing schemes



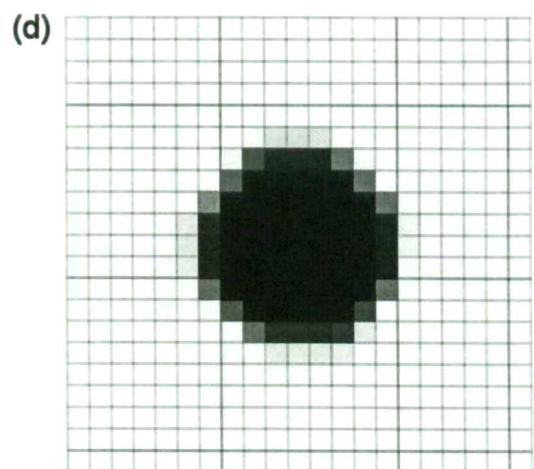
Operator splitting



Explicit



Euler implicit



Crank – Nicolson type of scheme

Figure 5.4: Enlargements of the final shapes predicted by Hyper-C with different temporal differencing schemes ($c_D=0.334$)

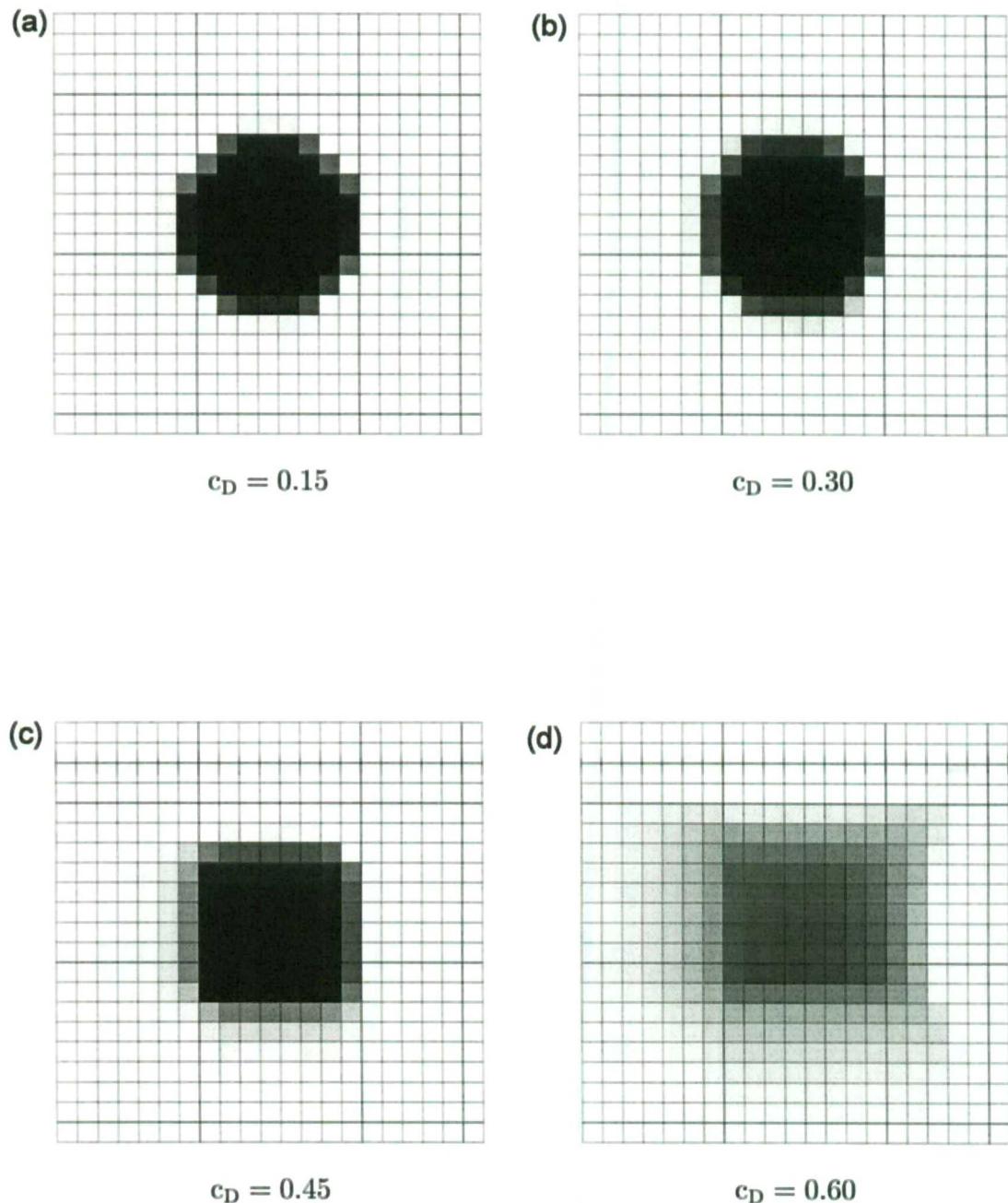


Figure 5.5: Enlargements of the final shapes for CICSAM using CN temporal discretisation with different Courant numbers

and numerical diffusion. For the remainder of test cases presented in this chapter the recommended value of $c_D=0.3$ will be used. This ensures acceptable numerical accuracy whilst making most efficient use of available computer resources.

5.3.2 A square 'bubble' in an oblique velocity field

Lafaurie *et al.* (1994) present numerical results for convection of a square 'bubble' in an oblique velocity field (see Figures 1.10 (a) and 4.9) with their method SURFER, which is in principle a modified VOF method. The calculation will be repeated here for CICSAM on both uniformly and non-uniformly spaced meshes.

The results for the uniformly spaced mesh are given in Figure 5.6. The mesh, consisting of 125x125 uniformly spaced cells, together with the velocity vectors and the original interface are shown in the top left panel. The square itself occupies 40 cells. At the final time step, $t=0.09s$, the square still has its original shape, with a lower left corner smeared over one additional cell.

The next case, presented in Figure 5.7, presents the convection of the square in an oblique velocity field on a non-uniform mesh at different time steps. The mesh used consists of 120 uniformly spaced cells in the vertical direction and 60 non-uniformly spaced cells in the horizontal direction. The first and last 30 cells are uniformly spaced, and the spacing between the remainder of the cells increases towards the centre of the flow domain. The volume of the largest cells is five times those of the smallest cells.

This case is interesting because it illustrates the previously stated phenomenon that the thickness of the interface is at least the size of one mesh cell, thus the higher the resolution of the mesh, the higher the resolution of the interface. CICSAM manages to restore the volume fraction square almost to its original configuration. However, it has difficulties in modelling interfaces with sharp corners on such meshes. Nonetheless, the overall performance is still much better than other schemes even on uniform meshes as shown in Figures 1.10 (a) and 4.9.

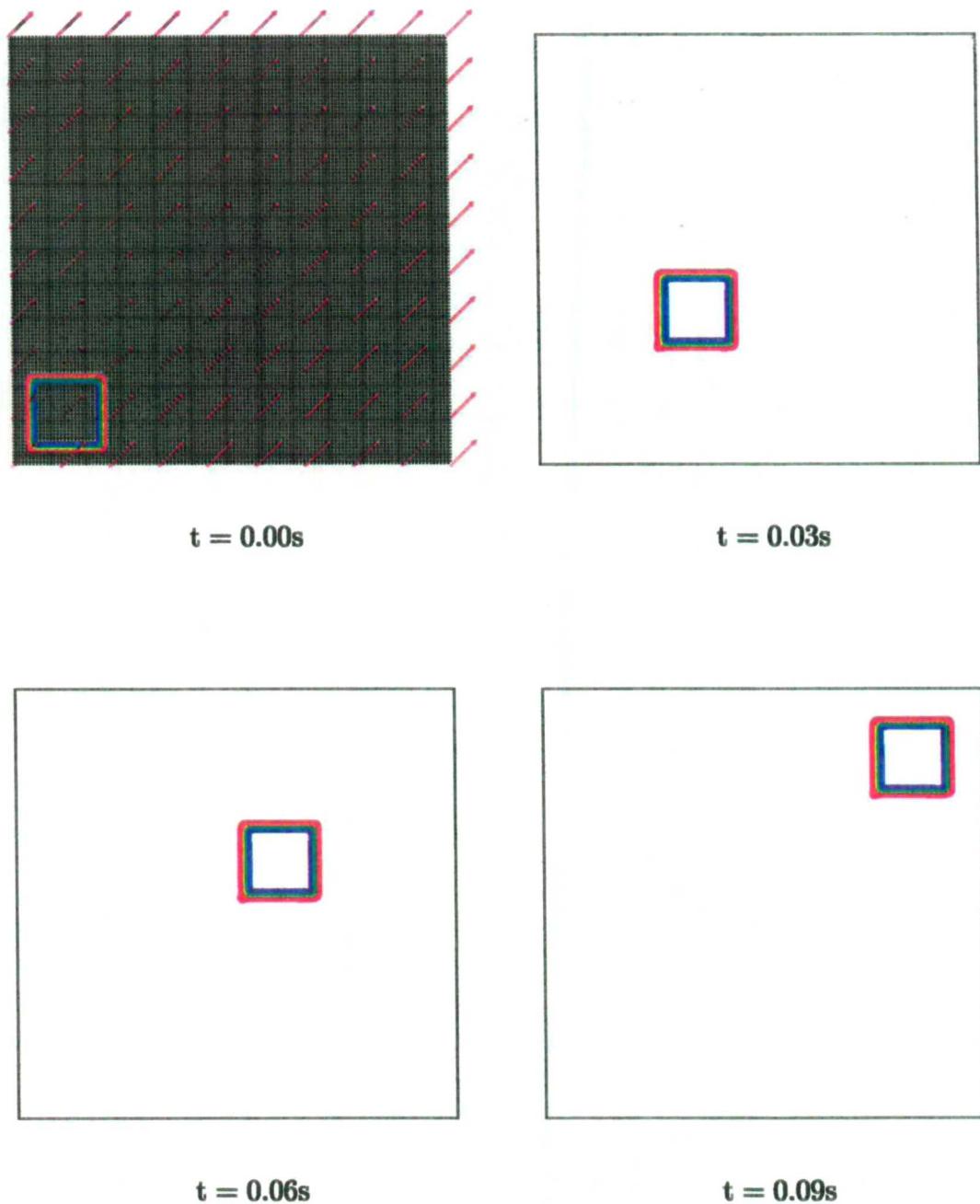
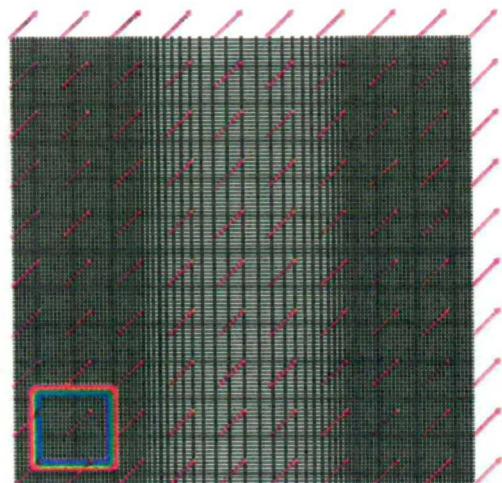
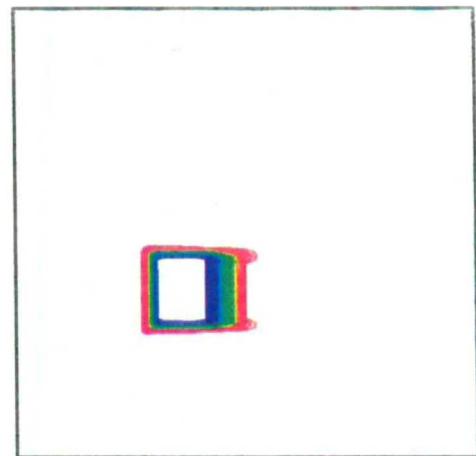


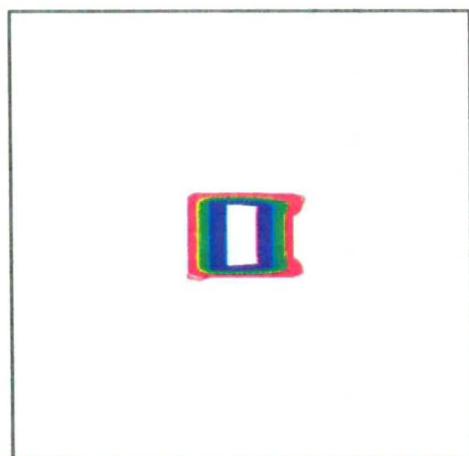
Figure 5.6: A volume fraction square convected with an oblique velocity field on a uniform mesh



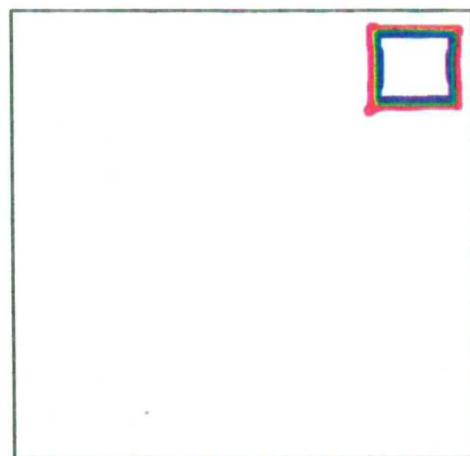
$t = 0.00s$



$t = 0.03s$



$t = 0.045s$



$t = 0.09s$

Figure 5.7: A volume fraction square convected with an oblique velocity field on a non-uniform mesh

5.3.3 A wave in an oblique velocity field

Another case, not encountered elsewhere, is that of an interface with a cosine profile placed in an oblique velocity field as shown in the insets of Figures 5.8 and 5.9. The volume fraction value for the area underneath the cosine profile is initialised with unity and the area above is given a value of zero. Instead of tracking the profile over a large domain and mapping the final shape back onto the original profile, use is made of cyclic boundaries, where the top/bottom and left/right boundaries are connected to each other. Thus, as the profile leaves the upper right corner, it enters at the lower left corner. After 350 time steps (for the 70x30 mesh) and 700 time steps (for the 140x60 mesh) the calculations are terminated - theoretically the final shape should exactly match the original shape.

The results obtained with the donor-acceptor formulation, as defined in Section 4.3 and CICSAM are shown in Figures 5.8 and 5.9 for a 70x30 and 140x60 mesh respectively.

The waviness introduced by using the donor-acceptor formulation on its own without taking into account the interface orientation is clearly visible at the turning points. Furthermore, the waviness worsens with higher mesh resolution. However, this waviness does not appear when CICSAM is used. CICSAM performs well, even in the case of the lower resolution mesh.

5.3.4 Rotation of a rectangle

In order to study the accuracy with which an interface can be captured in a non-uniform velocity field, Chorin (1980) introduced a rectangular interface in a vortical velocity field, corresponding to solid-body rotation, as shown in Figure 5.11. Lafaurie *et al.* (1994) used this case to evaluate the behaviour of SURFER, a CFD methodology based on the VOF method. Results obtained by them are shown in Figure 1.10 (b).

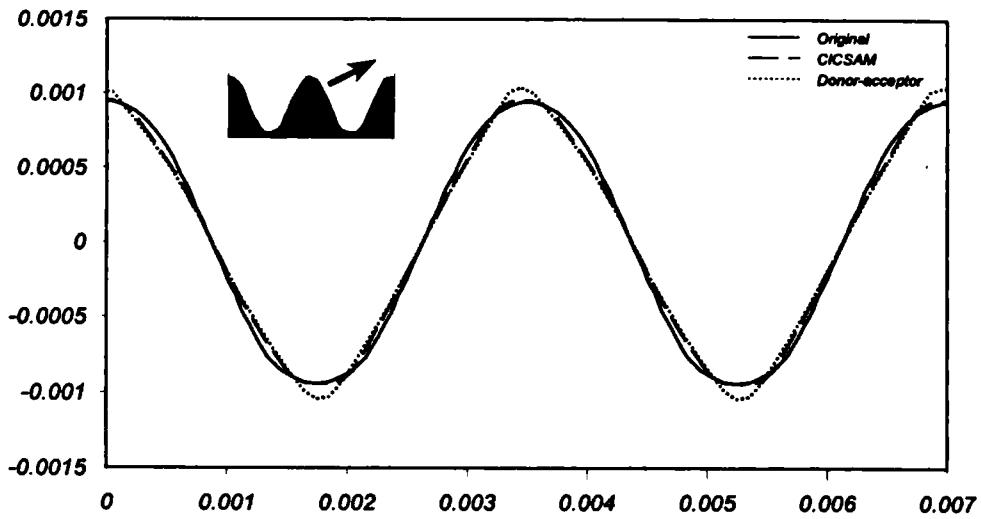


Figure 5.8: A volume fraction wave convected with a uniform oblique velocity field on a (70x30) mesh with cyclic boundary conditions.

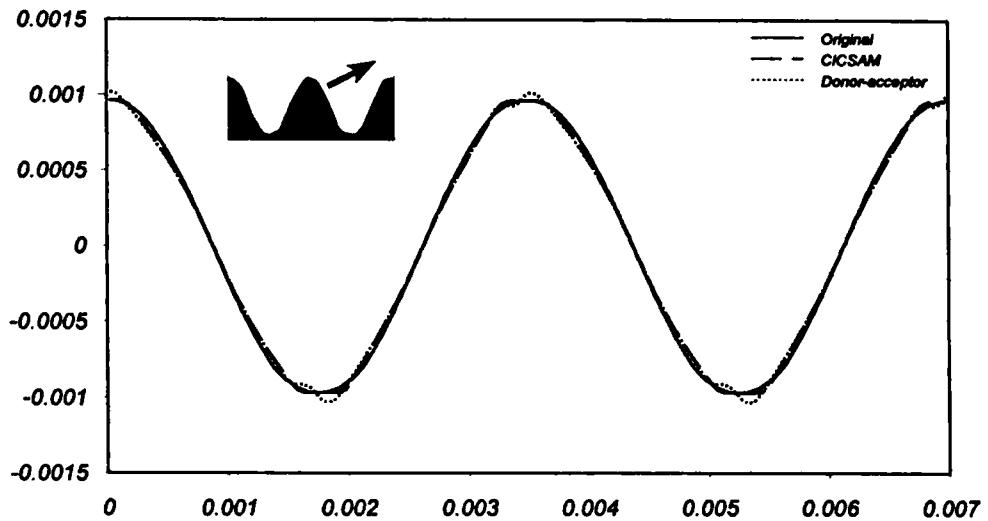


Figure 5.9: A volume fraction wave convected with a uniform oblique velocity field on a (140x60) mesh with cyclic boundary conditions.

The mesh used for the calculation of the results (shown in Figure 5.12) is composed of 100x100 uniformly spaced cells. Initially the rectangle occupies 60x30 cells in the centre of the flow domain. Figure 5.12 (a) shows the interface after half a revolution of the rectangle predicted with Hyper-C (Leonard (1991)). The characteristic waviness caused by a compressive scheme which does not take into account the interface orientation is already visible in the early stages of the calculation. Panel (b) shows the results obtained with CICSAM - the only visible deformation occurs at the corners of the rectangle.

In order to test CICSAM for non-rectangular meshes, the half revolution of the rectangular interface has also been computed on two additional meshes, consisting of hexagons and triangles respectively.

Figure 5.13 shows the results obtained on a mesh consisting of hexagons. Two effects are clearly visible, namely:

- the rounding of the corners and
- the diffusion introduced at the corners on the trailing edge of the rectangle.

The reason for this is explained with the aid of Figure 5.10 (a), showing three hexagons containing part of the interface. The donor and acceptor cells are denoted with a D and A respectively. It has not explicitly been mentioned before, but by determining the fluid distribution in a cell with either the VOF method or CICSAM, the assumption

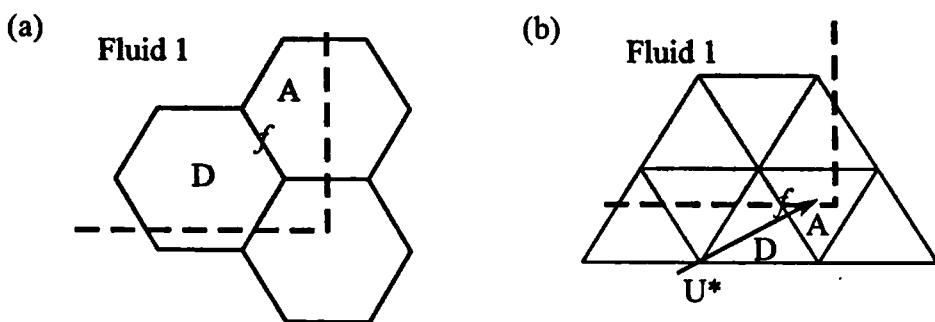


Figure 5.10: Schematic representation of the interface on (a) hexagons and (b) triangles

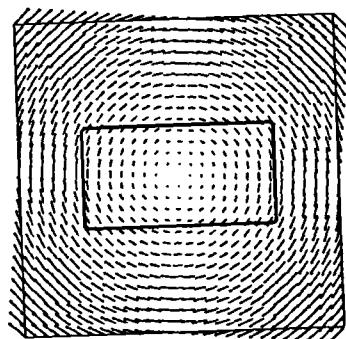


Figure 5.11: Volume fraction rectangle in a circular velocity field

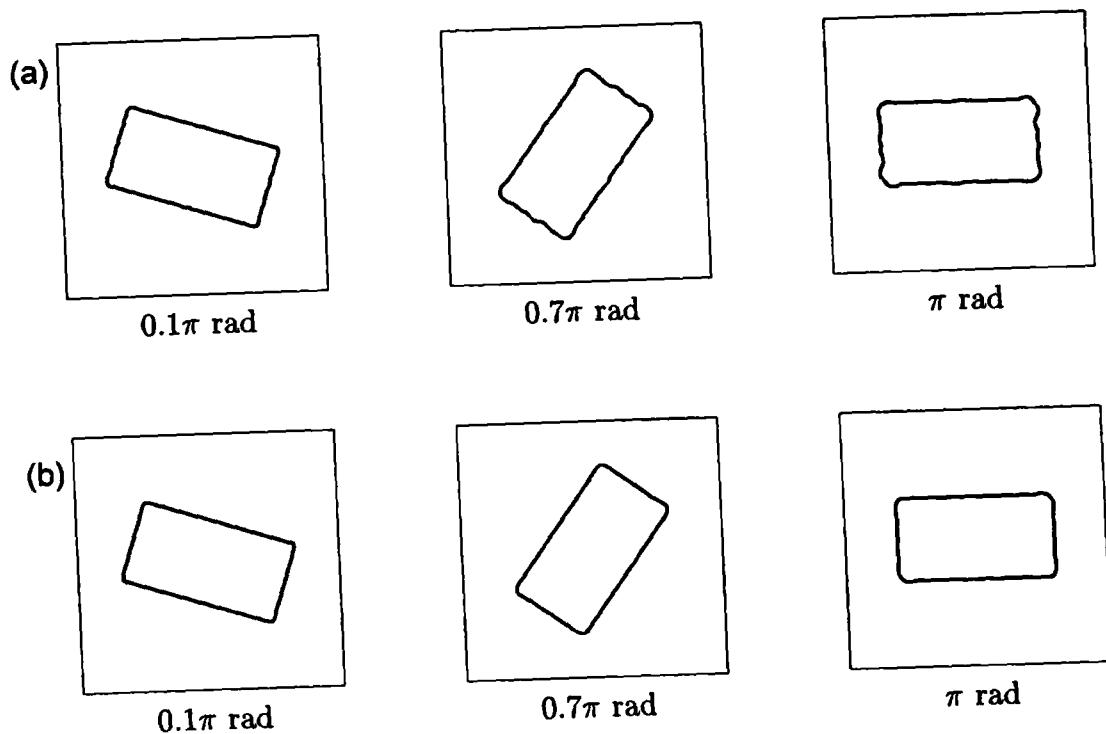
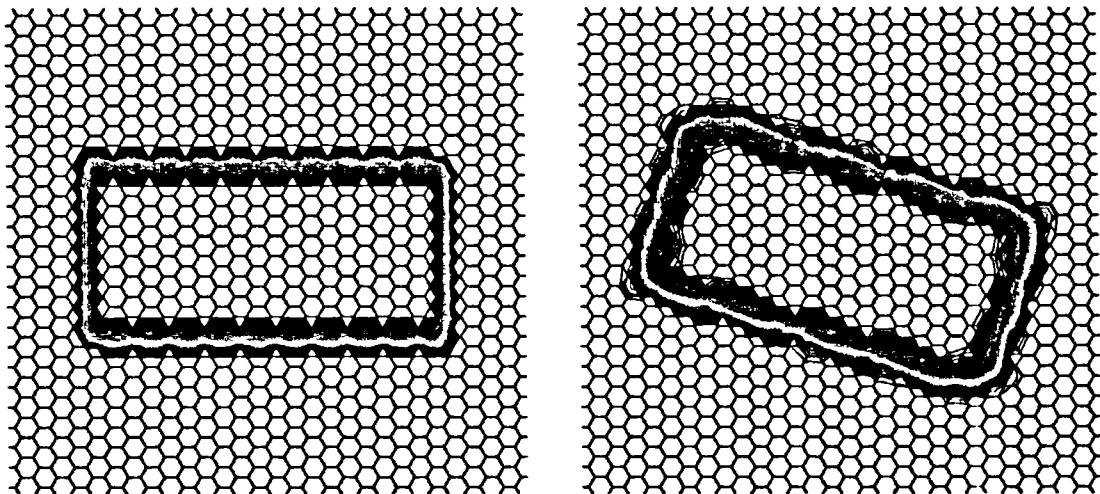
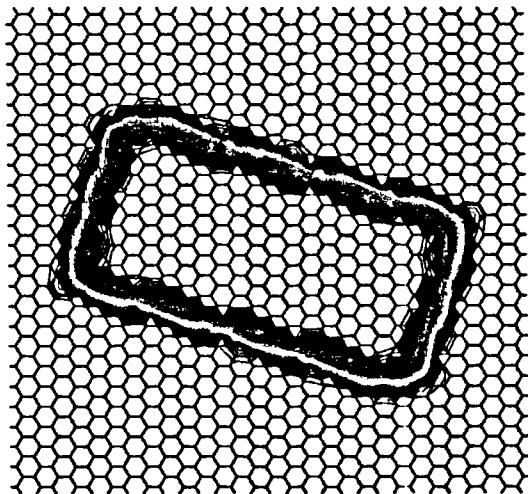


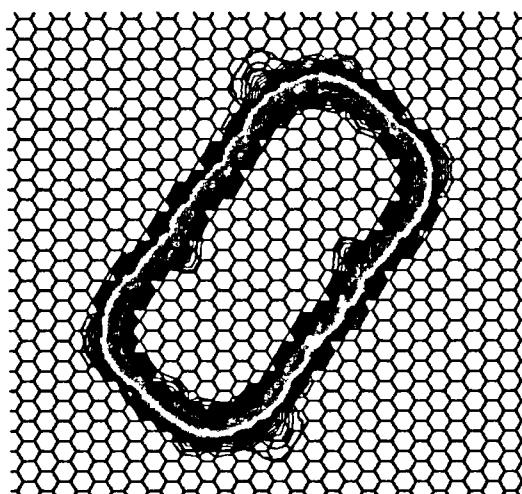
Figure 5.12: Results obtained with (a) Hyper-C and (b) CICSAM



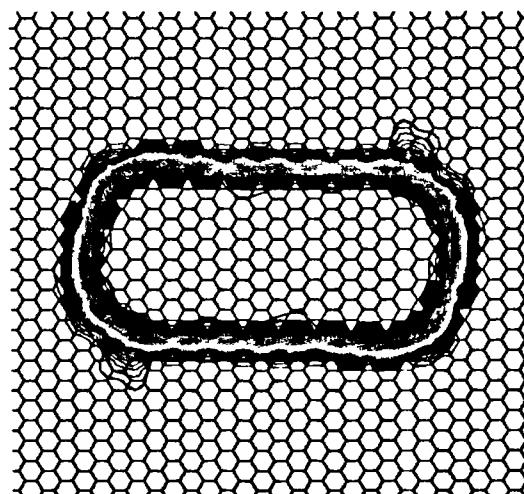
Initial configuration



0.1π rad

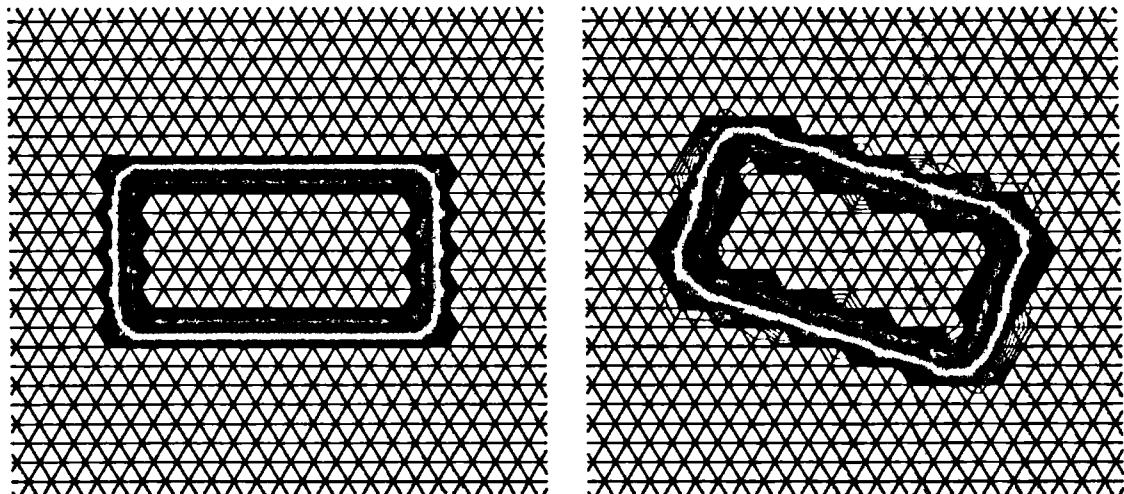


0.7π rad

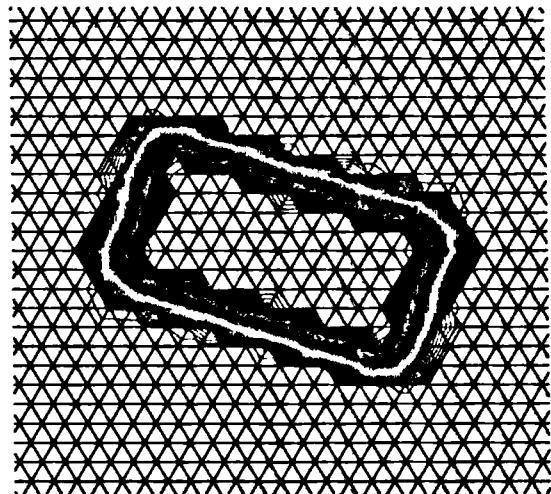


1.0π rad

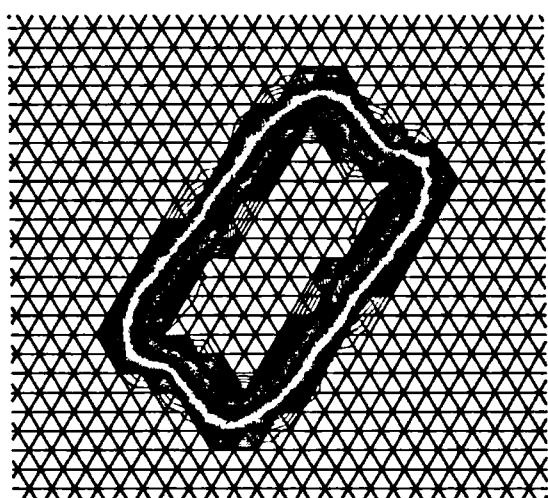
Figure 5.13: The rotation of a rectangle captured on a mesh consisting of hexagons



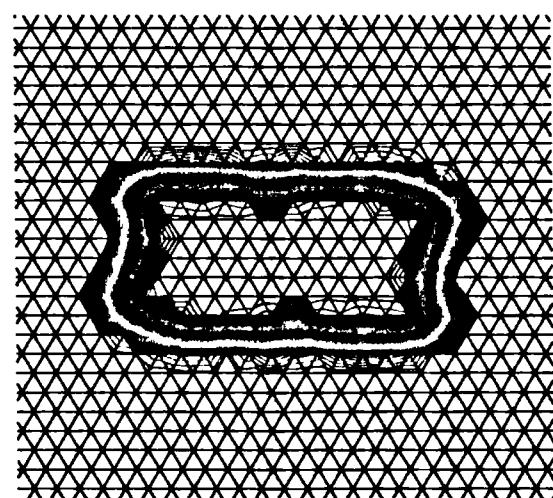
Initial configuration



0.1π rad

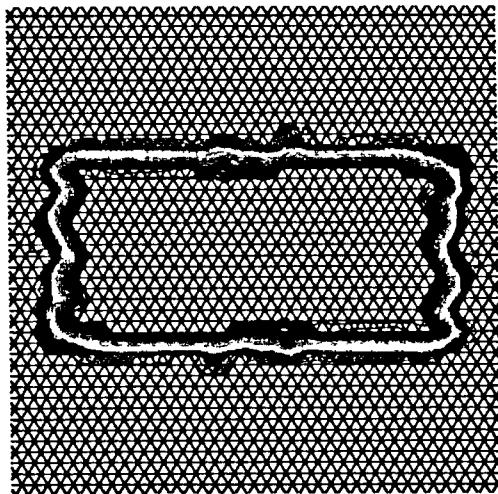


0.7π rad

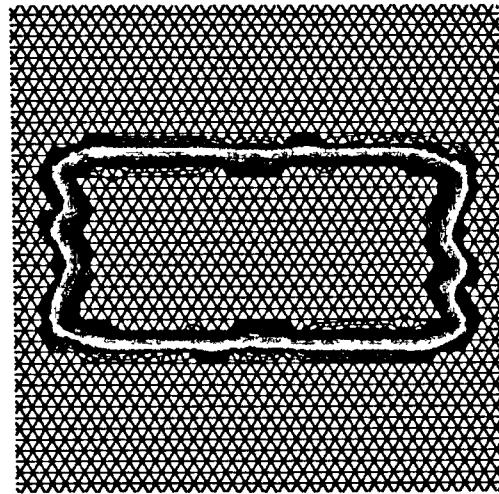


1.0π rad

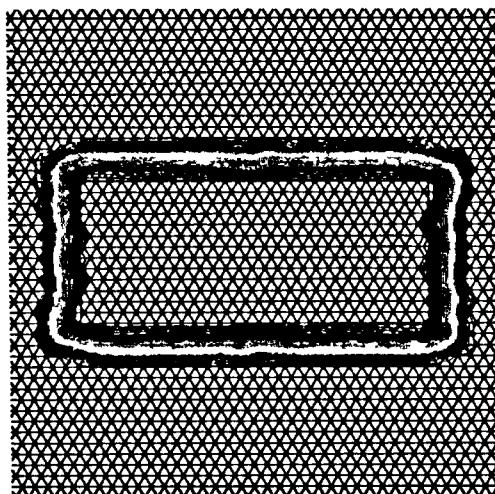
Figure 5.14: The rotation of a rectangle captured on a mesh consisting of triangles



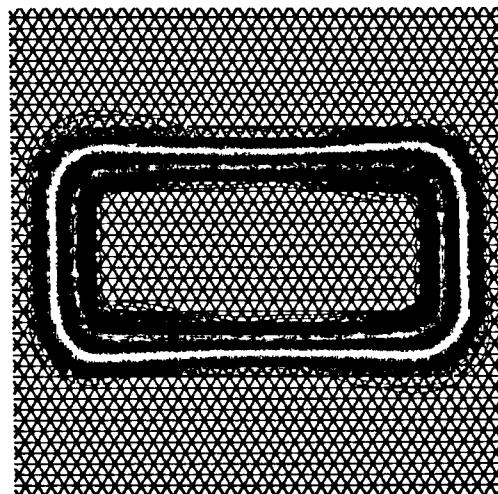
$k_\gamma = 10.0$



$k_\gamma = 1.0$



$k_\gamma = 0.8$



$k_\gamma = 0.0$

Figure 5.15: The half rotation of a rectangle obtained with different values of k_γ

is made that if two neighbouring cells contain part of the interface, then the interface cuts through the face shared by them. Theoretically this assumption is acceptable, because the mesh can always be refined to a level where it holds. As indicated in Figure 5.10 (a) this assumption is most prone to failure on meshes where the control volume faces meet at angles larger than 90° . In the case of the rotating rectangle, ignoring the existence of sharp corners eventually causes the rounding of the corners and their diffusion on the trailing edges.

Figure 5.14 shows the results of a rotating rectangle obtained on a mesh consisting of triangles. Waves seem to appear on the short sides of the rectangle. This worsens with a higher resolution mesh, as will be shown next, together with a remedy to reduce this effect.

In Section 4.6, k_γ was introduced as a constant to control the dominance of Hyper-C and UQ, where for large values of k_γ , CICSAM reduces to Hyper-C and for $k_\gamma = 0$, it reduces to UQ. The results obtained with different values of $k_\gamma \geq 0$ for a half revolution of the rectangle on a finer resolution mesh consisting of triangular cells are presented in Figure 5.15. From this figure it is clear that good results can be obtained by reducing the recommended value of $k_\gamma = 1$ to $k_\gamma = 0.8$, however too much reduction causes a diffused interface as shown in the lower right panel.

The reason for the waves on a mesh consisting of triangular control volumes can be explained with the aid of Figure 5.10 (b). As before, the donor and acceptor cells are denoted with a D and A respectively and U^* denotes the predicted upwind cell. For the fluid configuration shown in Figure 5.10 (b), the area around U^* is occupied by fluid 2 and too much downwinding removes fluid 1 from the donor cell resulting in a wavy pattern. A relaxation on the amount of downwinding used reduces the occurrences of the small waves on such meshes.

Evidently CICSAM is less accurate on meshes consisting of control volumes other than cubes, but is still more accurate on these meshes than the other methods discussed in the text, even when they are applied to cubic control volume meshes (see Figure 1.10 (b)).

5.3.5 Surface tension

As noted before, surface tension operates at the interface and is responsible for the spherical shape of small bubbles and droplets. A numerical test illustrating this phenomenon is to apply it to an initially square 'bubble' (Brackbill *et al.* (1992)). The non-uniform distribution of the curvature at the interface tends to make it change into a circle.

The time evolution of the changes is shown in Figure 5.16. Initially the largest velocities occur at the corners of the square as a result of the high surface forces produced at these positions, where the curvature is a maximum. The top left panel of Figure 5.16 shows the velocity vectors pulling inward on the diagonal of the flow domain and pulling outward at the other parts of the surface, trying to minimise the surface area of the interface. The lower left panel shows an intermediate stage of the evolution, where the interface *overshoots* the stable shape, thereby inducing an oscillation, which is eventually damped out by the presence of the diffusion term in the momentum equations. The last panel presents the circle with all the velocities damped out.

The area of the volume fraction square illustrated in Figure 5.16 is $\pi \times 10^{-6} \text{m}^2$, which is equivalent to the area of a circle with a radius $R=1\text{mm}$. The upper left panel of Figure 5.17 gives the one hundred volume fraction contours between zero and unity associated with the circle obtained in Figure 5.16. The upper right panel of Figure 5.17 presents the curvature for this circle, predicted with the CSF model (Brackbill *et al.* (1992)). As explained in Section 2.4, the curvature is negative because fluid 2 lies on the concave side of the interface. The curvature associated with this circle is $\kappa=1000\text{m}^{-1}$. The predicted mean value corresponds with this, but as noted before discretisation errors occur at local points, resulting in higher local curvature values. This is consistent with the results presented in Figure 3.2. The contours for the predicted pressure field are shown in the lower left panel of Figure 5.17 and the pressure distribution on a diagonal is shown in the lower right panel. The error in the prediction

of the pressure jump across the interface is 3%, which is in the range of the 1%-5% reported by Brackbill *et al.* (1992).

A problem associated with the higher localised curvature values outlined above, is the existence of a low amplitude velocity field surrounding the interface, even in the steady-state case. Lafaurie *et al.* (1994) refer to these velocities as inevitable parasite currents. The magnitude of these velocities scales with the viscosity and surface tension (Lafaurie *et al.* (1994)). A dimensional analysis of the viscous and surface forces reveals that:

$$K = \frac{u_{\max} \mu}{\sigma} \quad (5.1)$$

where K itself has no physical meaning, but is useful for the prediction of the behaviour of these parasite currents. A detailed analysis of K and the dimensionless radius of bubbles is presented elsewhere (Lafaurie *et al.* (1994)) and will not be repeated here. However, the existence of these parasite currents and their scaling with the surface tension coefficient and viscosity will be discussed next with the aid of Figure 5.18.

The upper left panel of Figure 5.18 shows the circle used in the previous example. The ratio $\mu/\sigma = 1$ holds for this case and small parasite currents are visible around the interface. The next panel presents the results obtained with a 10 times higher surface tension coefficient. Thus $\mu/\sigma = 0.1$ and the velocity vectors shown are ten times the size of the original vectors. The lower left panel presents the results obtained with a surface tension coefficient 100 times larger, resulting in a ratio of $\mu/\sigma = 0.01$ - the velocity vectors for this case are 100 times larger than the original vectors. The last panel shows the results obtained with both the surface tension coefficient and viscosity 100 times the original values, resulting in a ratio of $\mu/\sigma = 1$ - both this ratio and the magnitude of the velocity vectors correspond with the original vectors shown in the first panel.

These parasite currents increase with decreasing viscosity and increasing surface tension. Therefore the modelling of small scale bubbles and droplets is also restricted to this limitation (Tomiyama (1993b)) as explained later in Section 5.5.

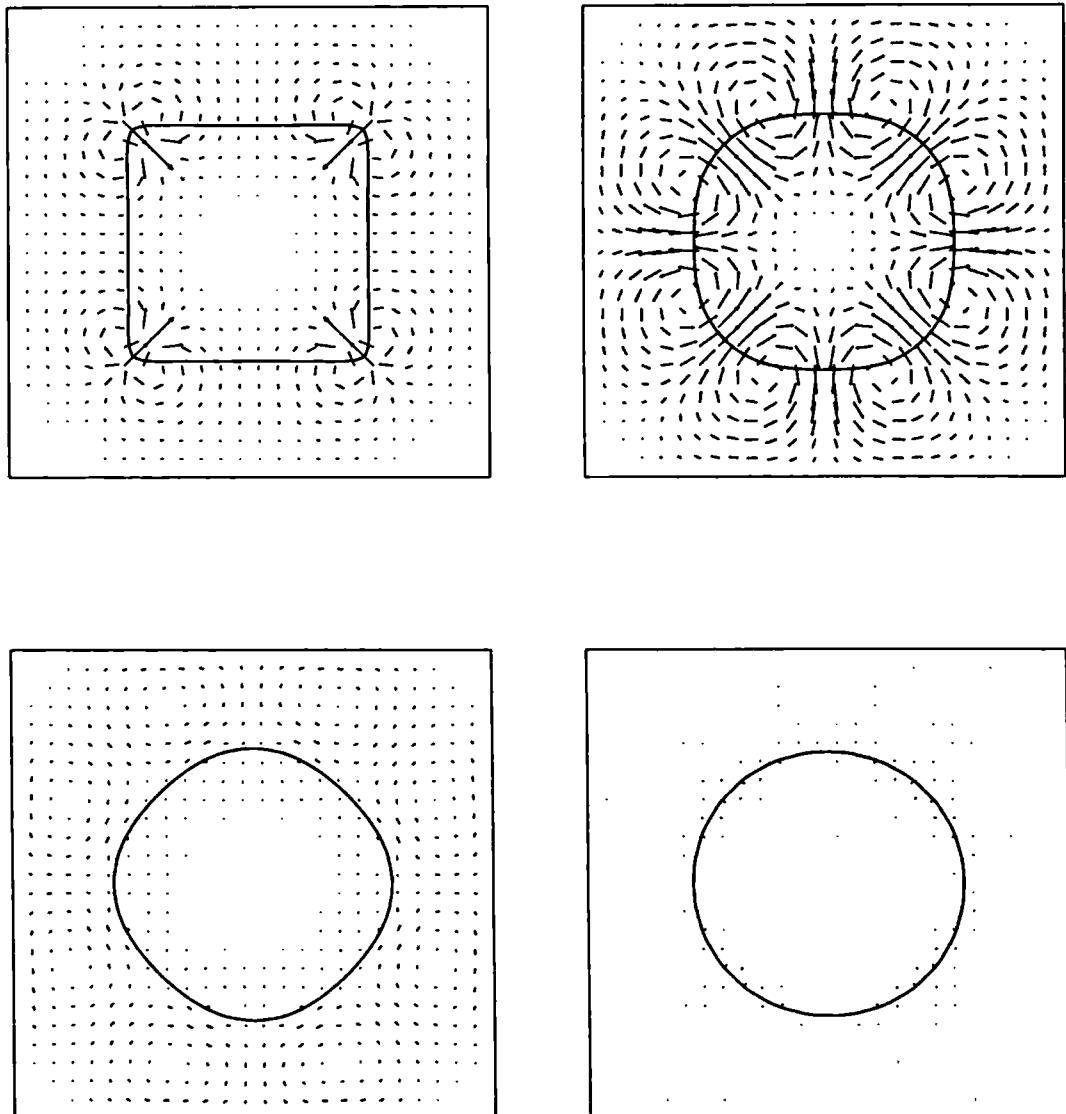


Figure 5.16: Time evolution of the shape change of a square subjected to surface tension forces

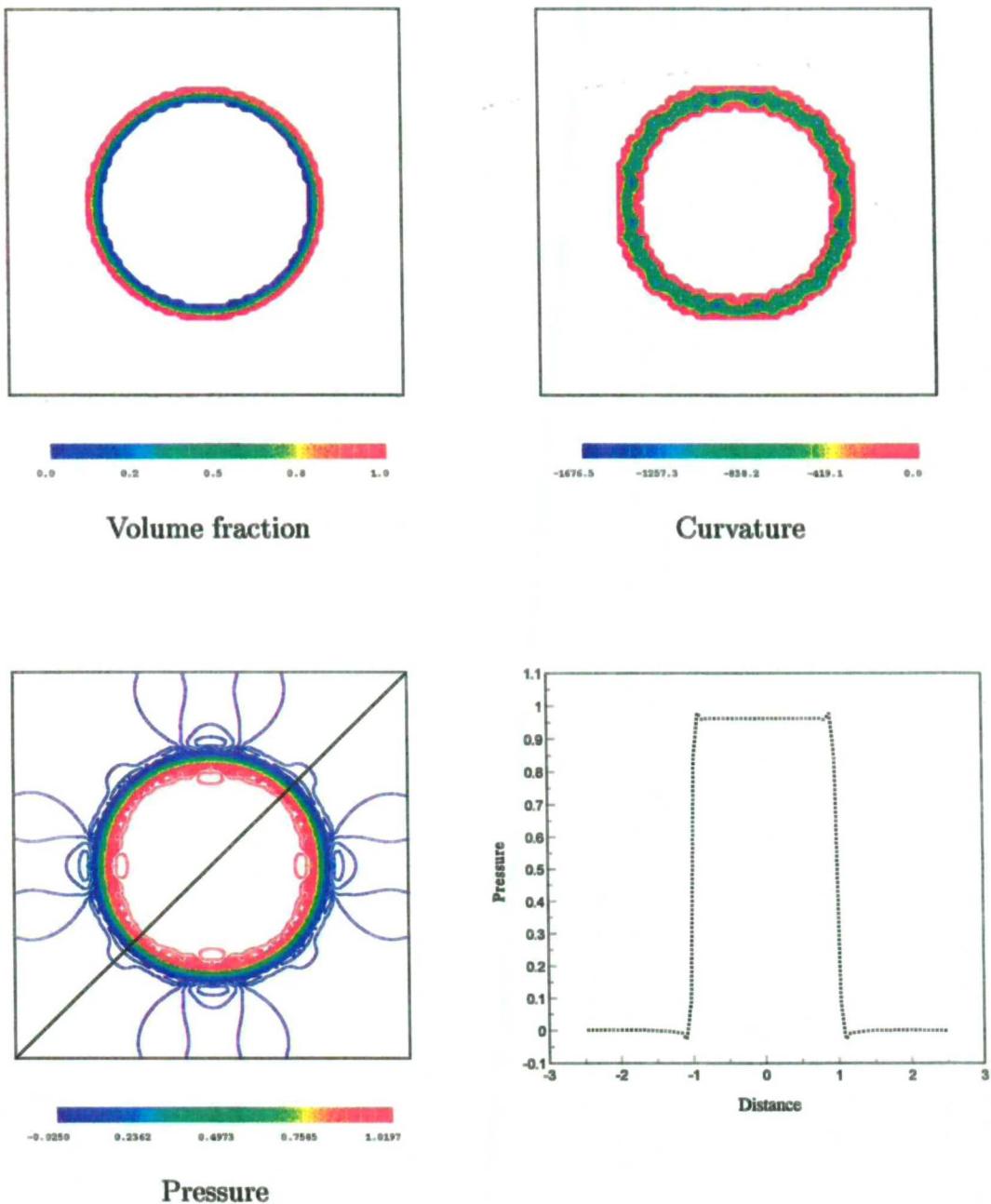
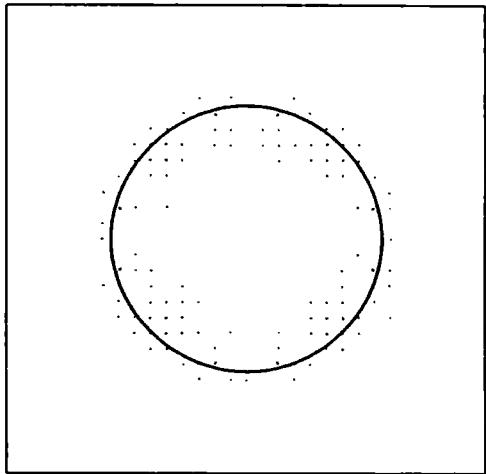
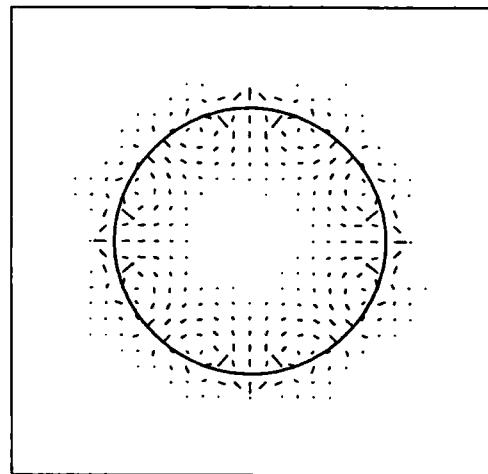


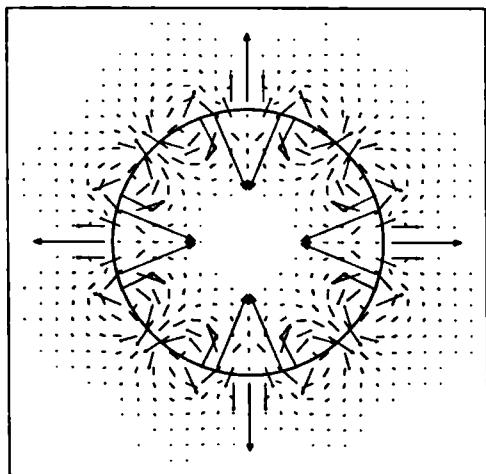
Figure 5.17: Pressure jump induced by surface tension: (a) the volume fraction contours of a circle with radius 1mm, (b) the curvature predicted with the CSF model, (c) pressure contours and (d) the pressure distribution on the diagonal



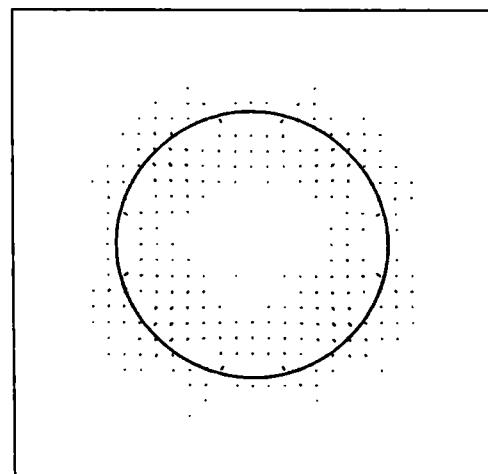
$$\sigma = 10^{-3}, \mu = 10^{-3}, \mu/\sigma = 1$$



$$\sigma = 10^{-2}, \mu = 10^{-3}, \mu/\sigma = 10^{-1}$$



$$\sigma = 10^{-1}, \mu = 10^{-3}, \mu/\sigma = 10^{-2}$$



$$\sigma = 10^{-1}, \mu = 10^{-1}, \mu/\sigma = 1$$

Figure 5.18: Parasite currents at the surface of a circle with radius 1mm for different surface tension coefficients and viscosity values

5.4 Sloshing

The sloshing of a liquid wave with a low amplitude under the influence of gravity was mathematically investigated by Tadjbakhsh & Keller (1960) and used as a test to evaluate the interface tracking methodology presented by Raad *et al.* (1995). It is useful for evaluating the CFD methodology with respect to two different phenomena present in the numerical prediction of interfacial flow, namely:

- the numerical dissipation introduced by the discretisation and
- the ability to conservatively transfer potential energy into kinetic energy and vice versa.

The situation considered is shown in Figure 5.19, which is the same as used by Raad *et al.* (1995). Initially, the quiescent fluid has an average depth of 0.05m, and its surface is defined by one half of a cosine wave with an amplitude of 0.005m. The computational domain used has a base length of 0.1m and a height of 0.065m. Its bottom and sides are treated as slip boundaries and at the top a fixed pressure boundary condition is applied. The viscosity of both fluids is taken as zero and the densities are defined as $\rho_1=1000$ and $\rho_2=1$. The domain is discretised with 160 cells in the horizontal direction and 104 cells in the vertical direction.

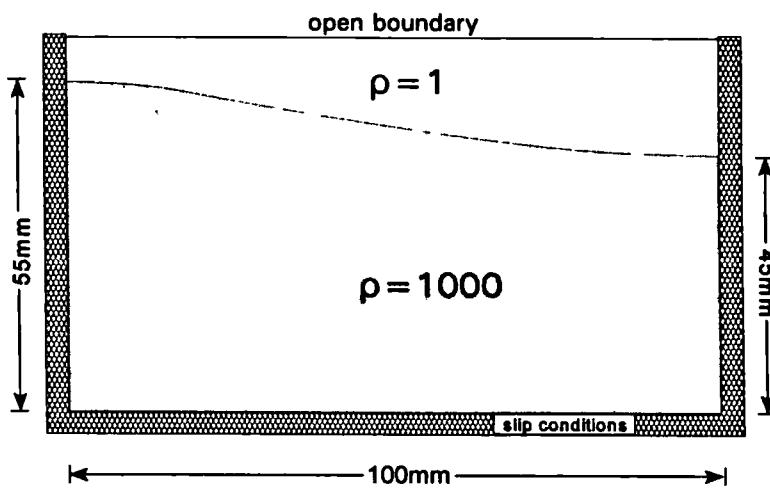


Figure 5.19: Initial geometry of the wave

The fluid begins to slosh solely under the influence of a constant gravitational field set as 9.8m/s^2 in the downward vertical direction. The theoretical period of sloshing of the first mode is (Raad *et al.* (1995)):

$$P = 2\pi\sqrt{gk \tanh(kh)} = 0.3739s \quad (5.2)$$

where k is the wave number and h the average fluid depth.

Waves with higher modes are also present in this system, whereof the most important is the second mode with half the above period. The other higher order modes become less important and do not significantly influence the behaviour of the first modes. Thus, the wave position for the even periods should correspond with each other and the odd periods should show similar behaviour (Tadjbakhsh & Keller (1960)).

The present numerical results obtained for the first period are shown in Figure 5.20. Initially the whole system is at rest. After a quarter of a period the potential energy of the system has been transferred to kinetic energy and the velocities reach their maxima. After a half period all the kinetic energy has been transferred back into potential energy with the velocity almost back to zero. The middle right and bottom left panels show the wave on its way back to its original position shown in the lower right panel.

Figure 5.21 shows plots of the position of the interface at the left boundary against time for the first six periods. The frequency corresponds with the theoretical one, so do the amplitudes of the even periods. Figure 5.22 shows the wave position for the first six periods superimposed on each other. The second mode on the odd periods is clearly visible. Evaluation of the accuracy of the results presented in the last two figures will be addressed next.

Since the exact solutions of the even periods are known, it is possible to quantify the accuracy of CICSAM by direct comparison of the interface positions at a given instant. Furthermore, the period of sloshing is also known, which makes it possible to quantify the temporal error as well.

Raad *et al.* (1995) used two norms, the temporal error and the root mean square (RMS) spatial error for assessing accuracy. The percent temporal error is defined as:

$$100 \frac{(t_s - t_t)}{t_t} \quad (5.3)$$

where t_s and t_t represent the simulation and theoretical times respectively. The simulation time is taken as the time when the interface reaches its highest point against the left boundary, and the theoretical time is the period of sloshing multiplied by the amount of periods.

The RMS spatial error is defined as:

$$\frac{100}{a\sqrt{M}} \left[\sum_{m=1}^M (y_s - y_t)^2 \right]^{\frac{1}{2}} \quad (5.4)$$

where m is the cell counter in the horizontal direction, M the number of cells in the horizontal direction, y_s the predicted interface height, y_t the theoretical interface height and a the wave amplitude.

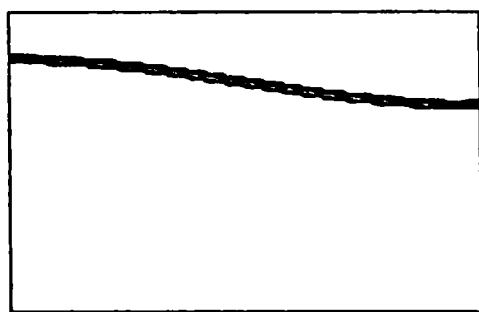
The errors calculated with the above norms for the first three even periods are presented in Table 5.1. The errors obtained by Raad *et al.* (1995) for an effective mesh resolution of 240 cells in the horizontal direction are also given in Table 5.1. Raad *et al.* (1995) used an interface tracking technique with a refined mesh at the interface. For the present study a uniform mesh with 160 cells in the horizontal direction has been used (50% coarser than that used by Raad *et al.* (1995)). The effective time step used in the present study is four times larger than the $\delta t = 2.5 \times 10^{-4}$ s used by Raad *et al.* (1995).

The growth of the Spatial error, possibly caused by the presence of the higher order modes, has not been investigated as the aim of this case has only been the investigation of the performance of the present methodology in comparison with those of others.

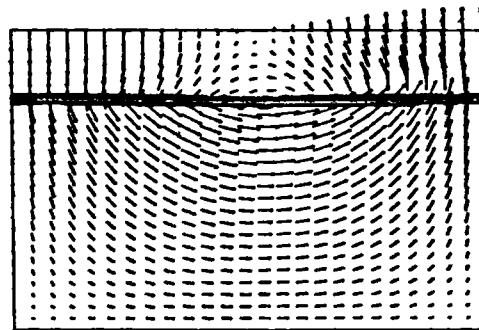
Theoretical time	Temporal error (%)		Spatial error (%)	
	Raad <i>et al.</i> (1995)	CICSAM	Raad <i>et al.</i> (1995)	CICSAM
2 P	-0.44	0.0	2.0	1.9
4 P	-0.39	-0.75	2.8	2.0
6 P	-0.29	0.04	3.5	3.7

Table 5.1: Comparison of temporal and spatial errors

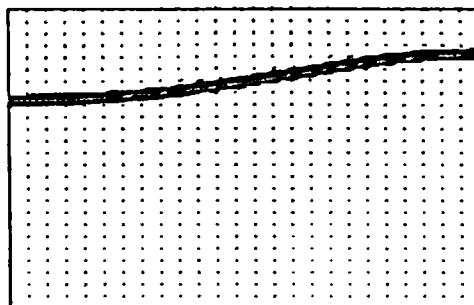
This test not only shows the accuracy with which a low amplitude wave can be captured with CICSAM, but also reveals that volume based methods produce results that compare very well with surface-based methods, whilst being computationally cheaper.



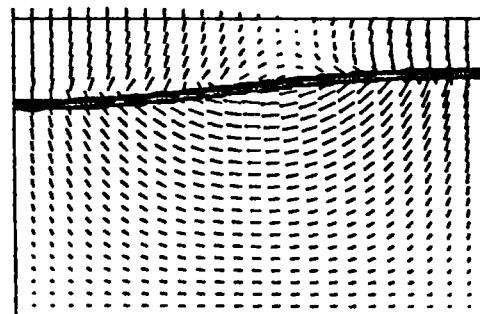
$t = 0.00s$



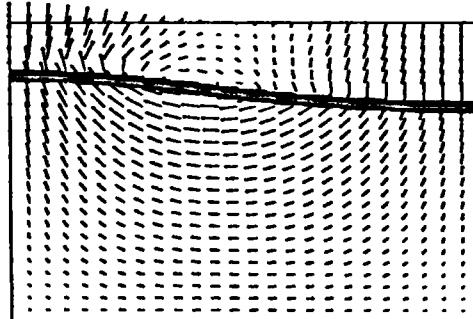
$\frac{1}{4}$ period



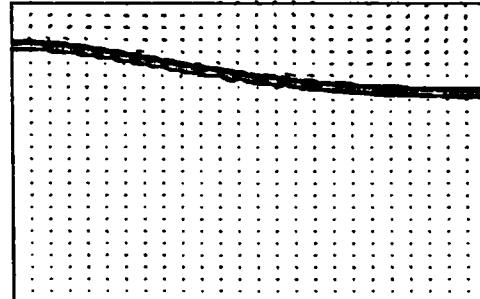
$\frac{1}{2}$ period



$\frac{5}{8}$ period



$\frac{7}{8}$ period



1 period

Figure 5.20: Plots of the wave position and velocity vectors for the first period of the sloshing of an inviscid liquid under the influence of gravity

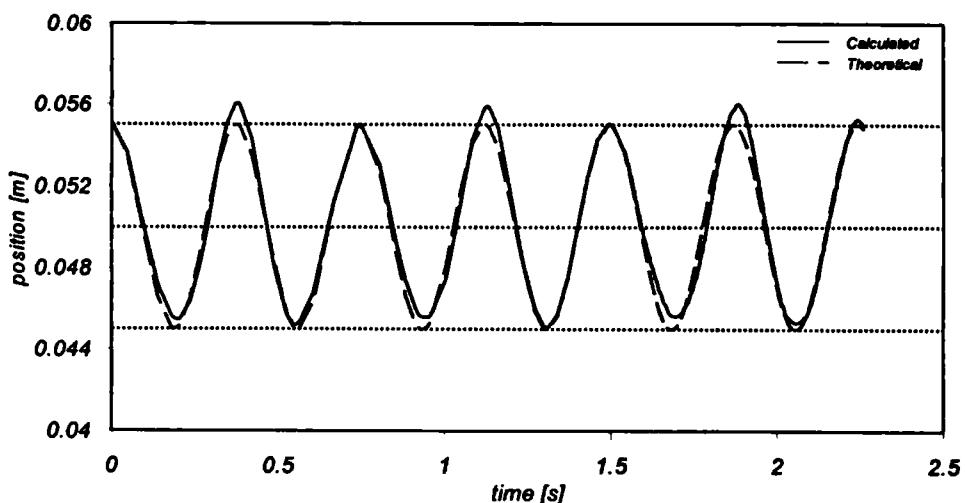


Figure 5.21: Position of the interface at the left boundary plotted against the time

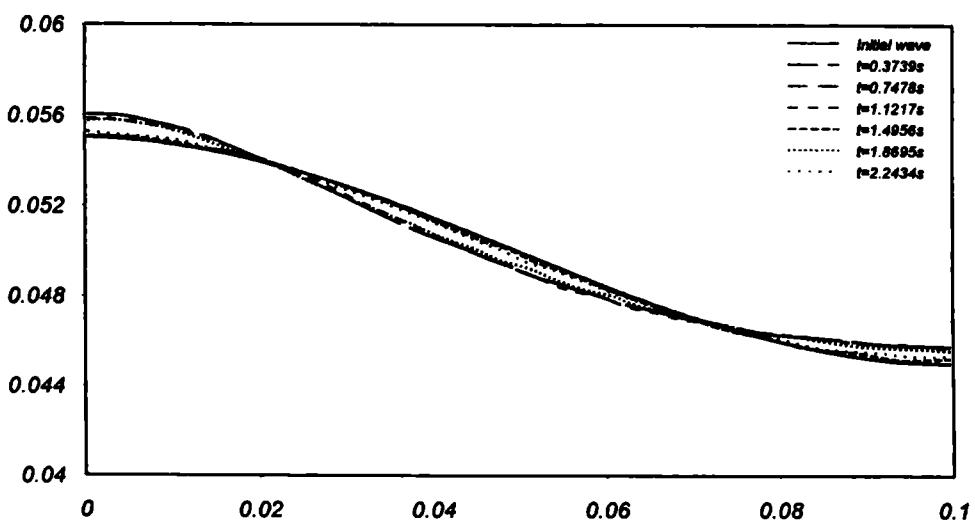


Figure 5.22: Wave positions for the first six periods

5.5 Taylor bubble

The case of a cylindrical gas bubble which occupies most of the cross section of a vertical tube filled with a liquid is another area of interest. The rise velocity of the bubble and the shape of the bubble depend on the fluids involved and the diameter of the tube. White & Beardmore (1962) conducted a dimensional analysis using a large amount of available experimental data and proposed a graphical correlation (shown in Figure 5.23) of the terminal rising velocity of cylindrical air bubbles in a vertical tube. The three dimensionless numbers involved are the Eötvös number defined as:

$$E_o = \frac{g(\rho_1 - \rho_2)D^2}{\sigma} \quad (5.5)$$

the Morton number defined as:

$$M = \frac{g\mu^4(\rho_1 - \rho_2)}{\rho_1^2 \sigma^3} \quad (5.6)$$

and the Froude number defined as:

$$F_r = \frac{u_r}{\sqrt{\frac{gD(\rho_1 - \rho_2)}{\rho_1}}} \quad (5.7)$$

where D is the diameter of the tube and u_r , the terminal rise velocity of the bubble. For the graphical correlation, shown in Figure 5.23, ρ_2 the density of the air is neglected.

The Eötvös number represents the contribution of the effects of surface tension and buoyancy. The Morton number, which is sometimes referred to as the property group, measures the relative importance of viscous and surface tension forces. The Froude number is the normalised bubble rise velocity relative to the inertially dominated value. According to Figure 5.23 the Froude number (the vertical axis) is uniquely determined by the Eötvös (the horizontal axis) and Morton (the contours on the graph) numbers.

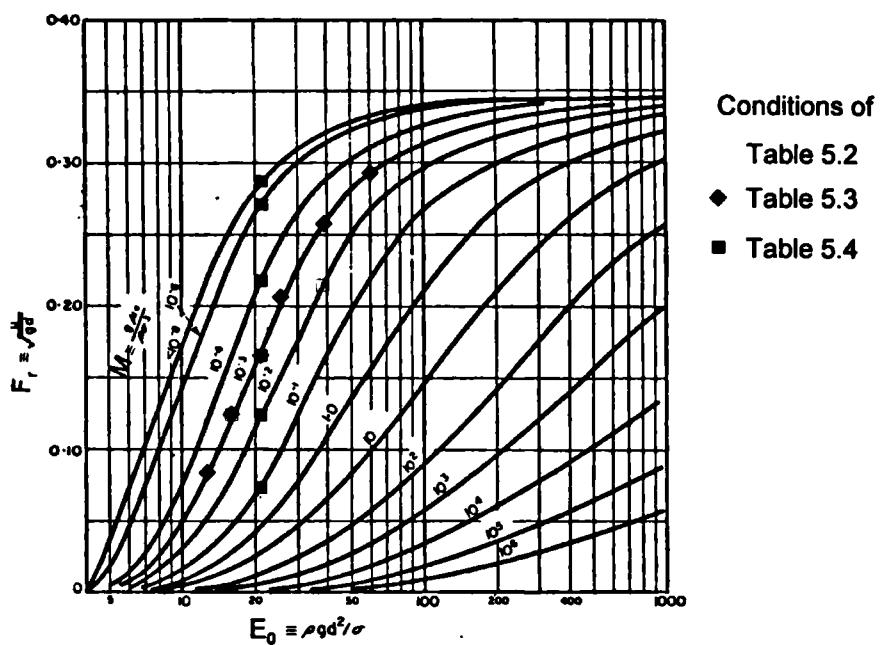


Figure 5.23: Correlation of the rise velocity of cylindrical bubbles in liquids in vertical tubes (from: White & Beardmore 1962, Chemical Engineering Science, Vol. 17, Fig. 5, p. 356).

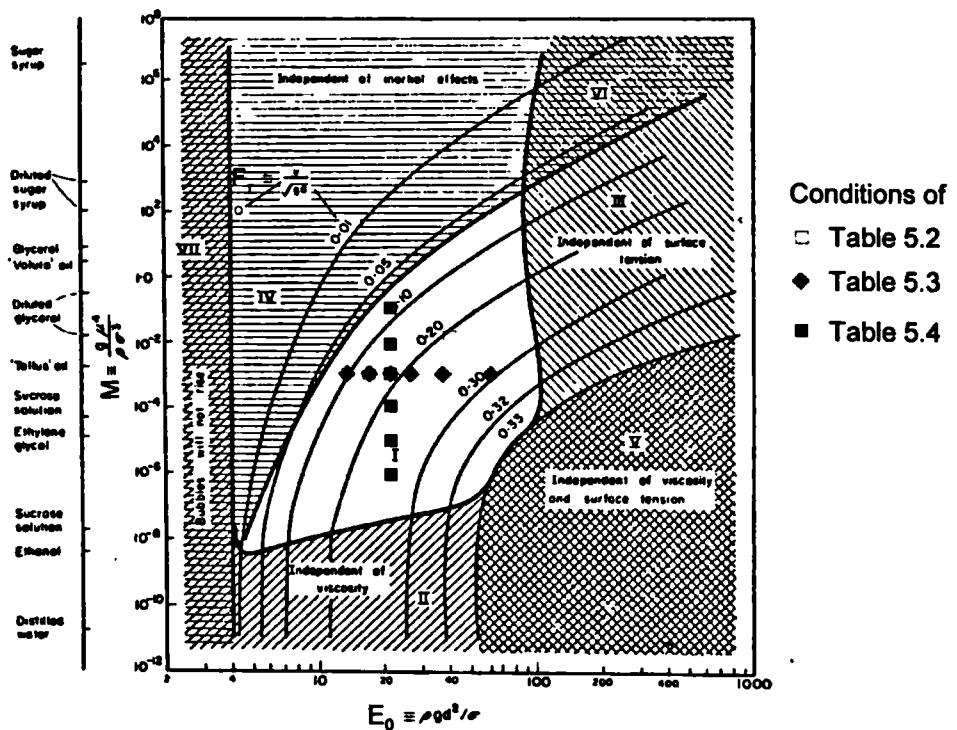


Figure 5.24: Cross plot showing the regions in which the effect of some variables become unimportant (from: White & Beardmore, 1962, Chemical Engineering Science, Vol. 17, Fig. 6 p. 357.)

Figure 5.24 presents a cross plot showing the regions in which the effect of some variables become unimportant. The Eötvös number is represented on the horizontal axis and the Morton numbers on the vertical axis and different Froude number contours are plotted on the graph. The white area in the centre of the graph (region I) represents comparable contributions from all the different flow parameters, namely, viscous effects, inertial effects and surface tension. Region II is independent of viscous effects, region III independent of surface forces, region IV independent of inertial effects, region V independent of viscosity and surface tension and region VI independent of inertial and interfacial forces. In the case of region VII, no bubble movement occurs. For instance, according to this figure, the rise of an air bubble in water (with a Morton number of $M=10^{-11}$) is always independent of viscous effects.

Experimental results on the shapes and rise velocities of Taylor bubbles within the laminar flow regime are presented by Tomiyama *et al.* (1994), who also use these results to evaluate the feasibility of using the VOF method for the prediction of the flow behaviour of a single air bubble in a vertical tube occupied by a stagnant liquid. In an earlier work on the numerical analysis of bubble motion with the VOF method, Tomiyama *et al.* (1993b) reported a difficulty in obtaining a numerically stable solution when $M<10^{-7}$. The reason for this does not lie within the VOF method's inability to predict such flows, but in the existence of the parasite currents introduced when the flow behaviour is dominated by surface forces, as explained in Section 5.3.5. According to Figure 5.24 flows with $M<10^{-7}$ mostly fall within the regime where the flow behaviour is independent of viscous effects.

As noted before, the study of Taylor bubbles belongs to the wider topic of slug flow, where a continuous string of bubbles is present in the tube. The flow pattern of such flows is complicated - in some cases small air bubbles are shed at the tail of the bubble and the next bubble is fed by these bubbles. In other instances the flow may be highly turbulent, or even transitional in nature - laminar at the nose of the bubble and turbulent at its tail. The present investigation of a single bubble in a stagnant liquid is not to

reveal new phenomena regarding such flows, but to evaluate the behaviour of CICSAM and its ability to predict these flows.

The cases presented in this section are divided into two groups:

- The shapes and rise velocity of two Taylor bubbles will be compared with experimentally determined data in order to illustrate the accuracy with which CICSAM can predict this type of flow.
- The converged solutions for different Eötvös and Morton numbers will be presented to illustrate the general behaviour CICSAM and to determine a flow regime within which the present methodology operates with reasonable accuracy.

The setup used in the present study is the same as used by Tomiyama *et al.* (1994). Normally the flow is kept stationary and the bubble rises through it. Here, the computational frame of reference is attached to the bubble, so in effect the bubble is kept stationary and the tube is moved instead. In order to keep the nose of the Taylor bubbles stationary, the inlet velocities are adjusted dynamically. Thus, the rise velocity is equal to the inlet velocity. The computational domain and boundary conditions are illustrated in Figure 5.25.

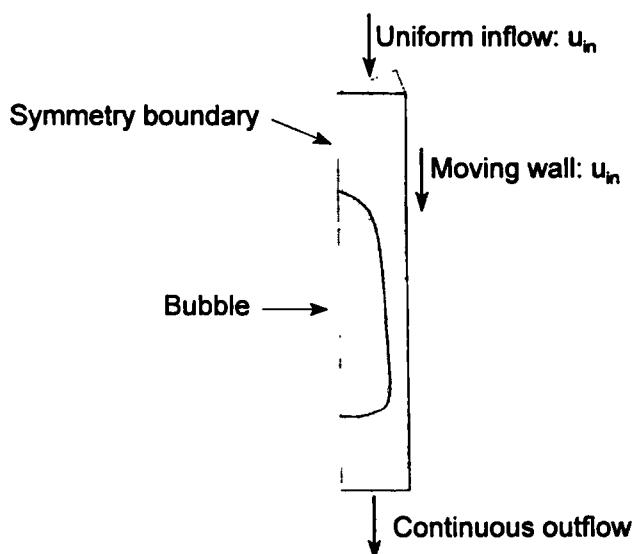


Figure 5.25: Schematic representation of the computational setup and the boundary conditions

The flow is axi-symmetric, enabling the tube to be modelled as a wedge. Since the velocity gradients are large in the liquid film region surrounding the Taylor bubble, a finer-resolution non-uniform mesh is employed in this region. The height to radius ratio of the flow domain is 9:1, with a lower mesh resolution in the vertical direction near the inlet and outlet. The mesh density (illustrated in the left panel of Figure 5.26) corresponds with that used by Tomiyama *et al.* (1994).

First, calculations for two Taylor bubbles shapes with experimentally determined shapes will be presented. The flow conditions as well as the error u_ϵ in the predicted rise velocities for these two cases are given in Table 5.2. They fall in regime I of Figure 5.24, i.e. the bubble rise is influenced by both viscous and surface tension forces.

Case	ρ_1	ρ_2	$\mu_1 \times 10^{-3}$	$\mu_2 \times 10^{-5}$	$\sigma \times 10^{-2}$	$D \times 10^{-3}$	M	E_o	$u_\epsilon \%$ error
1	1311	1.22	97	1.8	7.6	10.2	0.0015	17.59	2.1
2	1313	1.23	169	1.8	7.38	15.1	0.015	39.76	7.7

Table 5.2: Conditions for the comparison of bubble shapes and rise velocities

Figures 5.26 and 5.27 present the time evolution of the bubbles, starting from a simple assumed initial shape, for the two different flow conditions. In Figures 5.28 and 5.30 the final shapes of these two cases are compared with the experimental profiles presented by Tomiyama *et al.* (1994). The latter have been obtained by digitising enlargements of the shapes presented by Tomiyama *et al.* (1994), who in turn digitised them from enlarged photographs: this procedure undoubtedly introduces some uncertainty, which should be borne in mind in the comparison.

A more accurate accuracy measure can be made by comparing the predicted rise velocities with those presented by White & Beardmore (1962). The plots of the predicted velocities against time for the two cases, are presented in Figures 5.29 and 5.31 respectively and the errors u_ϵ in the prediction of the rise velocity are given in Table 5.2. These graphs clearly show that the calculation reaches a steady state.

The error in the prediction of the rise velocity is closely related to the magnitude of the Eötvös number, as will now be demonstrated. In order to investigate this, calculations are performed for:

- six different Eötvös numbers, ranging between 14 and 60 and a fixed Morton number $M=0.0015$ and
- six different Morton numbers ranging between 10^{-1} and 10^{-6} and a fixed Eötvös number $E_o=21.21$.

The flow conditions, as well as the error in the predicted rise velocities for the above two cases, are given in Tables 5.3 and 5.4 respectively. The final predicted shapes, as well as the velocity vectors for both the above case are presented in Figures 5.32 and 5.33 respectively.

The predicted tendencies of the bubble shapes for different Eötvös numbers, shown in Figure 5.32, correlate with the experimental observations of Tomiyama *et al.* (1994) who report a flattening of the bubble tail with increasing Eötvös number. For $E_o=14.31$ the predicted tail starts to diffuse numerically. The reason for this is because the bubble is kept stationary and divergent flow (that is flow in both directions away from the interface) occurs across the interface. These velocities eventually start to pull in both directions and pull the interface apart. This flow phenomenon occurs for example near a wall when the interface orientation is normal to the wall and fluid enters a control volume tangentially to the interface and diverts in both directions normal to the interface.

A possible remedy to this phenomenon is to apply a slightly higher κ_y in the vicinity of the tail. A more general approach would be to include a test for these types of flows and to overwrite the switch based on the interfacial orientation and direction of motion when such a scenario occurs.

For $E_o=60.0$ the real bubble starts to shed small air bubbles at its tail. This phenomenon cannot be resolved accurately on a wedge, because every bubble is in effect a three

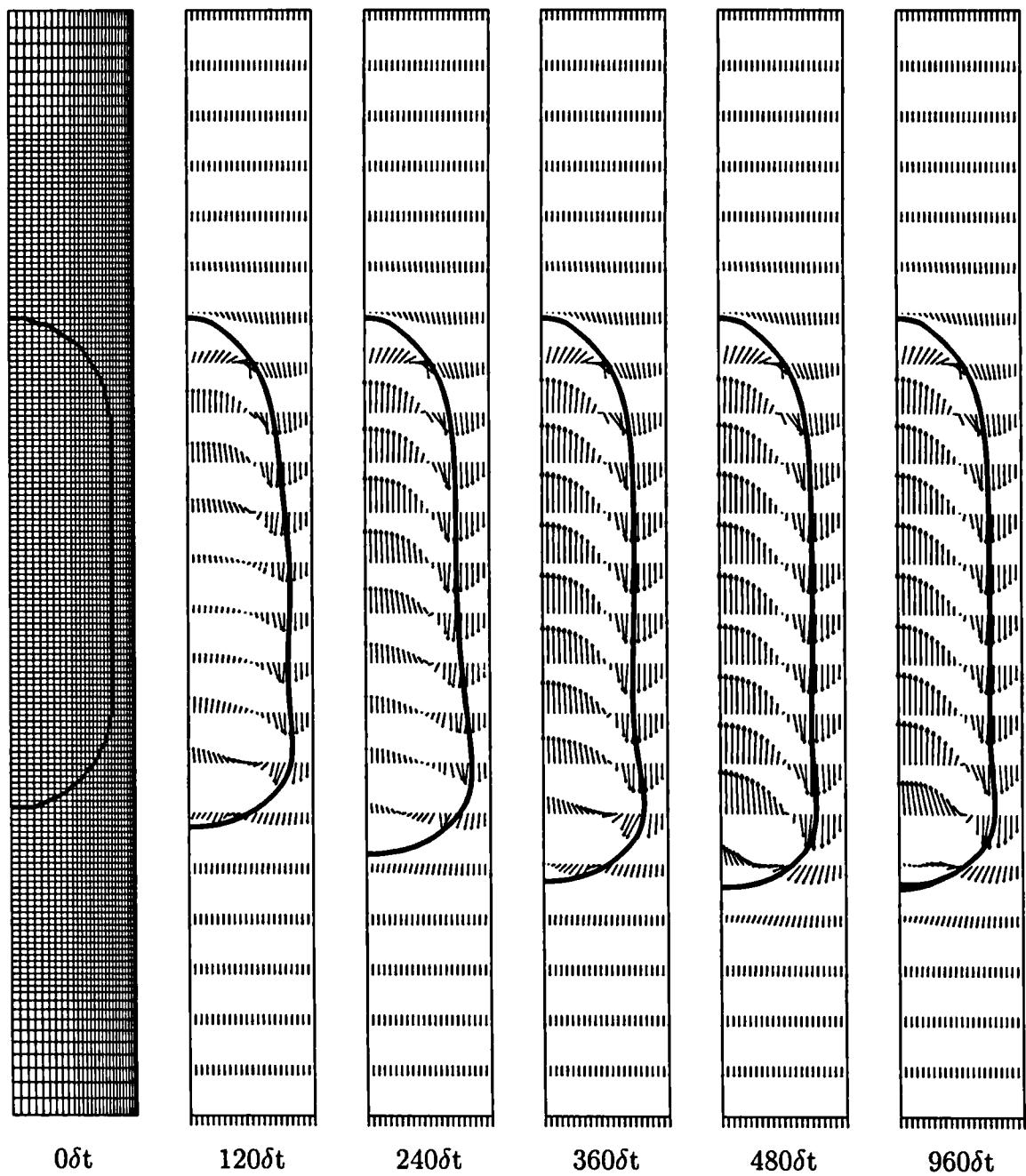


Figure 5.26: Time evolution of a Taylor bubble ($E_o=17.59$, $M=0.0015$)

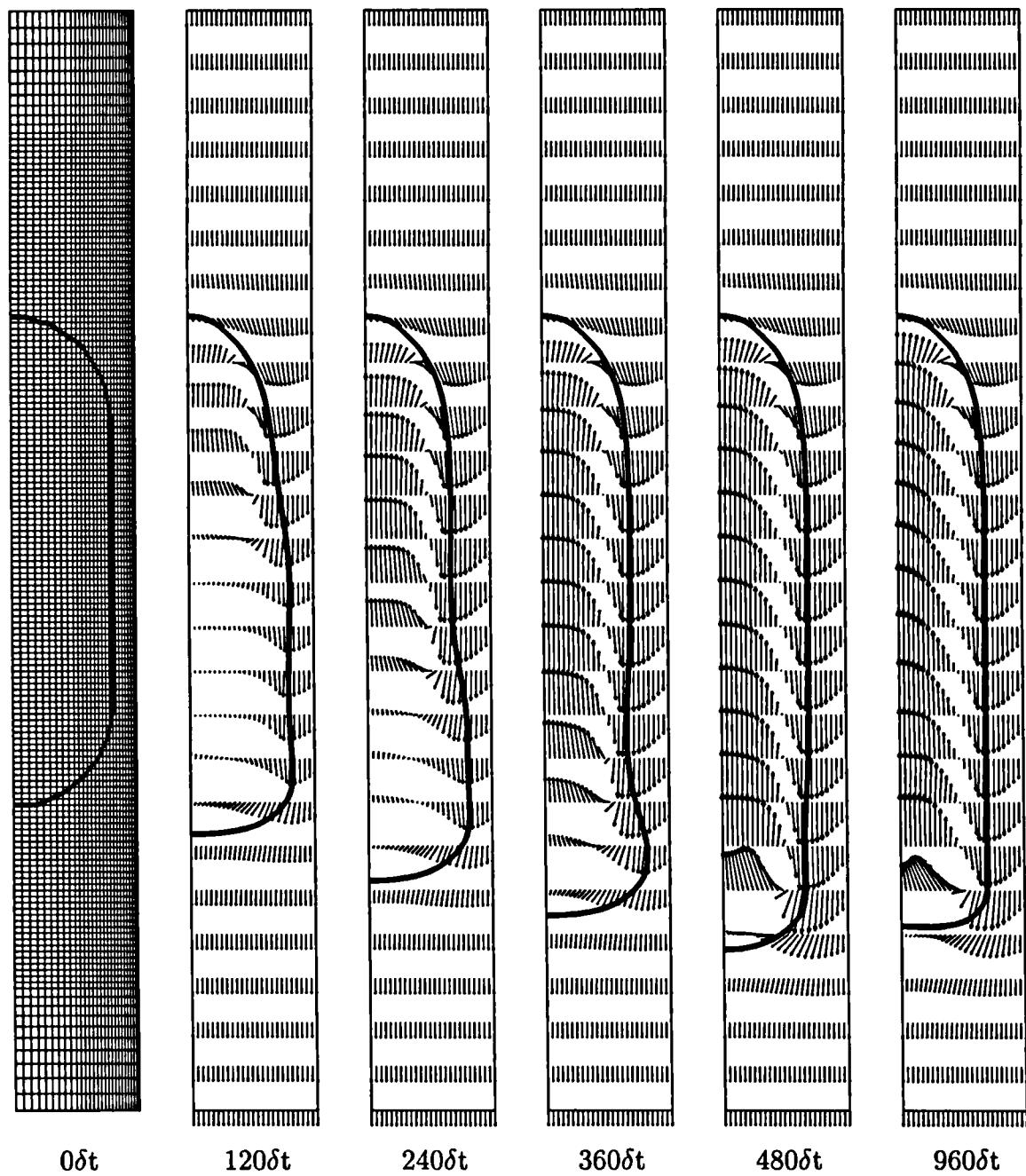


Figure 5.27: Time evolution of a Taylor bubble ($E_o=39.76$, $M=0.015$)

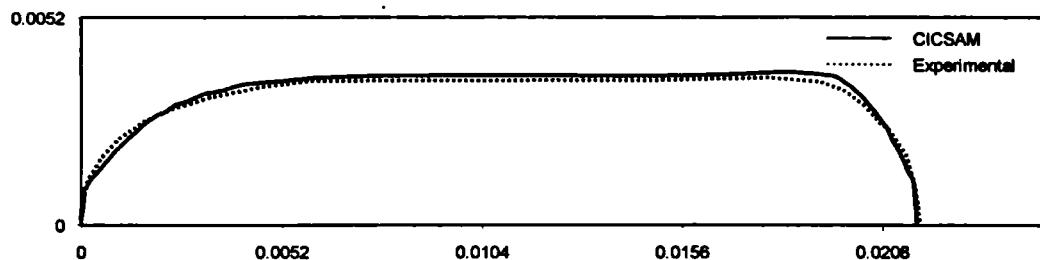


Figure 5.28: The shape of a Taylor bubble for $M=0.0015$ and $E_o=17.59$

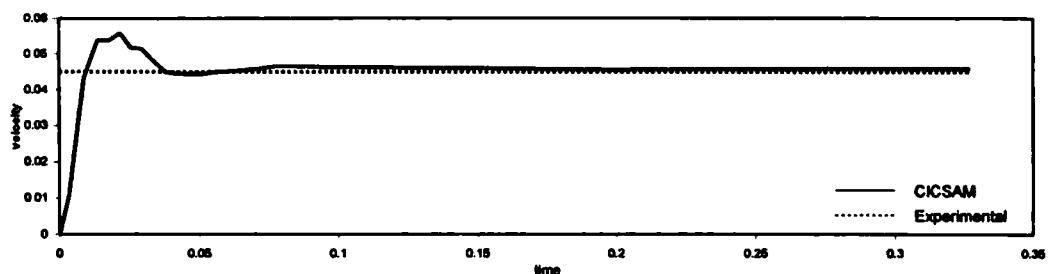


Figure 5.29: Rise velocity of a Taylor bubble for $M=0.0015$ and $E_o=17.59$



Figure 5.30: Shape of a Taylor bubble for $M=0.015$ and $E_o=39.76$

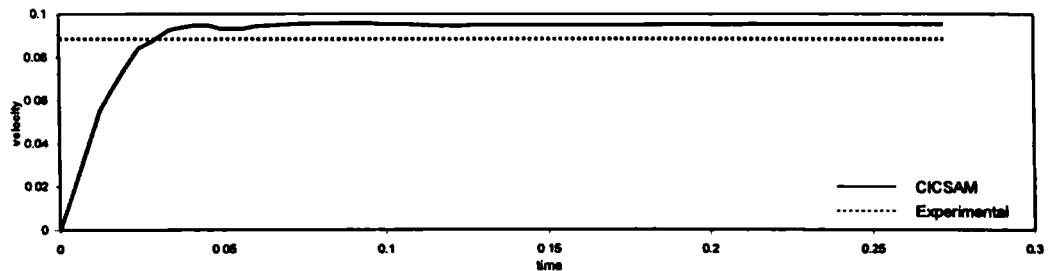


Figure 5.31: Rise velocity of a Taylor bubble for $M=0.015$ and $E_o=39.76$

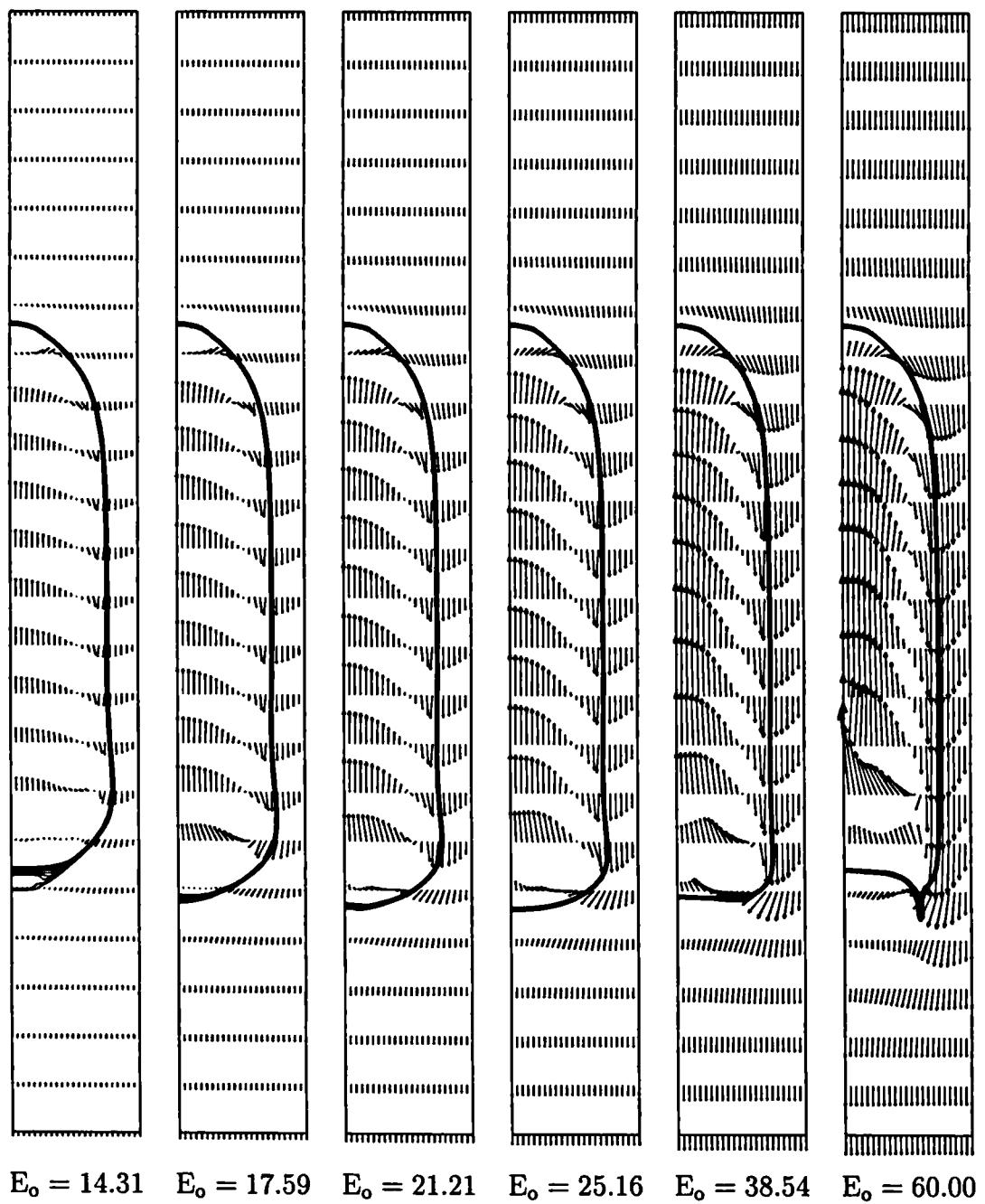


Figure 5.32: Predicted velocity vectors and bubble shapes for different Eötvös numbers with $M=0.0015$

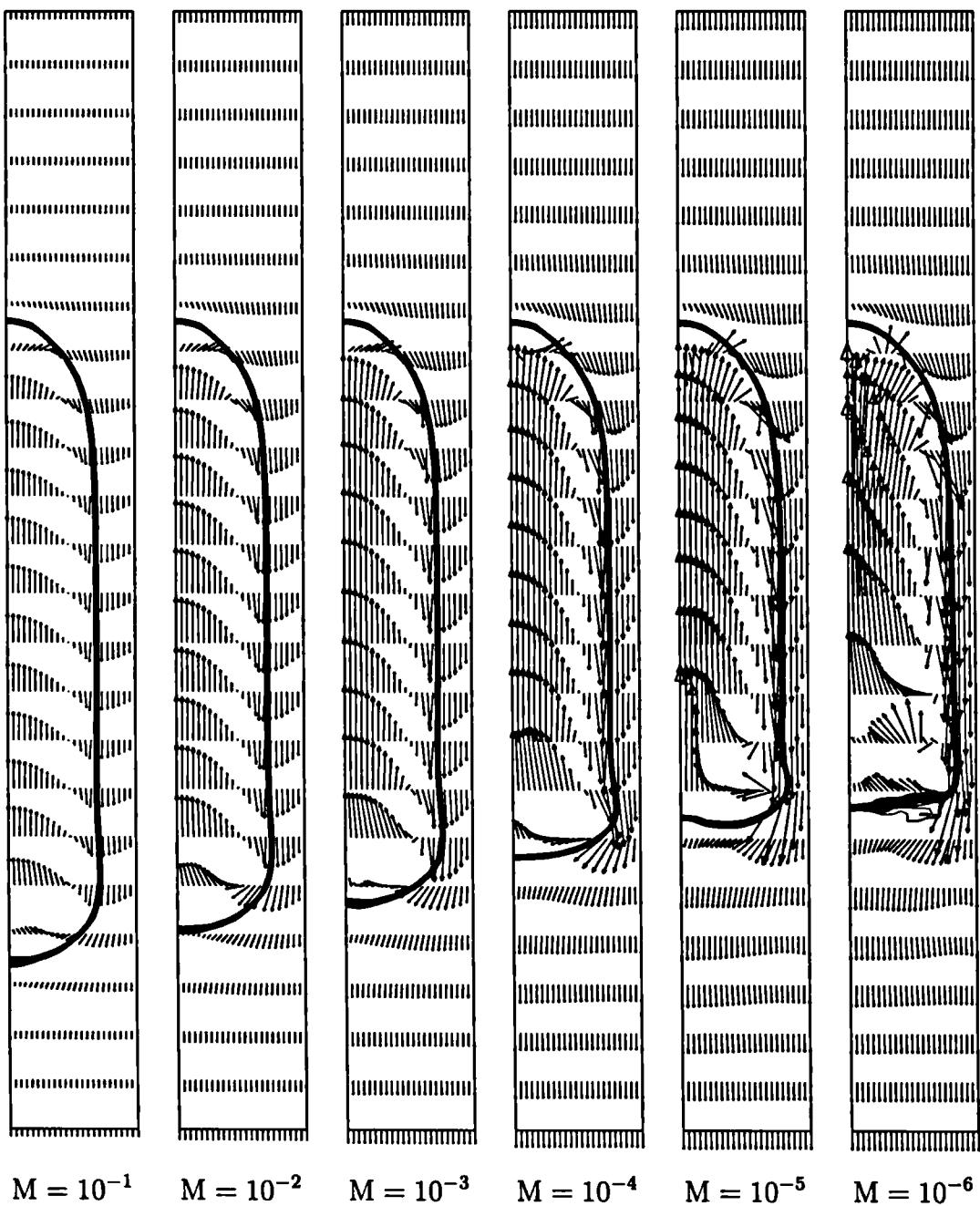


Figure 5.33: Predicted velocity vectors and bubble shapes for different Morton numbers with $E_o=21.21$

Case	ρ_1	ρ_2	μ_1 $\times 10^{-3}$	μ_2 $\times 10^{-5}$	σ $\times 10^{-2}$	D $\times 10^{-3}$	M	E_o	u_e	% error
1	1311	1.22	97	1.8	7.6	9.2	0.0015	14.31	1.9	
2	1311	1.22	97	1.8	7.6	10.2	0.0015	17.59	2.1	
3	1311	1.22	97	1.8	7.6	11.2	0.0015	21.21	4.6	
4	1311	1.22	97	1.8	7.6	12.2	0.0015	25.16	8.6	
5	1311	1.22	97	1.8	7.6	15.1	0.0015	38.54	7.2	
6	1311	1.23	97	1.8	7.6	18.8	0.0015	60.00	6.9	

Table 5.3: Conditions for the comparison of Taylor bubbles for different Eötvös numbers

Case	ρ_1	ρ_2	μ_1 $\times 10^{-3}$	μ_2 $\times 10^{-5}$	σ $\times 10^{-2}$	D $\times 10^{-3}$	M	E_o	u_e	% error
1	1311	1.22	276.7	1.8	7.6	11.2	10^{-1}	21.21	13.2	
2	1311	1.22	155.6	1.8	7.6	11.2	10^{-2}	21.21	4.3	
3	1311	1.22	97.0	1.8	7.6	11.2	10^{-3}	21.21	4.5	
4	1311	1.22	49.2	1.8	7.6	11.2	10^{-4}	21.21	1.5	
5	1311	1.22	27.7	1.8	7.6	11.2	10^{-5}	21.21	1.8	
6	1311	1.22	15.6	1.8	7.6	11.2	10^{-6}	21.21	2.5	

Table 5.4: Conditions for the comparison of Taylor bubbles for different Morton numbers

dimensional torus. Furthermore, at even higher Eötvös numbers the flow becomes turbulent and the shedding is more severe, an issue not addressed here. The increase in velocity with an increase in the Eötvös number is also clearly visible in Figure 5.32. However, the accuracy with which the present methodology predicts the terminal rise velocity, decreases with an increase in Eötvös number. The predicted and measured Froude numbers are plotted in Figure 5.34 (a).

The predicted tendencies of the bubble shapes for different Morton numbers, shown in Figure 5.33, correlate with the experimental observations of Tomiyama *et al.* (1994) who report a flattening of the bubble tail, a decrease in the film thickness and an increase

in the bluntness of the nose with decreasing Morton number. The errors on the prediction of the rise velocity are moderate (i.e. $\leq 4.5\%$), except in the case of $M=10^{-1}$, where the error rises to 13.2%. A graphical comparison between the predicted and measured Froude numbers is given in Figure 5.34 (b).

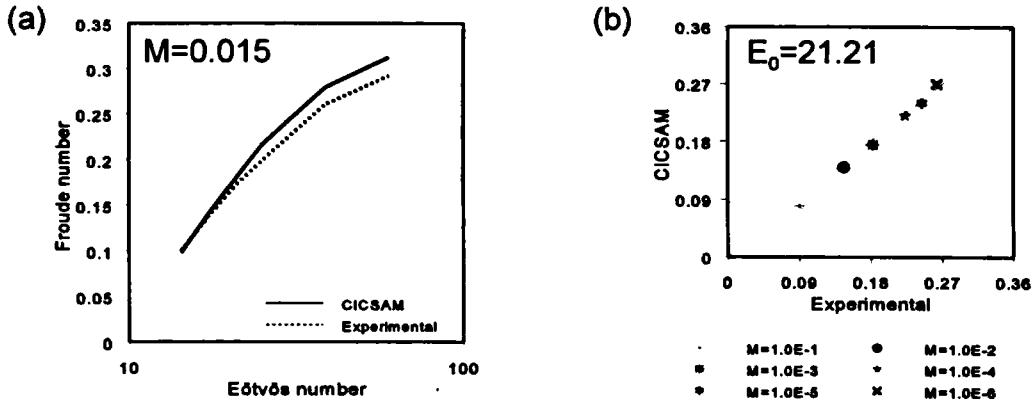


Figure 5.34: (a) Plots of Froude number against Eötvös number for $M=0.015$ and (b) comparison between experimental and calculated Froude numbers for $E_0=21.21$

For $M=10^{-6}$ the prediction of the terminal rise velocity is good, but due to the decrease of the viscous effects, the velocities increases to a level where it is difficult to distinguish between the real flow behaviour and the parasite currents introduced by the modelling of surface tension with the CSF model.

The results obtained for the rise of a Taylor bubble for different Eötvös and Morton numbers reveal that the present methodology can predict such flows accurately when the flow behaviour is determined by a balanced contribution of the different parameters such as viscous effects, surface forces and inertial effects. This corresponds with the centre of the white area on the cross plot presented in Figure 5.24. The further away from this centre, the less accurate the predictions of the present methodology.

5.6 Collapse of a liquid column

5.6.1 Collapse with a return wave

A classical experiment (Martin & Moyce (1952)) used in the validation of the mathematical modelling of free surface flows or two-fluid systems is the collapse of a liquid column (Harlow & Welch (1965), Nichols & Hirt (1971), Hirt & Nichols (1981), Ramaswamy & Kawahara (1987), Ghobadian (1991), Soulis (1992) and Koshizuka *et al.* (1995)). Measurements of the exact interface shape are not available, but some secondary data such as the speed of the wave front and the reduction of the column height are available. Photographs however, showing the time evolution of the collapsing column as well as the wave returning after hitting a wall on the opposite site are available for the purpose of evaluating the mathematical methodology on the basis of flow visualisation (Koshizuka *et al.* (1995)). Furthermore, an analytical solution (Stoker (1958)) of the interface evolution can be obtained. However, these are for inviscid flows and only of use for the initial stages.

Figure 5.35 shows a plane view of the experimental setup which is used for the current flow prediction. The dimensions of the tank and the water column correspond with those used in the experiment carried out by Koshizuka *et al.* (1995). The tank is made of glass, with a base length of 0.584m. The water column, with a base length of 0.146m and a height of 0.292m, is initially supported on the right by a vertical plate drawn up

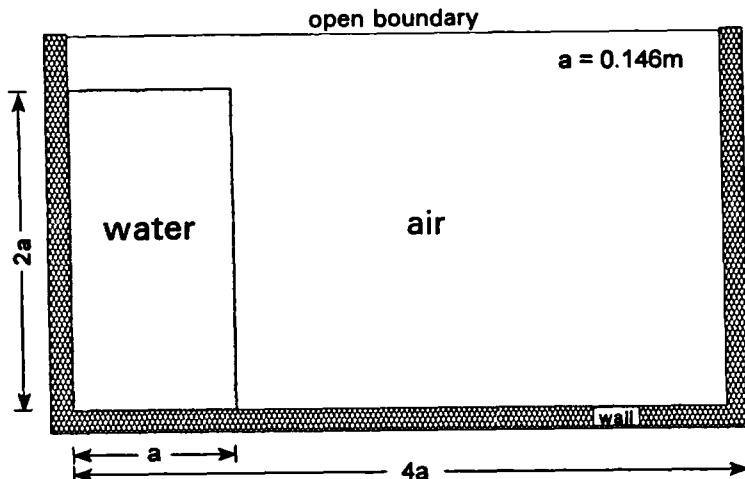


Figure 5.35: Initial geometry of the water column

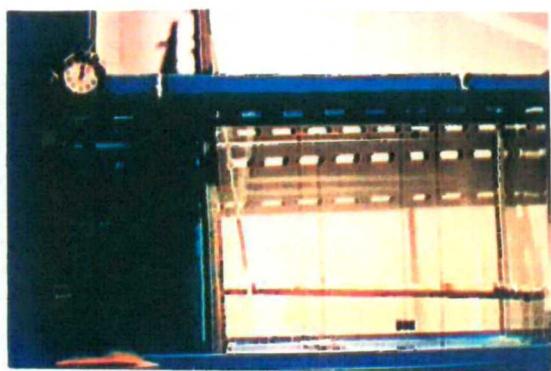
rapidly at time $t=0.0\text{s}$. The time evolution of the collapsing column has been captured by a video camera and snapshots shown in Figure 5.36, have been made available by Koshizuka (1995).

Gravitational acceleration causes the water column in the left of the tank to seek the lowest possible level of potential energy. Thus, the column will collapse and eventually come to rest and occupy the bottom of the tank. The initial stages of the flow are dominated by inertia forces with viscous effects increasing rapidly as the water comes to rest. On such a large scale, the effect of surface tension forces is unimportant.

For the numerical calculation no-slip boundary conditions have been applied to the bottom and sides of the tank. This experiment models a two-dimensional effect, therefore slip conditions have been applied to the front and back of the tank. The top boundary has been modelled as an open boundary with a fixed atmospheric pressure and zero normal gradient on the velocities. Both fluids are allowed to leave the computational domain through this boundary, but only the entering of air back into the domain has been allowed for. Thus, a zero gradient boundary condition on the volume fractions when the velocity is directed outward and a fixed value boundary condition when the velocity is directed inward are prescribed. This simulates the water leaving the container as seen in Figure 5.36 at time $t=0.6\text{s}$.

The predicted position of the interface together with the velocity vectors, predicted with CICSAM on a 120×70 uniformly spaced mesh, are shown in Figure 5.37. The times $t=0.0\text{s}, 0.2\text{s}, 0.4\text{s}, 0.6\text{s}, 0.8\text{s}, 1.0\text{s}$, correspond with the times of the photographs shown in Figure 5.36 and reveal the following:

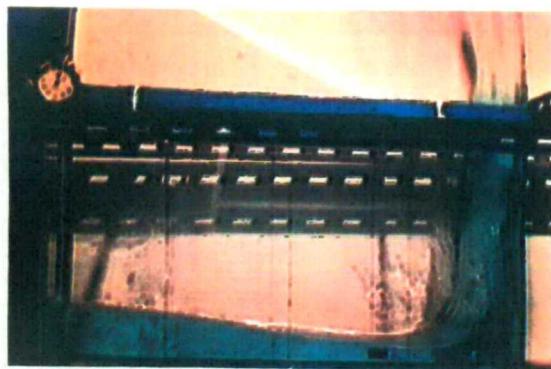
- Time $t=0.2\text{s}$: Approximate 75% of the base is covered with water.
- Time $t=0.4\text{s}$: The horizontal interface makes a small angle with the base of the tank and the water starts to leave the domain at the top right corner.
- Time $t=0.6\text{s}$: The horizontal interface is almost parallel to the base of the tank and the water against the right wall starts to fall back under the influence of gravity.



$t = 0.0s$



$t = 0.2s$



$t = 0.4s$



$t = 0.6s$

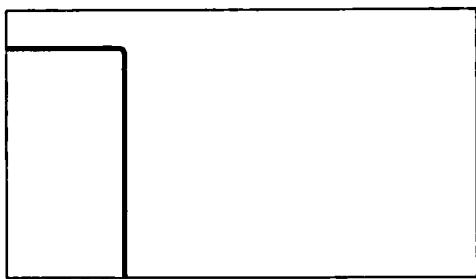


$t = 0.8s$

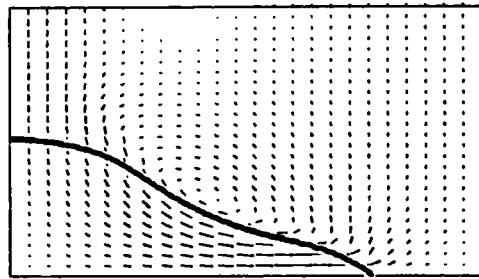


$t = 1.0s$

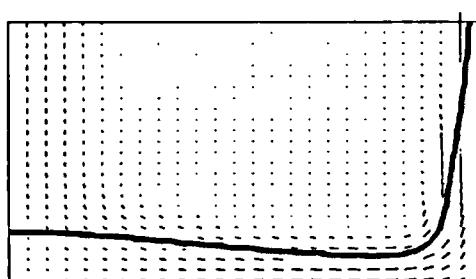
Figure 5.36: Experimental results of a collapsing water column



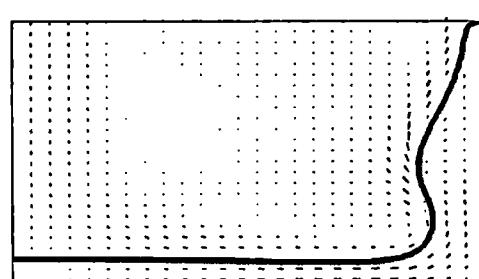
$t = 0.0s$



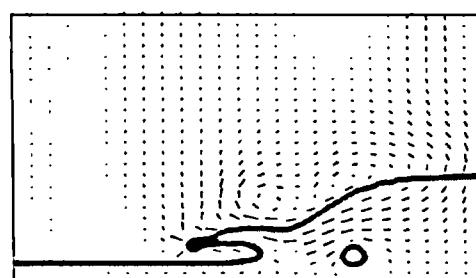
$t = 0.2s$



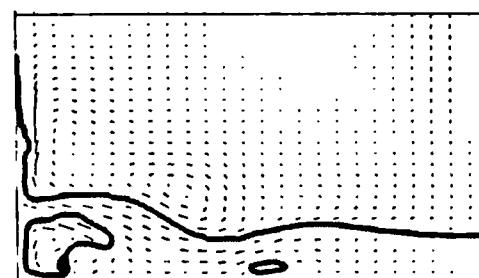
$t = 0.4s$



$t = 0.6s$



$t = 0.8s$



$t = 1.0s$

Figure 5.37: Numerical results of the collapsing water column

- Time $t=0.8\text{s}$: The backward moving wave has folded over and a small amount of air is trapped. In the experiment however, this air is present in form of small bubbles. The current methodology has been derived for sharp interfaces and tries to keep similar fluid regions together. To predict the behaviour of the small bubbles correctly, either a breakup model needs to be introduced or the mesh needs significant refinement to a resolution smaller than the bubble size. This failure to resolve bubbles or spray without the use of a prohibitively small grid results in the spray on the air side being incorporated in the tongue at the front of the backward moving wave. This results in the tongue being slightly heavier and therefore slightly lower than the position given in the photographs. In the experiment the water surface on the right is represented by two distinguishable interfaces, the top containing a dispersed phase and just underneath it an interface which represent the bulk of the water. The numerical prediction corresponds with the latter.
- Time $t=1.0\text{s}$: The tongue of the backward moving wave impinges upon the left wall, trapping a large air bubble. In the experiment this bubble also contains a dispersed phase not predicted with the current methodology. It is interesting to note that the predicted position of the impingement against the left wall agrees well with the experiment. The spray on the left above the surface shown in the experimental results is concentrated against the left wall in the case of the numerical prediction.

This calculation shows that CICSAM can predict the behaviour of complex interfacial structures. A more quantitative comparison for the early stages of this experiment can be made by using the reduction in height and the speed of the wave front. The experimental data shown were obtained by Martin & Moyce (1952).

The non-dimensional height of the collapsing water column at the left wall versus the non-dimensional time is shown in Figure 5.38. Numerical predictions for a coarse (40×25) mesh and a finer (120×70) mesh are shown. For both these calculations the predicted height is the same and it corresponds very well with the experimental data presented by Martin & Moyce (1952). The non-dimensional positions of the leading

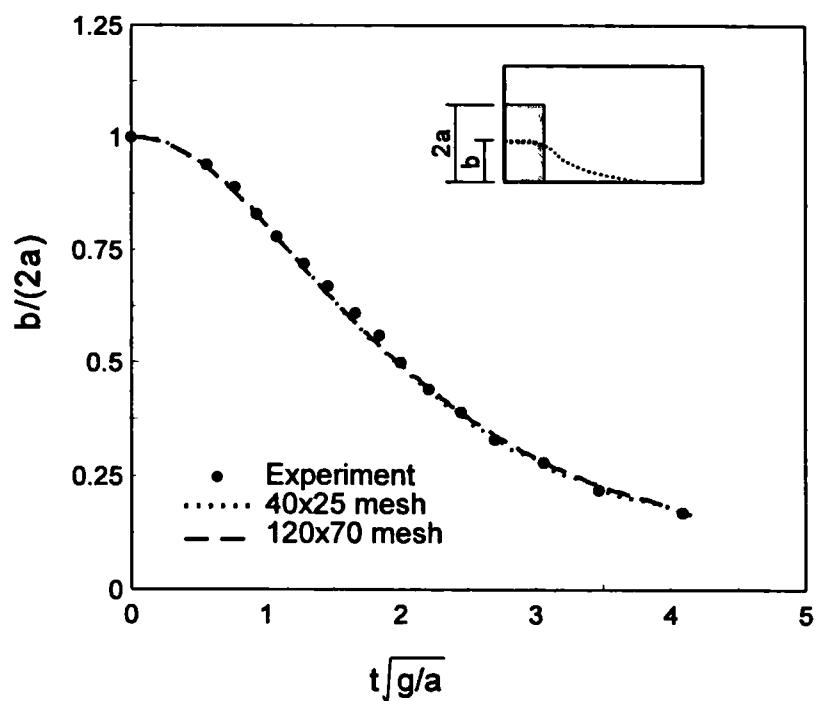


Figure 5.38: The height of the collapsing water column versus time

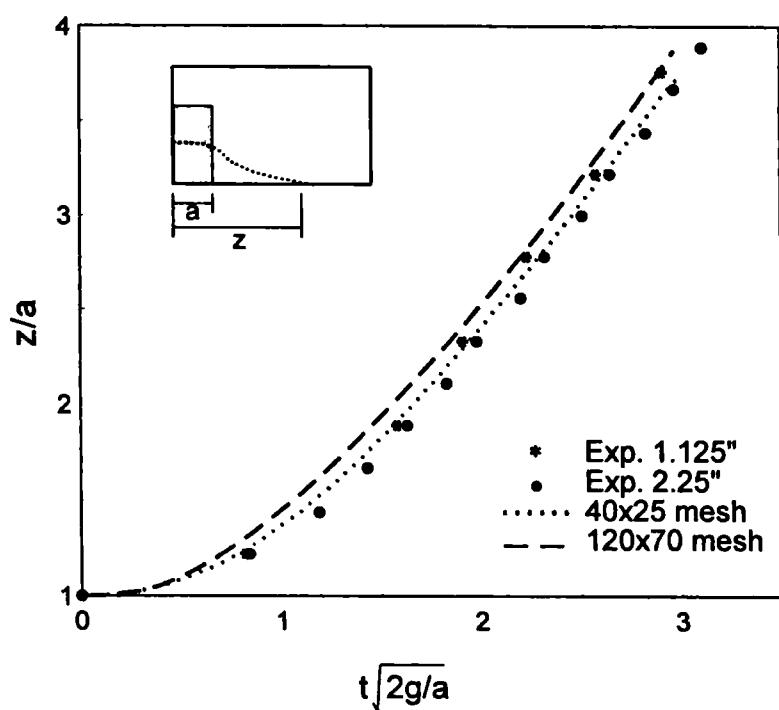


Figure 5.39: The position of the leading edge versus time

edge for the same cases are shown in Figure 5.39. The calculated results show that the leading edge moves faster when the resolution of the mesh increases. Results presented by other researchers show the same tendency (see Koshizuka *et al.* (1995)). The reason for this is the difficulty to determine the exact position of the leading edge. A thin layer shoots over the bottom and the rest of the bulk flow follows shortly behind it. The difficulty is also confirmed by Martin & Moyce (1952) who present two different sets of experimental data.

5.6.2 Collapse with an obstacle

A more interesting version of a collapsing water column occurs when a small obstacle is placed in the way of the wave front (Harlow & Amsden (1971) and Koshizuka *et al.* (1995)). Figure 5.40 shows the experimental setup.

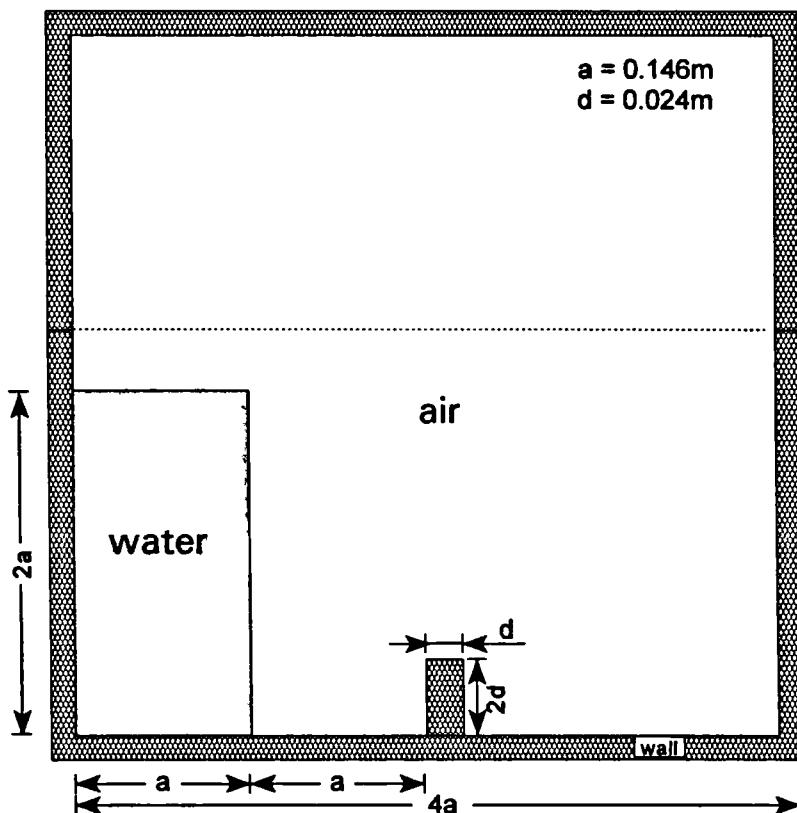


Figure 5.40: Geometry of collapsing water experiment with an obstacle

The dimensions of the tank are the same as those for the previous experiment. The obstacle (0.024m x 0.048m) is placed on the bottom of the tank, in the way of the moving front with its lower left corner in the centre of the tank.

The experiment has been carried out with an open-topped container as indicated by the dotted line in Figure 5.40. For the numerical modelling however, the computational domain has been increased in height and a closed domain is considered. This is done to determine whether the methodology presented in the previous chapters conserves mass throughout the calculation. This experiment also models a two-dimensional effect, therefore slip conditions have been applied to the front and back of the tank. No slip conditions have been applied to all the other boundaries.

The predicted position of the interface together with the velocity vectors, predicted with CICSAM on a 200 x 200 non-uniformly spaced mesh, are shown in Figure 5.42. The times $t=0.1\text{s}$, 0.2s , 0.3s , 0.4s , 0.5s , 0.6s , correspond with the times of the photographs shown in Figure 5.41 (Koshizuka (1995)), and show:

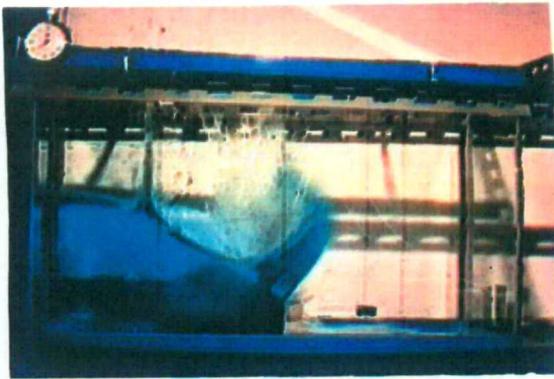
- Time $t=0.1\text{s}$: The leading edge of the collapsing water column has not yet reached the obstacle: however, in the case of the experiment the thin layer which rushes over the bottom of the tank has been blocked by the obstacle.
- Time $t=0.2\text{s}$: The movement of the leading edge has been obstructed by the obstacle and a tongue of water bounces up from the upper left corner of the obstacle in the direction of the opposite wall.
- Time $t=0.3\text{s}$: The tongue continues its movement towards the opposite wall. The correlation between the experimental and predicted fluid distributions is very good - the horizontal interface on the left has the same shape and the tongue is attached to the upper left corner of the obstacle.

- Time $t=0.4\text{s}$: The tongue impinges against the opposite wall, trapping air beneath it. The sheet of water formed in this way begins to fall under the action of gravity, but the trapped air provides resistance to this downward motion.
- Time $t=0.5\text{s}$: The sheet of water continues its downward motion. The trapped air will eventually burst through the water sheet above it. During this period a secondary tongue of water is generated at the obstacle and shoots into the trapped air space beneath the first sheet. The predicted length of the tongue is slightly shorter, an indication that the times of the experiment and calculation do not coincide exactly.
- Time $t=0.6\text{s}$: The sheet of water has burst open and the secondary tongue has impinged against the bottom wall, trapping a smaller bubble to the right of the obstacle. The shape of the horizontal interface on the left corresponds well with the experimental shape. The water on the right contains a high amount of air and continues its downward motion under the influence of gravity. This is also visible in the predictions.

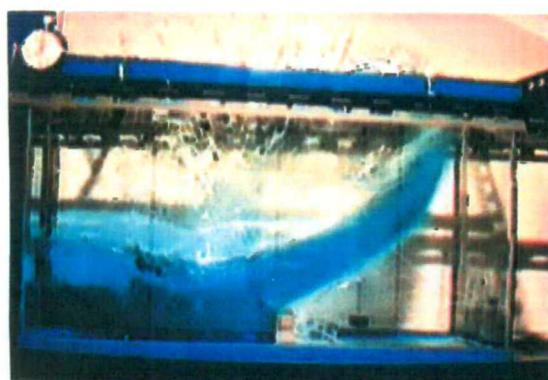
This calculation shows that CICSAM can easily cope with the dynamics of free surface flows.



$t = 0.1s$



$t = 0.2s$



$t = 0.3s$



$t = 0.4s$

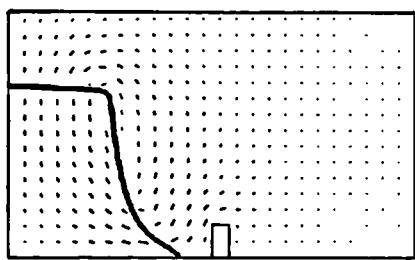


$t = 0.5s$

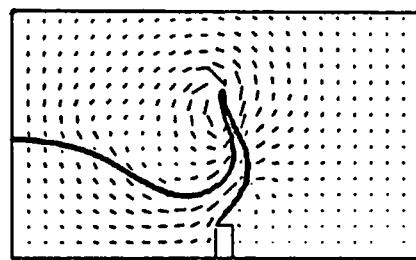


$t = 0.6s$

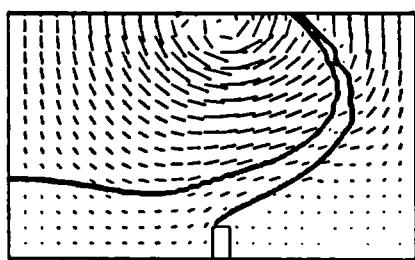
Figure 5.41: Experimental results of a collapsing water column hitting an obstacle (Koshizuka (1995))



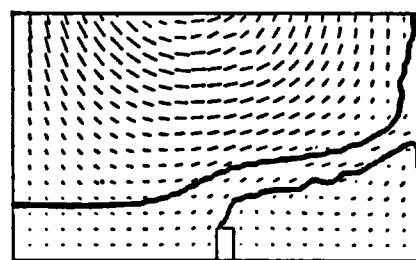
$t = 0.1s$



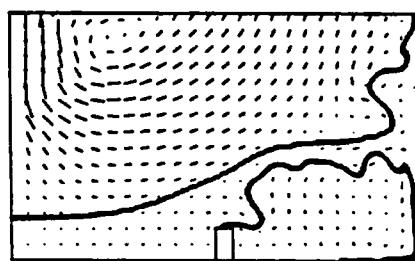
$t = 0.2s$



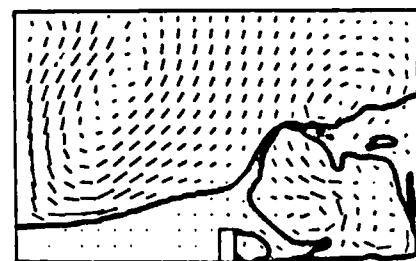
$t = 0.3s$



$t = 0.4s$



$t = 0.5s$



$t = 0.6s$

Figure 5.42: Numerical results of a collapsing water column

5.7 Hydraulic bore

A bore, or moving hydraulic jump, can be generated by pushing a horizontal layer of liquid into a vertical wall (Stoker (1958)). The step wave or bore, produced in this manner runs away from the wall over the horizontal layer which is moving towards the wall. The flow within the wave itself is very irregular and the exact interface shape within this regime is indeterminate, however, some secondary information regarding the height of the interface at the vertical wall is available (Stoker (1958)). The numerical modelling of a bore is demanding, because in the case of a breaking bore the interface folds continuously over itself (Hirt & Shannon (1968), Nichols & Hirt (1971 and 1973) and Lemos (1992)).

A schematic representation of the computational setup is shown in Figure 5.43. The base length of the domain is 12.5m and the height is defined as 4m. The domain is discretised with a uniform mesh of 50 cells in the horizontal direction and 30 cells in the vertical direction. The initial depth of the liquid is 1m and it is moving uniformly to the right with unit velocity. Slip conditions are applied at the top left, bottom and right boundary. At the lower left boundary, fluid enters the flow domain with a unit velocity in order to prevent any waves from being generated there. The top boundary is defined as a fixed pressure boundary. As in the case of the collapsing water column, both fluids are allowed to leave the computational domain at this boundary, but only fluid 2 is allowed to enter the flow domain through it.

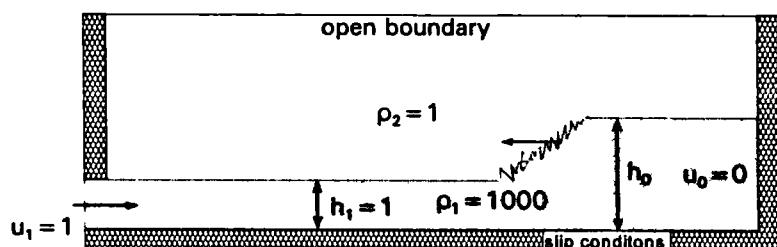


Figure 5.43: Schematic representation of the computational domain for the bore

The calculation has been repeated for two different gravity values, namely $g=1$ and $g=0.4548$ (numerical results for these two cases are also presented by Hirt & Nichols (1981)).

The viscosity of both fluids has been taken as zero, because the theoretical solution with which the calculations are compared is for inviscid flow. The densities of the liquid and gas are $\rho_1=1000$ and $\rho_2=1$ respectively. The theoretical height, predicted height and percentage error for the two case are presented in Table 5.5.

Case	g	Theoretical height	Predicted height	Error (%)
1	1.0	2.19	2.17	0.9
2	0.4548	2.9	2.8	3.4

Table 5.5: Comparison of the height of a bore

The numerical results for these two cases are shown in Figures 5.44 and 5.45 respectively. The breaking wave is clearly visible in the second panel of both sets of results. Thereafter the interface continuously folds (or rolls) over itself. As noted before, the exact shape of the breaking interface is not known. However the shape of the main interface between the two fluids is quite similar to the results presented by Hirt & Nichols (1981). A major difference between the two methodologies is that the current methodology includes both fluids in the calculation, while in the methodology employed by Hirt & Nichols (1981), the second fluid is assumed to be void. Thus the current methodology enables the modelling of the gas captured within the water, whilst the latter ignores the existence of this behaviour.

This case reveals that CICSAM can predict an interface which continuously folds over on itself without any additional computational effort. As in the case of the collapsing liquid column, this calculation can benefit by adding a breakup model to diffuse the large bubbles underneath the main interface.

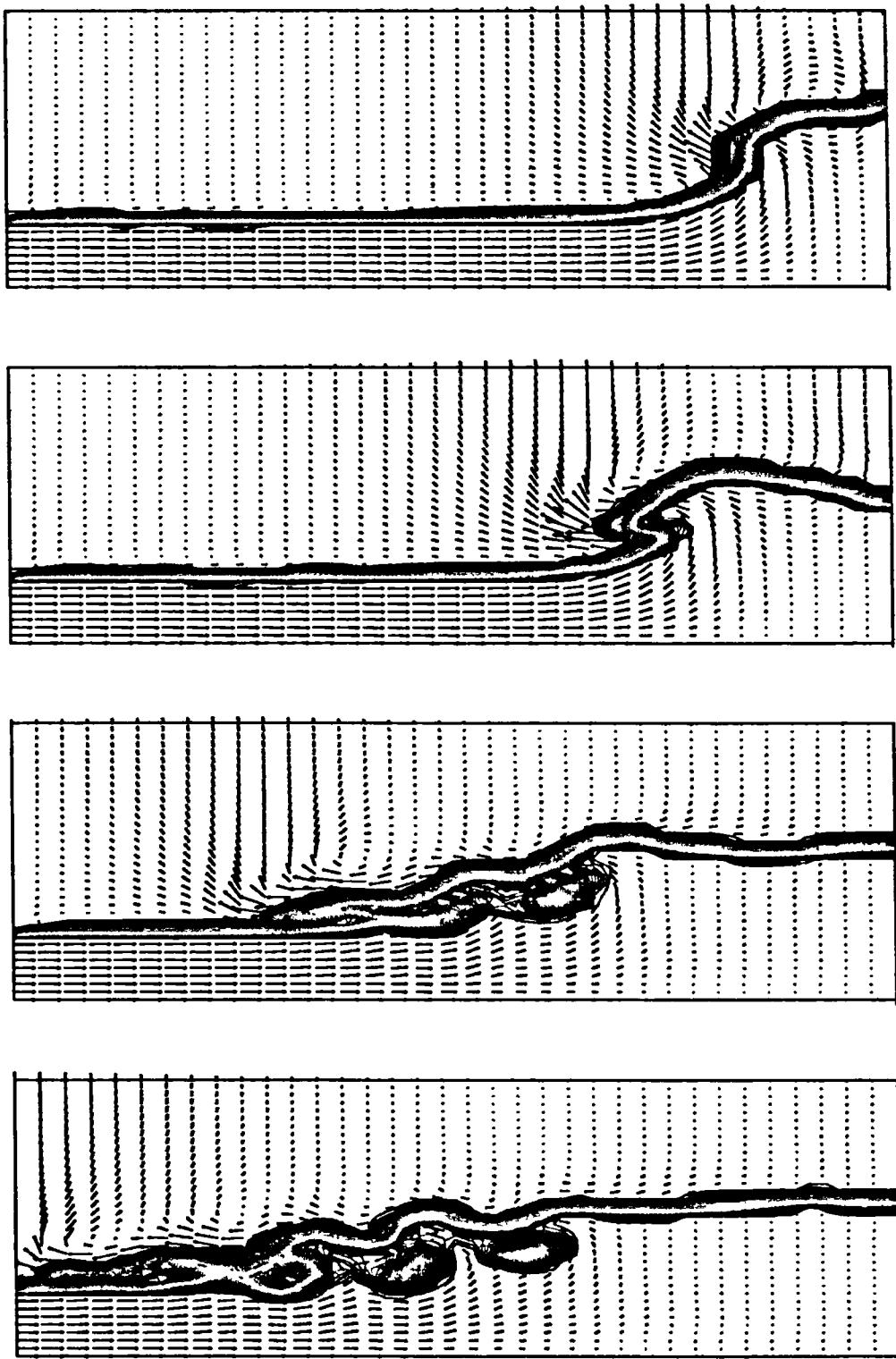


Figure 5.44: Velocity vectors and fluid configuration for the bore calculated with $g=1$ at times 3.50s, 4.90s, 7.94s, 9.88s

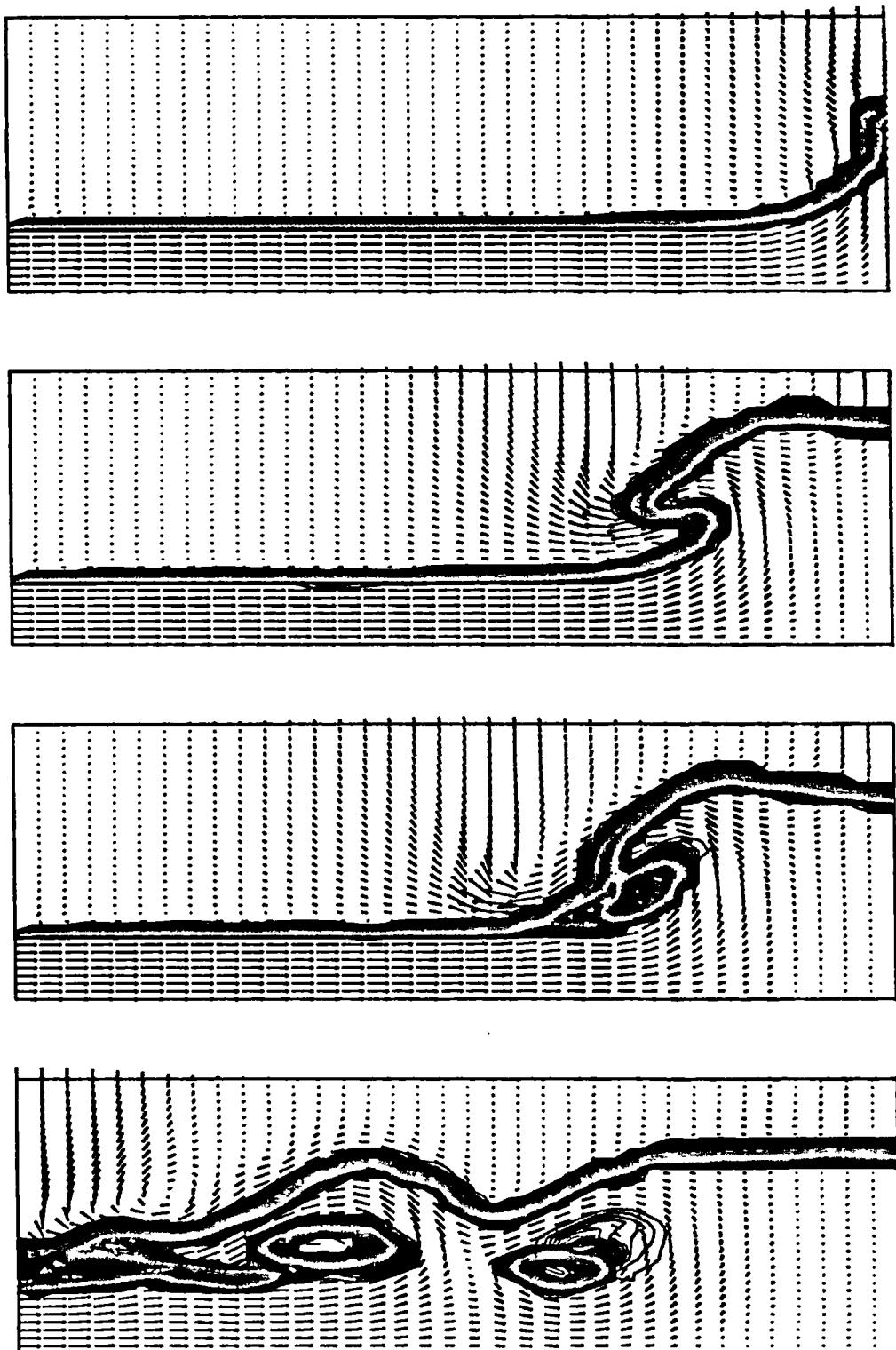


Figure 5.45: Velocity vectors and fluid configuration for the bore calculated with $g=0.4548$ at times
0.97s, 7.10s, 8.42s, 15.89s

5.8 Flow through a sluice-gate

As noted earlier, this case is included as an illustration of the wider applicability of CICSAM and no correlations with experimental or analytical data will be made.

The computational setup used in the present study is shown in Figure 5.46. Initially everything is at rest. At time $t=0$ the water on the left starts to flow through the gate at the bottom of the tank under the influence of gravity, generating a pulse which evolves into a wave. The height of the water in the tank on the left is kept constant throughout the calculation. The whole top boundary and the top right boundary are defined as fixed pressure boundaries.

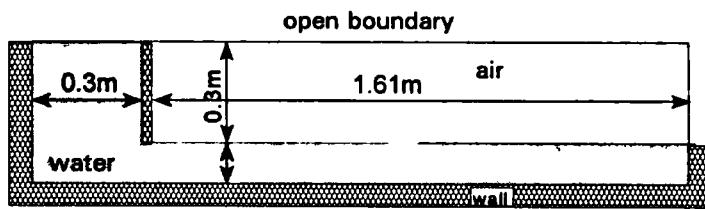


Figure 5.46: Schematic representation of the computational domain for the sluice-gate

The results obtained are presented in Figure 5.47. It is interesting to follow the initially created wave until it breaks near the outlet. Another interesting phenomenon is the entrainment of air at the top of the sluice gate. In the last panel, most of the air trapped in the liquid has either left the flow domain through the outlet or it has risen to the surface. Note also that the liquid level has risen above the gate opening, inhibiting further entrainment.

This case shows the ease with which CICSAM can predict the forming and propagation of a wave as well as the natural way in which the methodology can predict the entrainment of the one fluid into the other.

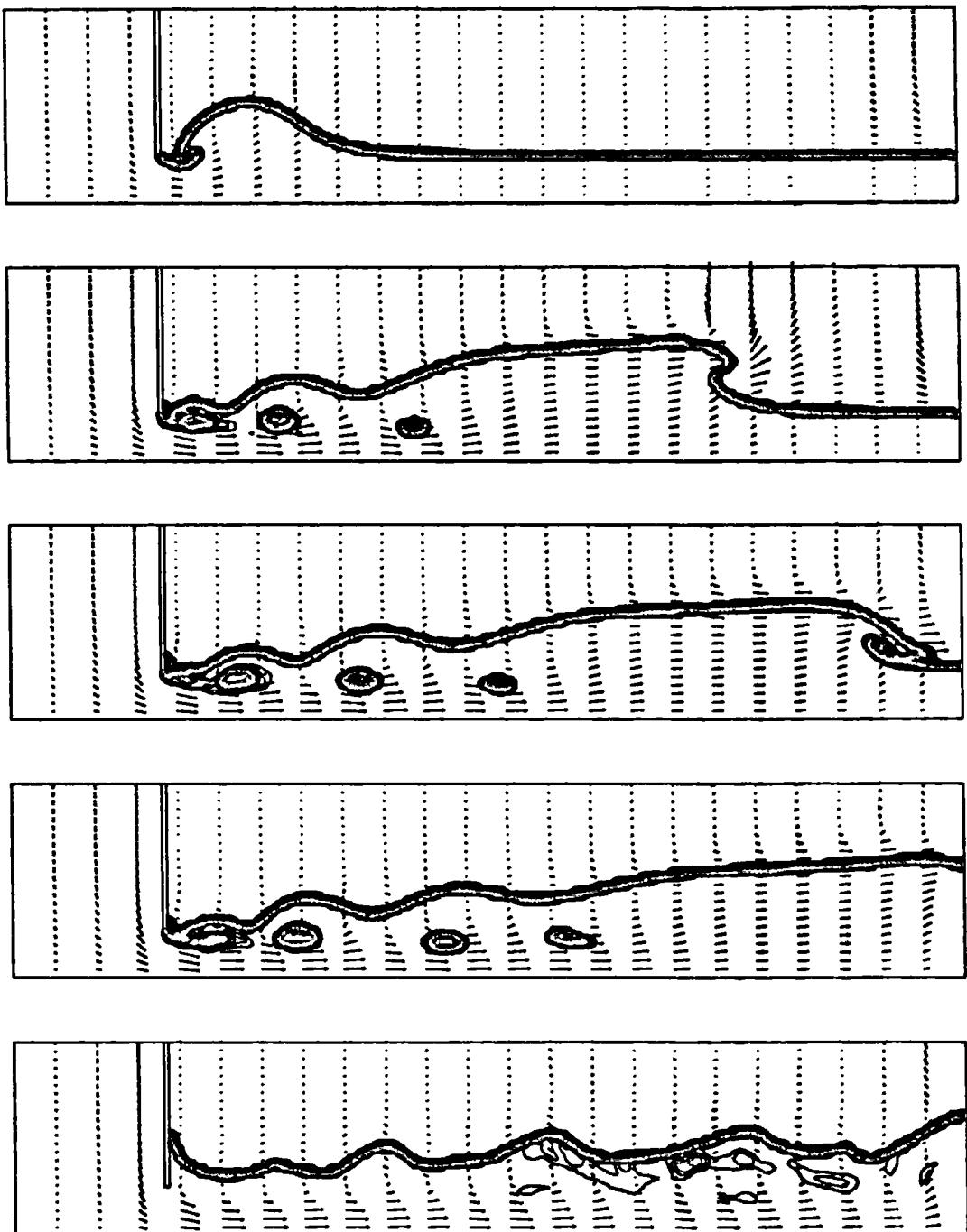


Figure 5.47: Velocity vectors and fluid distribution of the flow through a sluice-gate at different times

5.9 Splash of drop

The methodology presented in this thesis was derived for implementation into a multi-dimensional space. All the flow calculations presented in the previous sections were done for a two-dimensional space. This last case serves to illustrate that the methodology can predict three-dimensional interfacial flows.

This case predicts a common phenomenon, namely the splash of a water drop. Here it occurs into a cup partially filled with water and tilted at an angle of 30° . The computational mesh used for this calculation is presented in Figure 5.48. The irregularly shaped mesh is used to illustrate that the present methodology can predict interfacial flow on such meshes. The top view of the cup shows a fine resolution kernel which has been introduced to resolve the drop itself more accurately.

The results at different times for the first 0.66s are presented in Figure 5.49. The time step for this calculation is $\delta t=3.0\times 10^{-4}$ s. Initially the drop has a velocity of 1.0m/s in the direction normal to the base of the cup. Thus, the angle between the direction of motion and gravity which is acting downward is 30° . The first panel at $80\delta t$ shows the drop on its way to the stagnant liquid whilst the initial stages of the merging of the drop with the surface of the stagnant liquid is visible in the second panel ($120\delta t$). The next three panels at times $160\delta t$, $200\delta t$ and $260\delta t$ show the continuation of the downward moving drop and the development of the crater whereof the latter represents the maximum size of the crater. The next two panels at times $360\delta t$ and $450\delta t$ show the fluid motion of the back wave, whereof the latter represents the maximum height. The second last panel at time $600\delta t$ gives a continuation of the wave motion which starts to damp out. At $2200\delta t$ most of the waves have been damped and only a small ripple effect on the surface is visible. The overall behaviour of the merging drop with the stagnant water as predicted with CICSAM seems to be realistic.

This test has been included as a general application to illustrate that CICSAM can be used to predict flow encountered in everyday life, using irregular meshes in a three-dimensional space, without any additional computational efforts or difficulties.

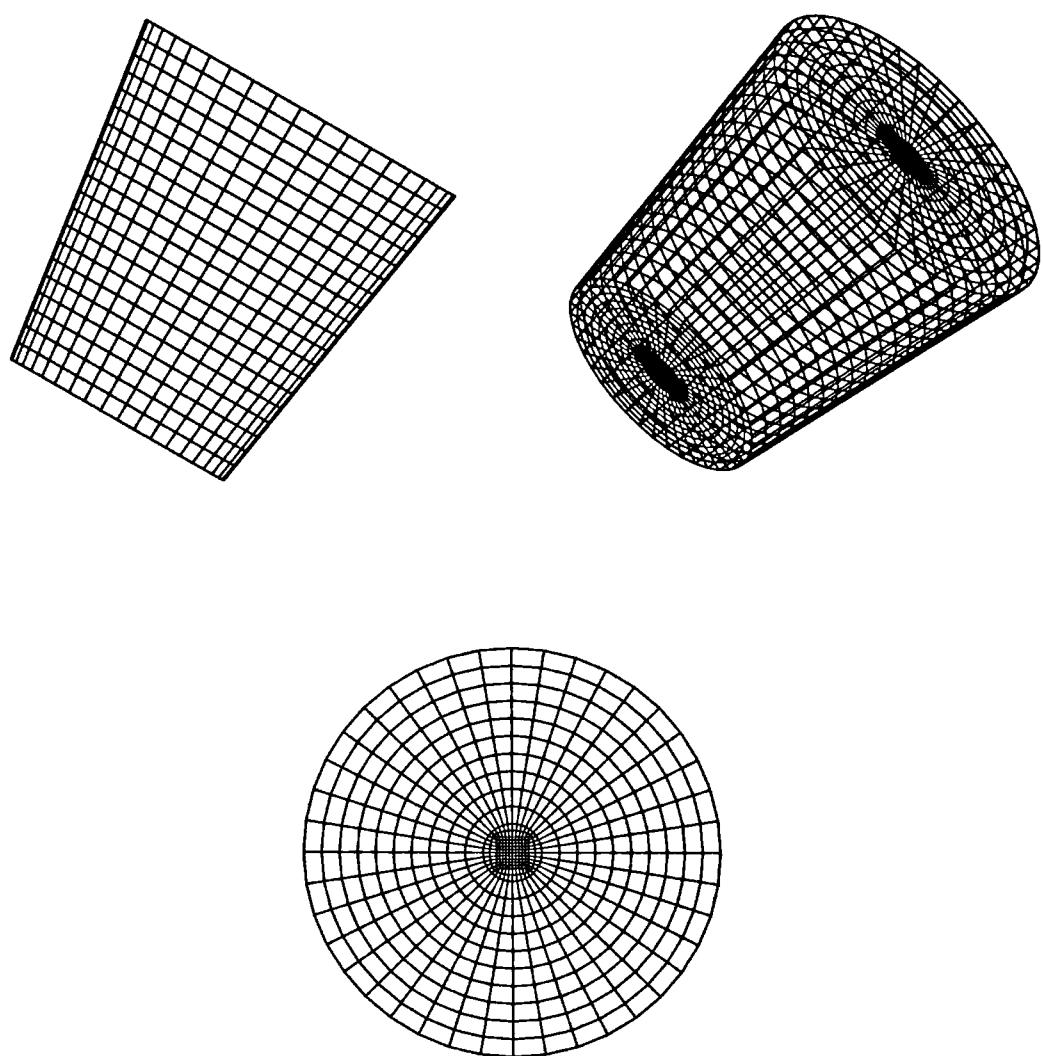


Figure 5.48: The computational mesh used for the prediction of the splash of a droplet

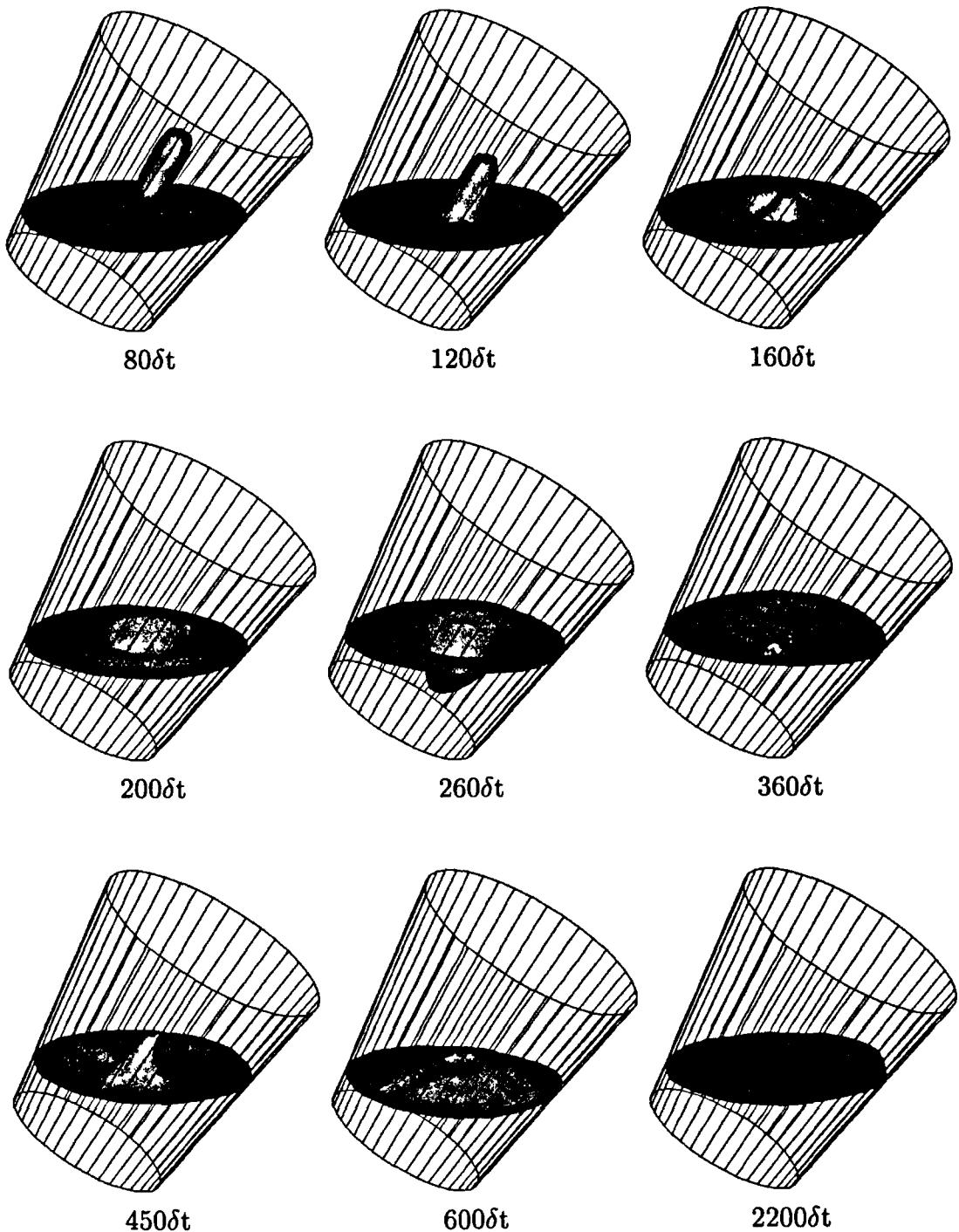


Figure 5.49: Three-dimensional splash of a droplet at different times

5.10 Closure

In this chapter the efficiency of the present CFD methodology for flows with internal interfaces was assessed for six different types of interfacial flow problems on a wide spectrum of different types of computational meshes in both two- and three-dimensional spaces. In most of the cases, good agreement with available experimental or theoretical data was obtained.

The uniform-density flow calculations revealed that CICSAM can capture sharp interfaces accurately, even on complex meshes. The shapes of the interfaces on uniform meshes are captured more accurately than other interface-capturing schemes employed previously. The level of accuracy regarding the interfacial shapes on complex meshes is acceptable, keeping in mind that its performance is still better than other interface-capturing schemes employed previously on uniform meshes.

The accuracy of the two-fluid system was tested extensively by the cases of low amplitude sloshing and the rise of a single Taylor bubble in a stagnant liquid. This test reveals that CICSAM is accurate, however, it starts to smear the interface when divergent flows occur (that is when both fluids flow away from the interface) in the case of steady flow calculations.

The cases of the collapsing water column, the bore, flow underneath a sluice-gate and the splash of the drop into a stagnant liquid illustrated the capability of the present methodology to cope in a natural way with merging interfaces and interfaces which fold over themselves.

This completes the assessment of the CFD methodology for interface capturing presented in this study. In the next chapter the main findings are summarised and some recommendations are given for future research.

6 Summary and conclusions

In the previous chapters a CFD methodology capable of capturing sharp fluid/fluid interfaces has been employed within a finite volume framework. A new compressive differencing scheme has been proposed and an implementation of it in multiple dimensions for irregular unstructured meshes has been presented. In particular, attention has been given to the simplicity, robustness and accuracy of the methodology.

This chapter concludes the thesis and is divided into two sections, namely a summary which includes the conclusions and suggestions for future work.

6.1 Summary

In the first chapter an extensive overview of the existing CFD methodology capable of predicting segregated flows has been given. An analysis of the advantages and disadvantages of these methods has been made in the light of the present objective, namely an accurate and robust methodology for the capturing of sharp interfaces on complex meshes. The most promising approach with regard to simplicity, accuracy and optimal use of computer resources has been shown to be the use of volume fractions.

In order to use volume fractions for interfacial flow calculations it is necessary to solve a convection transport equation, similar to the transport equations for the other flow quantities. An immediate difficulty which arises in the discretisation of the equation for the interface application is the diversity in the characteristics of the volume fraction gradient that needs to be resolved. Normal to the interface the gradient is a step defined over one mesh cell; and tangential to it a moderate gradient exists. A successful differencing scheme should cater for this diversity.

The most promising approaches available to achieve this, namely the VOF method and the less diffusive HR schemes, have been assessed and discussed extensively in the previous chapters. The HR schemes applied to interfacial calculations have shown to be too diffusive. In the case of VOF, instances where it introduces non-realistic deformation of the interfaces have been reported. The difficulties experienced previously in both these approaches have been identified and addressed. Based on this, a new HR scheme named CICSAM, specially tailored to capture well defined interfaces efficiently and accurately has been presented.

CICSAM is a blending of Hyper-C, the upper bound of the convection boundedness criteria (CBC), and ULTIMATE-QUICKEST, the transient bounded version of QUICK. The blending of the schemes is determined by the orientation of the interface and the flow direction. The novelty of CICSAM lies in the use of the normalised face value for the calculation of a weighting factor which is used to express the value at the face itself in terms of the two cells sharing a face. This enables an implicit implementation on arbitrary meshes, because all the information regarding the orientation of the interface, direction of motion and slope of interface is carried by the weighting factors of the different faces.

Additional to the above, an implicit implementation for multi-dimensional flow calculations has been derived which eliminates the need for operator splitting and the restriction associated with it. In order to preserve boundedness with the implicit implementation, a new CBC for multi-dimensions based on the Courant number of the donor cell has been derived.

This implicit implementation gives a more accurate prediction of the volume fraction values at the faces, which together with the solution algorithm presented in this study, enable the solution of the conservative form of the momentum equations. Previously, non-conservative formulations had to be used due to the difficulty in predicting accurate face values for the density.

In the derivation of methods to deal with volume fractions everything possible is done to obtain results which are physical. Unfortunately, this is only guaranteed in one-dimensional flow and special treatment is necessary to guarantee physical values at all times. Previous methods have assumed that the error is small enough and can be neglected by resetting non-physical values back to zero and unity. In the present study a better alternative in the form of a corrector step, based on the magnitude of the unbounded error, has been developed.

In Chapter 5 several calculated examples over a wide spectrum of segregated flows have been presented, illustrating the capability of the methodology presented in this thesis. The first cases have shown the accuracy with which it can predict the interface shape and the other cases have shown the ease with which it can predict merging or rupturing interfaces.

6.2 Future research

CICSAM has a Courant number limitation attached to it, especially in the vicinity of the interface. In transient calculations this is not necessarily a major drawback, but for steady state calculations, such as a Taylor bubble or flow over a weir, where the interface converges to a fixed shape, this limitation can be rather cumbersome. However, this is not insurmountable because it should be possible to apply a technique of sub-cycling where the time step of the main loop is divided into smaller steps in order to advect the volume fractions within the Courant number limitation of CICSAM. This has not yet been investigated and further research needs to be done.

The calculations obtained on the meshes consisting of triangles or hexagons show deformations similar to those reported with the VOF method on a mesh with rectangular cells. Two mechanisms can be responsible for this: Firstly, with the use of triangles a value of the 'upwind' cell does not exist. The use of the physical bounds, zero and unity, to bound the predicted upwind value reduces CICSAM to VOF. Secondly, in the derivation an assumption is made that if two neighbouring cells contain part of the interface, then the interface cuts through the face shared by them. With angles larger than 90° between the faces of a cell meeting at the vertices it is possible for two neighbouring cells to contain part of the interface without the interface itself cutting the face. Further investigation and refinement could probably improve the general behaviour of HR schemes on these types of meshes.

The CSF model has contributed significantly to the inclusion of surface tension effects within the volume fraction framework and has therefore been implemented in the present study. However, calculations on cases where the radius of curvature is very small have shown that the CSF model is sensitive to numerical errors in the prediction of the curvature. If the viscosity of the fluids is high enough the velocities used to create a homogeneous curvature will be damped out almost immediately. This phenomenon has also been reported by Lafaurie *et al.* (1994) and needs further attention in order to be able to predict the interface behaviour where surface tension forces dominate.

Turbulence is a phenomenon present in most flows, including the flow between two immiscible fluids. A variety of turbulence models exist and references are available where such models have been successfully implemented in conjunction with a mathematical methodology similar to that followed here. Preliminary tests, not reported here, have been carried out and indicate that further research needs to be done.

In the previous chapter examples where a dispersed phase such as a spray or small bubbles originate during the course of the calculation have been shown. The present technique has been developed to keep the fluid regions together and fails to predict this phenomenon accurately, unless the calculations can be done on a mesh with cells significantly smaller than the smallest droplets or bubbles. Therefore, the methodology

followed in this study can benefit by being employed with a suitable breakup model and dispersed phase flow model.

An immiscible two-fluid system commonly encountered is one consisting of a liquid and its vapour phase such as water and steam. Two phenomena are present, namely (i) a phase change occurs at the interface and (ii) the vapour phase is compressible. For the accurate modelling of such flows account must be taken of the phase change in all the equations, including the one for volume fraction. Furthermore, the solution algorithm also needs to be reformulated in order to make provision for this phase change and the compressibility of the gas.

6.3 Conclusions

This study should be regarded as a step towards the simulation of all the processes contained within multi-fluid flows on arbitrary meshes. Its aim was to overcome the problems experienced previously with segregated flows and to produce a robust, but accurate and simple to implement CFD methodology able of capturing well defined interfaces over a wide spectrum of industrial and engineering applications. However, the various numerical techniques presented in this study are not limited to segregated flow calculations only and can be implemented independently of each other. CICSAM is a HR scheme which operates at the upper bound of the CBC and therefore the numerical techniques employed could also be useful in the multi-dimensional implementation of other HR schemes on arbitrary meshes. In this respect, it is the hope of the author that the study will be of benefit to them who are in need of such methodology.

References

- Adamson, A.W., 1976, Physical chemistry of surfaces. New York: John Wiley & Sons.
- Ashgriz, N. and Poo, J.Y., 1991, FLAIR: Flux line-segment model for advection and interface reconstruction. *J. Comput. Phys.*, Vol. 93, p. 449-468.
- Batchelor, G.K., 1967, An introduction to fluid dynamics. Cambridge: Cambridge University Press.
- Brackbill, J.U., Kothe, D.B. and Zemach, C., 1992, A continuum method for modelling surface tension. *J. Comput. Phys.*, Vol. 100, p. 335-354.
- Caretto, L.S., Gosman, A.D., Patankar, S.V. and Spalding, D.B., 1972, Two calculation procedures for steady, three-dimensional flows with recirculation. In: Proceedings of Third Int. Conf. Numer. Methods Fluid Dyn., Paris.
- Chorin, A.J., 1980, Flame advection and propagation algorithms. *J. Comput. Phys.*, Vol. 35, p. 1-11.
- Clarke, A.P. and Issa, R.I., 1995, A numerical model of slug flow in vertical tubes. Dept. of Mechanical Engineering, Imperial College, London, internal report.
- Daly, B.J., 1967, Numerical study of two fluid Rayleigh-Taylor instability. *Phys. Fluids*, Vol. 10(2), p. 297-307.
- Daly, B.J., 1969, A technique for including surface tension effects in hydrodynamic calculations. *J. Comput. Phys.*, Vol. 4, p. 97-117.
- Darwish, M.S., 1993, A new high-resolution scheme based on the normalized variable formulation. *Numerical Heat Transfer, Part B*, Vol. 24, p. 353-371.
- Davis, S.F., 1994, Flux difference splittings and limiters for the resolution of contact discontinuities. *Appl. Mathematics and Computation*, Vol. 65, p. 3-18.
- Dervieux, A. and Thomasset, F., 1979, A finite element method for the simulation of a Rayleigh-Taylor instability. IRIA-LABORIA report, F-78150 Le Chesnay.
- Farmer, J., Martinelli, L. and Jameson, A., 1994, Fast multigrid method for solving incompressible hydrodynamic problems with free surfaces. *AIAA Journal*, Vol. 32, p. 1175-1182.
- Ferziger, J.H. and Perić, M., 1996, Computational methods for fluid dynamics. Heidelberg: Springer.

- Fletcher, C.A.J., 1991, Computational techniques for fluid dynamics, Volume 2, Specific techniques for different flow categories. 2nd ed. New York: Springer-Verlag.
- Fujibayashi, A., Taylor, A.M.K.P. and Whitelaw, J.H., 1995, Reynolds stress and heat flux measurements in a vertical bubble-driven plumes. In: Papers presented at the 2nd International Conference on multiphase flow, Kyoto, Japan, 3-7 April, 1995.
- Gaskell, P.H. and Lau, A.K.C., 1988, Curvature-compensated convective transport: SMART, a new boundedness-preserving transport algorithm. *Int. J. Numer. Methods Fluids*, Vol. 8, p. 617-641.
- Ghobadian, A., 1991, Development of a method for numerical simulation of flows with moving interfaces. National Power report TEC/L/0077/M91.
- Glimm, J., Grove, J., Lindquist, B., McBryan, O.A. and Tryggvason, G., 1988, The bifurcation of tracked scalar waves. *SIAM J. Sci. Stat. Comput.*, Vol. 9(1), p. 61-79.
- Glimm, J., McBryan, O., Menikoff, R. and Sharp, D.H., 1986, Front tracking applied to Rayleigh-Taylor instability. *SIAM J. Sci. Stat. Comput.*, Vol. 7(1), p. 230-251.
- Hagiwara, T. and Madarame, H., 1992, Two-dimensional numerical analysis of turbulent-flow with a free surface in a tank. *JSME Int. J. Series II*, Vol. 35(2), p. 174-178.
- Harlow, F.H. and Welch, J.E., 1965, Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Phys. Fluids*, Vol. 8(12), p. 2182-2189.
- Harlow, F.H. and Amsden, A.A., 1971, Fluid dynamics. Los Alamos National Laboratory report LA-4700.
- Harlow, F.H., Amsden, A.A. and Nix, J.R., 1976, Relativistic fluid dynamics calculations with the particle-in-cell technique. *J. Comput. Phys.*, Vol. 20, p. 119-129.
- Heinrich, P., 1991, Nonlinear numerical model of landslide-generated water waves. *Int. J. Eng. Fluid Mech.*, Vol. 4(4), p. 403-416.
- Hirsch, C., 1988, Numerical computation of internal and external flows, Volume 1, Fundamentals of numerical discretization. New York: John Wiley & Sons.
- Hirsch, C., 1990, Numerical computation of internal and external flows, Volume 2, Computational methods for inviscid and viscous flows. New York: John Wiley & Sons.
- Hirt, C.W. and Nichols, B.D., 1981, Volume of fluid (VOF) method for the dynamics of free boundaries. *J. Comput. Phys.*, Vol. 39, p. 201-225.
- Hirt, C.W. and Shannon, J.P., 1968, Free-surface stress conditions for incompressible-flow calculations. *J. Comput. Phys.*, Vol. 2, p. 403-411.
- Hyman, J.M., 1984, Numerical methods for tracking interfaces. *Physica*, Vol. 12D, p. 396-407.

- Ishii, M., 1975, Thermo-fluid dynamic theory of two-phase flow. Eyrolles -x-70-06741-1.
- Issa, R.I., 1986, Solution of the implicitly discretised fluid flow equations by operator-splitting. *J. Comput. Phys.*, Vol. 62(1), p. 40-65.
- Jasak, H., 1996, Error analysis and estimation for the finite volume method with applications to fluid flows. Phd Thesis, University of London.
- Jasak, H., Weller, H.C., Issa, R.I. and Gosman, A.D., 1996, High resolution NVD differencing scheme for arbitrarily unstructured meshes. To be published.
- Koshizuka, S., 1995, Personal communication.
- Koshizuka, S., Tamako, H. and Oka, Y., 1995, A particle method for incompressible viscous flow with fluid fragmentation. *Computational Fluid Dynamics JOURNAL*, Vol. 4(1), p. 29-46.
- Kothe, D.B. and Mjolsness, R.C., 1992, RIPPLE: A new method for incompressible flows with free surfaces. *AIAA Journal*, Vol. 30 (11), p. 2694-2700.
- Lafaurie, B., Nardone, C., Scardovelli, R., Zaleski, S. and Zanetti, G., 1994, Modelling merging and fragmentation in multiphase flows with SURFER. *J. Comput. Phys.*, Vol. 113, p. 134-147.
- Lai, C.J. and Yen, C.W., 1993, Turbulent free surface flow simulation using a multilayer model. *Int. J. Numer. Methods Fluids*, Vol. 16, p. 1007-1025.
- Launder, B.E. and Spalding, D.B., 1974, The numerical computation of turbulent flows. *Comp. Meth. in Appl. Mech. and Eng.*, Vol. 3, p. 269-289.
- Lemos, C.M., 1992, A simple numerical technique for turbulent flows with free surfaces. *Int. J. Numer. Methods Fluids*, Vol. 15, p. 127-146.
- Leonard, B.P., 1979, A stable and accurate convective modelling procedure based on quadratic upstream interpolation. *Comp. Meth. in Appl. Mech. and Eng.*, Vol. 19, p. 59-98.
- Leonard, B.P., 1991, The ULTIMATE conservative difference scheme applied to unsteady one-dimensional advection. *Comp. Meth. in Appl. Mech. and Eng.*, Vol. 88, p. 17-74.
- Li, X.L., 1993, Study of three-dimensional Rayleigh-Taylor instability compressible fluids through level set method and parallel computation. *Phys. Fluids A*, Vol. 5 (8), p. 1904-1913.
- Li, Y.S. and Zhan, J.M., 1993, An efficient three-dimensional semi-implicit finite element scheme for simulation of free surface flows. *Int. J. Numer. Methods Fluids*, Vol. 16, p. 187-198.
- Lötstedt, P., 1982, A front tracking method applied to Burger's equation and two-phase porous flow. *J. Comput. Phys.*, Vol. 47, p. 211-228.

- Mao, Z.S. and Dukler, A.E., 1989, An experimental study of gas-liquid slug flow. *Experiments in Fluids*, Vol. 8, p. 169-182.
- Martin, J.C. and Moyce, W.J., 1952, An experimental study of the collapse of liquid columns on a rigid horizontal plane. *Philos. Trans. Roy. Soc. London*, Vol. A244, p. 312-324.
- Massey, B.S., 1979, Mechanics of fluids. London: Van Nostrand Reinhold Company.
- Maxwell, T.T., 1977, Numerical modelling of free-surface flows. Phd Thesis, University of London.
- Muzafferija, S., Peric, M. and Seidl, V., 1995, Computation of flow around circular cylinder in a channel. Internal Report, Institut fur Schiffbau, University of Hamburg. To be published.
- Nichols, B.D. and Hirt, C.W., 1971, Improved free surface boundary conditions for numerical incompressible-flow calculations. *J. Comput. Phys.*, Vol. 8, p. 434-448.
- Nichols, B.D. and Hirt, C.W., 1973, Calculating three-dimensional free surface flows in the vicinity of submerged and exposed structures. *J. Comput. Phys.*, Vol. 12, p. 234-246.
- Noh, W.F. and Woodward, P., 1976, SLIC (Simple Line Interface Calculations). *Lecture Notes in Physics*, Vol. 59, p. 330-340.
- Osher, S. and Sethian, J.A., 1988, Fronts propagating with curvature-dependant speed: algorithms based on Hamilton-Jacobi formulations. *J. Comput. Phys.*, Vol. 79, p. 12-49.
- Patankar, S.V., 1980, Numerical heat transfer and fluid flow. New York: McGraw-Hill.
- Patankar, S.V. and Spalding, D.B., 1972, A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *Int. J. Heat and Mass Transfer*, Vol. 15, p. 1787-1806.
- Perić, M., 1985, A finite volume method for the prediction of three-dimensional flow in complex ducts. Phd Thesis, University of London.
- Pericleous, K.A. and Chan, K.S., 1994, The sea method for free surface problems with heat transfer and change of phase. In: Crowe, C.T. et al (eds) Numerical methods in multiphase flows 1994, presented at the 1994 ASME fluids engineering division summer meeting, Lake Tahoe, Nevada, June 19-23
- Prakash, C., 1981, A finite element method for predicting flow through ducts with arbitrary cross section. PhD Thesis, University of Minnesota.
- Raad, P.E., Chen, S. and Johnson, D.B., 1995, The introduction of micro cells to treat pressure in free surface fluid flow problems. *Journal of Fluids Engineering*, Vol. 117, p. 683-690.

- Raithby, G.D. and Schneider, G.E., 1979, Numerical solution of problems in incompressible fluid flow: treatment of the velocity-pressure coupling. *Numerical Heat Transfer*, Vol. 2, p. 417-440.
- Ramaswamy, B. and Kawahara, M., 1987, Lagrangian finite element analysis applied to viscous free surface fluid flow. *Int. J. Numer. Methods Fluids*, Vol. 7, p. 953-984.
- Ramshaw, J.D. and Trapp, J.A., 1976, A numerical technique for low-speed homogeneous two-phase flow with sharp interfaces. *J. Comput. Phys.*, Vol. 21, p. 438-453.
- Renardy, M. and Renardy, Y., 1991, On the nature of boundary conditions for flows with moving free surfaces. *J. Comput. Phys.*, Vol. 93, p. 325-335.
- Rhie, C.M. and Chow, W.L., 1983, A numerical study of the turbulent flow past an isolated airfoil with trailing edge separation. *AIAA Journal*, Vol. 21, p. 1525-1532.
- Richardson, S.M., 1989, Fluid mechanics. New York: Hemisphere publishing corporation.
- Sabersky, R.H. and Acosta, A.J., 1964, Fluid Flow. London: Collier-Macmillan Limited..
- Sethian, J.A., 1996, Level set methods: Evolving interfaces in geometry, fluid mechanics, computer vision and materials sciences. Cambridge University Press.
- Shepherd, D.G., 1965, Elements of fluid mechanics. New York: Harcourt, Brace & World, Inc..
- Soulis, J.V., 1992, Computation of two-dimensional dam-break flood flows. *Int. J. Numer. Methods Fluids*, Vol. 14, p. 631-664.
- Spalding, D.B., 1974, A method for computing steady and unsteady flows possessing discontinuities of density. CHAM Report 910/2..
- Stoker, J.J., 1958, Water waves. New York: John Wiley & Sons.
- Sussman, M., Smereka, P. and Osher, S., 1994, A level set approach for computing solutions to incompressible two-phase flow. *J. Comput. Phys.*, Vol. 114, p. 146-159.
- Tadjbakhsh, I. and Keller, J.B., 1960, Standing surface waves of finite amplitude. *J. Fluid Mech.*, Vol. 8, p. 442-451.
- Takizawa, A., Koshizuka, S. and Kondo, S., 1992, Generalization of physical component boundary fitted co-ordinate (PCBFC) method for the analysis of free-surface flow. *Int. J. Numer. Methods Fluids*, Vol. 15, p. 1213-1237.
- Tomiyama, A., Sou, A. and Sakaguchi, T., 1994, Numerical simulation of a Taylor Bubble in a stagnant liquid. In: Crowe, C.T. et al (eds) *Numerical methods in multiphase flows 1994*, presented at the 1994 ASME fluids engineering division summer meeting, Lake Tahoe, Nevada, June 19-23.

- Tomiyama, A., Sou, A., Minagawa, H. and Sakaguchi, T., 1993a, Numerical analysis of a single bubble by VOF method. JSME Int. J. Series B, Vol. 36, p. 51-56.
- Tomiyama, A., Zun, I., Sou, A. and Sakaguchi, T., 1993b, Numerical analysis of bubble motion with the VOF method. Nuclear Engineering and Design, Vol. 141, p. 69-82.
- Torrey, M.D., Mjolsness, R.C. and Stein, L.R., 1987, NASA-VOF3D: A three dimensional computer program for incompressible flows with free surfaces. Los Alamos National Laboratory report LA-11009-MS.
- Unverdi, S.O. and Tryggvason, G., 1992, A front-tracking method for viscous, incompressible, multi-fluid flows. J. Comput. Phys., Vol. 100, p. 25-37.
- Van Leer, B., 1977, Towards the ultimate conservative differencing scheme. IV. A new approach to numerical convection. J. Comput. Phys., Vol. 23, p. 276-299.
- Vanka, S.P., 1986, Block-implicit multigrid solution of Navier-Stokes equations in primitive variables. J. Comput. Phys., Vol. 65, p. 138-158.
- Versteeg, H.K. and Malalasekera, W., 1995, An introduction to computational fluid dynamics, the finite volume method. Harlow: Longman Scientific & Technical.
- White, E.T. and Beardmore, R.H., 1962, The velocity of rise of single cylindrical air bubbles through liquids contained in vertical tubes. Chemical Engineering Science, Vol. 17, p. 351-361.
- White, F.M., 1994, Fluid mechanics. 3rd ed. London: McGraw-Hill.
- Youngs, D.L., 1982, Time-dependent multi-material flow with large fluid distortion. In: Morton, K.W. and Baines M.J. (eds.) Numerical Methods for fluid dynamics. London: Academic Press. p. 273-285.
- Yuan, S.W., 1970, Foundations of fluid mechanics. London: Prentice-Hall International.
- Yule, A.J. and Filipovic, I., 1992, On the break-up times and lengths of diesel sprays. Int. J. Heat and Fluid Flow, Vol. 13 (2), p. 197-206.
- Zhou, G., 1995, Numerical simulations of physical discontinuities in single and multi-fluid flows for arbitrary mach numbers. PhD Thesis, Chalmers University of Technology, Gothenburg.