COBEM 99
XV Congresso Brasileiro de Engenharia Mecânica
15th Brazilian Congress of Mechanical Engineering
22 - 26 de Novembro de 1999 / November 22 - 26, 1999 Águas de Lindóia, São Paulo,

AN EVALUATION OF UPWIND-BASED SCHEMES FOR THREE-DIMENSIONAL FREE SURFACE FLOWS

V. G. Ferreira
Departmento de Estatistica, Matemática Aplicada e Computacional
Unesp de Rio Claro
M. F. Tomé
Instituto Superior Técnico de Lisboa, Departamento de Matemática
Av. Rovisco Pais,1, Lisboa CODEX, Lisboa, Portugal
A. O. Fortuna
A. Castelo F.
J. A. Cuminato
N. Mangiavacchi
Universidade de São Paulo
Departamento de Ciências de Computação e Estatística
Cx.P. 668 – 13560-161 – São Carlos, SP, Brasil

Abstract. *This work is concerned with the effects of different approximations to the convective terms on the simulation of incompressible three-dimensional free surface flows. A finite difference computer code for solving three-dimensional free surface flows is described. The Navier-Stokes equations together with the full stress boundary conditions are considered. The convective terms are treated explicitly employing four difference schemes: the usual centred difference, a first order upwind, the QUICK method and the HLPA (Hybrid Linear Parabolic Approximation) scheme. Results are presented showing the performance of these schemes applied to the simulation of an axisymmetric jet flowing into a three-dimensional container.*

Keywords: *Numerical simulation, Free-surface flows, Convective terms.*

1. INTRODUCTION

Accurate solution techniques for the Navier-Stokes equations have been extensively studied since the introduction of digital computers. Much of that study has been devoted to the accurate modeling of convection-dominated internal and external flows, since these form the bulk of engineering applications. While many discrete models for the convective terms have been developed, two models captured the attention of CFD practioners for

opposite reasons: second-order central difference (CD) and first-order upwind (FOU). In principle, the second-order accuracy of the CD scheme makes it an ideal candidate for accurate modeling of convection-dominated flows. Unfortunately, this scheme is prone to cause oscilations when the local cell Peclet number is greater than some critical value (normally taken as 2). Mesh refinement can alleviate the problem in the case of two-dimensional problems, but becomes impractical in three-dimensional simulations. FOU schemes do not cause such oscilations and are therefore employed to obtain oscilation-free solutions (e.g. Timin & Esmail, 1983). Due to their first order truncation error, however, they may introduce an unacceptable amount of "numerical diffusion" that can exceed the actual physical diffusion, thereby producing inaccurate solutions (Li & Baldacchino, 1995).

Different techniques were presented to deal with these problems. Spalding's HYBRID scheme (Spalding, 1972) employed the CD scheme everywhere but in regions where the Peclet number was greater than 2, in which case it reverted to FOU. Upstream-weighted differencing schemes were another attempt at minimizing the effects of numerical diffusion caused by pure first-order upwinding (Raithby & Torrance, 1974). Leonard's QUICK scheme (Leonard, 1979) represented one of the first attempts to develop a high-order method that did not display an oscillatory behaviour. Unfortunately, the QUICK scheme is prone to introduce undershootings and overshootings in regions surrounding steep gradients. This tendency to display under- and overshootings is not isolate but is shared with high-order methods. This unboundedness problem is sometimes dealt with using "flux limiters". These prevent the appearance of new maxima or minima in the solution, thus avoiding the development of unrealistic oscilations.

High-order schemes with and without flux limiting have been successfully applied to internal and external compressible/incompressible steady flows (Biagioli, 1998). However, their application to unsteady flow, specially free-surface flows, has been limited to two-dimensional problems. Armenio (1997) successfuly employed Zhu's HLPA (Zhu, 1992) for the solution of unsteady high Reynolds flows. Unfortunately, his paper is silent on the implementation of HLPA and the modeling of the convective term near free surface boundaries.

The objective of this work is to compare the performance of four convective schemes on the solution of three-dimensional free-surface flows governed by the Navier-Stokes equations. Schemes I and II are the standard first-order upwind scheme (FOU) and second-order central difference scheme (CD), respectively. QUICK method and HLPA were chosen as being scheme III and IV respectively. An explicit discretization of the convective and diffusion terms on a staggered grid was adopted. In order to apply these schemes to three-dimensional free surface flows we employ the Freeflow code developed by Tome *et al.* (1999).

2. BASIC EQUATIONS AND METHOD OF SOLUTION

The basic equations for incompressible Newtonian flows are the Navier-Stokes together with the mass conservation equation which in non-dimensional form can be written as

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\nabla p + \frac{1}{Re}\nabla^2 \mathbf{u} + \frac{1}{Fr^2}\mathbf{g}, \qquad \nabla \cdot \mathbf{u} = 0, \tag{1}$$

respectively. In the above equations $Re = UL/\nu$ is the Reynolds number and $Fr = U/\sqrt{Lg}$ is the Froude number. U and L are typical velocity and length scales, g is the gravitational constant and $\mathbf{g}$ is the unit gravitational field vector, $\mathbf{u} = (u, v, w)$ and p are the non-dimensional velocity and pressure fields.

2.1. Method of Solution

In order to solve equations Eqs. (1) we employ the GENSMAC3D (Tome *et al.*, 1999) algorithm which is the three-dimensional version of GENSMAC (Tome & McKee, 1994). This algorithm can be described as follows:
Let us suppose that at a given time, say t_n, the velocity field $\mathbf{u}(\mathbf{x}, t_n)$ 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 $t = t_n + \delta t_n$, we proceed as follows:

i: Let $\tilde{p}(\mathbf{x}, t_n)$ be a pressure field which satisfies the correct pressure condition on the free surface. This pressure field is computed according to the equations approximating the boundary conditions.

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

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

with $\tilde{\mathbf{u}}(\mathbf{x}, t_n) = \mathbf{u}(\mathbf{x}, t_n)$ using the correct boundary conditions for $\mathbf{u}(\mathbf{x}, t_n)$. Eq. (2) is solved by a finite difference method and the underlying difference equations will be given in Section 3.. It can be shown 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

$$\mathbf{u}(\mathbf{x}, t) = \tilde{\mathbf{u}}(\mathbf{x}, t) - \nabla \psi(\mathbf{x}, t) \tag{3}$$

with

$$\nabla^2 \psi(\mathbf{x}, t) = \nabla \cdot \tilde{\mathbf{u}}(\mathbf{x}, t). \tag{4}$$

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

iii: Solve the Poisson equation Eq. (4).

iv: Compute the velocity field Eq. (3).

v: Compute the pressure. It can be shown (Gaskell & Lau, 1988) that the pressure is given by

$$p(\mathbf{x}, t) = \tilde{p}(\mathbf{x}, t_n) + \frac{\partial \psi}{\partial t}(\mathbf{x}, t) \ . \tag{5}$$

2.2. Boundary conditions

We consider the flow of a fluid flowing into a passive atmosphere and neglect surface tension. In this case on the free surface we assume the following conditions

$$(\mathbf{T}.\mathbf{n}).\mathbf{n} = 0, \qquad (\mathbf{T}.\mathbf{n}).\mathbf{m_1} = 0, \qquad \text{and} \qquad (\mathbf{T}.\mathbf{n}).\mathbf{m_2} = 0, \tag{6}$$

where $\mathbf{T}$ is the Newtonian stress tensor, $\mathbf{n}$ is the local outward unit normal vector to the surface, and $\mathbf{m1}, \mathbf{m2}$ are local tangential vectors. On rigid boundaries the no-slip condition is applied, namely, $\mathbf{u} = \mathbf{0}$. For the Poisson equation we require $\frac{\partial \psi}{\partial n} = 0$ on rigid boundaries and $\psi = 0$ on the free surface.

3. FINITE DIFFERENCE DISCRETIZATION

To solve equations Eq. (2) – (5) we employ the finite difference method on a staggered grid. A typical cell is shown in Fig. 1. The velocity $\tilde{\mathbf{u}}$ is discretized at u, v and $w-$nodes, respectively. The time derivative is discretized using the forward difference while the viscous terms and the pressure gradients are 2nd-order approximated. The convective terms are approximated using the various differencing schemes presented in Section 3.3. Equation 2 is then approximated by

$$\tilde{u}_{i+\frac{1}{2},j,k} = u_{i+\frac{1}{2},j,k} - \delta t \left[CONV(u)_{i+\frac{1}{2},j,k} - \frac{1}{Fr^2} g_x - \frac{1}{Re} \left(\frac{u_{i-\frac{1}{2},j,k} - 2u_{i+\frac{1}{2},j,k} + u_{i+\frac{3}{2},j,k}}{\delta x^2} + \right.\right.$$

$$\left.\left. + \frac{u_{i+\frac{1}{2},j-1,k} - 2u_{i+\frac{1}{2},j,k} + u_{i+\frac{1}{2},j+1,k}}{\delta y^2} + \frac{u_{i+\frac{1}{2},j,k-1} - 2u_{i+\frac{1}{2},j,k} + u_{i+\frac{1}{2},j,k+1}}{\delta z^2} \right) + \frac{\tilde{p}_{i+1,j,k} - \tilde{p}_{i,j,k}}{\delta x} \right],$$

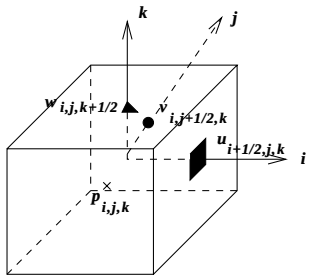

Figure 1: Typical cell in a GENSMAC3D calculation.

Expressions for $\tilde{v}_{i,j+\frac{1}{2},k}$ and $\tilde{w}_{i,j,k+\frac{1}{2}}$ are analogous.

In the above equation, $CONV$ represents the discretization of the convective terms $\left(\text{e.g. } CONV(u) = u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z}\right)$. The finite difference approximations for these derivatives will be given in Section 3.3.

The Poisson equation Eq. (4) is discretized at cell centres using the seven-point laplacian, which gives rise to a symmetric and positive definite linear system. This system is solved by the conjugate gradient method. The velocity at the advanced time t_{n+1} is obtained by discretizing Eq. (5) at the respective nodes. The time step size is selected according to the parabolic and hyperbolic stability restrictions.

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 wich do not contain fluid; Full (F) - Cells full of fluid; Surface (S) - Cells wich contain

fluid and have at least one face contiguous with an Empty cell; Boundary (B) - Cells which define a rigid boundary; Inflow (I) - Cells which define an inflow boundary. Details about the definition of these cells can be found in (Tome *et al.* 1999).

3.1. Boundary conditions approximation

To apply Eq. (6) we assume that the mesh cell is sufficiently small so that locally, the free surface can be adequately represented by a planar surface which is parallel to one of the coordinate axes, or at an angle of $\frac{\pi}{4}$ to two coordinate axes, or at an angle of $\frac{\pi}{6}$ to the three coordinate axes. These surfaces are identified by (S) cells having only one face contiguous with an empty cell (E) or only two surface cell faces contiguous with two empty cell faces or only three surface cell faces contiguous with three empty cell faces. Details of the finite difference equations corresponding to these approximations are given in (Tome *et al.* 1999).

The rigid boundary conditions currently implemented are of "no-slip" and "prescribed inflow" types. They are applied on the rigid boundaries (containers and inflow boundaries) in contact with the fluid and require the calculation of the intersections of lines parallel to the sides of the cells with the containers. These intersections are calculated only once and stored in a tree structure representing the cells in the structures Container and Inflow.

3.2. Particle movement

The fluid domain is represented by its boundaries. The fluid surface is defined by a piecewise linear surface composed of triangles and/or quadrilaterals containing marker particles on their vertices. The surface information is stored by a B-Rep data structure. The fluid surface is updated in three stages: firstly, the surface is moved to the new location according to the newly computed velocity field, in the second stage new particles are inserted if required and thirdly, particles are removed from cells which have accumulated too many. A fluid particle moves according to the equation $\frac{d\mathbf{x}_p}{dt} = \mathbf{u}_p$, where $\mathbf{u}_p$ is the velocity of the particle at time t_{n+1}. By using Euler's method, the particles are moved to the new position $x^{n+1} = x^n + u_p\delta t$, $y^{n+1} = y^n + v_p\delta t$, and $z^{n+1} = z^n + w_p\delta t$, where (x^n, y^n, z^n) is the position of particle p at time $t = t_n$. The particle velocities u_p, v_p, z_p are found by performing a tri-linear approximation using the eight nearest velocities. Details on the particle movement and on the insertion/deletion procedure can be found in (Castelo *et al.* 1998).

3.3. Approximation of convective terms

It is well known that the convective terms in the Navier-Stokes equations are primarily responsible for many of the complex flow phenomena. They are the major cause of numerical difficulties and various techniques for treating these terms have been developed (Timin & Esmail, 1983), (Spalding, 1972). In particular, the finite difference approximations to the derivatives deserves special attention if one seeks sufficiently accurate and stable schemes for solving realistic problems. The usual central difference scheme may lead to unphysical oscillatory behaviour in regions of the flow where convection dominates diffusion. First order upwind (Timin & Esmail, 1983) ensures stability of the calculation but introduces false diffusion (Li & Baldacchino, 1995). The QUICK scheme (Leonard, 1979) combines the accuracy of quadratic interpolation with stability, to obtain a diffusion free and a high order numerical scheme. However, since it is not a bounded scheme, it

can also introduce overshootings and undershootings. Recently, Zhu (1992) introduced a new scheme for treating convection dominated flows denominated HLPA (Hybrid Linear Parabolic Approximation). This scheme has the property of combining the high order accuracy of the QUICK scheme with an anti-diffusive term correcting the upwind scheme. In addition, it is a bounded scheme.

Consider Fig. 2 for computing the partial derivative $\frac{\partial \phi}{\partial s}$ of a generic variable ϕ at point P_0, where s is one of the coordinate axes and φ_A, φ_B are the convective velocities at the points P_A and P_B respectively. This derivative can be approximated by the expression

$$\left.\frac{\partial \phi}{\partial s}\right|_{P_0} = \frac{\phi_B - \phi_A}{\Delta s} \tag{7}$$

Using each of the schemes mentioned above, ϕ_A and ϕ_B are evaluated in terms of grid values $\phi_{-2}, \ldots, \phi_2$ by setting:

First Order Upwind (FOU):

$$\phi_A = \begin{cases} \phi_{-1} \text{ if } \varphi_A \geq 0 \\ \phi_0, \text{ otherwise} \end{cases} \qquad \phi_B = \begin{cases} \phi_0 \text{ if } \varphi_B \geq 0 \\ \phi_1, \text{ otherwise} \end{cases}$$

Central Difference (CD):

$$\phi_A = \frac{\phi_0 + \phi_{-1}}{2}, \qquad \phi_B = \frac{\phi_1 + \phi_0}{2}$$

QUICK:

$$\phi_A = \begin{cases} \frac{3}{8}\phi_0 + \frac{6}{8}\phi_{-1} - \frac{1}{8}\phi_{-2} \text{ if } \varphi_A \geq 0 \\ \frac{3}{8}\phi_{-1} + \frac{6}{8}\phi_0 - \frac{1}{8}\phi_1 \text{ if } \varphi_A < 0 \end{cases} \qquad \phi_B = \begin{cases} \frac{3}{8}\phi_1 + \frac{6}{8}\phi_0 - \frac{1}{8}\phi_{-1} \text{ if } \varphi_B \geq 0 \\ \frac{3}{8}\phi_0 + \frac{6}{8}\phi_1 - \frac{1}{8}\phi_2 \text{ if } \varphi_B < 0 \end{cases}$$

HLPA:

$$\phi_A = \begin{cases} \phi_{-1} + \gamma_A (\phi_0 - \phi_{-1}) \widehat{\phi}_A \text{ if } \varphi_A \geq 0 \\ \phi_0 + \gamma_A (\phi_{-1} - \phi_0) \widehat{\phi}_A \text{ if } \varphi_A < 0 \end{cases} \qquad \phi_B = \begin{cases} \phi_0 + \gamma_B (\phi_1 - \phi_0) \widehat{\phi}_B \text{ if } \varphi_B \geq 0 \\ \phi_1 + \gamma_B (\phi_0 - \phi_1) \widehat{\phi}_B \text{ if } \varphi_B < 0 \end{cases}$$

where

$$\gamma_A = \begin{cases} 1 \text{ if } \left|\widehat{\phi}_A - 0.5\right| < 0.5 \\ 0 \text{ otherwise} \end{cases} \qquad \gamma_B = \begin{cases} 1 \text{ if } \left|\widehat{\phi}_B - 0.5\right| < 0.5 \\ 0 \text{ otherwise} \end{cases}$$

and $\widehat{\phi}_A$ and $\widehat{\phi}_B$ are defined in terms of the upstream (ϕ_U), remote-upstream (ϕ_R) and downstream (ϕ_D) velocities at the points P_A and P_B, namely,

$$\widehat{\phi}_A = \left.\frac{\phi_U - \phi_R}{\phi_D - \phi_R}\right|_{PA}, \qquad \widehat{\phi}_B = \left.\frac{\phi_U - \phi_R}{\phi_D - \phi_R}\right|_{PB}.$$

3.4. Implementation of convective terms

We shall consider the application of the above schemes to three-dimensional flows. For brevity, only the discretization of the convective terms in the $x-$momentum equation

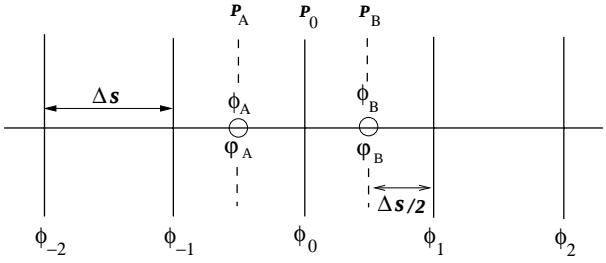

Figure 2: Stencil used for calculating ϕ_A and ϕ_B using various schemes.

is given; the other two-momentum components are treated similarly.

At node $\left(i+\frac{1}{2},j,k\right)$ the convective term $CONV(u)$ can be written as

$$\left[u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+w\frac{\partial u}{\partial z}\right]_{i+\frac{1}{2},j,k} = u_{i+\frac{1}{2},j,k}\left.\frac{\partial u}{\partial x}\right|_{i+\frac{1}{2},j,k}+v_{i+\frac{1}{2},j,k}\left.\frac{\partial u}{\partial y}\right|_{i+\frac{1}{2},j,k}+w_{i+\frac{1}{2},j,k}\left.\frac{\partial u}{\partial z}\right|_{i+\frac{1}{2},j,k} \quad (8)$$

Velocity components not defined at location $\left(i+\frac{1}{2},j,k\right)$ are obtained by averaging, e.g.,

$$v_{i+\frac{1}{2},j,k} = 0.25\left(v_{i,j+\frac{1}{2},k}+v_{i,j-\frac{1}{2},k}+v_{i+1,j+\frac{1}{2},k}+v_{i+1,j-\frac{1}{2},k}\right), \quad (9)$$

The derivatives in Eq. (8) are approximated by

$$\left.\frac{\partial u}{\partial x}\right|_{i+\frac{1}{2},j,k} = (u_{i+1,j,k}-u_{i,j,k})/\delta x, \quad (10)$$

$$\left.\frac{\partial u}{\partial y}\right|_{i+\frac{1}{2},j,k} = \frac{(u_{i+\frac{1}{2},j+\frac{1}{2},k}-u_{i+\frac{1}{2},j-\frac{1}{2},k})}{\delta y}, \quad \left.\frac{\partial u}{\partial z}\right|_{i+\frac{1}{2},j,k} = \frac{(u_{i+\frac{1}{2},j,k+\frac{1}{2}}-u_{i+\frac{1}{2},j,k-\frac{1}{2}})}{\delta z} \quad (11)$$

We present the finite difference approximation for the derivative $\left.\frac{\partial u}{\partial x}\right|_{i+\frac{1}{2},j,k}$. The corresponding difference equations for the other two derivatives are analogous. Firstly, let us define the following parameter

$$S_{i,j,k} = \begin{cases} 0, \text{ if } u_{i-\frac{1}{2},j,k}+u_{i+\frac{1}{2},j,k} \geq 0 \\ 1, \text{ otherwise} \end{cases} \quad (12)$$

The velocities appearing in Eq. (10) are given by:

First Order Upwind (FOU): $u_{i,j,k} := (1-S_{i,j,k})u_{i-\frac{1}{2},j,k}+S_{i,j,k}u_{i+\frac{1}{2},j,k}$

Central Difference (CD): $u_{i,j,k} := \frac{u_{i-\frac{1}{2},j,k}+u_{i+\frac{1}{2},j,k}}{2}$

QUICK:

$u_{i,j,k} := (1-S_{i,j,k})\left(\frac{3u_{i+\frac{1}{2},j,k}+6u_{i-\frac{1}{2},j,k}-u_{i-\frac{3}{2},j,k}}{8}\right)+S_{i,j,k}\left(\frac{3u_{i-\frac{1}{2},j,k}+6u_{i+\frac{1}{2},j,k}-u_{i+\frac{3}{2},j,k}}{8}\right)$

HLPA:

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

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

$$\gamma_{i,j,k} := \begin{cases} 1, \text{ if } |\widehat{\phi}_{i,j,k} - 0.5| < 0.5 \\ 0, \text{ otherwise} \end{cases}$$

Due to lack of information outside the domain the stencil of the convective term has to be kept small near the boundaries. Therefore, adjacent to the free surface and to solid boundaries, we adopted the hybrid discretization.

4. NUMERICAL EXAMPLES

The various schemes presented in Section 3.3 have been implemented into the Freeflow code. In order to compare the behaviour of these methods when applied to three-dimensional free surface flows, we have simulated the flow of a jet impinging onto a flat surface at increasing Reynolds numbers. We considered an empty open box and injected an axisymmetric jet of a viscous fluid into it at a prescribed velocity. The following input data were used:

- Domain dimensions: 2.5 cm × 2.5 cm × 2.5 cm.
- Mesh size: 50 × 50 × 100 cells ($\delta x = \delta y = 0.5$ mm, $\delta z = 0.25$ mm).
- Box dimensions: 2.5 cm × 2.5 cm × 1 cm.
- Inflow dimensions: diameter (D) = 4 mm and height (H) = 2 mm. The inflow is situated at a distance of 2.3 cm above the bottom of the box.
- Inflow velocity (U) = 1 ms^{-1}.
- Scaling parameters: U = 1 ms^{-1} and D = 4 mm.
- Gravity acting in z-direction with $g = -9.81$ ms^{-2}.
- Froude number ($Fr = \frac{U}{\sqrt{gL}}$) = 5.0482
- Poisson tolerance (EPS) = 10^{-8}

The Freeflow code simulated the problem described above using each of the schemes: FOU, CD, QUICK and HLPA. The following values of the kinematic viscosity (ν): 0.4×10^{-3}, 0.8×10^{-4}, 0.4×10^{-4}, 0.8×10^{-5}, and 0.4×10^{-5}, were used. These produced Reynolds numbers of 10, 50, 100, 500 and 1000, respectively. In total twenty runs have been performed, four runs for each of the Reynolds numbers. The only difference between one run and the other was the Reynolds number. The results of these runs are summarized in Fig 3.

Figure 3 displays the fluid flow visualization at t = 0.04 using each of the schemes above for each of the Reynolds numbers. Column CD shows the results obtained by the central difference and column FOU displays the results of the upwind scheme. Columns labelled QUICK and HLPA show the results of the QUICK and HLPA methods respectively. Due to numerical instabilities, simulations using CD and QUICK schemes were not able to reach the time $t = 0.04$ when the Reynolds number was sufficiently large (e.g. $Re = 500; 1,000$).

5. DISCUSSION

As an analytic solution for the problem studied in this paper is not known we shall compare the solutions obtained by the different schemes qualitatively. As we can see in Fig. 3, for Re = 10 and 50 the four schemes presented similar results not displaying any sign of numerical instability. However, as the Reynolds number increases, both CD and QUICK cause the numerical procedure to diverge due to numerical instability. This

is expected as those schemes are not bounded. On the other hand, FOU and HLPA produced acceptable results at high Reynolds numbers (e.g. $Re = 1,000$). Analysing the output obtained for this problem, FOU and HLPA are seen to produce similar results. This is due to the fact that the velocity field is mainly along coordinate lines, therefore the cross-stream diffusion introduced by FOU is small. However, observing Fig. 3d and 3e (FOU and HLPA), one can see that the height reached by the flow above the bottom surface is different for the two schemes. For HLPA, this height is greater than for FOU. This result is consistent with the lower aritificial viscosity introduced by HLPA, a second order-accurate scheme. This suggests the advantage of using HLPA for the simulation of free surface flows, since it is bounded and introduces less numerical diffusion than FOU.

Acknowledgments

We gratefully acknowledge the support given by Fapesp and CNPq.

REFERENCES

Armenio, V, 1997, An improved MAC method (SIMAC) for unsteady high-Reynolds free surface flows, Int. J. Num. Methods in Fluids, vol. 24, pp. 185-214.

Biagioli, F., 1998, Calculation of laminar flows with second-order schemes and collocated variable arrangement, Int. J. Num. Methods in Fluids, vol. 26, pp. 887-905.

Castelo F., A., Tome, M., César, C.N.L., McKee, S., and Cuminato, J.A., 1998, Freeflow: An Integrated Simulation System for Three-Dimensional Free Surface Flows, paper submitted to J. of Vis. Sim. Science.

Gaskell, P.H. and Lau, A.K.C, 1988, Curvature-compensated convective transport: SMART, a new boundedness-preserving transport algorithm, Int. J. Num. Methods in Fluids, vol. 8, pp. 617-641.

Leonard, B.P., 1979, An stable and accurate convective modelling procedure based on quadratic upstream interpolation, Comp. Meth. App. Mech. Eng., vol. 19, pp. 59-98.

Li, Y. and Baldacchino, L., 1995, Implementation of some higher-order convection schemes of non-uniform grids, Int. J. Num. Methods in Fluids, vol. 21, pp. 1201-1220.

Raithby, G.D. and Torrance, K.E., 1974, Upstream-weighted differencing schemes and their application to elliptic problems involving fluid flow, Comp. Fluids, vol. 2, pp. 191-206.

Spalding, D.B., 1972, A novel finite difference formulation for differential expressions involving both first and second derivatives, Int. J. Num. Methods in Engineering, vol. 4, pp. 551-559.

Timin, T. and Esmail, N., 1983, A comparative study of central and upwind difference schemes using the primitive variables, Int.J. Num. Methods in Fluids, vol. 3, pp. 295-305.

Tome, M.F., Castelo F., A., Cuminato, J.A., and McKee, S, 1999, GENSMAC3D: A Numerical Method for Solving Three-dimensional Free Surface Flows. Paper submitted to the Int. J. Num. Methods in Fluids.

Tome, M.F. and McKee, S., 1994, GENSMAC: A Computational Marker-and-Cell Method for Free Surface Flows in General Domains, J. Comp. Physics, vol. 110, pp. 171-186.

Zhu, J., 1992, On the higher-order bounded discretization schemes for finite volume computations of incompressible flows, Comp. Meth. Applied Mech. Eng., vol. 98, pp. 345-360.

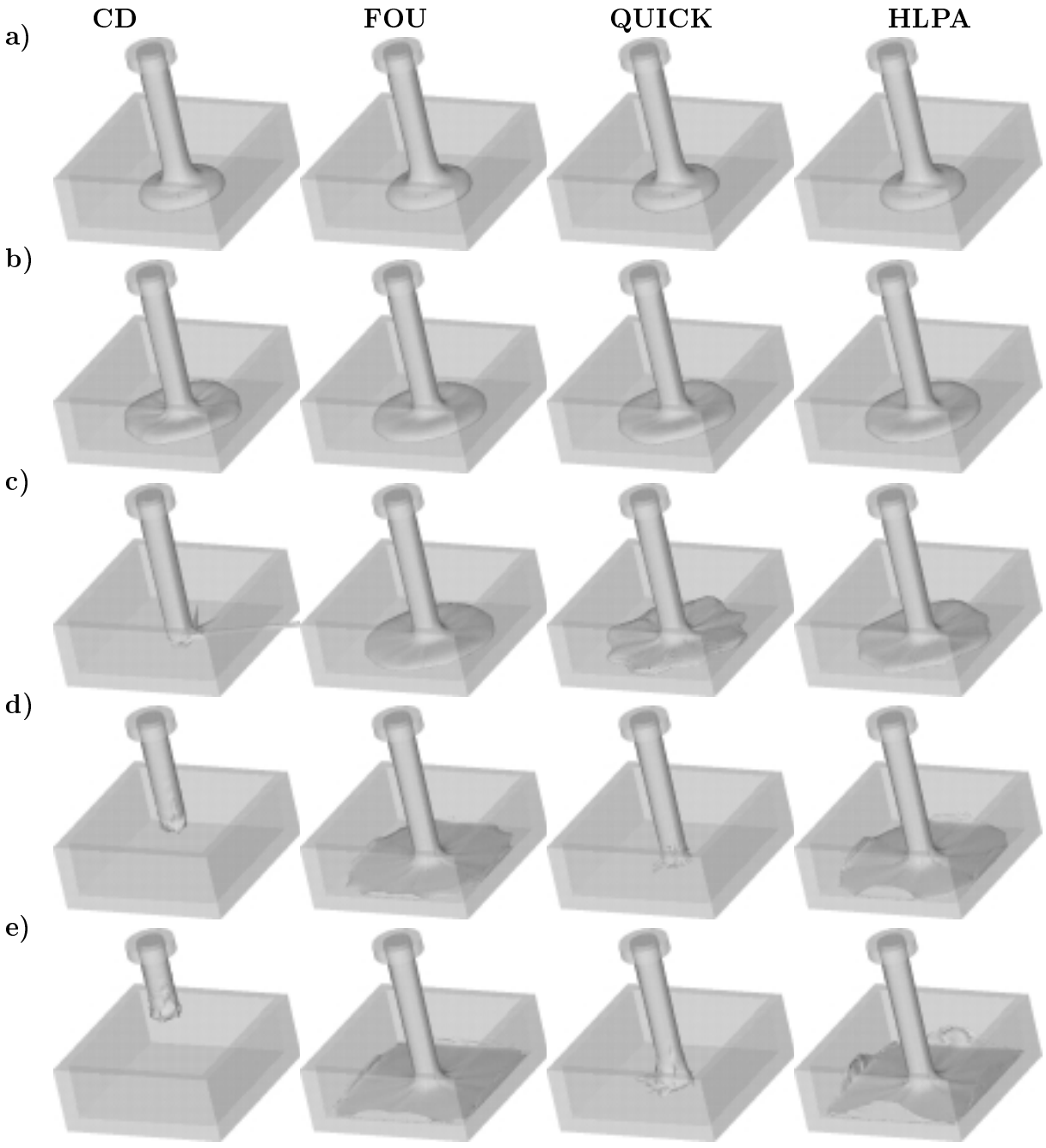

Figure 3: Fluid flow visualization of the simulation of a jet impinging onto a flat surface at increasing Reynolds numbers using different schemes. Columns are as follows: **A:** CD, **B:** FOU, **C:** QUICK, **D:** HLPA. Rows from top to bottom are as follows: **a)** Re = 10, **b)** Re = 50, **c)** Re = 100, **d)** Re = 500, **e)** Re = 1,000.