UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

Recent Developments in the MAC Technique

M. F. Tomé
J. A. Cuminato
A. Castelo
V. G. Ferreira
S. McKee

Nº 74

NOTAS

São Carlos - SP

UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação
ISSN 0103-2577

Recent Developments in the MAC Technique

M. F. Tomé
J. A. Cuminato
A. Castelo
V. G. Ferreira
S. McKee

Nº 74

NOTAS

Série Computação

São Carlos – SP
Set./2003

Recent Developments in the MAC Technique

M. F. Tomé, J. A. Cuminato, A. Castelo, V. G. Ferreira
Department of Computer Science and Statistics
USP – University of São Paulo
São Carlos – SP – Brazil

S. McKee
Department of Mathematics
University of Strathclyde
Glasgow - UK

Abstract

In this article recent developments in the MAC technique will be reviewed. The MAC method dates back to the early sixties at the Los Alamos Laboratories and this paper starts with a historical review, and then a summary of related techniques. Improvements since the early days of MAC (and the Simplified MAC – SMAC) include automatic time-stepping, the use of the conjugate gradient method to solve the Poisson equation for the corrected velocity potential, greater efficiency through stripping out all particles (markers) other than those near the free surface, more accurate approximations of the free surface boundary conditions, the addition of a bounded high accuracy upwinding for the convected terms (thereby being able to solve higher Reynolds number flows), and a (dynamic) flow visualization facility. This article will concentrate, in the main, on a three-dimensional version of the SMAC method. It will show how to approximate curved boundaries by considering one configurational example in detail; the same will also be done for the free surface. The article will avoid validation, but rather focus on many of the examples and applications that the MAC method can solve from turbulent flows to rheology. It will conclude with some speculative comments on the future direction of the methodology.

Resumo

No presente artigo, desenvolvimentos recentes na técnica MAC são examinados. Inicia-se com uma visão histórica da metodologia MAC, e, então, resumem-se técnicas relacionadas. Melhoramentos desde a infância da técnica MAC (e do MAC Simplificado – SMAC) incluem o controle automático no tamanho do passo temporal, o uso do método de gradientes conjugados para resolver eficientemente a equação de Poisson, o emprego de aproximações mais precisas para as condições de contorno na superfície livre do fluido, a inclusão de um esquema "upwind" limitado e de alta ordem para representar os termos não-lineares nas equações do movimento (permitindo simular numericamente escoamentos a altos volores do número de Reynolds), e o uso de um ambiente de visualização da dinâmica do escoamento. O artigo dá atenção especial em uma versão tridimensional do método SMAC. Mostra-se, também, como aproximar contornos irregulares, considerando-se em detalhes um exemplo representativo. Considerações análogas, são feitas para a superfície livre. Evita-se neste artigo validação. Entretanto, apresentam-se exemplos e aplicações de escoamentos turbulentos e viscoelásticos. Conclui-se o trabalho com alguns comentarios especulativos, no que diz respeito ao futuro da metodologia.

1 INTRODUCTION

The purpose of this article is to provide an overview of the recent developments in the Marker-and-Cell (MAC) method. Although it is a technology dating back to the early sixties at the Los Alamos Laboratories, with the considerably greater computing power that we now enjoy, it is witnessing a revival and, for free surface fluid flow problems, showing itself the equal of any of the competing methods.

This review article will provide a brief history of the MAC technique, and even briefer description of related methods but will include a fairly complete description of the techniques and ideas behind the method. It will show that, with high order monotone upwinding approximation of the inertial terms, high Reynolds number flows are attainable and this will be illustrated by the hydraulic jump and a turbulent jet in a pool. Viscous jet flow will be studied, particularly viscous jet buckling, examples of container filling will be given, and a liquid drop splashing onto a quiescent fluid will be simulated. Finally, we shall show how to extend the MAC method to viscoelastic fluid flow, thus opening up the field of computational free surface rheology.

2 THE HISTORY AND BACKGROUND OF THE MAC METHOD

Based on a staggered grid system, the MAC is a finite difference solution technique for investigating the dynamics of an incompressible viscous fluid. It was first introduced by Harlow and Welch (see [82]) in 1965. It employs the primitive variables of pressure and velocity and has particular application to the modelling of fluid flows with free surfaces. One of the key features is the use of Lagrangian virtual particles whose coordinates are stored and which move from one cell to the next according to the latest computed velocity field. If a cell contains a particle, it is deemed to contain fluid, thus providing flow visualization of the free surface.

In 1970, Amsden and Harlow [1] subsequently developed a simplified MAC method (SMAC), which circumvented difficulties with the original method by splitting the calculation cycle into two parts, namely: a provisional velocity field calculation, followed by a velocity revision employing an auxiliary potential function to ensure incompressibility throughout. Amsden and Harlow [1] describe a specific program, ZUNI, which embodies the SMAC methodology. ZUNI was used to calculate two-dimensional flows in rectangular or cylindrical coordinates. It could deal with free surface flows with free-slip and/or no-slip conditions applied on the rigid boundaries. In addition, provision is made for prescribed inflows and outflows, and a rectangular obstacle can be incorporated into the region. Fundamentally, though, the fluid domain was required to be rectangular and it was restricted to two dimensions.

There have been a number, or developments, of the SMAC code over the intervening years. Due to the limitation on the time step-size, Deville [19], Pracht [57] and Golafshani [27] developed an implicit scheme. In particular, Pracht [57] developed an implicit treatment of the velocity similar to the implicit-fluid Eulerian (ICE) method (see Harlow and Welch [29]), using an Arbitrary Lagrangian-Eulerian (ALE) computational mesh. This, in principle, allowed the calculation of flows involving curved or moving boundaries. Hirt and Shannon [31] and Nichols and Hirt [51] were concerned with loss of accuracy at the free surface and suggested an improved

treatment for the free surface stress conditions whereby the normal stress is applied on the actual fluid surface rather than at the centre of the surface cells.

The 1980s saw many applications of the SMAC method: Sicilian and Hirt [64] developed a Containment Atmosphere Prediction (CAP) code using a Particle-In-Cell (PIC) approach to model the flow of a containment; Miyata and Nishimura [48] and Miyata [49] used SMAC for the simulation of water waves generated by ships and breaking waves over circular and elliptical bodies. At the same time, the UK Electricity industry (then the Central Electricity Generating Board) was taking an interest in the MAC method. McQueen and Rutter [46] and Markham and Proctor [45] described modifications to the original fluid flow code ZUNI to provide enhanced performance. Essentially, they employed an automatic time-stepping routine and a preconditioned conjugate gradient solver for the Poisson equation in a rectangular region.

In the 1990s, many authors considered different, but related methods, like Volume Of Fluid (VOF) [33], Euler-Lagrangian tracking methods and, more recently, Level Set methods, a survey about which is given in the next section. Nevertheless, there was a revival of the MAC method, principally by a research group in Brazil. Motivated initially by a research contract from Unilever Research plc to simulate container filling, Tomé and McKee [72] substantially extended the work of McQueen and his group [45,46], developing an improved version for general regions called GENSMAC. An adaptation for generalized Newtonian flow then followed (Tomé et al. [73]), viscous jet buckling was studied using GENSMAC (see Tomé and McKee [75]) and apparatus was built to experimentally validate the code (Tomé et al. [76]).

At the turn of the millenium saw the development of a solid modelling shell for enhanced visual input and output (see Castelo et al. [11]). An axisymmetric version (Tomé et al. [77]) permitted the simulation of G.I. Taylor experiments on jets impinging on a quiescent fluid, a full three-dimensional version was produced [11], [78] and high order monotone upwinding techniques were introduced to permit the simulation of a hydraulic jump (see Ferreira et al. [22]). In particular the code was shown capable of simulating effects such as the double roller (two vortices at the transition layer) effect predicted by the experiments of Craik et al. [14].

More recently, the GENSMAC code has been extended (in two dimensions only) to solve problems in viscoelastic (Tomé et al. [79]) and turbulent flows (Ferreira et al. [23]). In particular, the Oldroyd-B fluid model has been simulated and several classical problems have been solved.

3 REVIEW OF RELATED TECHNIQUES

Rather than employing virtual marker particles the volume of fluid (VOF) approach (see, eg. [34], [52], [80]), the volume fraction $F(\omega, t)$ is used to represent the free surface. Typically ω represents a computational cell $\omega_{i,j}$, eg. $\omega_{i,j} = \{\mathbf{x}; \mathbf{x}_i \leq \mathbf{x} \leq \mathbf{x}_{i+1}\}$. If $F(\omega, t) = 0$ it is void of fluid. If $0 \leq F(\omega, t) \leq 1$ then ω contains some fluid.

An advantage of representing the free surface as a volume fraction is the fact that one can write accurate algorithms for advecting the volume fraction so that mass is conserved, while still maintaining a sharp representation of the free surface. On the other hand, a disavantage of the volume of fluid approach is the fact that it is difficult to compute accurately local curvature from volume fractions. Some authors (eg. [66, 67]) have attempted to overcome this difficulty,

but it still remains a problem.

Another approach that has found favour in recent years is the level set method. This was originally introduced by Osher and Sethian [53], and Sethian has produced two useful books on the subject [62], [63]. In the level set method a smooth function, called the level set function, is used to represent the free surface. Liquid regions are regions which have $\phi(\mathbf{x}, t) > 0$, while gas (or void) regions are regions in which $\phi(\mathbf{x}, t) < 0$. The free surface is the set of points such that $\phi(\mathbf{x}, t) = 0$. One of the advantages of the level set method is its simplicity, especially when computing the curvature on a moving surface. By representing the free surface in this way, the unit normal vector $\mathbf{n}$ and the mean curvature k are simply

$$\mathbf{n} = \frac{\nabla\phi}{|\nabla\phi|}$$

and

$$\mathrm{k} = \nabla \cdot \frac{\nabla\phi}{|\nabla\phi|},$$

respectively. The level set method has the principal disavantage that the discretisation of the equation to advect the level set function

$$\phi_t + \mathbf{u} \cdot \nabla\phi = 0$$

where $\mathbf{u}$ is the underlying velocity field, is likely to incur more numerical errors than a front-tracking volume-of-fluid method, particularly if the free surface experiences rapid changes in curvature. A common problem is that mass is not conserved. In an attempt to counter this difficulty, Sussman and Puckett [67] have introduced a hybrid level set/VOF technique (CLSVOF); their particular interest was two-phase flows. Sussman et al. [66] also considered an adaptive level set approach with the view to obtaining higher resolution on the free surface with a minimum of additional expense.

The boundary integral method is another approach that may be used to resolve the free surface and can be effective when inertia forces are negligible. Baker and Moore [3] and Tsai and Miksis [69] have used this approach to solve axisymmetric bubbles rising in a liquid. Impressive results were also obtained for the problem of bursting bubbles by Boulton and Blake (see [5]).

Front or shock-capturing methods are usually associated with compressible fluids; these methods are now extremely sophisticated, explicity enforcing monotonicity through a nonlinear step while simultaneously maintaining high order. The reader is referred to, for instance, the books by Hirsch [30] and LeVeque [41]. The implementation of these ideas is usually credited to Glimm et al. [26]. They represent the moving front by a connected set of points, which form a moving internal boundary. To calculate the evolution inside the fluid in the vicinity of the interface, an irregular grid is constructed and a special finite difference stencil is used on these irregular grids. Authors who have used this approach to different flow regimes include Chenn et al. [12], Darip et al. [18], Moretti [50] and Peskin [56] (see also Fanci and Peskin [21] and Fogelson and Peskin [25]).

There are clearly many approaches to resolving the free surface of a dynamic fluid. Most are modifications, adaptations and refinements of the original idea put forward by Harlow and Welch [28]. Most of these new methods no longer rely on virtual particles. However, with greater computing power, the simplicity and robustness of the SMAC methodology is now having a greater appeal. Certainly, it is highly effective for low viscosity flows, particularly for non-Newtonian viscoelastic fluids. As we shall see it is also able to cope with medium to high Reynolds number flows and with flows that have a high rate of curvature.

4 THE EMPHASIS OF THE REVIEW

Having provided a historical backdrop, the purpose of this paper is to describe the essence of the modern marker and cell approach in sufficient detail to convey the flavour of the method, while not overburdening the reader with all the minutiae associated with the methodology. For example, in three dimensions to approximate the free surface in a reasonably accurate manner it is necessary to consider 24 cases; in this review we feel it is more useful to consider one in some detail. The same is also true of how many approximations are made on the boundary. Equally of the many bounded upwinding schemes available, we focus on the Variable-Order Non-Oscillatory Scheme (VONOS) (Varonos & Bergeles [81]). This paper contains no validation; extensive validation may be found in many of the references cited. However, the review does include a number of applications, by no means exhaustive, to illustrate the potential of the method. The paper concludes with a speculative discussion on the future of the MAC method and the sort of applications that might be expected in the next twenty years or so.

5 GOVERNING EQUATIONS

The basic equations governing the flow of an incompressible Newtonian fluid are the non-dimensional Navier-Stokes equations

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla.(\mathbf{uu}) = -\nabla p + \frac{1}{Re}\nabla^2\mathbf{u} + \frac{1}{Fr^2}\mathbf{g} \tag{4}$$

and the mass conservation equation

$$\nabla.\mathbf{u} = 0. \tag{5}$$

In the above equations, t is the time, $\mathbf{u} = (u, v, w)$ is the non-dimensional velocity field and p is the (non-dimensional) pressure divided by the density. The non-dimensional parameters $Re = UL/\nu$ and $Fr = U/\sqrt{L|\mathbf{g}|}$ denote the associated Reynolds and Froude numbers, respectively, in which U is a characteristic velocity scale, L is a length scale, and ν is the kinematic viscosity of the fluid. g is the gravitational constant and $\mathbf{g}$ is the unit gravitational field vector. With suitable initial and boundary conditions, equations (4) – (5) form a closed system of partial differential equations for the unknowns $\mathbf{u}$ and p.

5.1 Boundary Conditions

The boundary conditions at the mesh boundary can be of several types, namely, no-slip, free-slip, prescribed inflow, prescribed outflow and continuative outflow. The application of these conditions for the three-dimensional case is a direct generalization of the two-dimensional case. For clarity, we shall present the equations for no-slip and prescribed inflow boundaries.

Let u_n, u_{m1} and u_{m2} denote the normal and tangential velocities to the boundary, respectively. Then, for a no-slip boundary we have

$$u_n = 0 \,, \qquad u_{m1} = 0 \,, \qquad u_{m2} = 0$$

and for a prescribed inflow

$$u_n = U_{inf} \,, \qquad u_{m1} = 0 \,, \qquad u_{m2} = 0$$

respectively. For the Poisson equation we require

$$\frac{\partial \psi}{\partial n} = 0 \qquad \text{on rigid boundaries and} \qquad \psi = 0 \qquad \text{on the free surface.}$$

In the equations above, the subscript n, $m1$ and $m2$ denote the normal and the two tangential directions to the boundary respectively.

5.2 Free Surface Stress Conditions

Three-dimensional free surface flows are highly dependent on how the stress conditions are imposed. However, in the literature various techniques have been proposed in which the stress conditions are replaced by the kinematic condition while the pressure on the free surface is set to zero [40], [52]. The tangential stresses are usually neglected (e.g. [2]).

The appropriate boundary conditions on the free surface, in the absence of surface tension, are (see Batchelor [4])

$$\mathbf{n} \cdot \boldsymbol{\sigma} \cdot \mathbf{n} = 0 \tag{6}$$

$$\mathbf{m1} \cdot \boldsymbol{\sigma} \cdot \mathbf{n} = 0 \tag{7}$$

$$\mathbf{m2} \cdot \boldsymbol{\sigma} \cdot \mathbf{n} = 0 \tag{8}$$

where $\boldsymbol{\sigma}$ is the stress tensor given by

$$\boldsymbol{\sigma} = -p\mathbf{I} + \frac{1}{Re}\left[\nabla \mathbf{u} + (\nabla \mathbf{u})^{\mathrm{T}}\right],$$

and $\mathbf{n} = (n_1, n_2, n_3)$ is the local outward unit normal vector to the surface; $\mathbf{m1}, \mathbf{m2}$ are local tangential vectors.

6 NUMERICAL PROCEDURE

To solve equations (4)–(5) the following procedure is adopted.

Let us suppose that at a given time, say t_0, the velocity field $\mathbf{u}(\mathbf{x}, t_0)$ is known and boundary conditions for the velocity and pressure are given. To compute the velocity field and the pressure field at the advanced time $t = t_0 + \delta t$, we proceed as follows:

Step 1: Let $\tilde{p}(\mathbf{x}, t_0)$ be a pressure field that satisfies the correct pressure condition on the free surface. This pressure field is computed according to the equations approximating the stress conditions and are given in Subsection 6.3

Step 2: Calculate the intermediate velocity field, $\tilde{\mathbf{u}}(\mathbf{x}, t)$, from

$$\frac{\partial \tilde{\mathbf{u}}}{\partial t} = -\nabla.(\mathbf{uu}) - \nabla\tilde{p} + \frac{1}{Re}\nabla^2\mathbf{u} + \frac{1}{Fr^2}\mathbf{g} \tag{9}$$

with $\tilde{\mathbf{u}}(\mathbf{x}, t_0) = \mathbf{u}(\mathbf{x}, t_0)$ using the correct boundary conditions for $\mathbf{u}(\mathbf{x}, t_0)$. Equation (9) is solved by an explicit finite difference method and the underlying difference equations will be given in Subsection 6.1. It can be shown [73] that $\tilde{\mathbf{u}}(\mathbf{x}, t)$ possesses the correct vorticity at time t. However, $\tilde{\mathbf{u}}(\mathbf{x}, t)$ does not satisfy $\nabla \cdot \tilde{\mathbf{u}}(\mathbf{x}, t) = 0$. Let

$$\mathbf{u}(\mathbf{x}, t) = \tilde{\mathbf{u}}(\mathbf{x}, t) - \nabla\psi(\mathbf{x}, t) \tag{10}$$

with

$$\nabla^2\psi(\mathbf{x}, t) = \nabla \cdot \tilde{\mathbf{u}}(\mathbf{x}, t). \tag{11}$$

Thus $\mathbf{u}(\mathbf{x}, t)$ now conserves mass, and the vorticity remains unaltered.

Step 3: Solve the Poisson equation (11).

Step 4: Compute the velocity field (10).

Step 5: Compute the pressure. It can be shown [73] that the pressure is given by

$$p(\mathbf{x}, t) = \tilde{p}(\mathbf{x}, t_0) + \frac{\psi(\mathbf{x}, t)}{\delta t}\ . \tag{12}$$

6.1 Basic Finite Difference Equations

In order to solve equations (9)–(12), the following approach is employed. A staggered grid is used. A typical cell is shown in figure 1. The velocity $\tilde{\mathbf{u}}$ is discretized at u, v and w-nodes respectively.

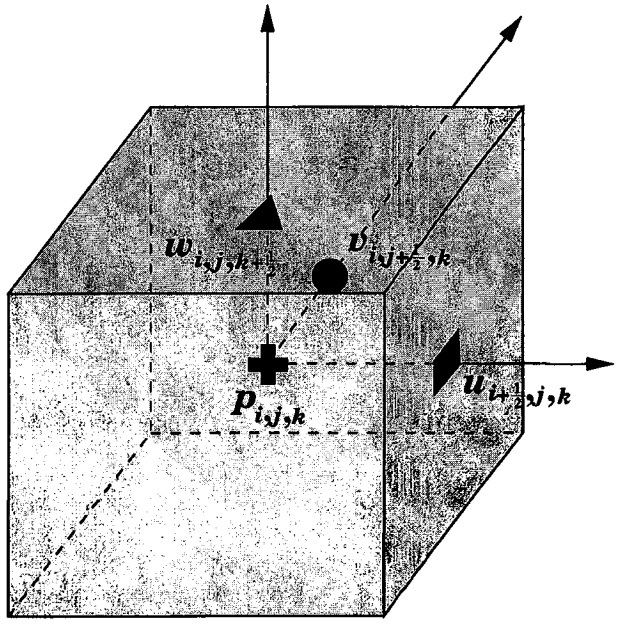

Figure 1. Typical cell in a GENSMAC3D calculation.

For instance, if equation (9) is considered, the discretization is performed as follows: the time derivative is discretized explicitly, while the spatial derivatives are approximated by central differences. The convective terms are approximated by the high order upwinding VONOS scheme. Details of these approximations will be given in Subsection 6.8. For instance, the discretized form of the x-component of equation (9) is given by

$$\tilde{u}_{i+\frac{1}{2},j,k} = u_{i+\frac{1}{2},j,k} + \delta t \left[\mathbf{conv}(uu)|_{i+\frac{1}{2},j} + \mathbf{conv}(vu)|_{i+\frac{1}{2},j} + \mathbf{conv}(wu)|_{i+\frac{1}{2},j} - \frac{\tilde{p}_{i+1,j,k} - \tilde{p}_{i,j,k}}{\delta x} \right.$$
$$+ \frac{1}{Re} \left(\frac{u_{i-\frac{1}{2},j,k} - 2u_{i+\frac{1}{2},j,k} u_{i+\frac{3}{2},j,k}}{\delta x^2} + \frac{u_{i+\frac{1}{2},j-1,k} - 2u_{i+\frac{1}{2},j,k} + u_{i+\frac{1}{2},j+1,k}}{\delta y^2} \right.$$
$$\left. \left. + \frac{u_{i+\frac{1}{2},j,k-1} - 2u_{i+\frac{1}{2},j,k} + u_{i+\frac{1}{2},j,k+1}}{\delta z^2} \right) + \frac{1}{Fr^2} g_x \right] , \tag{13}$$

where $\mathbf{conv}(uu)|_{i+\frac{1}{2},j}$, $\mathbf{conv}(vu)|_{i+\frac{1}{2},j}$ and $\mathbf{conv}(wu)|_{i+\frac{1}{2},j}$ represent the convective terms and will be dealt with in Subsection 6.8. The discretized form of the y and z-components of (9) are obtained similarly. The Poisson equation (11) is discretized at cell centres using the seven-point Laplacian giving

$$\frac{\psi_{i+1,j,k} - 2\psi_{i,j,k} + \psi_{i-1,j,k}}{\delta x^2} + \frac{\psi_{i,j+1,k} - 2\psi_{i,j,k} + \psi_{i,j-1,k}}{\delta y^2}$$
$$+ \frac{\psi_{i,j,k+1} - 2\psi_{i,j,k} + \psi_{i,j,k-1}}{\delta z^2} = \tilde{D}_{i,j,k} \tag{14}$$

where

$$\tilde{D}_{i,j,k} = \frac{\tilde{u}_{i+\frac{1}{2},j,k} - \tilde{u}_{i-\frac{1}{2},j,k}}{\delta x} + \frac{\tilde{v}_{i,j+\frac{1}{2},k} - \tilde{v}_{i,j-\frac{1}{2},k}}{\delta y} + \frac{\tilde{w}_{i,j,k+\frac{1}{2}} - \tilde{w}_{i,j,k-\frac{1}{2}}}{\delta z} .$$

The velocity at the advanced time t_{n+1} is obtained by discretizing (10) at the respective nodes, namely,

$$\begin{cases} u_{i+\frac{1}{2},j,k} = \tilde{u}_{i+\frac{1}{2},j,k} - \left(\dfrac{\psi_{i+1,j,k} - \psi_{i,j,k}}{\delta x} \right), \\ v_{i,j+\frac{1}{2},k} = \tilde{v}_{i,j+\frac{1}{2},k} - \left(\dfrac{\psi_{i,j+1,k} - \psi_{i,j,k}}{\delta y} \right), \\ w_{i,j,k+\frac{1}{2}} = \tilde{w}_{i,j,k+\frac{1}{2}} - \left(\dfrac{\psi_{i,j,k+1} - \psi_{i,j,k}}{\delta z} \right). \end{cases} \tag{15}$$

Thus, a calculational cycle consists of solving equations (13)–(15) at each time step.

6.2 Cell Flagging

The cells within the mesh can be of several types and a scheme for identifying them, similar to the 2D case, is employed. The cells within the mesh can be:

- Empty (E) - Cells that do not contain fluid.
- Full (F) - Cells full of fluid. These cells do not have any face contiguous with an Empty cell.
- Surface (S) - Cells that contain fluid and have at least one face contiguous with an Empty cell face. These cells contain the free surface.
- Boundary (B) - Cells that define a rigid boundary. In these cells the no-slip condition is applied.
- Inflow (I) - Cells that define an inflow boundary.

Figure 2 illustrates the cell structure for a two–dimensional slice of a trapezoidal tub at a given instant of time. For clarity, the empty cells are left blank.

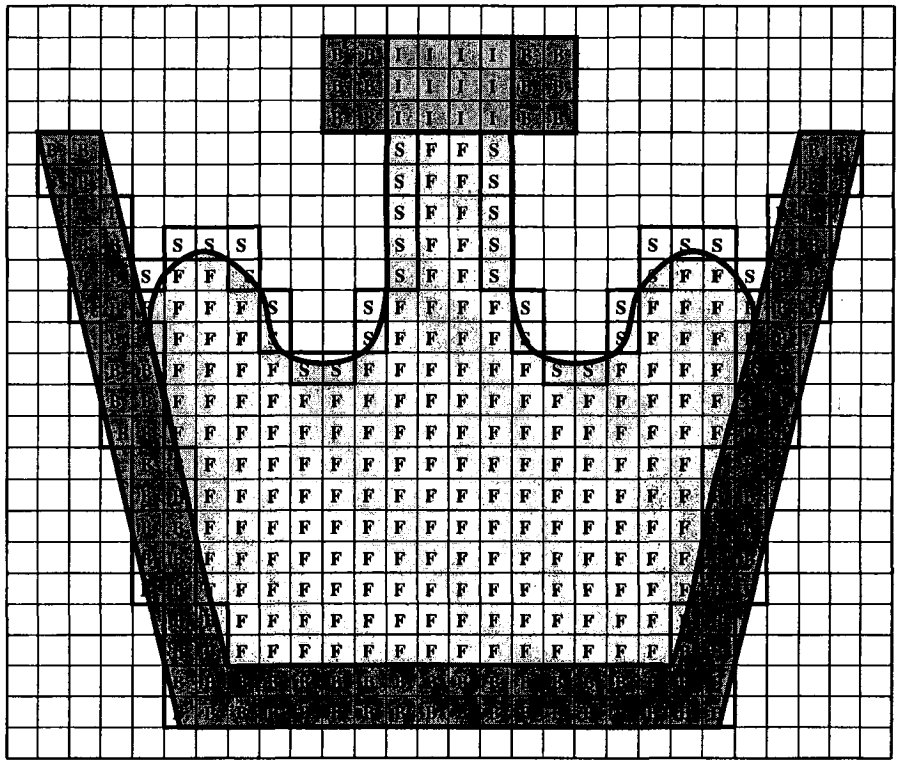

Figure 2. A two-dimensional slice indicating the types of cells used in a GENSMAC3D calculation.

6.3 Approximate Free Surface Stress Conditions

To apply the stress conditions (6)–(8), the ideas presented in GENSMAC [72] are extended as follows.

Let us suppose that the mesh spacing is small enough so that the free surface can be approximated by a planar surface. Then (6)–(8) can be approximated by local finite differences. In this review, the focus will be upon the $\mathbf{60^0}$-sloped plane. The reader is referred to Tomé et al. [78] for more details.

6.3.1 60^0–sloped planar surface: These surfaces are defined to have the local unit vector making $\mathbf{60^0}$ with the coordinate axes. They are identified by surface cells having three adjacent faces contiguous with empty cell faces (see figure 3). For these surfaces, the normal vector takes the form $\mathbf{n} = \left(\pm\frac{\sqrt{3}}{3}, \pm\frac{\sqrt{3}}{3}, \pm\frac{\sqrt{3}}{3}\right)$. It can be seen that there are 8 possible different positions for these planar surfaces. The approximating equations for one particular case will be given here; for details of each case see [74].

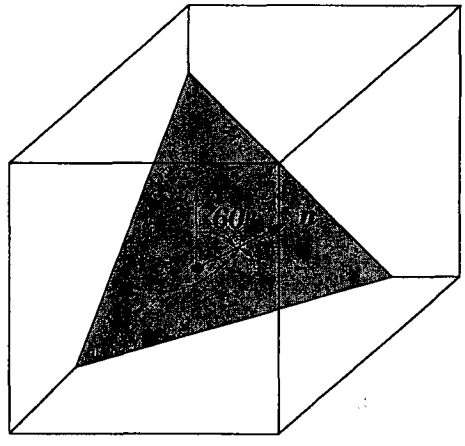

Figure 3. An example of a 60^0–planar surface.

Let us consider the surface cell in figure 4. For this cell we assume the local unit vectors take the form:

$$\mathbf{n} = \left(\frac{\sqrt{3}}{3}, \frac{\sqrt{3}}{3}, \frac{\sqrt{3}}{3}\right) \quad \mathbf{m1} = \left(0, \frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2}\right) \quad \mathbf{m2} = \left(-2\frac{\sqrt{6}}{6}, \frac{\sqrt{6}}{6}, \frac{\sqrt{6}}{6}\right) .$$

Introducing **n**, **m1** and **m2** into (6)–(8), we obtain a set of three equations. Adding (7) to (8) yields

$$-4\frac{\partial u}{\partial x} + 4\frac{\partial v}{\partial y} - 2\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right) + 2\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right) = 0 . \tag{16}$$

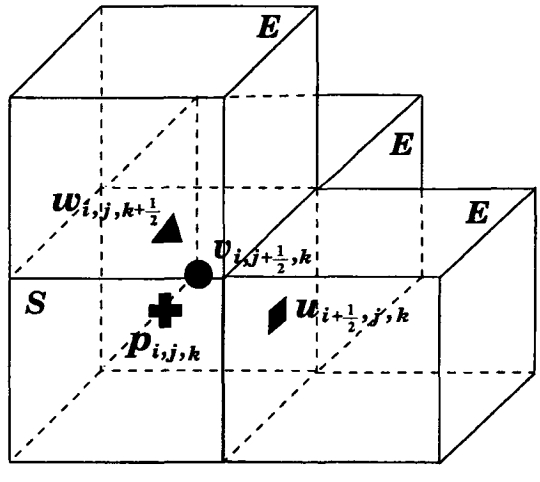

Figure 4. **S**–cell with the $(i+\frac{1}{2})$ and $(j+\frac{1}{2})$ and $(k+\frac{1}{2})$–faces contiguous with **E**–cell faces.

It can be seen that when calculating the tilde velocities through (9) the values of $u_{i+\frac{1}{2},j,k}$, $v_{i,j+\frac{1}{2},k}$ and $w_{i,j,k+\frac{1}{2}}$ are required. They can be obtained by applying finite differences to (7), (16) and the mass conservation equation (5) as follows. First, applying (5) at the surface cell centre we have

$$\frac{u_{i+\frac{1}{2},j,k} - u_{i-\frac{1}{2},j,k}}{\delta x} + \frac{v_{i,j+\frac{1}{2},k} - v_{i,j-\frac{1}{2},k}}{\delta y} + \frac{w_{i,j,k+\frac{1}{2}} - w_{i,j,k-\frac{1}{2}}}{\delta z} = 0$$

which can be written as

$$u_{i+\frac{1}{2},j,k} + \frac{\delta x}{\delta y} v_{i,j+\frac{1}{2},k} + \frac{\delta x}{\delta z} w_{i,j,k+\frac{1}{2}} = b_1 \tag{17}$$

where

$$b_1 = u_{i-\frac{1}{2},j,k} + \frac{\delta x}{\delta y} v_{i,j-\frac{1}{2},k} + \frac{\delta x}{\delta z} w_{i,j,k-\frac{1}{2}} \ .$$

Now, applying (7) at the surface cell centre gives

$$\begin{aligned}
&2\left(\frac{v_{i,j+\frac{1}{2},k} - v_{i,j-\frac{1}{2},k}}{\delta y}\right) - 2\left(\frac{w_{i,j,k+\frac{1}{2}} - w_{i,j,k-\frac{1}{2}}}{\delta z}\right) + \frac{1}{2}\left(\frac{u_{i+\frac{1}{2},j,k} + u_{i-\frac{1}{2},j,k}}{}\right. \\
&\left.\frac{-u_{i+\frac{1}{2},j-1,k} - u_{i-\frac{1}{2},j-1,k}}{\delta y} + \frac{v_{i,j+\frac{1}{2},k} + v_{i,j-\frac{1}{2},k} - v_{i-1,j+\frac{1}{2},k} - v_{i-1,j-\frac{1}{2},k}}{\delta x}\right) \\
&-\frac{1}{2}\left(\frac{u_{i+\frac{1}{2},j,k} + u_{i-\frac{1}{2},j,k} - u_{i+\frac{1}{2},j,k-1} - u_{i-\frac{1}{2},j,k-1}}{\delta z} + \frac{w_{i,j,k+\frac{1}{2}} + w_{i,j,k-\frac{1}{2}}}{}\right. \\
&\left.\frac{-w_{i-1,j,k+\frac{1}{2}} - w_{i-1,j,k-\frac{1}{2}}}{\delta x}\right) = 0
\end{aligned}$$

which can be re-written as

$$\left(1 - \frac{\delta y}{\delta z}\right) u_{i+\frac{1}{2},j,k} + \left(4 + \frac{\delta y}{\delta x}\right) v_{i,j+\frac{1}{2},k} - \left(4\frac{\delta y}{\delta z} + \frac{\delta y}{\delta x}\right) w_{i,j,k+\frac{1}{2}} = b_2 \tag{18}$$

where

$$\begin{aligned}
b_2 \ = \ & -u_{i-\frac{1}{2},j,k} + u_{i+\frac{1}{2},j-1,k} + u_{i-\frac{1}{2},j-1,k} + \left(\frac{\delta y}{\delta z}\right)\left(u_{i-\frac{1}{2},j,k} - u_{i+\frac{1}{2},j,k-1} - u_{i-\frac{1}{2},j,k-1}\right) \\
& +4v_{i,j-\frac{1}{2},k} - \left(\frac{\delta y}{\delta x}\right)\left(v_{i,j-\frac{1}{2},k} - v_{i-1,j+\frac{1}{2},k} - v_{i-1,j-\frac{1}{2},k}\right) \\
& -4\left(\frac{\delta y}{\delta z}\right) w_{i,j,k-\frac{1}{2}} + \left(\frac{\delta y}{\delta x}\right)\left(w_{i,j,k-\frac{1}{2}} - w_{i-1,j,k+\frac{1}{2}} - w_{i-1,j,k-\frac{1}{2}}\right) \ .
\end{aligned}$$

Similarly, discretizing (16) at surface cell position (i, j, k), we obtain

$$\begin{aligned}
&-4\left(\frac{u_{i+\frac{1}{2},j,k} - u_{i-\frac{1}{2},j,k}}{\delta x}\right) 4\left(\frac{v_{i,j+\frac{1}{2},k} - v_{i,j-\frac{1}{2},k}}{\delta y}\right) p - \left(\frac{u_{i+\frac{1}{2},j,k} + u_{i-\frac{1}{2},j,k}}{}\right. \\
&\left.\frac{-u_{i+\frac{1}{2},j,k-1} - u_{i-\frac{1}{2},j,k-1}}{\delta z} + \frac{w_{i,j,k+\frac{1}{2}} + w_{i,j,k-\frac{1}{2}} - w_{i-1,j,k+\frac{1}{2}} - w_{i-1,j,k-\frac{1}{2}}}{\delta x}\right) \\
&+\left(\frac{v_{i,j+\frac{1}{2},k} + v_{i,j-\frac{1}{2},k} - v_{i,j+\frac{1}{2},k-1} + v_{i,j-\frac{1}{2},k-1}}{\delta z} + \frac{w_{i,j,k+\frac{1}{2}} + w_{i,j,k-\frac{1}{2}}}{}\right. \\
&\left.\frac{-w_{i,j-1,k+\frac{1}{2}} - w_{i,j-1,k-\frac{1}{2}}}{\delta y}\right) = 0
\end{aligned}$$

which gives

$$-\left(4 + \frac{\delta x}{\delta z}\right) u_{i+\frac{1}{2},j,k} + \left(4\frac{\delta x}{\delta y} + \frac{\delta x}{\delta z}\right) v_{i,j+\frac{1}{2},k} + \left(\frac{\delta x}{\delta y} - 1\right) w_{i,j,k+\frac{1}{2}} = b_3 \tag{19}$$

where

$$
\begin{aligned}
b_3 \;=\; & -4u_{i-\frac{1}{2},j,k} + + \left(\frac{\delta x}{\delta z}\right)\left(u_{i-\frac{1}{2},j,k} - u_{i+\frac{1}{2},j,k-1} - u_{i-\frac{1}{2},j,k-1}\right) \\
& +4\frac{\delta x}{\delta y}v_{i,j-\frac{1}{2},k} - \left(\frac{\delta x}{\delta z}\right)\left(v_{i,j-\frac{1}{2},k} - v_{i,j+\frac{1}{2},k-1} - v_{i,j-\frac{1}{2},k-1}\right) \\
& +w_{i,j,k-\frac{1}{2}} - w_{i-1,j,k+\frac{1}{2}} - w_{i-1,j,k-\frac{1}{2}} - \left(\frac{\delta x}{\delta y}\right)\left(w_{i,j,k-\frac{1}{2}} - w_{i,j-1,k+\frac{1}{2}} - w_{i,j-1,k-\frac{1}{2}}\right) .
\end{aligned}
$$

Equations (17)–(19) provide a linear system for the unknowns $u_{i+\frac{1}{2},j,k}$, $v_{i,j+\frac{1}{2},k}$ and $w_{i,j,k+\frac{1}{2}}$, which in matrix form is given by

$$
\begin{bmatrix}
1 & \frac{\delta x}{\delta y} & \frac{\delta x}{\delta z} \\
\left(1-\frac{\delta y}{\delta z}\right) & \left(4+\frac{\delta y}{\delta x}\right) & -\left(4\frac{\delta y}{\delta z}+\frac{\delta y}{\delta x}\right) \\
-\left(4+\frac{\delta x}{\delta z}\right) & \left(4\frac{\delta x}{\delta y}+\frac{\delta x}{\delta z}\right) & \left(-1+\frac{\delta x}{\delta y}\right)
\end{bmatrix}
\begin{bmatrix} u_{i+\frac{1}{2},j,k} \\ v_{i,j+\frac{1}{2},k} \\ w_{i,j,k+\frac{1}{2}} \end{bmatrix}
=
\begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}
\tag{20}
$$

The system (20) can be easily solved by Gaussian elimination. Once the values of $u_{i+\frac{1}{2},j,k}$, $v_{i,j+\frac{1}{2},k}$ and $w_{i,j,k+\frac{1}{2}}$ have been computed, the pressure follows from (6) applied at the surface cell centre, giving

$$
\begin{aligned}
\tilde{p}_{i,j,k} = \frac{1}{3Re}\Bigg[\Bigg(&\frac{u_{i+\frac{1}{2},j,k} + u_{i-\frac{1}{2},j,k} - u_{i+\frac{1}{2},j-1,k} - u_{i-\frac{1}{2},j-1,k}}{\delta y} + \frac{v_{i,j+\frac{1}{2},k} + v_{i,j-\frac{1}{2},k}}{} \\
&\frac{-v_{i-1,j+\frac{1}{2},k} - v_{i-1,j-\frac{1}{2},k}}{\delta x}\Bigg) + \Bigg(\frac{u_{i+\frac{1}{2},j,k} + u_{i-\frac{1}{2},j,k} - u_{i+\frac{1}{2},j,k-1} + u_{i-\frac{1}{2},j,k-1}}{\delta z} \\
&+ \frac{w_{i,j,k+\frac{1}{2}} + w_{i,j,k-\frac{1}{2}} - w_{i-1,j,k+\frac{1}{2}} - w_{i-1,j,k-\frac{1}{2}}}{\delta x}\Bigg) + \Bigg(\frac{v_{i,j+\frac{1}{2},k} + v_{i,j-\frac{1}{2},k}}{} \\
&\frac{-v_{i,j+\frac{1}{2},k-1} - v_{i,j-\frac{1}{2},k-1}}{\delta z} + \frac{w_{i,j,k+\frac{1}{2}} + w_{i,j,k-\frac{1}{2}} - w_{i,j-1,k+\frac{1}{2}} - w_{i,j-1,k-\frac{1}{2}}}{\delta y}\Bigg)\Bigg] .
\end{aligned}
$$

The remaining configurations of surface cells having three adjacent faces contiguous with empty cells are treated similarly. For details of each case see Tomé et al. [74].

6.4 Boundary Conditions on Curved Surfaces

When the discretized Navier-Stokes equation (9) is applied at nodes adjacent to a boundary cell (B-cell), the components of the velocity field u, v and w on the boundary cell faces are required. If no-slip conditions are imposed on the boundary surface, these values can be estimated in terms of values at internal nodes and values on the boundary by linear interpolation. It is supposed that the mesh spacing is sufficiently small that the curved boundary cuts the cell faces according only to three cases:

- **1-axis surface**: These surfaces are assumed to cut the B-cell passing through only one of its axes (see figure 5 **a**). These surfaces are identified by B-cells having only one face contiguous with an interior cell.

- **2-axes surface**: A 2-axes surface is defined to be one that cuts a B-cell passing through two of its coordinate axes (see figure 5 **b**). These surfaces are identified by B-cells having only two adjacent faces contiguous with interior cells.
- **3-axes surface**: These surfaces are defined to cut the three coordinate axes of the cell (see figure 5 **c**). These surfaces are identified by B-cells having three adjacent faces contiguous with interior cells.

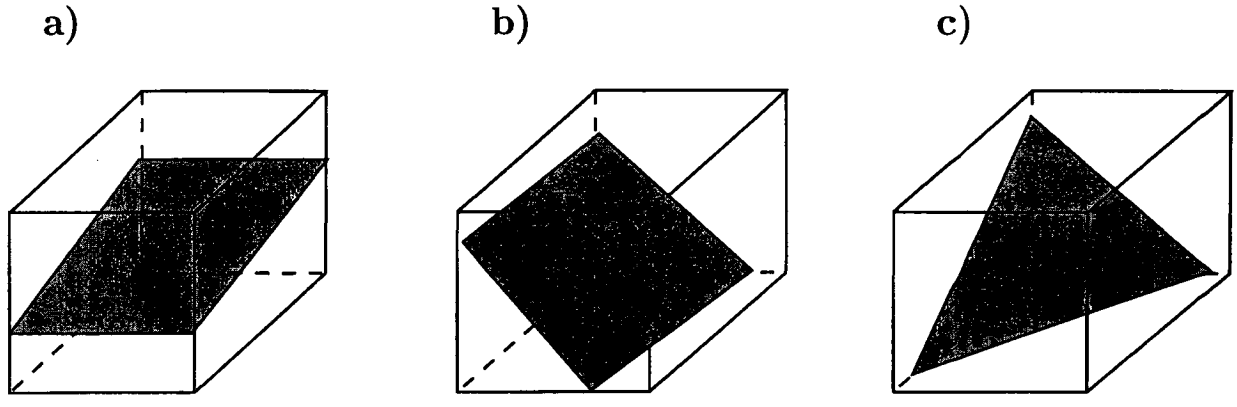

Figure 5. Examples of 1, 2 and 3–axes surfaces.

It can be seen that the boundary cells can have only one, two or three faces contiguous with interior cells. More specifically, there are 6 possible configurations of B–cells with only one face contiguous with an interior cell, 12 cases of B–cells with two adjacent faces contiguous with interior cells and 8 cases of B–cells having three adjacent faces contiguous with interior cells. In this Section, we give the equations for calculating the velocities on the boundary cell faces by considering various B–cell configurations as follows. Like the approximation of the free surface conditions, the implementation of the boundary conditions on a curved boundary is both complicated and tedious, but not essentially difficult; full details may be found in Tomé et al. [74], [78].

6.5 Free Surface Movement

One of the main challenges in generalizing the ideas of the two-dimensional GENSMAC code to three dimensions was in dealing with the free surface. This is because in two dimensions virtual particles were used to represent the fluid, and this technique cannot be carried over to three dimensions due to the very large number of virtual particles that would be needed to represent the fluid to photographic precision. Tomé et al. [77] overcame this problem by introducing a new procedure whereby marker particles were employed on the fluid surface only. This brought substantial savings in storage and computing time, making the extension of GENSMAC feasible for solving full three–dimensional problems efficiently. In three dimensions, it is desirable to employ solid modelling techniques whereby the fluid is modelled by a B-Rep structure [44]. The fluid surface is represented by a piecewise linear surface composed of quadrilaterals and triangles containing marker particles on their vertices. A procedure for inserting and deleting

particles on the free surface is employed. More details on the free surface treatment can be found in [11]. At the end of the calculational cycle these vertices (i.e. the coordinates of the virtual particles) are moved to a new position according to

$$\mathbf{x}^{n+1} = \mathbf{x}^n + \mathbf{u}_p \delta t^{n+1}$$

where $\mathbf{x}^n$ is the position of the particle under consideration. The particle velocity $\mathbf{u}_p$ is found by performing a tri-linear interpolation using the eight nearest velocities.

6.6 Time-Stepping Procedure

A time-stepping procedure for computing the appropriate time-step size for every cycle is employed. It is based on the stability conditions (written in non-dimensional form)

$$\delta t < \frac{\delta \mathbf{x}}{||\mathbf{u}||} \tag{21}$$

$$\delta t < \frac{\delta x^2 \delta y^2 \delta z^2}{\delta x^2 \delta y^2 + \delta x^2 \delta z^2 + \delta y^2 \delta z^2} \frac{Re}{2} \tag{22}$$

where the first inequality is understood componentwise. The restriction (21) requires that no particles should cross more than one cell boundary in a given time interval; this is an accuracy requirement. The second restriction (22) comes from the explicit discretization of the Navier-Stokes equations [56] and is essentially a local von Neumann stability requirement. Since low Reynolds number flows ($0 \leq Re \leq 10$) are the primary concern, it is anticipated that (22) is generally the more restrictive condition. In order to implement these equations, similar ideas to that of GENSMAC were employed.

Let δt^n be the time-step employed in the previous calculational cycle and define

$$\delta t_{visc} = \frac{1}{2} \frac{\delta x^2 \delta y^2 \delta z^2}{\delta x^2 \delta y^2 + \delta x^2 \delta z^2 + \delta y^2 \delta z^2} \frac{Re}{2} \tag{23}$$

$$\delta t_u = \lambda_1 \cdot \frac{1}{2} \cdot \frac{\delta x}{|\widetilde{u}_{max}|} \tag{24}$$

$$\delta t_v = \lambda_2 \cdot \frac{1}{2} \cdot \frac{\delta x}{|\widetilde{v}_{max}|} \tag{25}$$

$$\delta t_w = \lambda_3 \cdot \frac{1}{2} \cdot \frac{\delta x}{|\widetilde{w}_{max}|} \tag{26}$$

where $|\widetilde{u}_{max}|$, $|\widetilde{v}_{max}|$ and $|\widetilde{w}_{max}|$ are maximum of the tilde velocities computed through (7)–(9) and $0 < \lambda_i \leq 1, \quad i = 1, 2, 3$. The extra factor of 0.5 in (24)–(26) has been introduced as a conservative measure to allow for the fact that only local stability analyses have been performed. The time-step employed in the calculation is then chosen to be

$$\delta t^{n+1} = \lambda \cdot min\{\delta t_{visc}, \delta t_u, \delta t_v, \delta t_w\} \tag{27}$$

where $0 < \lambda \leq 1$.

The factor λ in (27) is necessary since the values of $|u_{max}^{n+1}|$, $|v_{max}^{n+1}|$ and $|w_{max}^{n+1}|$ at the begining of the calculational cycle are not known. It is computationally more efficient to use the tilde velocities with the factor λ acting as a compensating measure than using u_{max}^{n+1}, v_{max}^{n+1}, and w_{max}^{n+1}. However, if δt_u, δt_v or δt_w is less than δt^n then the tilde velocities are recalculated and the time-step is revised. This time-stepping procedure has proved to be an efficient automatic means of adjusting the size of the time-step.

6.7 Poisson Solver

At every calculational cycle, the discrete Poisson equation (14) requires to be solved subject to Dirichlet and Neumann boundary conditions. For three-dimensional problems, this can be extremely time consuming and the choice of an appropriate algorithm is vital. For two-dimensional problems, the conjugate gradient method proved to be the best solver, only taking a few iterations to converge to a prescribed tolerance. This would appear to be due to the small time-steps employed in the calculational cycle, which makes the solution from the previous calculational cycle a good approximation for the next. For the conjugate gradient solver, it was necessary to write a routine for assembling the matrix and a routine to perform the product of this matrix with a given vector. For details of the matrix assembly see Tomé and McKee [72].

6.8 Finite Difference Approximation of the Convective Terms

Here, for brevity, only the discretization of the convective terms in the $x-$momentum equation will be given; the other two–momentum components will be treated similarly.

The convective terms $\mathbf{conv}(uu)|_{i+\frac{1}{2},j,k}$, $\mathbf{conv}(vu)|_{i+\frac{1}{2},j,k}$ and $\mathbf{conv}(wu)|_{i+\frac{1}{2},j,k}$ can be written as

$$\left[\frac{\partial(uu)}{\partial x}+\frac{\partial(vu)}{\partial y}+\frac{\partial(wu)}{\partial z}\right]_{i+\frac{1}{2},j,k}=\left.\frac{\partial(uu)}{\partial x}\right|_{i+\frac{1}{2},j,k}+\left.\frac{\partial(vu)}{\partial y}\right|_{i+\frac{1}{2},j,k}+\left.\frac{\partial(wu)}{\partial z}\right|_{i+\frac{1}{2},j,k}. \tag{28}$$

The derivatives in (28) are approximated by

$$\left.\frac{\partial(uu)}{\partial x}\right|_{i+\frac{1}{2},j,k} = (\bar{u}_{i+1,j,k}u_{i+1,j,k}-\bar{u}_{i,j,k}u_{i,j,k})/\delta x, \tag{29}$$

$$\left.\frac{\partial(vu)}{\partial y}\right|_{i+\frac{1}{2},j,k} = (\bar{v}_{i+\frac{1}{2},j+\frac{1}{2},k}u_{i+\frac{1}{2},j+\frac{1}{2},k}-\bar{v}_{i+\frac{1}{2},j-\frac{1}{2},k}u_{i+\frac{1}{2},j-\frac{1}{2},k})/\delta y, \tag{30}$$

$$\left.\frac{\partial(wu)}{\partial z}\right|_{i+\frac{1}{2},j,k} = (\bar{w}_{i+\frac{1}{2},j,k+\frac{1}{2}}u_{i+\frac{1}{2},j,k+\frac{1}{2}}-\bar{w}_{i+\frac{1}{2},j,k-\frac{1}{2}}u_{i+\frac{1}{2},j,k-\frac{1}{2}})/\delta z\,. \tag{31}$$

The convective velocities appearing in (29)–(31) are obtained by averaging, namely

$$\bar{u}_{i,j,k}=0.5(u_{i+\frac{1}{2},j,k}+u_{i-\frac{1}{2},j,k}), \qquad \bar{u}_{i+1,j,k}=0.5(u_{i+\frac{3}{2},j,k}+u_{i+\frac{1}{2},j,k}),$$
$$\bar{v}_{i+\frac{1}{2},j+\frac{1}{2},k}=0.5(v_{i,j+\frac{1}{2},k}+v_{i+1,j+\frac{1}{2},k}), \qquad \bar{v}_{i+\frac{1}{2},j-\frac{1}{2},k}=0.5(v_{i,j-\frac{1}{2},k}+v_{i+1,j-\frac{1}{2},k}),$$
$$\bar{w}_{i+\frac{1}{2},j,k+\frac{1}{2}}=0.5(w_{i,j,k+\frac{1}{2}}+w_{i+1,j,k+\frac{1}{2}}), \qquad \bar{w}_{i+\frac{1}{2},j,k-\frac{1}{2}}=0.5(w_{i,j,k-\frac{1}{2}}+w_{i+1,j,k-\frac{1}{2}}).$$

A finite difference approximation for computing the first derivative in (29), which is upwinding and monotone, will now be presented. This upwinding scheme is due to Varonos and Bergeles [81]. There are of course other schemes (see, for details, Ferreira et al. [22]), but this one was found to be the most effective among those considered. The corresponding difference equations for calculating the other derivatives in (30) and (31) are obtained similarly. Firstly, let us define the following parameters:

$$S_{i+1,j,k} = \begin{cases} 0, & \text{if } \bar{u}_{i+1,j,k} \geq 0 \\ 1, & \text{otherwise,} \end{cases} \qquad S_{i,j,k} = \begin{cases} 0, & \text{if } \bar{u}_{i,j,k} \geq 0 \\ 1, & \text{otherwise.} \end{cases} \tag{32}$$

The VONOS scheme is implemented by using the following difference equations:

$$\widehat{\phi}_{i+1,j,k} := (1 - S_{i+1,j,k}) \left(\frac{(u_{i+\frac{1}{2},j,k} - u_{i-\frac{1}{2},j,k})}{(u_{i+\frac{3}{2},j,k} - u_{i-\frac{1}{2},j,k})} \right) + S_{i+1,j,k} \left(\frac{(u_{i+\frac{3}{2},j,k} - u_{i+\frac{5}{2},j,k})}{(u_{i+\frac{1}{2},j,k} - u_{i+\frac{5}{2},j,k})} \right),$$

$$u_{i+1,j,k} := (1 - S_{i+1,j,k}) \begin{cases} u_{i+\frac{1}{2},j,k}, & \text{if } \widehat{\phi}_{i+1,j,k} \notin [0,\ 1] \\ 10u_{i+\frac{1}{2},j,k} - 9u_{i-\frac{1}{2},j,k}, & \text{if } \widehat{\phi}_{i+1,j,k} \in [0,\ 3/74) \\ \frac{3}{8}u_{i+\frac{3}{2},j,k} + \frac{6}{8}u_{i+\frac{1}{2},j,k} - \frac{1}{8}u_{i-\frac{1}{2},j,k}, & \text{if } \widehat{\phi}_{i+1,j,k} \in [3/74,\ 1/2) \\ 1.5u_{i+\frac{1}{2},j,k} - 0.5u_{i-\frac{1}{2},j,k}, & \text{if } \widehat{\phi}_{i+1,j,k} \in [1/2,\ 2/3) \\ u_{i+\frac{3}{2},j,k}, & \text{if } \widehat{\phi}_{i+1,j,k} \in [2/3,\ 1] \end{cases}$$

$$+ S_{i+1,j,k} \begin{cases} u_{i+\frac{3}{2},j,k}, & \text{if } \widehat{\phi}_{i+1,j,k} \notin [0,\ 1] \\ 10u_{i+\frac{3}{2},j,k} - 9u_{i+\frac{5}{2},j,k}, & \text{if } \widehat{\phi}_{i+1,j,k} \in [0,\ 3/74) \\ \frac{3}{8}u_{i+\frac{1}{2},j,k} + \frac{6}{8}u_{i+\frac{3}{2},j,k} - \frac{1}{8}u_{i+\frac{5}{2},j,k}, & \text{if } \widehat{\phi}_{i+1,j,k} \in [3/74,\ 1/2) \\ 1.5u_{i+\frac{3}{2},j,k} - 0.5u_{i+\frac{5}{2},j,k}, & \text{if } \widehat{\phi}_{i+1,j,k} \in [1/2,\ 2/3) \\ u_{i+\frac{1}{2},j,k}, & \text{if } \widehat{\phi}_{i+1,j,k} \in [2/3,\ 1] \end{cases}.$$

$$\widehat{\phi}_{i,j,k} = (1 - S_{i,j,k}) \left(\frac{(u_{i-\frac{1}{2},j,k} - u_{i-\frac{3}{2},j,k})}{(u_{i+\frac{1}{2},j,k} - u_{i-\frac{3}{2},j,k})} \right) + S_{i,j,k} \left(\frac{(u_{i+\frac{1}{2},j,k} - u_{i+\frac{3}{2},j,k})}{(u_{i-\frac{1}{2},j,k} - u_{i+\frac{3}{2},j,k})} \right),$$

$$u_{i,j,k} := (1 - S_{i,j,k}) \begin{cases} u_{i-\frac{1}{2},j,k}, & \text{if } \widehat{\phi}_{i,j,k} \notin [0,\ 1] \\ 10u_{i-\frac{1}{2},j,k} - 9u_{i-\frac{3}{2},j,k}, & \text{if } \widehat{\phi}_{i,j,k} \in [0,\ 3/74) \\ \frac{3}{8}u_{i+\frac{1}{2},j,k} + \frac{6}{8}u_{i-\frac{1}{2},j,k} - \frac{1}{8}u_{i-\frac{3}{2},j,k}, & \text{if } \widehat{\phi}_{i,j,k} \in [3/74,\ 1/2) \\ 1.5u_{i-\frac{1}{2},j,k} - 0.5u_{i-\frac{3}{2},j,k}, & \text{if } \widehat{\phi}_{i,j,k} \in [1/2,\ 2/3) \\ u_{i+\frac{1}{2},j,k}, & \text{if } \widehat{\phi}_{i,j,k} \in [2/3,\ 1] \end{cases}$$

$$+ S_{i,j,k} \begin{cases} u_{i+\frac{1}{2},j,k}, & \text{if } \widehat{\phi}_{i,j,k} \notin [0,\ 1] \\ 10u_{i+\frac{1}{2},j,k} - 9u_{i+\frac{3}{2},j,k}, & \text{if } \widehat{\phi}_{i,j,k} \in [0,\ 3/74) \\ \frac{3}{8}u_{i-\frac{1}{2},j,k} + \frac{6}{8}u_{i+\frac{1}{2},j,k} - \frac{1}{8}u_{i+\frac{3}{2},j,k}, & \text{if } \widehat{\phi}_{i,j,k} \in [3/74,\ 1/2) \\ 1.5u_{i+\frac{1}{2},j,k} - 0.5u_{i+\frac{3}{2},j,k}, & \text{if } \widehat{\phi}_{i+1,j,k} \in [1/2,\ 2/3) \\ u_{i-\frac{1}{2},j,k}, & \text{if } \widehat{\phi}_{i,j,k} \in [2/3,\ 1] \end{cases}.$$

7 THE FREEFLOW3D ENVIRONMENT SYSTEM

FreeFlow3D is an integrated suite of programs which solve 3D incompressible generalized Newtonian flows with free surfaces. The flow domain can be quite arbitrary and needs only be connected. The Freeflow3D system has been designed in three distinct modules:

- Modflow3D (initial specification) - This module is an interactive system for specifying the domain and the initial fluid. It includes elements such as the container shapes, the injection nozzle(s) and the initial position, velocity and pressure of the fluid. These geometric objects are then created using solid modelling techniques.
- Simflow3D (flow simulator) - This system is the central part of FreeFlow3D and implements the discretization of the governing equations and boundary conditions. It uses the GENSMAC3D method [78], which is a variant of the GENSMAC method, with marker particles on the free surface only [77].
- Visflow3D (flow visualization) - An interactive system for the visualization of the output of Simflow3D using visualization techniques.

Visflow3D is responsible for the presentation of the output produced by Simflow3D and Modflow3D. This output comprises data from the flow model, object representation and properties of the fluid flow saved at pre-defined instants of time. Time-independent data are stored in a single file, while those which are time-dependent are stored in a sequence of files containing information on the geometry of objects, flow properties, cell configuration, time and cycle number. Visflow3D allows the user to view the geometry (containers, inflows, outflows and the fluid), the fluid flow itself and also the flow properties (such as velocity and pressure). Like Modflow3D, Visflow3D uses rendering techniques for object presentation in order to make user-understanding of the pictures more immediate.

The three modules were written using the programming language C under the operating system UNIX. The graphic interfaces to Modflow3D and Visflow3D use the windowing system Xview under X-Windows. The data structure was designed to permit easy access and data interdependency, in order to simplify software maintenance and facilitate extensions. The modules communicate with each other through data files.

8 APPLICATIONS

The purpose of this section is simply to display three applications of the three-dimensional code: jet buckling, a splashing drop and a hydraulic jump. The first arises when a thin film of a viscous fluid lands on a flat surface and, due to its viscosity, cannot disperse sufficiently rapidly laterally resulting in an accummulation of fluid at the point of impingement followed by fluid jet buckling. The second application involves a falling spherical drop of fluid onto a quiescent pool of fluid with the resulting splash and with radial waves travelling outwards, eventually being reflected off the sides of the container. The third application is concerned with a cylindrical jet of fluid impinging on a flat surface at some speed ($Re \approx 1,000$). A thin layer

of fluid travels radially outwards rapidly and unhindered by its relatively low viscosity until it reaches a transition point or hydraulic jump at which one or two vortices occur allowing the fluid to make the transition from a shallow fast flowing fluid to a deeper, considerably slower moving fluid.

8.1 Jet Buckling

When a jet flows onto a rigid plate, a phenomenon known as jet buckling can occur if the Reynolds number is smaller than a prescribed value. This phenomenon has attracted the attention of a number of researchers, and it has been studied both experimentally and numerically. Cruickshank and Munson [16] and Cruickshank [17] have presented both experimental and theoretical results for Newtonian fluid jets. From their study, they obtained estimates for when jet buckling occurs: these are based upon the Reynolds number and the ratio H/D (H is the height of the inlet to the rigid plate and D is the jet diamenter). In particular, for an axisymmetric jet they found that if the conditions

$$Re < 1.2 \qquad \text{and} \qquad H/D > 7.2 \tag{33}$$

are satisfied, then the jet will buckle. Tomé et al. [78] have applied the FreeFlow3D code to simulate this problem. They considered a thin jet of diameter $D = 5\text{mm}$ issuing from an inlet situated at a height $H = 5\text{cm}$ above the plate (so that $H/D = 10$), flowing onto a rigid plate of size 5cm $\times$ 5cm $\times$ 3mm (see figure 6 for details). A uniform input velocity of $U = 1\text{ms}^{-1}$ was set at the inlet. The fluid viscosity coefficient was $\nu = 0.02\text{m}^2\text{s}^{-1}$. The scaling paramenters were U, D, ν so that $Re = 0.25$ and $1/F_r^2 = 0.049$. A mesh size of $\delta x = \delta y = \delta z = 0.005\text{mm}$ ($100 \times 100 \times 100$ computational cells) was used. Figure 7 displays the fluid flow configuration at different times.

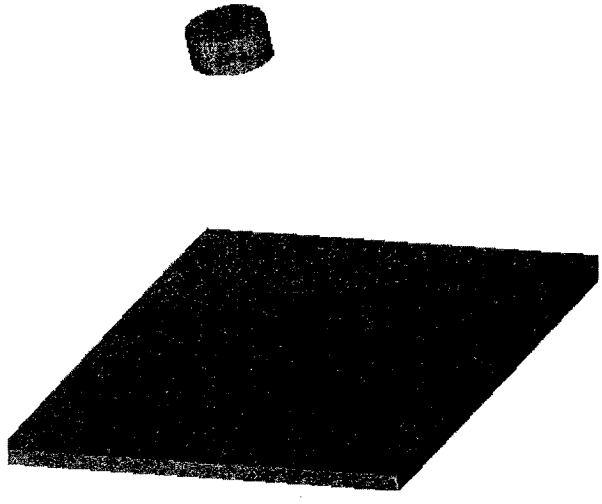

Figure 6. Domain specification for the simulation of jet buckling.

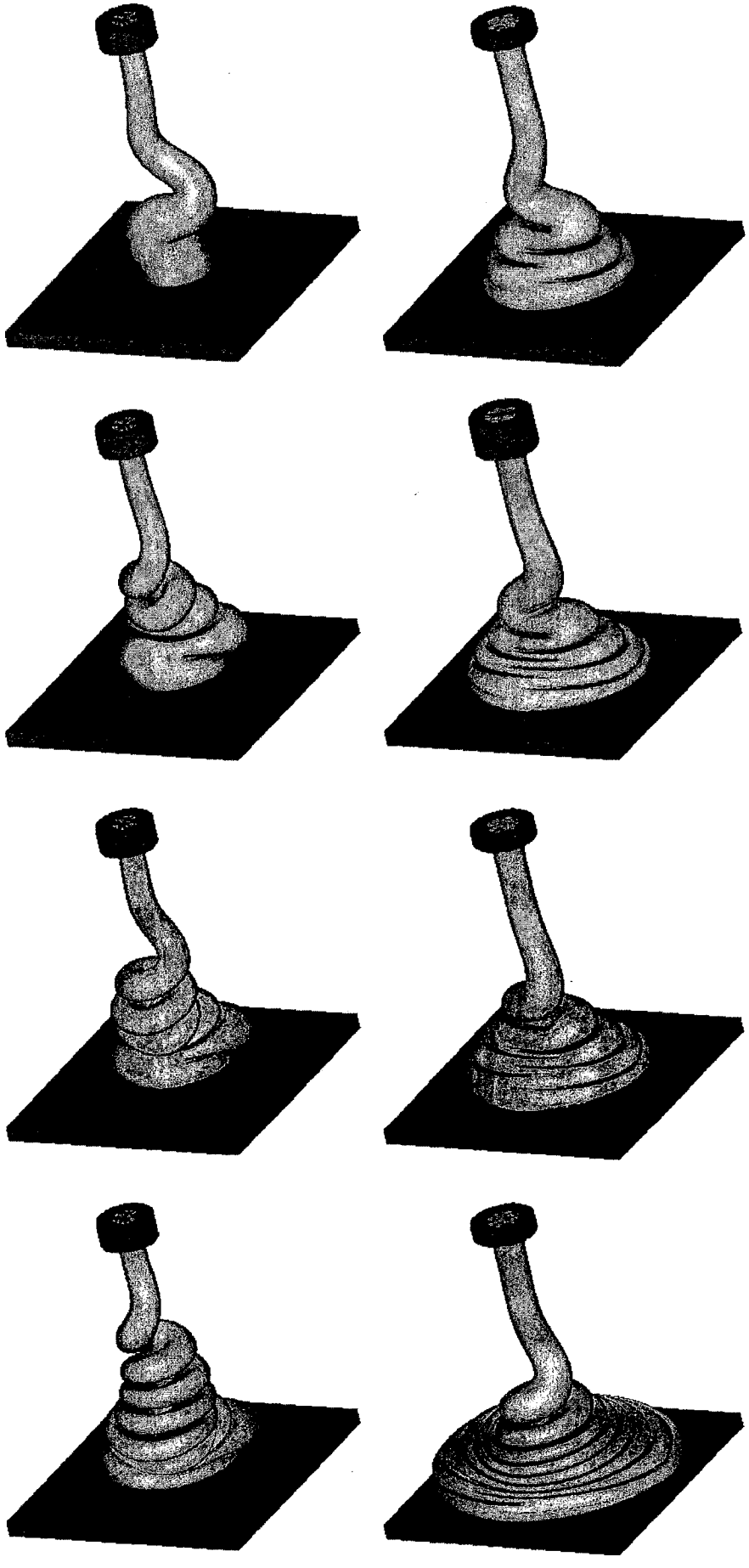

Figure 7. 3D simulation of jet buckling, $Re = 0.25$. Visualization at various times - 3D view.

8.2 Splashing Drop

A spherical drop of fluid of diameter $D = 10\text{mm}$ is given an initial velocity of $U = 1\text{ms}^{-1}$ and released from a height of 4cm above a square container of dimensions 10cm $\times$ 10cm $\times$ 10cm containing a quiescent fluid. The value of the viscosity was $\nu = 10^{-6}\text{m}^2\text{s}^{-1}$ so that the Reynolds number $Re = UD/\nu = 1,000$. The initial indentation may be observed with the formation of a travelling wave. Low pressure beneath this indentation causes fluid to travel upwards (in the form of a splash). As the splash reaches its peak surface waves being reflected from the sides of the container can be observed (see figure 8).

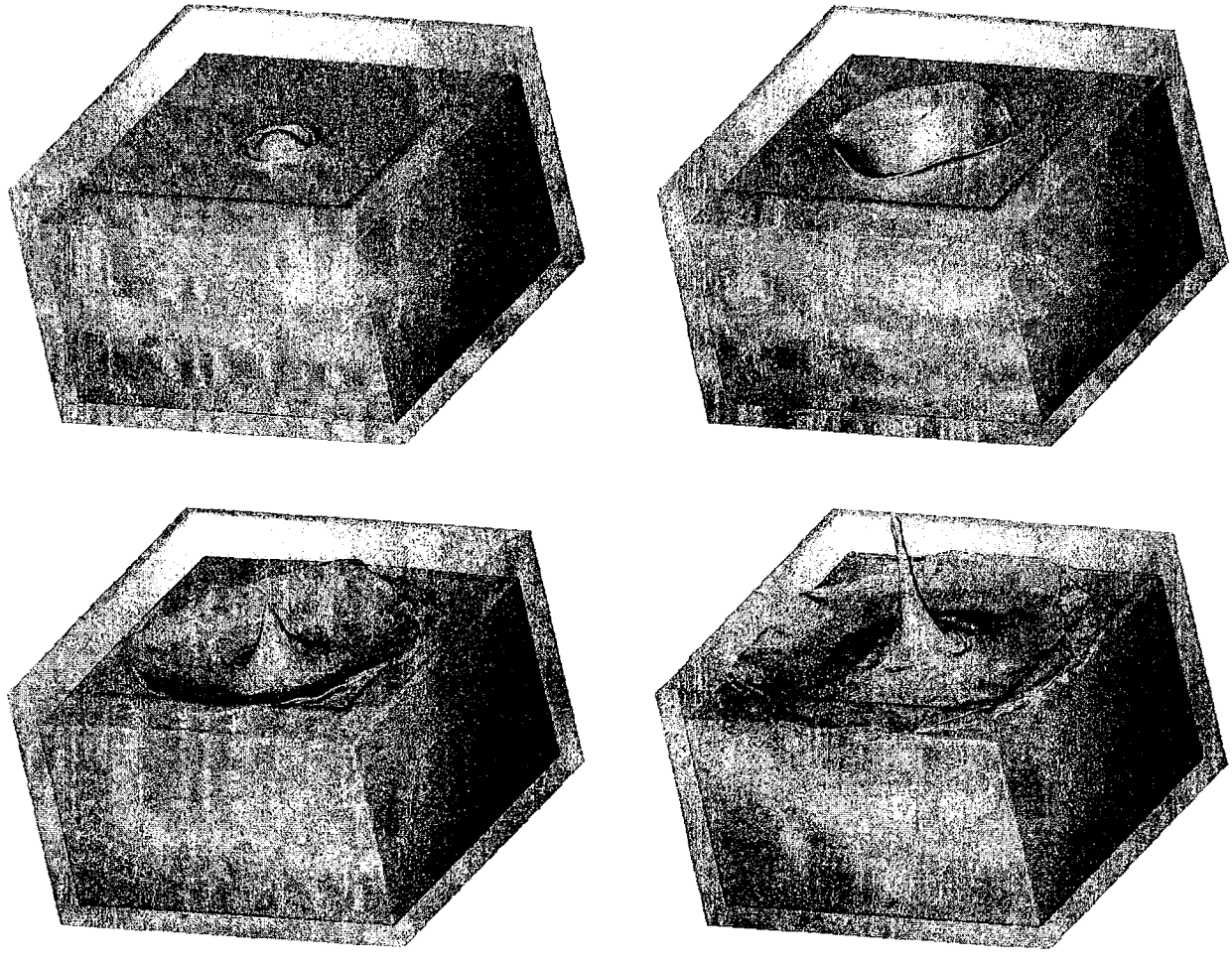

Figure 8. 3D numerical simulation the splashing drop at different times. 3D view.

8.3 Hydraulic Jump

The FreeFlow3D code has been applied to simulate a circular hydraulic jump. We considered a cylindrical jet of a viscous fluid flowing rapidly onto a rigid horizontal surface. The following input data were employed

- Domain dimensions: 10cm × 10cm × 10cm;
- Mesh size: 100 × 100 × 100 cells ($\delta x = \delta y = \delta z = 1$mm);
- Solid surface dimensions: 10cm × 10cm × 1.5mm;
- Inflow dimensions: diameter (D) = 8mm and height (H) = 2cm. The inflow nozzle is situated at a distance of 4.8cm above of the solid surface;
- Inflow velocity: U = 0.5ms^{-1};
- Scaling parameters: U = 0.5ms^{-1}, D = 8mm;
- Gravity was acting in the z-direction with $g = -9.81\text{ms}^{-2}$;
- Kinematic viscosity of liquid: $\nu = 10^{-5}\text{m}^2\text{s}^{-1}$;
- Froude number: $Fr = 1.785$;
- Reynolds number: $Re = 400$;
- Poisson tolerance: EPS = 10^{-8};

The FreeFlow3D code solved this problem with the above input data. Figure 9 exhibits snapshots of this run at selected times.

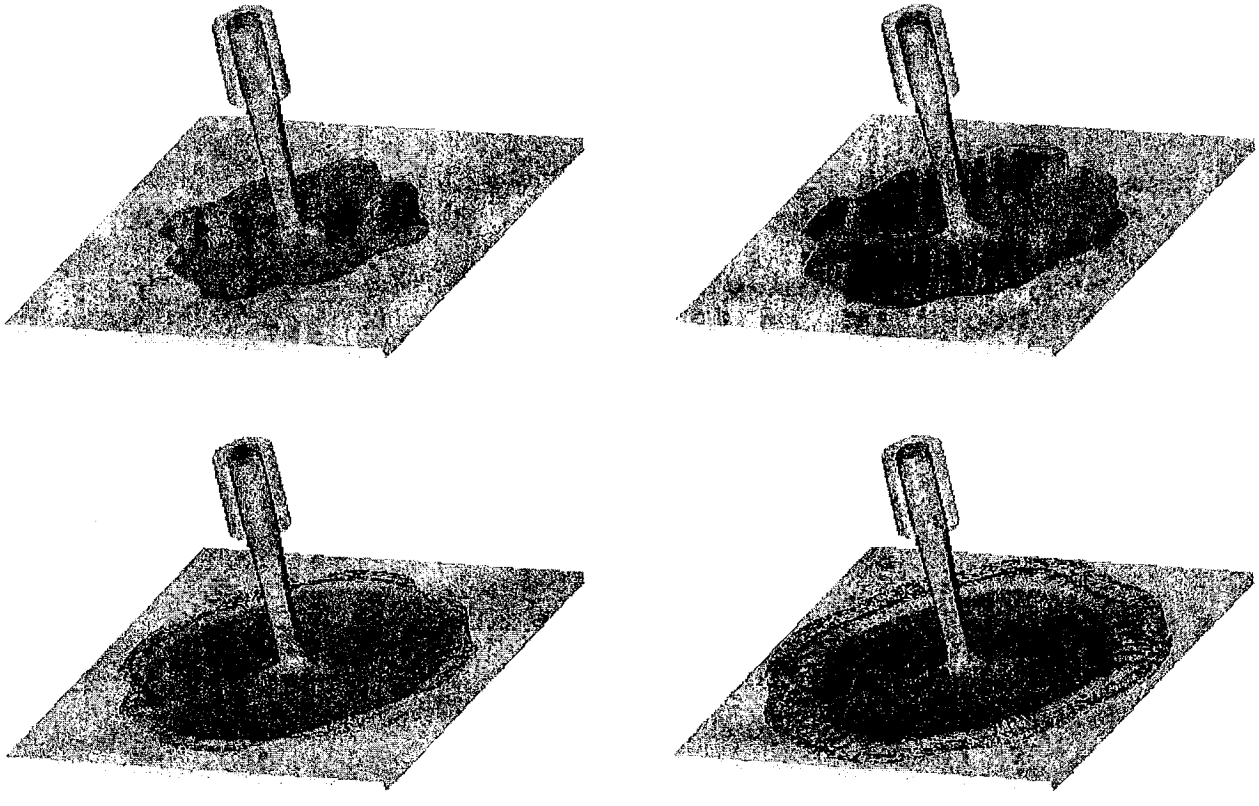

Figure 9. 3D numerical simulation of a circular hydraulic jump at different times, $Re = 400$ and $F_r = 1.785$. 3D view.

9 TURBULENT FREE SURFACE FLOWS

Turbulent free surface flows are of great technological interest and yet in 1987 Ferziger [24] in a fairly substantial review on the calculation of incompressible turbulent flows did not mention this challenging topic. It is interesting to note the changes that have been taken place since then. In 1996 there were no less than three review articles on turbulent free surface flow problems and related topics (Melville [47], Sarpkaya [61] and Tsai and Yue [70]). Other examples of numerical studies of turbulence at a free surface include Borne, Orszag and Staroselsky [8], Pan and Banerjee [55] and Tsai [71]; all three sets of authors employed time-dependent full Navier-Stokes and continuity equations, that is Direct Numerical Simulation (DNS).

One of the main difficulties is what to choose for the boundary conditions at the free surface. This is hardly surprising when one considers the different types of turbulence that can occur (see Brocchini and Peregrini [9], [10]). Recently, Ferreira et al. [23] developed an adaptation of the MAC method to a two-dimensional variant of the $\kappa - \varepsilon$ model. This section will briefly describe this method in which the incorporation of the VONOS scheme [81] into the MAC method (see Subsection 6.8) proved to be essential. Subsection 9.1 describes the particular $\kappa - \epsilon$ model, while Subsection 9.2 simulates a rapid jet entering parallel to and beneath the free surface.

9.1 Fluid Flow Equations

The conservation equations for time-dependent, viscous, isothermal incompressible turbulent Newtonian fluid flow are the time-averaged Navier-Stokes equations, the mass conservation equation, and the κ and ε equations. In conservative form, these fluid flow equations, omitting averaging symbols, can be written respectively, as

$$\frac{\partial u}{\partial t} + \frac{1}{r^\alpha}\frac{\partial (r^\alpha uu)}{\partial r} + \frac{\partial (uv)}{\partial z} = -\frac{\partial p_e}{\partial r} + \frac{1}{Re}\frac{\partial}{\partial z}\left(\frac{\partial u}{\partial z} - \frac{\partial v}{\partial r}\right) + \frac{1}{Fr^2}g_r$$
$$+ \frac{1}{Re}\left[2\frac{1}{r^\alpha}\frac{\partial}{\partial r}\left(r^\alpha \nu_t \frac{\partial u}{\partial r}\right) + \frac{\partial}{\partial z}\left(\nu_t\left(\frac{\partial u}{\partial z} + \frac{\partial v}{\partial r}\right)\right) - 2\alpha\frac{\nu_t u}{r^{2\alpha}}\right] \tag{34}$$

$$\frac{\partial v}{\partial t} + \frac{1}{r^\alpha}\frac{\partial (r^\alpha vu)}{\partial r} + \frac{\partial (vv)}{\partial z} = -\frac{\partial p_e}{\partial z} - \frac{1}{Re}\frac{1}{r^\alpha}\frac{\partial}{\partial r}\left(r^\alpha\left(\frac{\partial u}{\partial z} - \frac{\partial v}{\partial r}\right)\right) + \frac{1}{Fr^2}g_z$$
$$+ \frac{1}{Re}\left[2\frac{\partial}{\partial z}\left(\nu_t\frac{\partial v}{\partial z}\right) + \frac{1}{r^\alpha}\frac{\partial}{\partial r}\left(r^\alpha\nu_t\left(\frac{\partial u}{\partial z} + \frac{\partial v}{\partial r}\right)\right)\right] \tag{35}$$

$$\frac{1}{r^\alpha}\frac{\partial (r^\alpha u)}{\partial r} + \frac{\partial v}{\partial z} = 0 \tag{36}$$

$$\frac{\partial \kappa}{\partial t} + \frac{1}{r^\alpha}\frac{\partial (r^\alpha u\kappa)}{\partial r} + \frac{\partial (v\kappa)}{\partial z} = \frac{1}{Re}\left[\frac{1}{r^\alpha}\frac{\partial}{\partial r}\left(r^\alpha(1+\nu_t/\sigma_\kappa)\frac{\partial \kappa}{\partial r}\right) + \frac{\partial}{\partial z}\left((1+\nu_t/\sigma_\kappa)\frac{\partial \kappa}{\partial z}\right)\right] + P - \varepsilon \tag{37}$$

$$\frac{\partial \varepsilon}{\partial t} + \frac{1}{r^\alpha}\frac{\partial (r^\alpha u\varepsilon)}{\partial r} + \frac{\partial (v\varepsilon)}{\partial z} = \frac{1}{Re}\left[\frac{1}{r^\alpha}\frac{\partial}{\partial r}\left(r^\alpha(1+\nu_t/\sigma_\varepsilon)\frac{\partial \varepsilon}{\partial r}\right) + \frac{\partial}{\partial z}\left((1+\nu_t/\sigma_\varepsilon)\frac{\partial \varepsilon}{\partial z}\right)\right]$$
$$+ \quad (C_{1\varepsilon}P - C_{2\varepsilon}\varepsilon)/T_t + \beta E \tag{38}$$

In the above equations, $u = u(r,z,t)$ and $v = v(r,z,t)$ are, respectively, the components in the r and z directions of the local time-averaged velocity vector field $\mathbf{u} = \mathbf{u}(r,z,t)$, $\kappa = \kappa(r,z,t)$ is the local time-averaged turbulent kinetic energy of the fluctuating motion, $\varepsilon = \varepsilon(r,z,t)$ is the dissipation rate of κ, $p_e = p + \frac{2}{3}\frac{1}{Re}\kappa$ is the effective scalar pressure field divided by the density, and $\mathbf{g} = (g_r, g_z)$ is the gravitational acceleration.

The isotropic eddy viscosity ν_t, the turbulent shear stress production P, the turbulence time scale T_t, and the gradient dissipation E are, respectively, defined as

$$\nu_t = \kappa C_\mu f_\mu T_t \tag{39}$$

$$P = \nu_t \left(2\left(\frac{\partial u}{\partial r}\right)^2 + 2\left(\frac{\partial v}{\partial z}\right)^2 + 2\alpha\left(\frac{u}{r^\alpha}\right)^2 + \left(\frac{\partial v}{\partial r} + \frac{\partial u}{\partial z}\right)^2 \right) \tag{40}$$

$$T_t = (1-\beta) Min\left\{ \frac{\kappa}{\varepsilon}, \frac{2}{3C_\mu}\sqrt{\frac{3}{8|\mathbf{S}|^2}} \right\} + \beta\left\{ \frac{\kappa}{\varepsilon} + \left(\frac{1}{\varepsilon}\right)^{1/2} \right\} \tag{41}$$

$$E = \frac{2\nu_t}{Re}\left(\left(\frac{\partial^2 v}{\partial r^2}\right)^2 + \left(\frac{\partial^2 u}{\partial z^2}\right)^2 \right) \tag{42}$$

where, in (41), $|\mathbf{S}|^2 = \mathbf{D}:\mathbf{D}$ [20], with $\mathbf{D} = \nabla\mathbf{u} + (\nabla\mathbf{u})^{\mathrm{T}}$ rate of deformation tensor. The model constants C_μ, $C_{1\varepsilon}$, $C_{2\varepsilon}$, σ_κ and σ_ε, and the parameter β in (38) and (41) are used to specify the two-equation $\kappa - \varepsilon$ turbulence models considered here. When $C_\mu = 0.09$, $C_{1\varepsilon} = 1.44$, $C_{2\varepsilon} = 1.92$, $\sigma_\kappa = 1.0$, $\sigma_\varepsilon = 1.3$ and $\beta = 0$, we are dealing with the standard high-Reynolds number form of the $\kappa - \varepsilon$ model [39] (which we shall denote by HRe $\kappa - \varepsilon$ model), with a time scale proposed by Durbin [20] to take account of the stagnation-point anomaly. The damping function f_μ in (39) assumes the value $f_\mu = 1$ in the case of the HRe $\kappa - \varepsilon$ model, and takes the following expression in the case of the LRe $\kappa - \varepsilon$ model

$$f_\mu = \left(1 - exp(-a_1 Re_{z_w} - a_3 Re_{z_w}^3 - a_5 Re_{z_w}^5)\right)^{1/2} \tag{43}$$

where a_1, a_3 and a_5 are constants given by $a_1 = 1.5 \times 10^{-4}$, $a_3 = 5.0 \times 10^{-7}$, $a_5 = 1.0 \times 10^{-10}$ [83], and Re_{z_w} is the local Reynolds number defined as $Re_{z_w} = z_w Re \kappa^{1/2}$, z_w being the normal distance from the nearest rigid-boundary to a point in the flow. The parameter α in (34) through (40) is used to specify the coordinate system, namely: when $\alpha = 0$, plane Cartesian coordinates are considered (r is to be interpreted as x and z as y); and when $\alpha = 1$, cylindrical polar coodinates are assumed.

9.2 Rapid Jet Parallel to the Free Surface

The GENSMAC code embodied with the $\kappa - \varepsilon$ model was applied to simulate free surface turbulent flows using the HRe $\kappa-\varepsilon$ model. The problem consists of a horizontal jet penetrating a quiescent fluid from an entry port at a depth, denoted by H, beneath the free surface. The geometrical configuration together with the parameters employed for this free surface fluid flow are shown in figure 10. In this computation, the associated Reynolds and Froude numbers are $Re = DU_0/\nu = 5.0 \times 10^4$ and $Fr = U_0/\sqrt{gD} \approx 1.8$, respectively. The value of the H was set to 0.5. The results of this simulation are displayed in figure2 11 and 12.

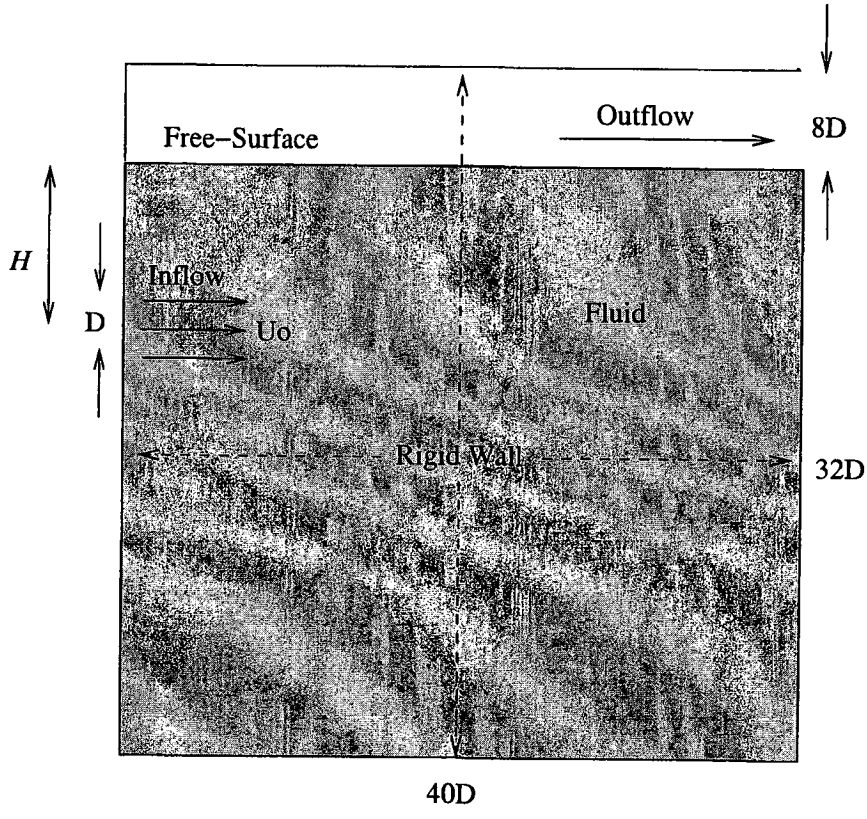

Figure 10. Geometry and parameters for the free surface fluid flow of a penetrating planar jet in a pool: $U_0 = 2.0$ ms^{-1}; $D = 0.25$ m; $H = 0.5$ m.

As we can see from figure 11 and 12, this simulation corresponds to a shallow jet that has a strong interaction with the free surface. From the beginning, the vortices interact with the free surface causing the formation of undulations on the free surface. As time progresses, the undulations become more intense until, at later times, the strengths of the vorticities decrease dramatically away from the injection point causing a reduction in the surface roughness.

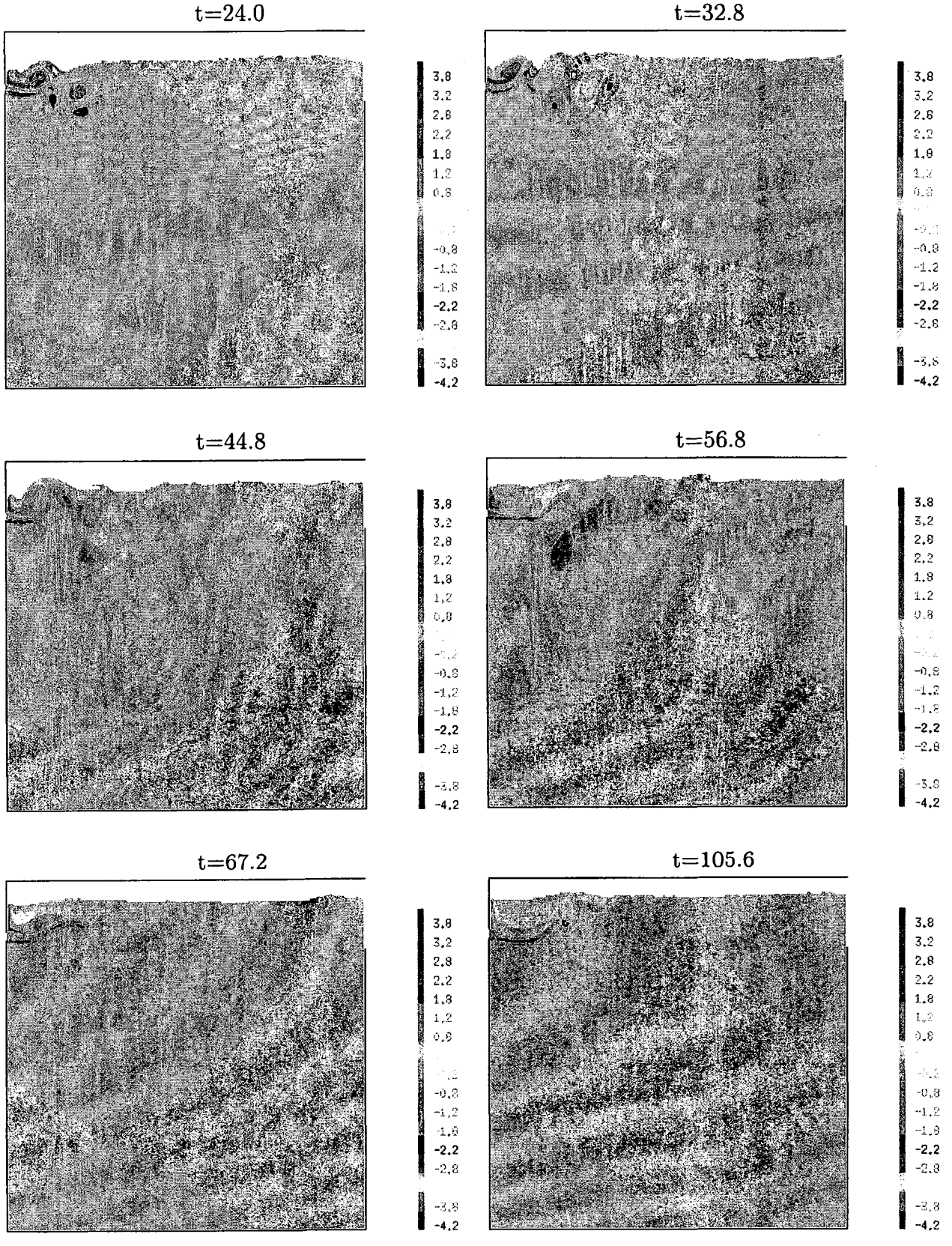

Figure 11. Non-dimensional time evolution of the non-dimensional vorticity contours of a two-dimensional numerical simulation, using the HRe $\kappa - \varepsilon$ model. $H = 0.5$ m.

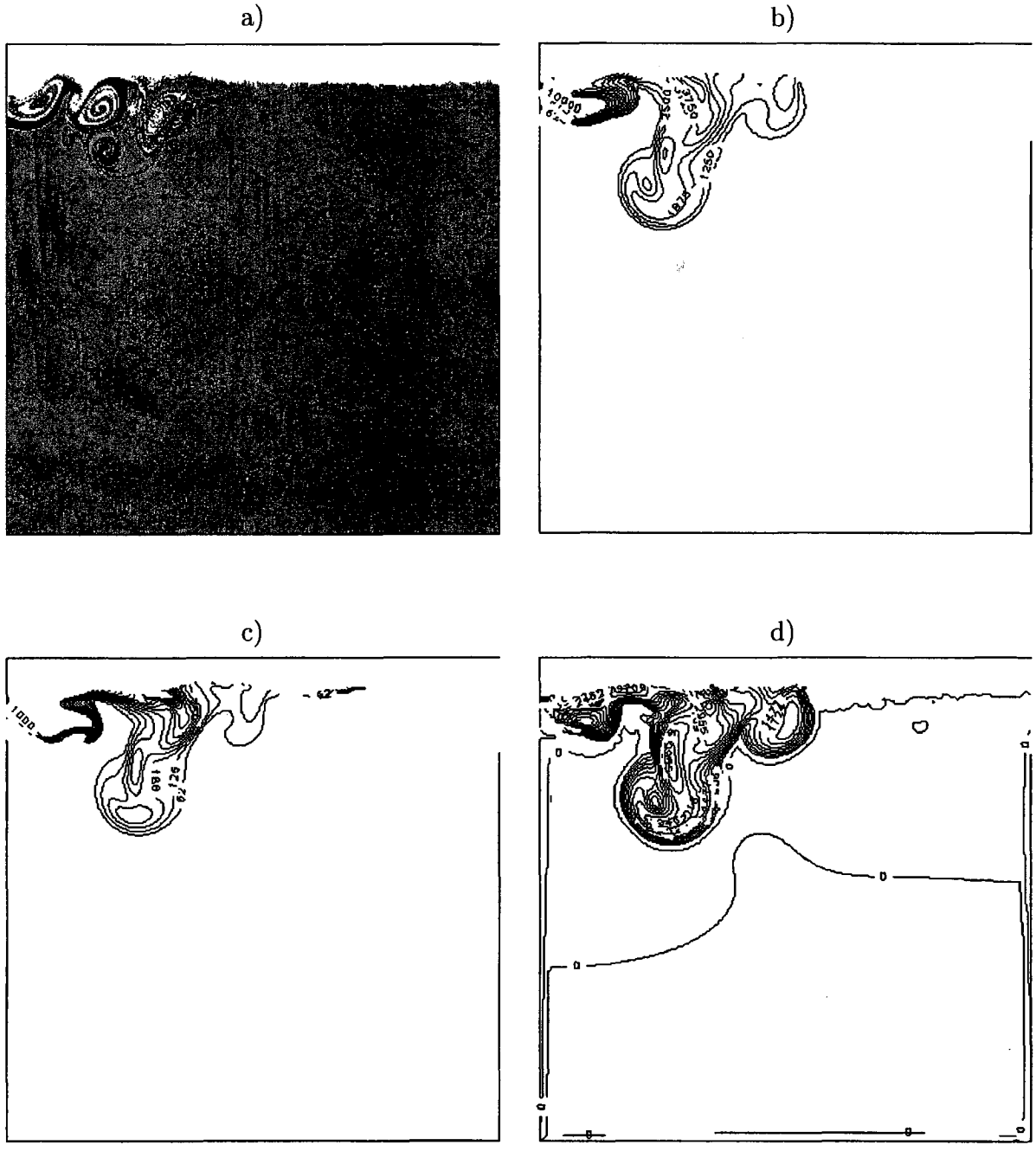

Figure 12. Jet in a pool at time $t = 33$ (case $H = 0.5$ m) showing: a) fluid flow visualization, b) kinetic energy contours, c) turbulence dissipation rate contours, and d) eddy viscosity contours.

10 VISCOELASTIC FLOWS

Viscoelastic free surface flow problems have been studied for 30 years: an early attempt at simulating extrudate swell of a Maxwell fluid was made by Tanner [68] in 1970 (see also Ryan and Dutta [59]). Crochet and Keunings [15] solved an Oldroyd-B fluid for both a circular and slit die-swell. Bousfield, Keunings and co-workers (see [6], [7], [36], [37]) wrote a number of papers dealing with free surface viscoelastic flows in the late 1980s. Kolte et al. [38] used a Lagrangian approach to simulate the transient filament stretching rheometer, while Yao and McKinley [84] treated transient extensional deformation. More recently Cormenzana et al. [13] extended their code, CONNFESSITT, to free surface flows and effected a comparison with POLYFLOW.

10.1 Oldroyd-B Model

The basic equations governing the flow of an Oldroyd-B fluid can be written as

$$\boldsymbol{\tau} + We\, \overset{\nabla}{\boldsymbol{\tau}} = \left(1 - \frac{\lambda_2}{\lambda_1}\right) \mathbf{D} \tag{44}$$

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla.(\mathbf{uu}) = -\nabla p + \frac{2}{Re}\left[\left(\frac{\lambda_2}{\lambda_1}\right) \nabla^2 \mathbf{u} + \nabla \cdot \boldsymbol{\tau}\right] + \frac{1}{Fr^2}\mathbf{g} \tag{45}$$

$$\nabla.\mathbf{u} = 0 \tag{46}$$

respectively, where $We = \lambda_1(U/L)$ is the Weissenberg number and $\mathbf{D}$ is the rate of deformation tensor (see Subsection 9.1). The upper convected derivative, $\overset{\nabla}{\boldsymbol{\tau}}$, is given by

$$\overset{\nabla}{\boldsymbol{\tau}} = \frac{\partial \boldsymbol{\tau}}{\partial t} + \mathbf{u} \cdot \nabla \boldsymbol{\tau} - (\nabla \mathbf{u})^T \cdot \boldsymbol{\tau} - \boldsymbol{\tau} \cdot (\nabla \mathbf{u}).$$

One of the major difficulties in attempting to solve the above set of differential equations is determining the boundary values for the non-Newtonian part of the extra-stress tensor $\boldsymbol{\tau}$. Details will not be given here, but may be found in Tomé et al. [79].

The non-Newtonian contribution to the extra-stress tensor $\boldsymbol{\tau}$ is placed at the centre of a cell and the SMAC algorithm is now the same solution sequence as before augmented with another step wherein $\boldsymbol{\tau}$ is explicity computed by an Euler-type method exactly analogous to the preliminary velocity update (see Tomé et al. [79]).

10.2 Elastic Jet

As an illustration of the power and flexibility of the SMAC approach when applied to viscoelastic fluids, two calculations are presented that simulate the buckling of a thin viscoelastic jet hitting a rigid plane; these are compared with the equivalent Newtonian flow. The latter has been investigated by a number of researchers eg. [16], [17], [75].

To illustrate that a viscoelastic fluid modelled by Oldroyd-B has a considerable effect on the buckling phenomenon, two calculations are presented that contrast Newtonian and non-Newtonian behaviour; in both cases, the conditions put forward by Cruickshank and Munson [16] are satisfied, predicting buckling of a Newtonian jet. The calculations concern a thin jet injected into a rectangular cavity of size 5cm, with an inlet slot size of $D = 5\text{mm}$ (so that $H/D = 20$). A mesh size of $\delta x = \delta y = 0.0005\text{mm}$ was employed (giving 100×200 computational cells). At the inlet, we imposed the uniform input velocity $U = 0.5\text{ms}^{-1}$ and the components of the non-Newtonian extra-stress tensor were set to zero (see [79]). Gravity is considered to be acting in the negative y-direction with $g = 9.81\text{ms}^{-2}$. The fluid parameters were chosen to be

$$\nu = 0.01\text{m}^2\text{s}^{-1} , \qquad \lambda_1 = 0.01\text{s} , \qquad \lambda_2 = 0.001\text{s} .$$

The scaling parameters were U, D, ν, giving $We = 1$ ($We_{\textbf{effect}} = 0.9$), $Re = 0.25$ and $Fr = 2.258$. We point out that the only data differing between these two calculations is the ratio λ_2/λ_1, which took the value of 0.1 in the non-Newtonian calculation and was set equal to 1 producing $We_{\textbf{effect}} = 0$ for the Newtonian computation. Figure 13 displays a comparison of the Newtonian and viscoelastic fluid configurations at different times. The difference between the two flows as displayed in figure 13 is dramatic. As the Newtonian jet hits the plate it becomes thicker, while the non-Newtonian jet produces a wave that travels upwards along the jet creating an instability which causes the jet to buckle much earlier than the Newtonian jet. We observe that at time $t = 30.0$ the non-Newtonian viscoelastic jet thins to almost a filament, then accelerates the fluid behind the filament at time $t = 36.25$, and thereafter behaves more like an "ordinary" Newtonian jet. As this is occurring the Newtonian fluid jet continues to thicken at the base with only a hint of initial buckling at time $t = 30.0$. At the time the Newtonian jet starts to buckle the non-Newtonian jet has already produced many folds.

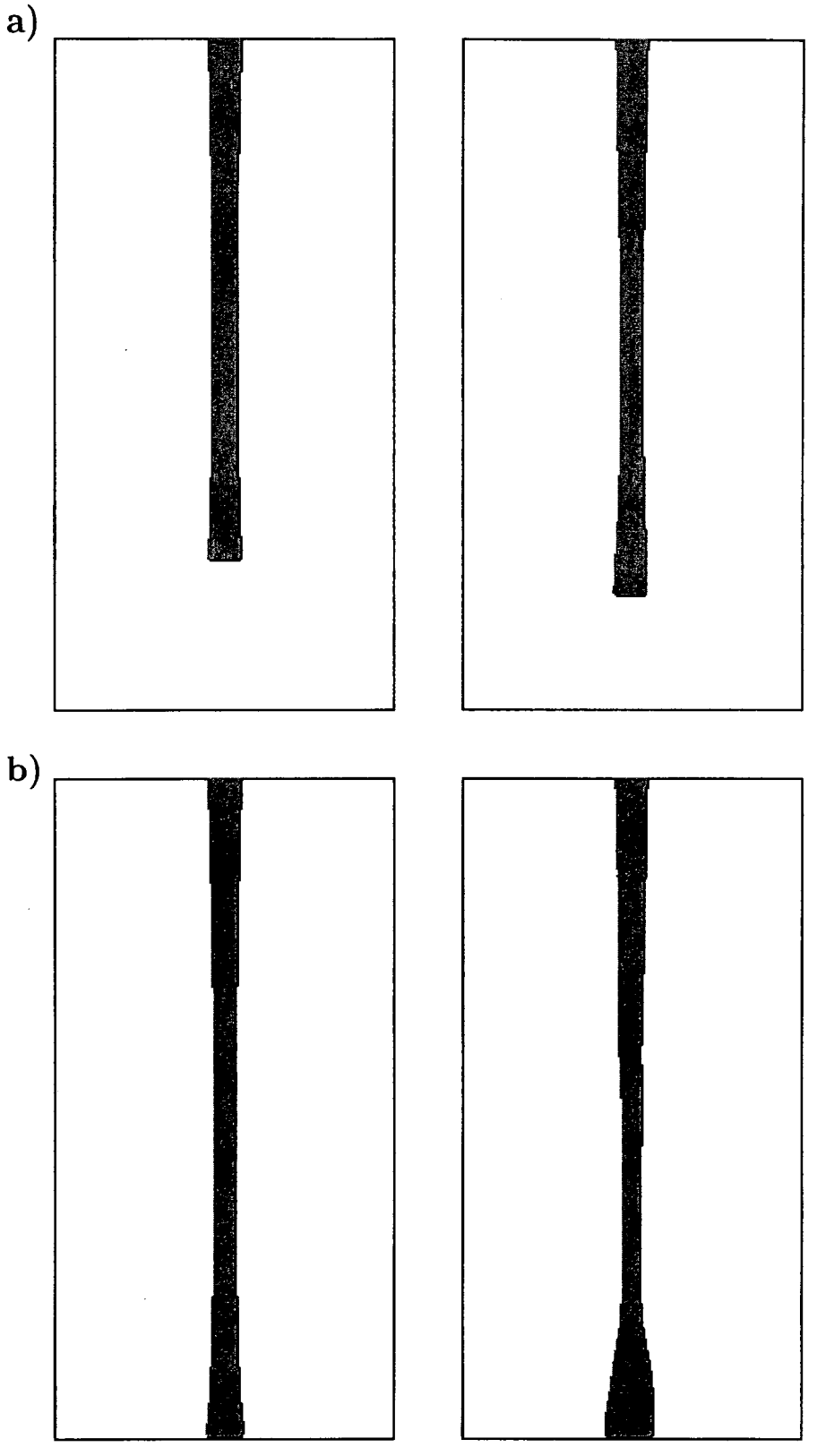

Figure 13. Numerical simulation of viscoelastic jet buckling. Fluid flow visualization at different times: Newtonian jet (on the left) and non-Newtonian (on the right). Times shown are ($\frac{U}{D}t$): **a)** 12.5, **b)** 15.0, **c)** 17.5, **d)** 23.75, **e)** 27.5, **f)** 30.0, **g)** 36.25, **h)** 42.5, **i)** 48.75, **j)** 55.0.

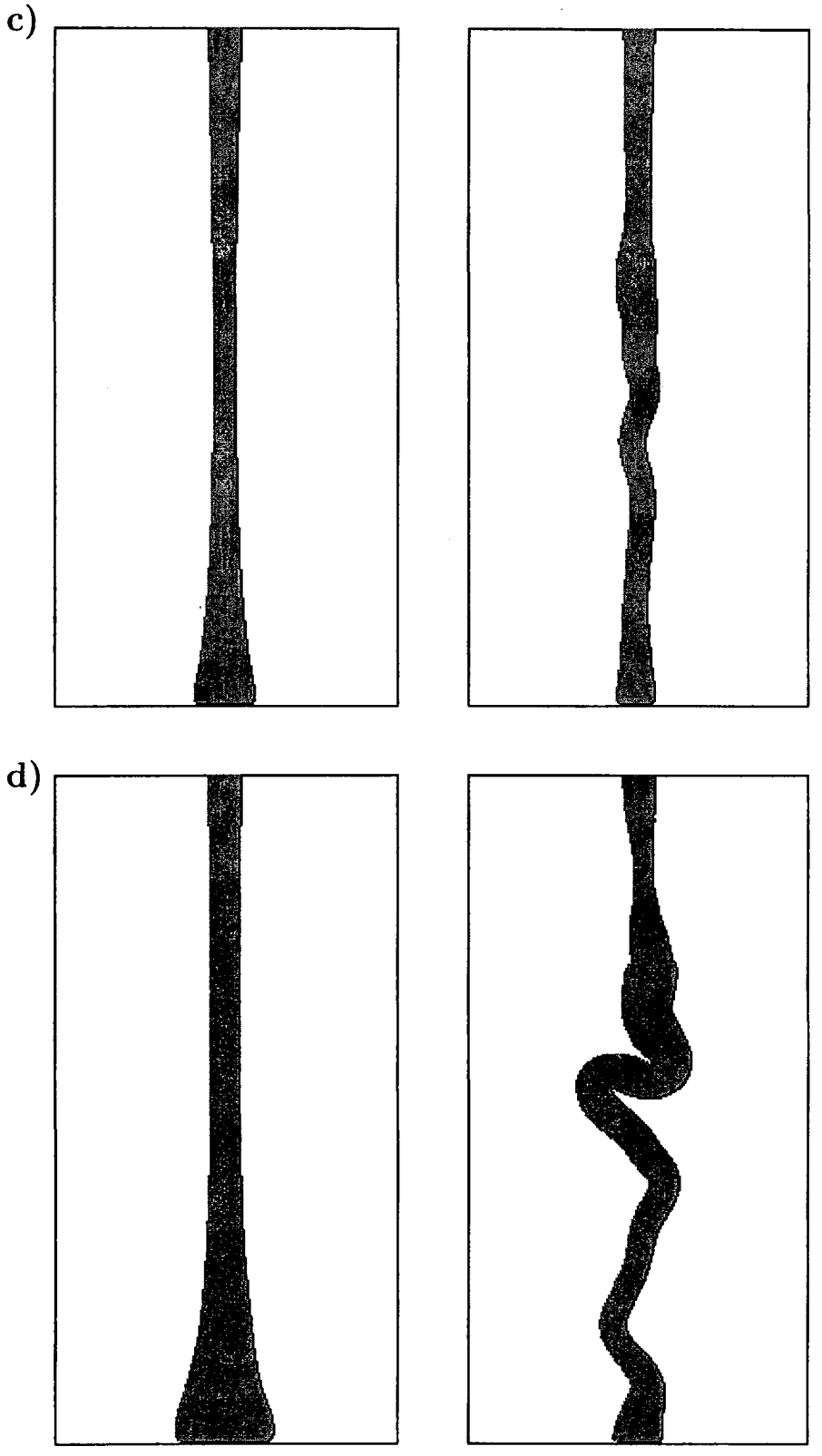

Figure 13. Continued.

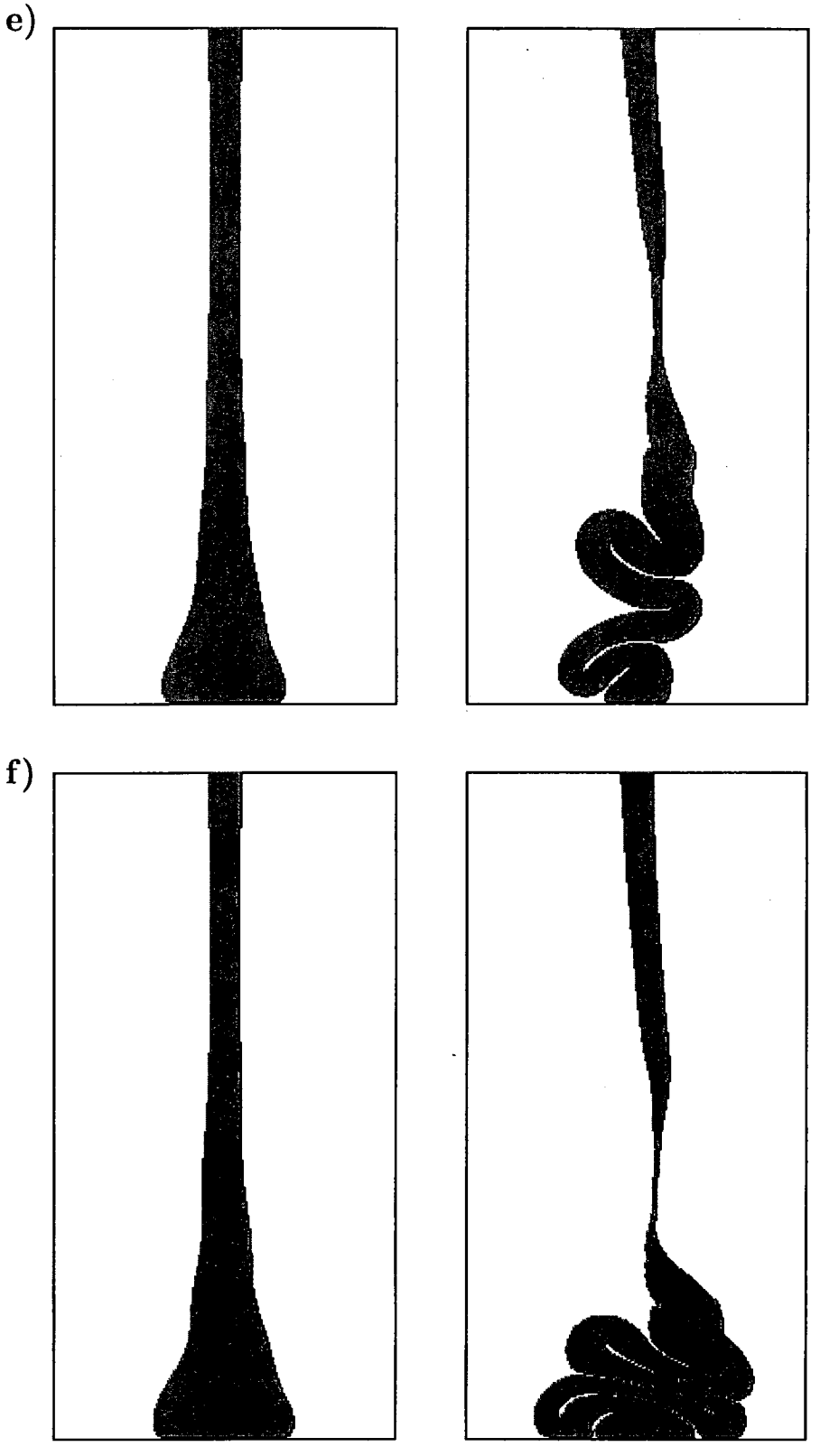

Figure 13. Continued.

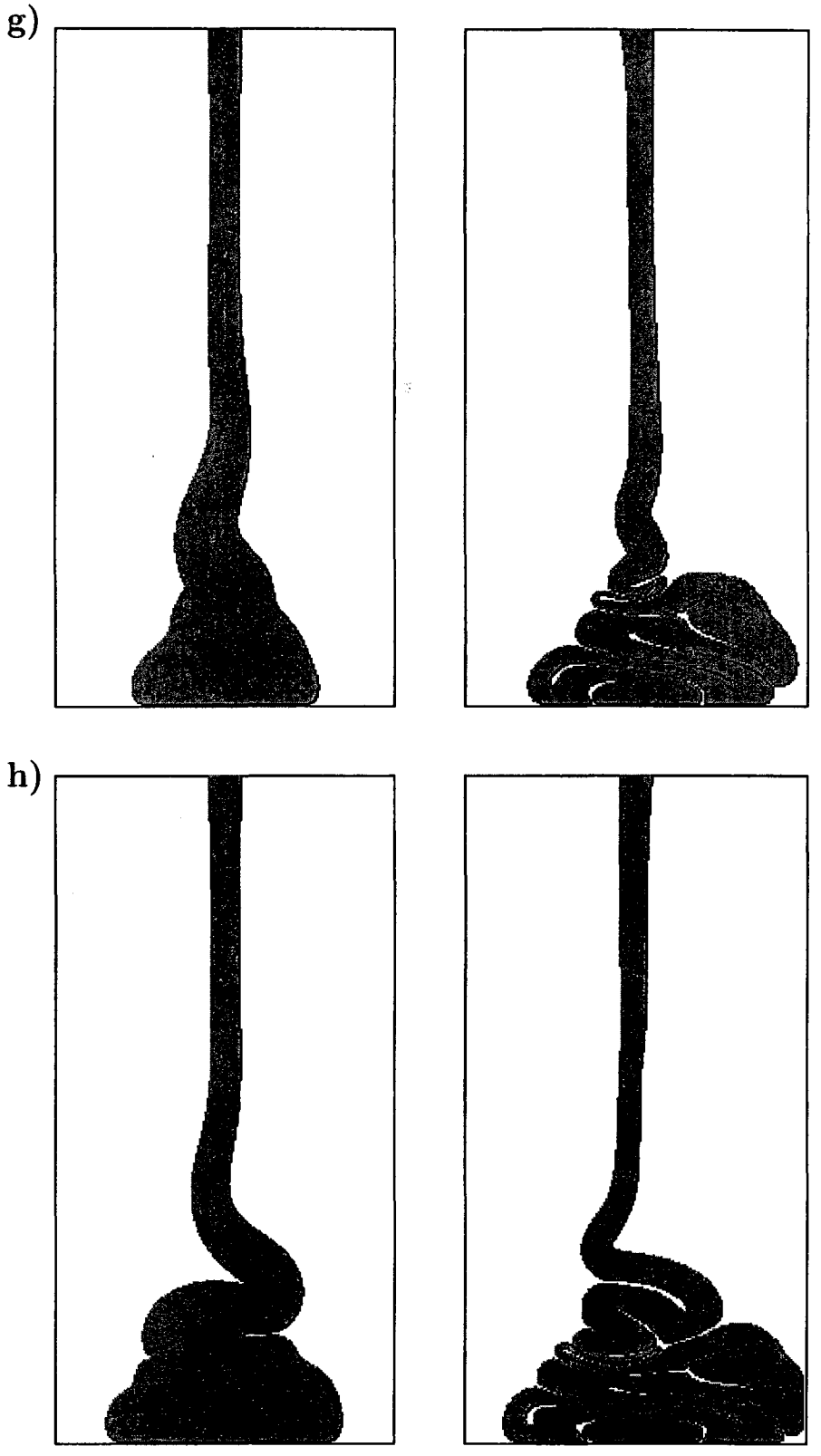

Figure 13. Continued.

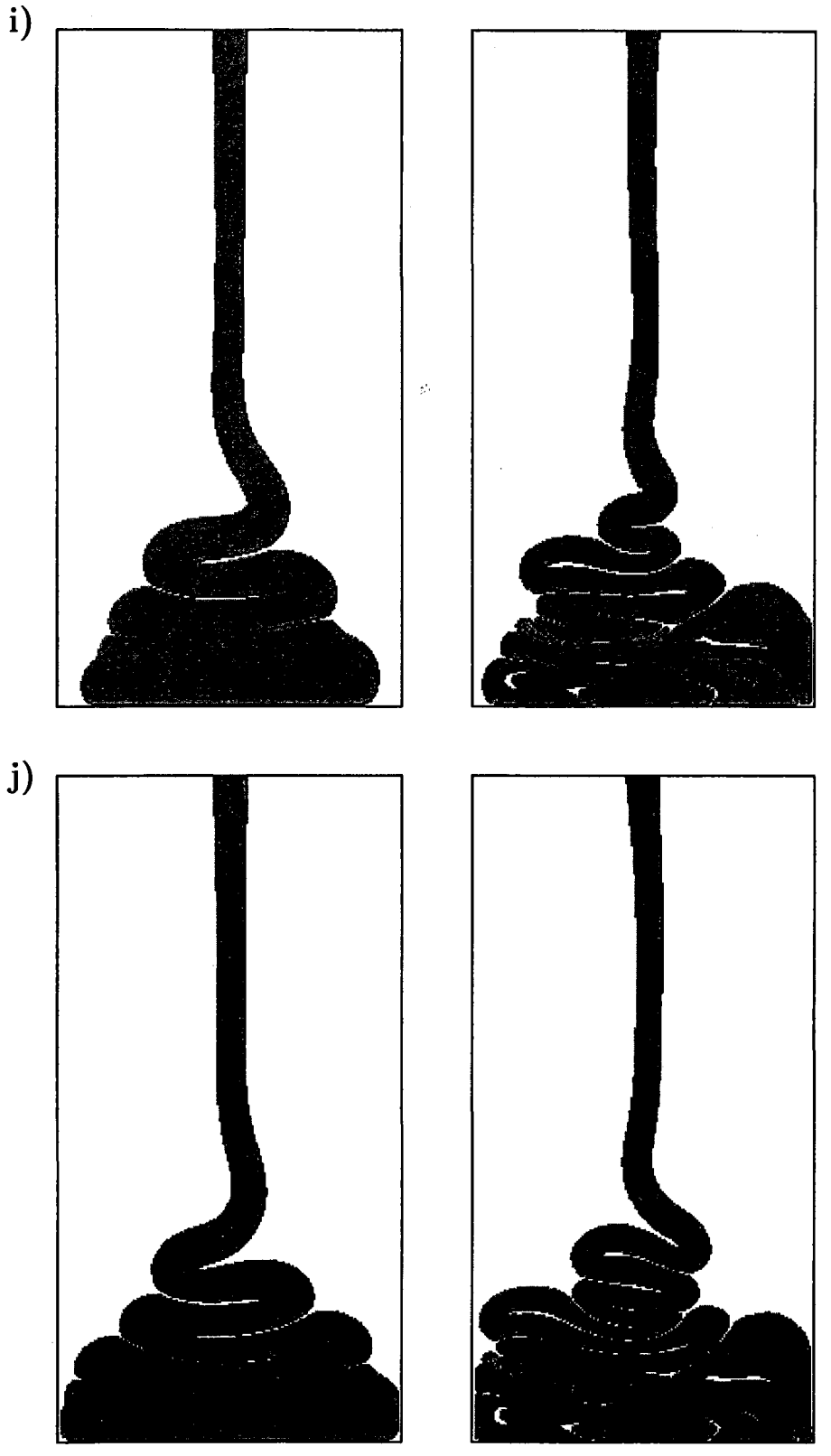

Figure 13. Continued.

11 THE FUTURE OF THE MAC METHOD

When the MAC method was introduced in 1965 it was severely limited by the computational power available at that time. Consequently, it was restricted to small two-dimensional problems. Even in 1972 when Hirt and Cook [32] introduced a three-dimensional version, the method was still extremely limited. Today this "toy" method of the 1960s is already becoming a useful scientific and technological tool, able to compute quite large three-dimensional problems (with grids of size of $1,000,000$ computational cells) with a modest computer and display the results as a realistic video sequence. This leads one to speculate what we might expect over the next forty years.

In the near future the proper implementation of surface tension is required, problems with dynamic contact angles are important and ubiquitous and multiphase flow problems particularly involving a large number of bubbles (foaming flows) are of interest. Progress in these areas is already underway (see [43, 60, 65]). As we have seen from section 10, the MAC method is particularly effective at solving viscoelastic fluids. We might therefore anticipate major advances over the next ten years in computational rheology. Oldroyd-B is only one such model. Implementation of the Maxwell model, second order fluids and the Phan-Thien Tanner (PTT) model (see Owens and Phillips [54]) are required together with a proper appreciation of which are "best" for what problems. Three-dimensional implementation will be the norm.

One of the aspects of the MAC method is its simplicity: the grid is fixed (Eulerian) - it does not suffer from the complexity of adaptivity; finite difference methods are used and these, for the most part, are first order; robustness is achieved through the use of a small, often very small (in the case of low Reynolds number flows) time step; the conjugate gradient solver does not have, nor does it need to have, preconditioning. Yet to achieve higher Reynolds number flows, it was found to be imperative to include high order monotone upwinding in the convective terms. To obtain improved efficiency and accuracy, greater computing power in itself may not be sufficient: it will probably be necessary to sacrifice the essential simplicity of the MAC approach and introduce adaptivity and higher order methods or other numerical methods such as finite elements or spectral collocation. The ultimate goal is virtual reality Computational Fluid Dynamics (CFD) whereby the output is in the form of a dynamic hologram in real time and is essentially indistinguishable from the "real thing". Applications include ink jet printing, injection moulding, coating processes, gas-oil centrifugal separators, foaming flows and general filling processes. Other applications include fast ships, Tsumi waves, lava flows and the design of harbour complexes. With regard to the first, we are thinking of large hydrofoils that can travel at up to 200 knots. The trick is to design the shape of the hydrofoil to minimize, or even eliminate, the bow wave which is the greatest source of drag. Even today the MAC method is, in principle, capable of simulating these flows but in 20 to 40 years time, when 10^9 cells can be employed with ease, it will become routine.

Indeed, the next forty years will be exciting times not just for free surface flow CFD codes, but for fluid codes generally. The greatest prize scientifically will be the undertanding of turbulence. It is the authors' belief that such codes will be the catalyst for a fundamental understanding of this phenomenon: it is likely that there are many mechanisms at work depending on the particular flow regime – for example, periodic doubling vortex cascades, cavitation and air

entrainment. All these routes to turbulence will be understood ultimately in terms of higher dimensional nonlinear dynamics, but the insight will arise from numerical experimentation with this powerful scientific tool.

If the MAC method is to survive against its many competitors, then it must prove that it can solve difficult problems more succesfully. There is a great need for a set of test problems. To initiate this discussion the authors would like to conclude by proposing 6 test problems that all good free surface fluid codes should be able to solve:

1. 3D jet buckling problem (Newtonian and viscoelastic).
2. 3D splashing drop (Newtonian and viscoelastic): tungsten ball dropped from different heights – various configurations with satellite drops should be achievable.
3. 3D extrudate-swell problem (Newtonian and viscoelastic): results of swelling should agree with experimental results (see eg. [42]).
4. 3D bursting bubble: a bubble bursting at the surface of a fluid gives rise to a jet or splash from its base.
5. A large number of bubbles (> 100) of different sizes, densities and viscosities rising (and interacting) in a fluid.
6. Rod climbing of a viscoelastic fluid at moderate Weissenberg number.

Currently the MAC method cannot solve all these test problems; however, they should all be achievable in the relatively near future.

ACKNOWLEDGEMENTS

We acknowledgment support from the *FAPESP (Fundação de Amparo à Pesquisa do Estado de São Paulo)*, Contract No. 00/03385-0.

References

[1] Amsden, A. and Harlow, F. (1970), *The SMAC method: a numerical technique for calculating incompressible fluid flows*, Technical Report LA-4370 (Los Alamos National Laboratory).

[2] Armenio, V. (1997), "An improved MAC method (SIMAC) for unsteady high-Reynolds free surface flows", *Int. J. Numer. Meth. Fluids* **24**, 185-214.

[3] Baker, G.R. and Moore, D.W. (1989), "The rise and distorsion of a two-dimensional gas bubble in an inviscid liquid", *Phys. Fluids A*, **9**, 1451-1459.

[4] Batchelor, G.K. (1967), *An Introduction to Fluid Dynamics*, Cambridge University Press, Cambridge.

[5] Boulton-Stone, J.M. and Blake, J.R. (1993), "Gas bubbles bursting at a free surface", *J. Fluid Mech.*, **254**, 437-466.

[6] Bousfield, D.W., Keunings, R. and Denn, M.M. (1986), "Nonlinear analysis of the surface tension driven breakup of viscoelastic filaments", *J. Non-Newtonian Fluid Mech.*, **21**, 79-97.

[7] Bousfield, D.W., Keunings, R. and Denn, M.M. (1988), "Transient deformation of an inviscid inclusion in a viscoelastic extensional flow", *J. Non-Newtonian Fluid Mech.*, **27**, 205-221.

[8] Borue, V., Orszag, S.A. and Staroselsky, I. (1995), "Interaction of surface waves with turbulence: direct numerical simulations of turbulent open-channel flow", *J. Fluid Mech.*, **286**, 1-23.

[9] Brocchini, M. and Peregrine, D.H. (2001), "The dynamics of strong turbulence at free surfaces. Part I. Description", *J. Fluid Mech.*, **449**, 225-254.

[10] Brocchini, M. and Peregrine, D.H. (2001), "The dynamics of strong turbulence at free surfaces. Part 2. Free surface boundary conditions", *J. Fluid Mech.*, **449**, 255-290.

[11] Castelo, A., Tomé, M.F., Cesar, C.N.L., McKee, S. and Cuminato, J.A. (2000), "Freeflow-3D: and integrated simulation system for three-dimensional free surface flows", *J. Comput. Visual. Science*, **2**, 199-210.

[12] Chenn, I.-L., Glimm, J., McBryan, O., Plohr, B. and Yaniv, S. (1986), "Front tracking for gas dynamics, *J. Comput. Phys.* **62**, 83-110.

[13] Cormenzana, J., Ledda, A., Laso, M. and Debbaut, B. (2001), "Calculation of free surface flows using CONNFESSITT", *J. Rheol.*, **45**, 237-258.

[14] Craik, A.D., Latham, R.C., Fawkes, M.J. and Gribbon, W.F. (1981), "The circular hydraulic jump", *J. Fluid Mech.*, **112**, 347-362.

[15] Crochet, M.J. and Keunings, R. (1982), "Finite element analysis of die-swell of a highly elastic fluid", *J. Non-Newton. Fluid Mech.*, **10**, 339-356.

[16] Cruickshank, J.O. and Munson, B.R. (1981), "Viscous-fluid buckling of plane and axisymmetric jets", *J. Fluid Mech.*, **113**, 221-239.

[17] Cruickshank, J.O. (1988), "Low-Reynolds-number instabilities in stagnating jet flows, *J. Fluid Mech.* **193**, 111-127.

[18] Darip, P., Glimm, J., Lindquist, B., Maesumi, M. and McBryan, O. (1988), *On the Simulation of Heterogeneous Petroleum Reservoirs*, in Numerical Simulation in Oil recovery, edited by M. Wheeler, Springer Verlag, New York.

[19] Deville, M.O. (1974), "Numerical experiments on the MAC code for slow flow", *J. Comput. Phys.*, **15**, 362-374.

[20] Durbin, P.A. (1996), "On the $\kappa - \varepsilon$ stagnation point anomaly", *J. Heat and Fluid Flow*, **17**, 89-90.

[21] Fauci, L.J. and Peskin, C.S. (1988), "A computational model of aquatic animal locomotion", *J. Comput. Phys.*, **77**, 85-108.

[22] Ferreira, V.G., Tomé, M.F., Mangiavacchi, N., Castelo, A., Cuminato, J.A., Fortuna, A.O. and McKee, S. (2002), "High order upwinding and the hydraulic jump", *Int. J. Numer. Meth. Fluids*, **39**, 549-583.

[23] Ferreira, V.G., Mangiavacchi, N., Tomé, M.F., Castelo, A., Cuminato, J.A. and McKee, S. (2003), "Numerical simulation of turbulent free surface flow with two-equation k-ε eddy-viscosity models", *Int. J. Numer. Meth. Fluids* (to appear).

[24] Ferziger, J.H. (1987) "Simulation of incompressible turbulent flows", *J. Comput. Phys.*, **69**, 1-48.

[25] Fogelson, A.L. and Peskin, C.S. (1988), "A fast numerical method for solving the three-dimensional Stokes equations in the presence of gas suspended particles", *J. Comput. Phys.* **79**, 50-69.

[26] Glimm, J., Grove, J., Lindquist, B., McBryan, O. and Tryggvason, G. (1988), "The bifurcation of tracked scalar waves", *SIAM J. Sci. Stat. Comput.*, **9**, 61-79.

[27] Golafshani, M. (1988), "A simple numerical technique for transient creep flows with free surfaces", *Int. J. Numer. Meth. Fluids*, **8**, 897-912.

[28] Harlow, F. and Welch, J.E. (1965), "Numerical calculation of time-dependent viscous incompressible flow of fluid with a free surface", *Phys. Fluids*, **8**, 2182-2189.

[29] Harlow, F.H and Welch, J.E. (1981), "A fast ICE solution procedure for flows with largely invariant compressiblity", *J. Comput. Phys.*, **40**, 254-261.

[30] Hirsch, C. (1988), *Numerical Computation of Internal and External Flows*, Vols. 1 and 2, Wiley series in Numerical Methods in Engineering.

[31] Hirt, C.W. and Shannon, J.P. (1968), "Free-surface stress conditions for incompressible-flow calculations", *J. Comput. Phys.*, **2**, 403-411.

[32] Hirt, C.W. and Cook, J.L. (1972), "Calculating three-dimensional flows around structures and over rough terrain", *J. Comput. Phys.*, **10**, 324-340.

[33] Hirt, C.W. and Nichols, B.D. (1981), "Volume of Fluid (VOF) method for the dynamics of free boundaries", *J. Comput. Phys.* **39**, 201-225.

[34] Hirt, C.W. (1988), *Flow3D Users Manual*, Flow Science Inc.

[35] Hoffman, G. (1975), "Improved form of the low Reynolds number $\kappa-\varepsilon$ turbulence model", *Phys. Fluids*, **18**, 309-312.

[36] Keunings, R. (1986), "An algorithm for the simulation of transient viscoelastic flows with free surfaces", *J. Comput. Phys.*, **62**, 199-220.

[37] Keunings, R. and Bousfield, D.W. (1987), "Analysis of surface tension driven leveling in horizontal viscoelastic films", *J. Non-Newton. Fluid Mech.*, **22**, 219-233.

[38] Kolte, M.I., Rasmussen, H.K. and Hassager, O. (1997), "Transient filament stretching rheometer. 2. Numerical simulation", *Rheol. Acta* **36**, 285-302.

[39] Launder, B.E. and Spalding, D.B. (1974), "The numerical computation of turbulent flows", *Int. J. Numer. Meth. Fluids*, **15**, 127-146.

[40] Lemos, C. (1996), "Higher-order schemes for free surface flows with arbitrary configurations", *Int. J. Numer. Meth. Fluids*, **23**, 545-566.

[41] LeVeque, R.J. (1992), *Numerical Methods for Conservation Laws*, Lectures in Mathematics, ETH, Zurich.

[42] Liang, Y., Ozetkin, A. and Neti, S. (1999), "Dynamics of viscoelastic jets of polymeric liquid extrudate", *J. Non-Newton. Fluid Mech.*, **81**, 105-132.

[43] Mangiavacchi, N, Castelo, A., Tomé, M.F., Cuminato, J.A., Bambozzi de Oliveira, M.L. and McKee, S., "A numerical technique for including surface tension effects for axisymmetric and planar flows using the GENSMAC method", (submitted for publication).

[44] Mäntylä, M. (1988), *An Introduction to Solid Modeling*, Computer Science Press, Rockville.

[45] Markham, G. and Proctor, M.V. (1983), "Modifications to the two-dimensional incompressible fluid flow code ZUNI to provide enhanced performance", C.E.G.B. report TPRD/L/0063/M82.

[46] McQueen, J.F. and Rutter, P. (1981), "Outline description of a recently implemented fluid flow code ZUNI", C.E.G.B. report LM/PHYS/258.

[47] Melville, W.K., (1996), "The role of surface-wave breaking in air-sea interaction", *Annual Rev. Fluid Mech.*, **28**:279-323.

[48] Miyata, H. and Nishimura, S. (1985), "Finite-difference simulation of nonlinear waves generated by ships of arbitrary three-dimensional configuration", *J. Comput. Phys.* **60**, 391-436.

[49] Miyata, H. (1986), "Finite-difference simulation of breaking waves", *J. Comput. Phys.* **65**, 179-214.

[50] Moretti, G. (1987), "Computation of flows with shocks", *Annual Rev. Fluid Mech.* **19**, 313-337.

[51] Nichols, B.D. and Hirt, C.W. (1971), "Improved free surface boundary conditions for numerical incompressible flow calculations", *J. Comput. Phys.*, **8**, 434.

[52] Nichols, B.D., Hirt, C.W. and Hotchkins, R.S. (1988) *SOLA-VOF: A solution algorithm for transient fluid flow with multiple free boundaries*, Technical Report LA-8355 (Los Alamos Scientific Laboratory).

[53] Osher, S. and Sethian, J.A. (1988), "Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations", *J. Comput. Phys.* **79 1**, 1-49.

[54] Owens, R.G. and Phillips, T.N. (2002), *Computational Rheology*, Imperial College Press.

[55] Pan, Y. and Banerjee, S. (1995), "A numerical study of free surface turbulence in channel flow", *Phys. Fluids*, **7**, 1649-1664.

[56] Peskin, C.S. (1977), "Numerical analysis of blood flow in the heart", *J. Comp. Phys.*, **25**, 220-252.

[57] Pracht, W.E. (1971), "A Numerical method for calculating transient creeping flows", *J. Comput. Phys.*, **7**, 46-60.

[58] Pracht, W.E. (1975), "Calculating three-dimensional fluid flows at all speeds with an Eulerian-Lagrangian computing mesh", *J. Comput. Phys.*, **17**, 132-159.

[59] Ryan, M.E. and Dutta, A. (1981), " A finite Difference simulation of Extrudate Swell, Proceedings of the 2nd World Congr. Chem. Eng.", **VI**, 277-281, Montreal.

[60] Santos, F.L.P., Mangiavacchi, N, Castelo, A., Tomé, M.F., Cuminato, J.A. and McKee, S., "A novel technique for free surface 2D multiphase flows", (submitted for publication).

[61] Sarpkaya, T. (1996), "Vorticity, free-surface and surfactants", *Annual Rev. Fluid Mech.*, **28**, 88-128.

[62] Sethian, J.A. (1995), *Theory, Algorithmns, and Applications of Level Set Methods for Propagating Interfaces*, Acta Numerica, C. U. P., Cambridge, UK.

[63] Sethian, J.A. (1996), *Level Set Methods: Evolving Interfaces in Geometry*, Fluid Mechanics, Computer Vision and Material Sciences, C. U. P., Cambridge, UK.

[64] Sicilian, J.M. and Hirt, C.W. (1984), "An efficient computation scheme for tracking contaminant concentrations in fluid flows", *J. Comput. Phys.*, **56**, 428-447.

[65] Sousa, F.S., Mangiavacchi, N, Nonato, L.G., Castelo, A., Tomé, M.F., Ferreira, V.G., Cuminato, J.A. and McKee, S., "A front tracking finite difference method for solve 3D Navier-Stokes equations for multi-fluid flows with free surfaces", (submitted for publication).

[66] Sussman, M. and Smereka, P. (1997), "Axisymmetric free boundary problems", *J. Fluid Mech.*, **341**, 269-294.

[67] Sussman, M. and Puckett, E.G. (2000), "A coupled level set and volume-of-fluid method for computing 3D and axisymmetric incompressible two-phase flows". *J. Comput. Phys.*, **162**, 301-337.

[68] Tanner, R.I. (1970), "A theory of die-swell", *J. Polymer Science*, **8**, 2067-2078.

[69] Tsai, T.M. and Miksis, M.J. (1994), "Dynamics of a drop in a constricted capillary tube", *J. Fluid Mech.*, **274**, 197-217.

[70] Tsai, W.-T. and Yue, D.K.P. (1996), "Computations of nonlinear free-surface flows", *Annual Rev. Fluid Mech.*, **28**, 249-278.

[71] Tsai, W.-T. (1998), "A numerical study of the evolution and structure of a turbulent shear layer under a free surface", *J. Fluid Mech.*, **354**, 239-276.

[72] Tomé, M.F. and McKee, S. (1994), "GENSMAC: A computational marker-and-cell method for free surface flows in general domains", *J. Comput. Phys.* **110**, 171-186.

[73] Tomé, M.F., Duffy, B. and McKee, S. (1996), "A numerical technique for solving unsteady non-Newtonian free surface flows", *J. Non-Newton. Fluid Mech.*, **62**, 9-34.

[74] Tomé, M.F., Castelo, A., Cuminato, J.A. and McKee, S. (1996) *GENSMAC3D: Implementation of the Navier-Stokes Equations and Boundary Conditions for 3D Free Surface Flows*, Universidade de São Paulo, Departamento de Ciência de Computação e Estatística, Notas do ICMC, No. 29.

[75] Tomé, M.F. and McKee, S. (1999), "Numerical simulation of viscous flow: buckling of planar jets", *Int. J. Numer. Meth. Fluids*, **29**, 705-718.

[76] Tomé, M.F. and McKee, S., Barratt, L., Jarvis, D.A. and Patrick, A.J. (1999), "An experimental and numerical investigation of container filling: viscous liquids", *Int. J. Numer. Meth. Fluids*, **31**, 1333-1353.

[77] Tomé, M.F., Castelo, A., Cuminato, J.A., Murakami, J., Minghim, R. and Oliveira, C.F. and McKee, S. (2000), "Numerical simulation of axisymmetric free surface flows", *J. Comput. Phys.*, **157**, 441-472.

[78] Tomé, M.F., Castelo, A., Cuminato, J.A. adn McKee, S. (2001), "GENSMAC3D: A numerical method for solving unsteady three-dimensional free surfaces flows", *Int. J. Numer. Meth. Fluids*, **37**, 747-796.

[79] Tomé, M.F., Mangiavacchi, N., Cuminato, J.A., Castelo, A. and McKee, S. (2002), "A numerical technique for solving unsteady viscoelastic free surface flows", *J. Non-Newton. Fluid Mech.*, **106**, 61-106.

[80] Torrey, M.D., Mjolsness, R.C. and Stein, L.R. (1987), *NASA-VOF3D: A three dimensional computer program for incompressible flows with free surfaces*, Technical Report LA-11009-MS (Los Alamos National Laboratory).

[81] Varonos, A. and Bergeles, G. (1998), "Development and assessment of a variable-order non-oscillatory scheme for convection term discretization", *Int. J. Numer. Meth. Fluids*, **26**, 1-16.

[82] Welch, J.E., Harlow, F.H., Shannon, J.P. and Daly, B.J. (1965), *The MAC method*, Technical Report LA-3425 (Los Alamos National Laboratory)

[83] Yang, Z., and Shih, H. (1993), "New time scale based $\kappa-\varepsilon$ model for near-wall turbulence", *AIAA J.*, **7**, 1191-1198.

[84] Yao, M.W. and McKinley, G.H. (1998), "Numerical simulation of extensional deformations of viscoelastic liquid bridges in filament stretching devices", *J. Non-Newton. Fluid Mech.*, **74**, 47-88.

NOTAS DO ICMC

SÉRIE COMPUTAÇÃO

073/2003 SOUZA, F.S.; MANGIAVACCHI, N.; NONATO, L.G.; CASTELO FILHO, A.; TOMÉ, M.F.; FERREIRA, V.G.; CUMINATO, J.A.; McKEE, S. – A front-tracking finite difference method to solve the 3D navier-stokes equations for multi-fluid flows with free surfaces.

072/2003 MANGIAVACCHI, N.; CASTELO FILHO, A.; TOMÉ, M.F.; CUMINATO, J.A.; OLIVEIRA, M.L.B.; McKEE, S. – A numerical technique for incluiding surface tension effects for axisymmetric and planar flows using the gensmac method.

071/2003 TOMÉ, M.F.; GROSSI, L.; CASTELO, A.; CUMINATO, J.A.; MANGIAVACCHI, N.; FERREIRA, V.G.; SOUSA, F.S.; McKEE, S. – A numerical method for solving three-dimensional generalized Newtonian free surface flows.

070/2003 SANTOS, F.L.P.; MANGIAVACCHI, N.; CASTELO, A.; TOMÉ, M.F.; CUMINATO, J.A. – A novel technique for free surface 2D multiphase flows.

069/2003 VIANNA, A.C.G.; ARENALES, M.N.; GRAMANI, M.C.N. – Two-stage and constrained two-dimensional guillotine cutting problems.

068/2003 ARAUJO, S.A.; ARENALES, M.N.; CLARK, A.R. – A lot-sizing and scheduling problem in a foundry.

067/2003 ARAUJO, S.A.; ARENALES, M.N. – Dimensionamento de lotes e programação do forno numa fundição automatizada de porte médio.

066/2002 VALERIO NETTO, A.; OLIVEIRA, M.C.F. – Industrial application trends and market perspectives for virtual reality and visual simulation.

065/2002 VALERIO NETTO, A.; OLIVEIRA, M.C.F. – Desenvolvimento de um protótipo de um torno CNC utilizando realidade virtual.

064/2002 MARQUES, F.P.; ARENALES, M.N. – O problema da mochila compartimentada e aplicações.