UNIVERSIDADE DE SÃO PAULO

NUMERICAL SIMULATION OF AXISYMMETRIC FREE SURFACE FLOWS

M. .F. TOMÉ
A. CASTELO FILHO
J. MURAKAMI
J. A. CUMINATO
R. MINGHIM
M. C. F. OLIVEIRA

Nº 30

NOTAS

Instituto de Ciências Matemáticas de São Carlos

Instituto de Ciências Matemáticas de São Carlos

ISSN - 0103-2577

NUMERICAL SIMULATION OF AXISYMMETRIC FREE SURFACE FLOWS

M. .F. TOMÉ
A. CASTELO FILHO
J. MURAKAMI
J. A. CUMINATO
R. MINGHIM
M. C. F. OLIVEIRA

Nº 30

NOTAS DO ICMSC
Série Computação

São Carlos
Set./1996

Numerical Simulation of Axisymmetric Free Surface Flows

Abstract

Free surface flows appear in various industrial applications such as cavity filling and injection moulding products. A number of researchers have tackled this class of problems but the development of efficient algorithms for simulating such flows yet present some challenges: the flow is transient, non-isothermal, non-Newtonian, three-dimensional and involves free surfaces in complex geometries. In this work we present a finite difference technique for solving axisymmetric free surface flows based on the SMAC (Simplified-Marker-and-Cell) method. The methodology employed is an extension to that embodied in the GENSMAC code for solving two-dimensional free surface flows in general domains. Numerical calculations demonstrating the applicability of the method for solving cavity filling problems are presented. Furthermore, results validating the technique and simulation of the splashing drop problem are included.

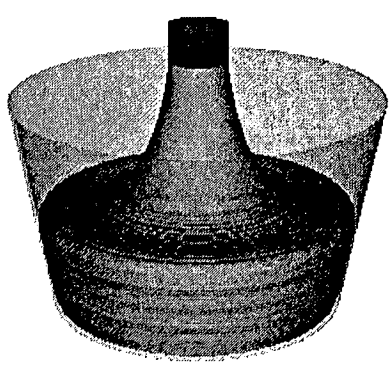
Resumo

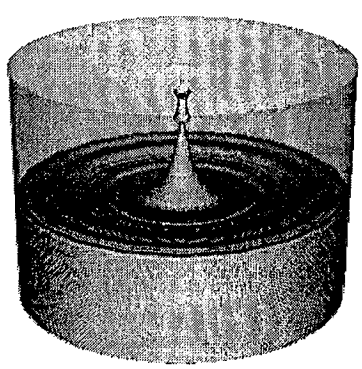
Escoamentos axisimetricos estão presentes em muitas aplicações industriais tail como enchimentos de moldes com geometria complexa. Neste trabalho apresentamos um método numérico, utilizando diferenças finitas, para resolver escoamentos axisimetricos com fronteira livre baseada no método SMAC (Simplified Marker-and-Cell). A metodologia empregada é uma extensão daquela utilizada em GENSMAC, para resolver escoamentos bidimensionais com superfícies livres em domínios arbitrários. Resultados numéricos ilustrando a aplicação do método para resolver problemas de enchimento de moldes complexos bem como comparação com resultados experimentais são apresentados.

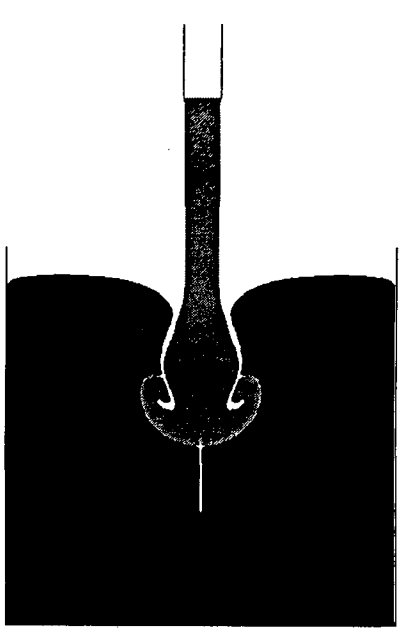
Numerical Simulation of Axisymmetric Free Surface Flows

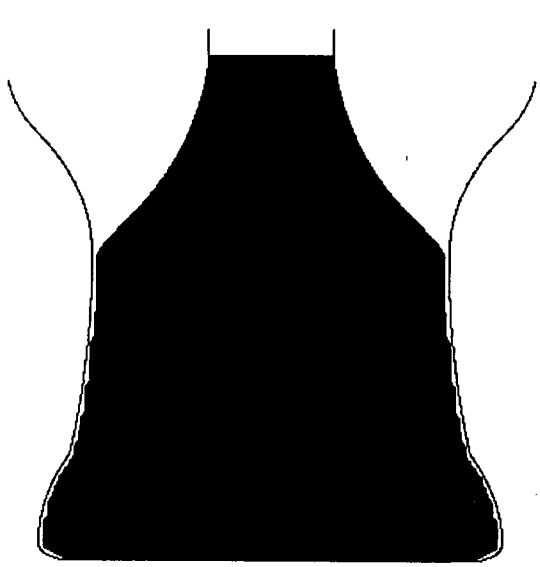
M. F. Tome, A. Castelo Fo., J. Murakami
J. A. Cuminato, R. Minghim and M. C. F. Oliveira

ICMSC - USP de São Carlos
Departamento de Ciência da Computação e Estatística
Av. Dr. Carlos Botelho, 1465 - Caixa Postal 668
13560-970 - São Carlos - SP - Brazil









1. Introduction

Modelling of injection moulding has been an area of intensive research over the past two decades, and yet present various challenges: the flow is unsteady, non-isothermal, non-newtonian and involves multiple free surfaces moving through cavities of complex shape. However research has resulted in a variety of computational techniques which have improved the design and manufacture of injection-moulded products. Among the various techniques employed for solving free surface problems, the Marker-and-Cell method (MAC) [1] was one of the first techniques to successfully simulate free surface flow. It is a finite difference solution technique based on a staggered grid for investigating the dynamics of an incompressible viscous fluid. It employs the primitive variables of pressure and velocity, and has particular application to the modelling of fluid flows with free surfaces. It has been developed by various researchers [2], [3], [4], [5], (to cite a few) and a detailed description of the technique can be found in Tome (1993). Recently Tome and McKee (1994) have developed the GENSMAC code which is an updated Marker-and-Cell methodology for solving free surface flows in general domains. It is an extention to the SMAC (Amsden and Harlow, 1971) (Simplified Mark-and-Cell method) for calculating time-dependent free surface flow problems employing pressure and velocity as the primary dependent variables. More specifically, GENSMAC calculates incompressible fluid flows in Cartesian co-ordinates within a general two-dimensional domain.The code is designed to deal with free surfaces flows having free-slip or no-slip rigid boundaries; a number of inflows and outflows can be handled as can any number of internal obstacles. However, although GENSMAC has a wide range of applicability it cannot deal with axisymmetric flows which are often encountered in many industrial applications. In this work we describe the implementation of cylindrical coordinates into the GENSMAC code with the aim of simulating axisymmetric free surface flows. Results demonstrating the applicability of the code to industrial applications and comparison with experimental data are presented. In addition, a graphic interface providing flow visualization for the code has been developed.

2. Basic Equations

We consider incompressible axisymmetric Newtonian flows. The governing equations are the non-dimensional mass and momentum equations which in cylindrical coordinates conservative form can be written as (Amsden and Harlow, 1971)

$$\frac{1}{r}\frac{\partial(ru)}{\partial r} + \frac{\partial v}{\partial z} = 0 \tag{1}$$

$$\frac{\partial u}{\partial t} + \frac{1}{r}\frac{\partial(ru^2)}{\partial r} + \frac{\partial(uv)}{\partial z} = -\frac{\partial p}{\partial r} + \frac{1}{Re}\frac{\partial}{\partial z}\left(\frac{\partial u}{\partial z} - \frac{\partial v}{\partial r}\right) + \frac{1}{F_r^2}g_r \tag{2}$$

$$\frac{\partial v}{\partial t} + \frac{1}{r}\frac{\partial(ruv)}{\partial r} + \frac{\partial v^2}{\partial z} = -\frac{\partial p}{\partial z} - \frac{1}{Re}\frac{1}{r}\frac{\partial}{\partial r}\left(r\left(\frac{\partial u}{\partial z} - \frac{\partial v}{\partial r}\right)\right) + \frac{1}{F_r^2}g_z \tag{3}$$

where $Re = UL/\nu$ and $F_r = U/\sqrt{Lg}$ denote the Reynolds number and the Froude number respectively. L and U are the length and velocity scales respectively, ν is a reference viscosity and $g = |\mathbf{g}|$. Further, $\mathbf{u} = (u, v)^T$ are the radial and vertical components of velocity while p is the non-dimensional pressure per unit density.

3. Method of Solution

For solving eqs. (1)–(3) we employ the GENSMAC methodology. A full description for two-dimensional flows can be found in [7].

It is supposed that at a given time t_0, the velocity field $\mathbf{u}(r, z, t_0)$ is known and boundary conditions for the velocity and pressure are given. The updated velocity field $\mathbf{u}(r, z, t)$ at $t = t_0 + \delta t$ is calculated as follows:

1. Let $\tilde{p}(r, z, t)$ be a pressure field which satisfies the correct pressure condition on the free surface. This pressure field is computed according to the stress conditions given in Section 4.2.

2. Calculate the intermediate velocity field $\tilde{\mathbf{u}}(r, z, t)$ from the explicity discretised form of

$$\frac{\partial u}{\partial t} = \left[-\frac{1}{r}\frac{\partial (ru^2)}{\partial r} - \frac{\partial (uv)}{\partial z} - \frac{\partial \tilde{p}}{\partial r} + \frac{1}{Re}\frac{\partial}{\partial z}\left(\frac{\partial u}{\partial z} - \frac{\partial v}{\partial r}\right) + \frac{1}{F_r^2} g_r \right]_{t=t_0} \tag{4}$$

$$\frac{\partial v}{\partial t} = \left[-\frac{1}{r}\frac{\partial (ruv)}{\partial r} - \frac{\partial (v^2)}{\partial z} - \frac{\partial \tilde{p}}{\partial z} - \frac{1}{Re}\frac{1}{r}\frac{\partial}{\partial r}\left(r\left(\frac{\partial u}{\partial z} - \frac{\partial v}{\partial r}\right)\right) + \frac{1}{F_r^2} g_z \right]_{t=t_0} \tag{5}$$

with $\tilde{\mathbf{u}}(r, z, t_0) = \mathbf{u}(r, z, t_0)$ using the correct boundary conditions for $\mathbf{u}(r, z, t_0)$. It can be shown (Tome et al., 1996) that $\tilde{\mathbf{u}}(r, z, t)$ possesses the correct vorticity at time t. However, $\tilde{\mathbf{u}}(r, z, t)$ does not satisfy (1). Let

$$\mathbf{u}(r, z, t) = \tilde{\mathbf{u}}(r, z, t) - \nabla \psi(r, z, t) \tag{6}$$

with

$$\nabla^2 \psi(r, z, t) = \nabla . \tilde{\mathbf{u}}(r, z, t) \; . \tag{7}$$

Thus, $\mathbf{u}(r, z, t)$ now satisfies (1) and the vorticity remains unchanged. Therefore, $\mathbf{u}(r, z, t)$ is identified with the updated velocity field at time t.

3. Solve the Poisson equation (7)

4. Compute the velocity (6)

5. Compute the pressure. It can be shown (see Tome et al., 1996) that the pressure is given by

$$p(r,z,t) = \tilde{p}(r,z,t) + \psi(r,z,t)/\delta t$$

6. Update the marker-particles positions

   The last step in the calculation involves moving the marker particles to their new positions. These are virtual particles whose coordinates are stored and updated at the end of each cycle by solving

$$\frac{dr}{dt} = u \ , \qquad \frac{dz}{dt} = v$$

   by Euler's method. This provides a particle with its new coordinates, allowing us to determine whether or not it moved to a new computational cell or if it leaved the containment region through an outlet.

3.1 Boundary Conditions

Boundary conditions must be imposed both on fixed boundaries and on free surfaces. On fixed boundaries we can impose no-slip, free-slip, prescribed inflow, prescribed outflow and continuative outflow (for details see [6,7]). The implementation of these boundary conditions is performed in the same way as in the GENSMAC code.
The appropriate free surface boundary conditions are the vanishing of the normal and the tangential stresses which in the absence of surface tension are (Batchelor, 1967):

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

$$\mathbf{m} \cdot \sigma \cdot \mathbf{n} = \mathbf{0} \tag{9}$$

where $\mathbf{n}$ and $\mathbf{m}$ are local unit normal and tangential vectors and σ is the stress tensor. These conditions are applied by making accurate local finite difference approximations on the free surface (Tome and McKee, 1994) and will be given in Section 4.2. The appropriate boundary condition for the Poisson equation (7) (see Amsden and Harlow [3]) is

$$\frac{\partial \psi}{\partial n} = 0 \quad \text{on fixed boundaries and} \quad \psi = 0 \quad \text{on the free surface} \tag{10}$$

4. Basic Finite Difference Equations

To implement the method presented in Section 3 we employ the finite difference method as follows.

A staggered grid is employed. A typical cell is as shown in Fig. 1.

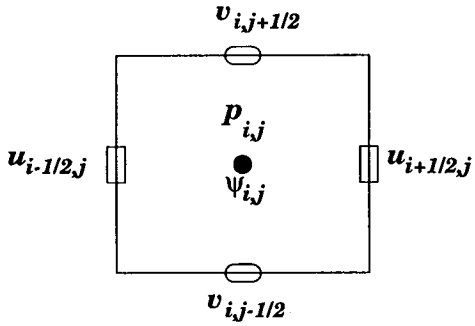


Fig. 1. Computational cell.

The variables pressure $\tilde{p}_{i,j}$ and the added velocity potential $\psi_{i,j}$ are positioned at the cell centre while $u_{i,j}$ and $v_{i,j}$ are staggered by a translation of $\delta r/2$ and $\delta z/2$ respectively.

The momentum equations (4) and (5) are discretized and applied at the u-nodes and v-nodes respectively. A forward difference in time is used for the time derivatives and the linear spatial terms on the right-hand side are approximated by central differences; for the convection terms in (4) and (5) the ZIP (see [3]) form is adopted. For the flux terms (uv), simple averages are performed, for instance

$$(uv)_{i+\frac{1}{2},j+\frac{1}{2}} = \frac{u_{i+\frac{1}{2},j} + u_{i+\frac{1}{2},j+1}}{2} \frac{v_{i,j+\frac{1}{2}} + v_{i+1,j+\frac{1}{2}}}{2}$$

Thus, the finite difference approximations to (4) and (5) become [3]

$$\tilde{u}_{i+\frac{1}{2},j} = u_{i+\frac{1}{2},j} + \delta t \left[\frac{u_{i+\frac{1}{2},j}}{r_{i+\frac{1}{2}}\delta r} \left(r_i u_{i-\frac{1}{2},j} - r_{i+1} u_{i+\frac{3}{2},j} \right) + \frac{\tilde{p}_{i,j} - \tilde{p}_{i+1,j}}{\delta r} + \frac{1}{F_r^2} g_r \right.$$
$$+ \frac{1}{4} \left((u_{i+\frac{1}{2},j} + u_{i+\frac{1}{2},j-1})(v_{i,j-\frac{1}{2}} + v_{i+1,j-\frac{1}{2}}) - (u_{i+\frac{1}{2},j} + u_{i+\frac{1}{2},j+1})(v_{i,j+\frac{1}{2}} + v_{i+1,j+\frac{1}{2}}) \right)$$
$$\left. + \frac{1}{Re} \left(\frac{u_{i+\frac{1}{2},j+1} + u_{i+\frac{1}{2},j-1} - 2u_{i+\frac{1}{2},j}}{\delta z^2} - \frac{v_{i+1,j+\frac{1}{2}} - v_{i+1,j-\frac{1}{2}} - v_{i,j+\frac{1}{2}} + v_{i,j-\frac{1}{2}}}{\delta r \delta z} \right) \right] \quad (11)$$

$$\tilde{v}_{i,j+\frac{1}{2}} = v_{i,j+\frac{1}{2}} + \delta t \left[\frac{v_{i,j+\frac{1}{2}} (v_{i,j-\frac{1}{2}} - v_{i,j+\frac{3}{2}})}{\delta z} + \frac{\tilde{p}_{i,j} - \tilde{p}_{i,j+1}}{\delta z} + \frac{1}{F_r^2} g_z \right.$$
$$+ \frac{1}{4 r_i \delta r} \left(r_{i-\frac{1}{2}} (u_{i-\frac{1}{2},j} + u_{i-\frac{1}{2},j+1})(v_{i-1,j+\frac{1}{2}} + v_{i,j+\frac{1}{2}}) - r_{i+\frac{1}{2}} (u_{i+\frac{1}{2},j} + u_{i+\frac{1}{2},j+1}) \right.$$
$$\left. (v_{i+1,j+\frac{1}{2}} + v_{i,j+\frac{1}{2}}) \right) - \frac{1}{Re} \frac{1}{r_i \delta r} \left(r_{i+\frac{1}{2}} \left(\frac{u_{i+\frac{1}{2},j+1} - u_{i+\frac{1}{2},j}}{\delta z} - \frac{v_{i+1,j+\frac{1}{2}} - v_{i,j+\frac{1}{2}}}{\delta r} \right) \right.$$
$$\left. \left. - r_{i+\frac{1}{2}} \left(\frac{u_{i-\frac{1}{2},j+1} - u_{i-\frac{1}{2},j}}{\delta z} - \frac{v_{i,j+\frac{1}{2}} - v_{i-1,j+\frac{1}{2}}}{\delta r} \right) \right) \right] \quad (12)$$

The Poisson equation (7) in cylindrical coordinates becomes

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial \psi}{\partial r}\right)+\frac{\partial^2 \psi}{\partial z^2}=\tilde{D} \qquad \text{where} \qquad \tilde{D}=\frac{1}{r}\frac{\partial (r\tilde{u})}{\partial r}+\frac{\partial \tilde{v}}{\partial z}\,. \tag{13}$$

We point out that the direct discretization of (13) would lead to a nonsymmmetric linear system which would be solved by the bi-conjugate gradient method. However, by rewriting (13) as

$$\frac{\partial}{\partial r}\left(r\frac{\partial \psi}{\partial r}\right)+r\frac{\partial^2 \psi}{\partial z^2}=r\tilde{D} \tag{14}$$

and discretizing (14) at the surface cell centre we obtain

$$-r_i\psi_{i,j-1}-r_{i-\frac{1}{2}}\psi_{i-1,j}+4r_i\psi_{i,j}-r_{i+\frac{1}{2}}\psi_{i+1,j}-r_i\psi_{i,j+1}=-r_i\tilde{D}_{i,j} \tag{15}$$

where

$$\tilde{D}_{i,j}=\frac{1}{r_i}\left(\frac{r_{i+\frac{1}{2}}\tilde{u}_{i+\frac{1}{2},j}-r_{i-\frac{1}{2}}\tilde{u}_{i-\frac{1}{2},j}}{\delta r}\right)+\frac{\tilde{v}_{i,j+\frac{1}{2}}-\tilde{v}_{i,j-\frac{1}{2}}}{\delta z}\,.$$

It is easily seen that now (15) leads to a linear system possessing a symmmetric positive definite matrix. We have employed the conjugate gradient method as implemented in GENSMAC for solving this linear system.

4.1 Cell Flagging

As the fluid is continuously moving, a procedure for identifying the fluid region and the free surface is employed. To accommodate this, the cells within the mesh are flagged according to whether they are surface cells (S), full cells (F), empty cells (E), boundary cells (B), inflow cells (I) and outflow cells (O). A detailed description can be found in Tome and McKee [7]. Figure 2. illustrates these types of cells within the mesh.

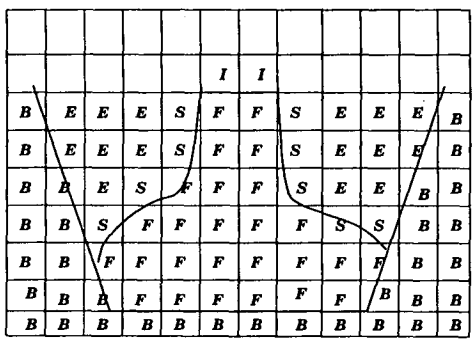


Fig. 2. Types of cells whithin the mesh.

4.2 Free Surface Stress Conditions

The stress conditions (8)–(9) can be written as

$$p - \frac{2}{Re}\left[\frac{\partial u}{\partial r}n_r^2 + \frac{\partial v}{\partial z}n_z^2 + \left(\frac{\partial u}{\partial z} + \frac{\partial v}{\partial r}\right)n_r n_z\right] = 0\,, \tag{16}$$

$$\frac{2}{Re}\left[\left(\frac{\partial u}{\partial r} - \frac{\partial v}{\partial z}\right)n_r n_z + \left(\frac{\partial u}{\partial z} + \frac{\partial v}{\partial r}\right)(n_r^2 - n_z^2)\right] = 0\,. \tag{17}$$

In order to apply these conditions we follow the approximations adopted by GENSMAC [7]; namely we consider two types of free surface orientations as follows:

a) **Horizontal/vertical surfaces:** These surfaces are identified by surface cells having only one side contiguous with empty cells. For these cells we assume the normal vector is pointing to the empty cell in which case we take $\mathbf{n} = (n_r, 0)$ or $\mathbf{n} = (0, n_z)$. The choice is made according to which side is contiguous with the empty cell. For instance, if a surface cell has only the top side contiguous with an empty cell (see Fig. 3.) then we take $\mathbf{n} = (0, 1)$. Equations (16) and (17) then reduce to

$$p - \frac{2}{Re}\left(\frac{\partial v}{\partial z}\right) = 0 \tag{18}$$

$$\left(\frac{\partial u}{\partial z} + \frac{\partial v}{\partial r}\right) = 0 \tag{19}$$

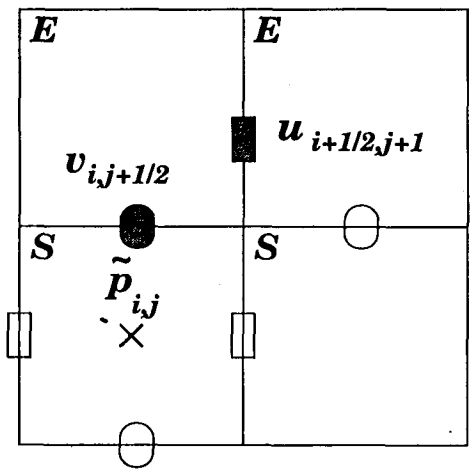


Fig. 3. Surface cell with only the top side contiguous with an empty cell.

It can be seen that when computing the tilde velocities through eqs. (11) and (12) the pressure $\tilde{p}_{i,j}$ and the values of $u_{i+\frac{1}{2},j+1}$ and $v_{i,j+\frac{1}{2}}$ at the empty cell faces are required (see Fig. 3). These can be obtained from eqs. (18), (19)

and the mass conservation equation (1) as follows: First, by applying (1) at the surface cell centre we get

$$\frac{1}{r_i}\left(\frac{r_{i+\frac{1}{2}}u_{i+\frac{1}{2},j} - r_{i-\frac{1}{2}}u_{i-\frac{1}{2},j}}{\delta r}\right) + \frac{v_{i,j+\frac{1}{2}} - v_{i,j-\frac{1}{2}}}{\delta z} = 0$$

which gives

$$v_{i,j+\frac{1}{2}} = v_{i,j-\frac{1}{2}} - \frac{\delta z}{\delta r}\frac{1}{r_i}\left(r_{i+\frac{1}{2}}u_{i+\frac{1}{2},j} - r_{i-\frac{1}{2}}u_{i-\frac{1}{2},j}\right) .$$

Now, discretizing (19) at position $(i+\frac{1}{2}, j+\frac{1}{2})$ we obtain

$$\frac{u_{i+\frac{1}{2},j+1} - u_{i+\frac{1}{2},j}}{\delta z} + \frac{v_{i+1,j+\frac{1}{2}} - v_{i,j+\frac{1}{2}}}{\delta r} = 0$$

which yields

$$u_{i+\frac{1}{2},j+1} = u_{i+\frac{1}{2},j} - \frac{\delta z}{\delta r}\left(v_{i+1,j+\frac{1}{2}} - v_{i,j+\frac{1}{2}}\right) .$$

Once the velocities have been computed the pressure $\tilde{p}_{i,j}$ follows from (18) applied at the surface cell centre, giving

$$\tilde{p}_{i,j} = \frac{2}{Re}\left(\frac{v_{i,j+\frac{1}{2}} - v_{i,j-\frac{1}{2}}}{\delta z}\right) .$$

Other types of configurations of surface cells having only one side contiguous with empty cells are treated similarly.

b) **45^0–sloped surface:** These surfaces are identified by surface cells having two adjacent faces contiguous with empty cells. For these cells we assume that the normal vector makes 45^0 with the axes in which case we take $\mathbf{n} = (\pm\frac{\sqrt{2}}{2}, \pm\frac{\sqrt{2}}{2})$. The sign is chosen according to which faces are contiguous with empty cells.

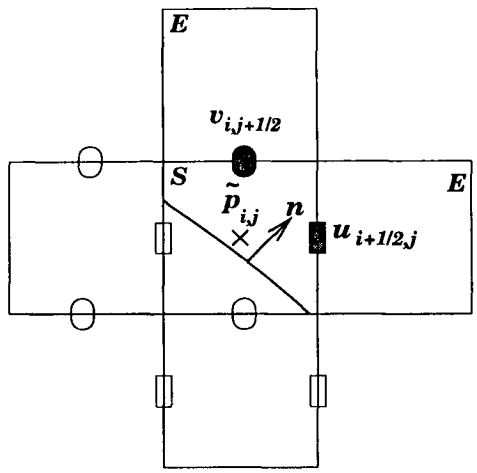


Fig. 4. Surface cell having the top and right sides contiguous with empty cells.

For instance, let us consider the surface cell in Fig. 4. For this cell we take $\mathbf{n} = (\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2})$ and introducing it into (16) and (17) we obtain

$$p - \frac{1}{Re}\left[\frac{\partial u}{\partial r} + \frac{\partial v}{\partial z} + \left(\frac{\partial u}{\partial z} + \frac{\partial v}{\partial r}\right)\right] = 0\,, \tag{20}$$

$$\left(\frac{\partial u}{\partial r} - \frac{\partial v}{\partial z}\right) = 0\,, \tag{21}$$

respectively. As we can see in Fig. 4, the values of $u_{i+\frac{1}{2},j}$ and $v_{i,j+\frac{1}{2}}$ are required. These are obtained by applying (21) and the mass conservation (1) at the surface cell centre which gives

$$\frac{u_{i+\frac{1}{2},j} - u_{i-\frac{1}{2},j}}{\delta r} + \frac{v_{i,j+\frac{1}{2}} - v_{i,j-\frac{1}{2}}}{\delta z} = 0\,, \tag{22}$$

$$\frac{1}{r_i}\left(\frac{r_{i+\frac{1}{2}}u_{i+\frac{1}{2},j} - r_{i-\frac{1}{2}}u_{i-\frac{1}{2},j}}{\delta r}\right) + \frac{v_{i,j+\frac{1}{2}} - v_{i,j-\frac{1}{2}}}{\delta z} = 0\,. \tag{23}$$

Eqs. (22) and (23) provide a (2×2)–linear system for $u_{i+\frac{1}{2},j}$ and $v_{i,j+\frac{1}{2}}$, yielding

$$u_{i+\frac{1}{2},j} = \frac{r_i + r_{i-\frac{1}{2}}}{r_i + r_{i+\frac{1}{2}}}u_{i-\frac{1}{2},j}\;;\quad v_{i,j+\frac{1}{2}} = v_{i,j-\frac{1}{2}} + \frac{\delta z}{\delta r}\left(u_{i+\frac{1}{2},j} - u_{i-\frac{1}{2},j}\right)\,. \tag{24}$$

Once the velocities at the empty cell faces have been computed, the pressure follows from (20) applied at the surface cell centre, giving

$$\tilde{p}_{i,j} = \frac{1}{Re}\left[\frac{u_{i+\frac{1}{2},j} - u_{i-\frac{1}{2},j}}{\delta r} + \frac{v_{i,j+\frac{1}{2}} - v_{i,j-\frac{1}{2}}}{\delta z} + \frac{1}{2}\left(\frac{u_{i+\frac{1}{2},j} + u_{i-\frac{1}{2},j}}{}\right.\right.$$
$$\left.\left.\frac{-u_{i+\frac{1}{2},j-1} - u_{i-\frac{1}{2},j-1}}{\delta z} + \frac{v_{i,j+\frac{1}{2}} + v_{i,j-\frac{1}{2}} - v_{i-1,j+\frac{1}{2}} - v_{i-1,j-\frac{1}{2}}}{\delta r}\right)\right]\,. \tag{25}$$

Other configurations of surface cells having two adjacent sides contiguous with empty cell the values of u, v and $\tilde{p}$ are obtained similarly.

c) **Surface cells with three sides or two opposite sides contiguous with empty cells:** These cells do not provide enough information to obtain an appoximation for the unit normal **n**. If they appear during a calculation we set the pressure equal to zero and adjust at least one velocity on one of the empty cell faces so that mass is conserved. To avoid the appearence of such cells a finer mesh should be employed.

4.3 Particle Movement

In accessing the most time consuming parts of GENSMAC it turned out that the particle movement routine took about fourty percent of the total time. This is due to the staggering number of particles needed to represent the fluid properly. A new technique was devised based on representing the fluid by its boundary using a set of ordered lists which defines the interior and exterior of the region containing the fluid. Each list stores a connected component of the fluid and each of its nodes stores information concerned with the position of the particle the type of the cell it is located in and the type of movement it can be performed on it. For instance a node of type "inflow" cannot move, whereas a node of type "surface" can move freely according to the velocity field. The fluid movement is obtained by solving the equations $\dot{r} = u(r, z)$, $\dot{z} = v(r, z)$, where u, v are the velocities in the r and z directions respectively. As u and v are defined on a staggered grid, the velocity in each node is obtained from bilinear interpolation using the 4 nearest velocities. This new technique has enhanced the code regarding its speed up and performance and enabled the code to cope with the jet flow and the splashing drop presented in Section 6.

5. VisFreeFlow

Computational Fluid Dynamics is characterised by the ample use of Computer Graphics resources in order to analyse and validate results from its simulations. In connection with the GENSMAC fluid flow code developed by Tome and Mckee [7], a visualisation package has been implemented that provides facilities for collecting domain data and flow parameters as well as displaying many results of the simulation graphically.

The package we have developed (VisFreeFlow) comprises modules for modelling the flow domain, for parameter reading, and for visualisation of the results.

The modelling module allows for definition of the flow domain. This is realised by a graphical editor comprising a drawing area, command buttons for interaction with the model, configuration options, and file management tools. Each flow domain may be designed using polylines and arcs. Inflows and outflows are specified and displayed using specific colours. Additional interaction techniques are available (like "dragging", "rubberbanding", and reference grid). Domain data can also be input

by typing segments in the format accepted by GENSMAC. The domain editor works independently of the simulation process, therefore allowing the same mould to be used in connection with distinct simulations.

The parameter reading module of VisFreeFlow provides a graphical user interface for inputing data, like viscosity, length and velocity scales, etc. This module is based on the XWindows windowing system, with Motif-like dialogue structures. Data validation, data filing, data retrieving, changing and association of flow data options are features readily available within this reading module. This program is also responsible for triggering the simulation process.

The output of the simulation can be graphically displayed with the visualisation module of VisFreeFlow. The possible visualisations include time varying display of flow particles, pressure, temperature and velocity, as well as the cell grid. The fluid flow is visualised using several techniques: particle tracing, boundary tracing and space-filling. Pressure can be visualised using isolines, as well as colour mapping of pressure ranges. The velocity field is visualized by vector plots that represent direction and intensity. Intensity can also be visualised using isolines as for pressure plots. The parameters for all the visualisation options can be interactively changed to achieve a good display. Multiple viewing is available by use of different windows or by "layering" plots together. Zoom in and out can also be obtained to support analysis of details.

6. Numerical Examples and Comparison with Experimental Data

As we can see in the procedure described in Section 3, the inclusion of cylindrical coordinates into the GENSMAC code will only affect steps 1, 2 and 3. Steps 4, 5 and 6 remain the same as the equations do not change from those for two-dimensional flows. Thus, a routine for calculating the pressure field $\tilde{p}(r, z, t)$ according to the equations for the stress conditions (see Section 4.2) was written as well as a routine for computing the tilde velocities through eqs. (11) and (12). The discretized Poisson equation (15) has been solved by the conjugate gradient routine employed by GENSMAC; for this it was necessary to write a routine for assembling the matrix and the corresponding right hand vector. Finally, regarding the boundary conditions for $\mathbf{u}(r, z, t)$, a routine for computing the velocities on empty cell faces according to the equations described in Section 4.2 was written. The equations for calculating the tangential velocities at the free surface are the same for two-dimensional flows. The boundary conditions on rigid boundaries have been handled in the same manner as for two-dimensional free surface flows as implemented in GENSMAC (for details see [7]).

6.1 Jet Flow Experiment

Due to their importance in industrial applications, low Reynolds number flows have been the subject of study of various researchers. In particular, Taylor [10] has

produced experimental data concerning jet flows. In his experiments he showed what happen if a jet is injected into a box containing the same fluid. To visualize the flow produced by the jet, the fluid in the box was uncoloured while the jet was coloured. Four experiments were performed using four fluids having different viscosities, the difference between the flows observed in the four cases were only due to differences in Reynolds number; the diameters D and velocities V were kept the same in all cases. The experiments were performed for the following Reynolds numbers: $Re = 0.05, Re = 10, Re = 200$ and $Re = 3000$. The case of $Re = 0.05$ was shown that the high viscosity fluid (syrup was used) produced a vertical stress which destroyed the vertical velocity of the jet. This in turn caused the jet to buckle as it reached the box, not penetrating the mass of fluid contained in the box. For $Re = 3000$ the jet was destroyed as it became turbulent. For the other two experiments the jet was steady having a mushroomlike head formed at its front. For the case of $Re = 10$ was reported that the jet penetrated only a few jet diameters before spreading out into a mushroom like head while for $Re = 200$ the jet penetrated the fluid until the bottom of the box was reached.

To validade the technique presented in this report we have performed one run on jet flows to compare with the experimental data presented by Taylor [10]. It is assumed that the flow is axisymmetric. We have considered a square box of 5 cm $\times$ 5 cm containing a coloured fluid. Above it, we set an inlet of 5 mm diameter from which a coloured jet falls vertically into the box at a prescribed velocity of 1 ms^{-1} (see Fig. 6.). The viscosity was set to $\nu = 0.0005$ m^2s^{-1}. This gave a Reynolds number of

$$Re = \frac{VD}{\nu} = 10 \text{ and a Froude number of } F_r = \frac{V}{\sqrt{gD}} \approx 0.2215$$

Figures 7 displays one snapshot taken from this run to compare with the experimental result of Taylor. As we can see the agreement with the experiment is formidable. A detailed evolution of the flow is shown in figure 8.

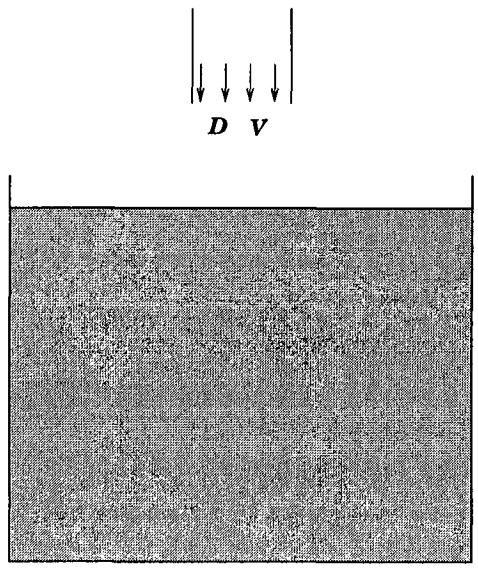


Fig. 6. Jet flow experiment.

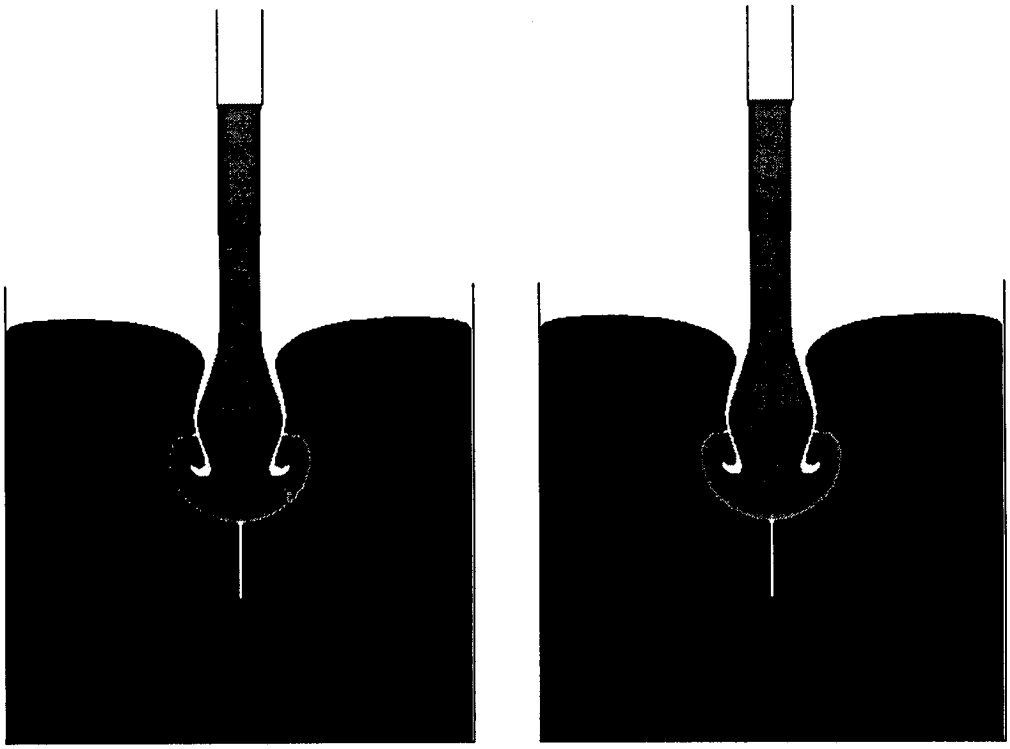


Fig. 7. Jet flow experiment: Re = 10.

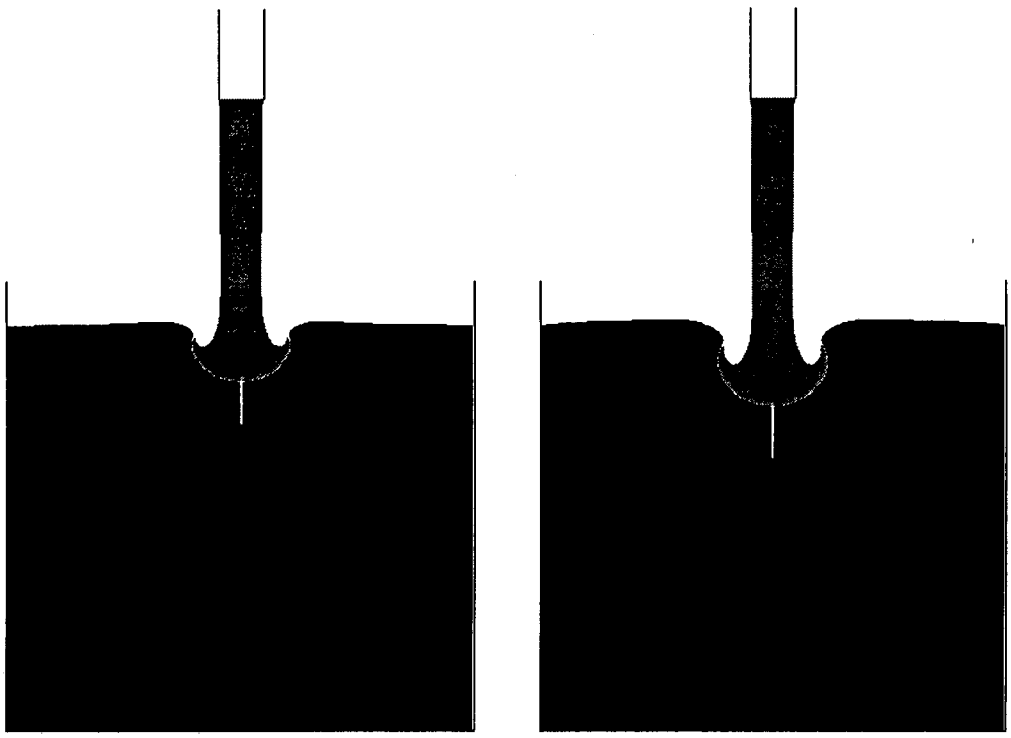


Fig. 8. Jet Flow experiment – $Re = 10$. Flow configuration at different times.

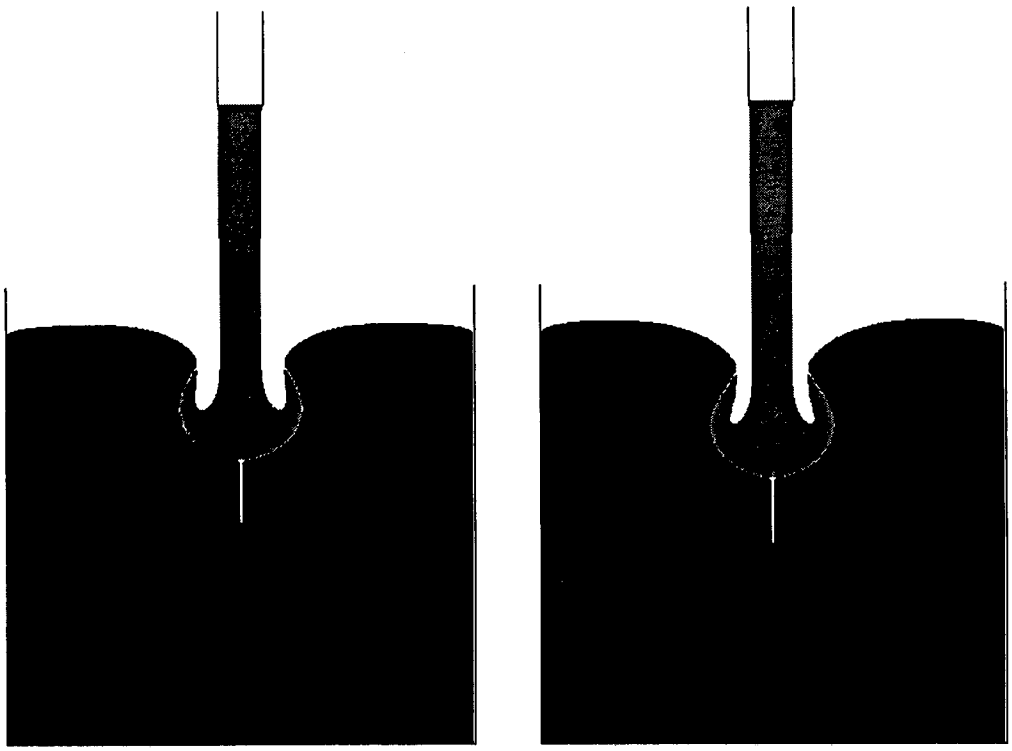


Fig. 8. Continued.

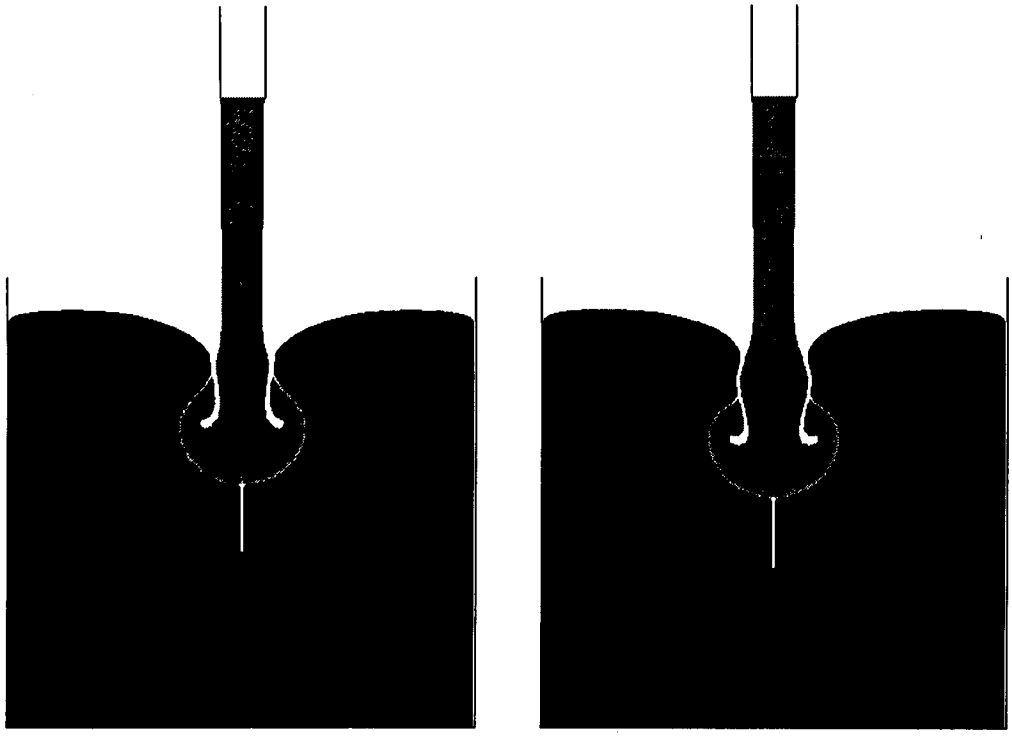


Fig. 8. Continued.

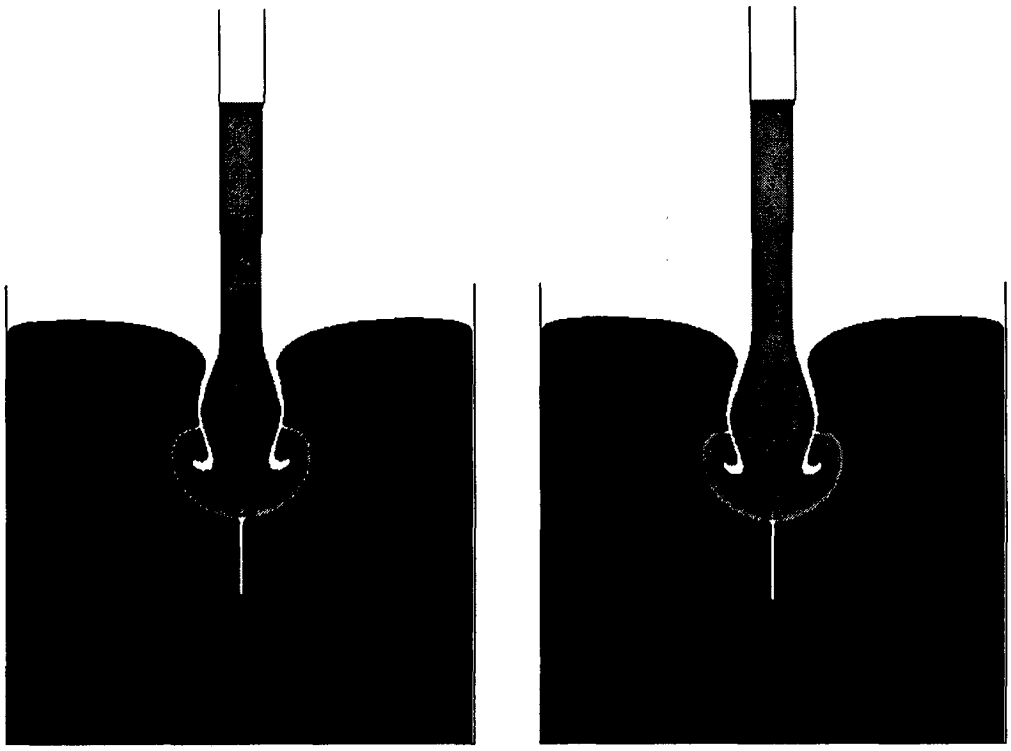


Fig. 8. Continued.

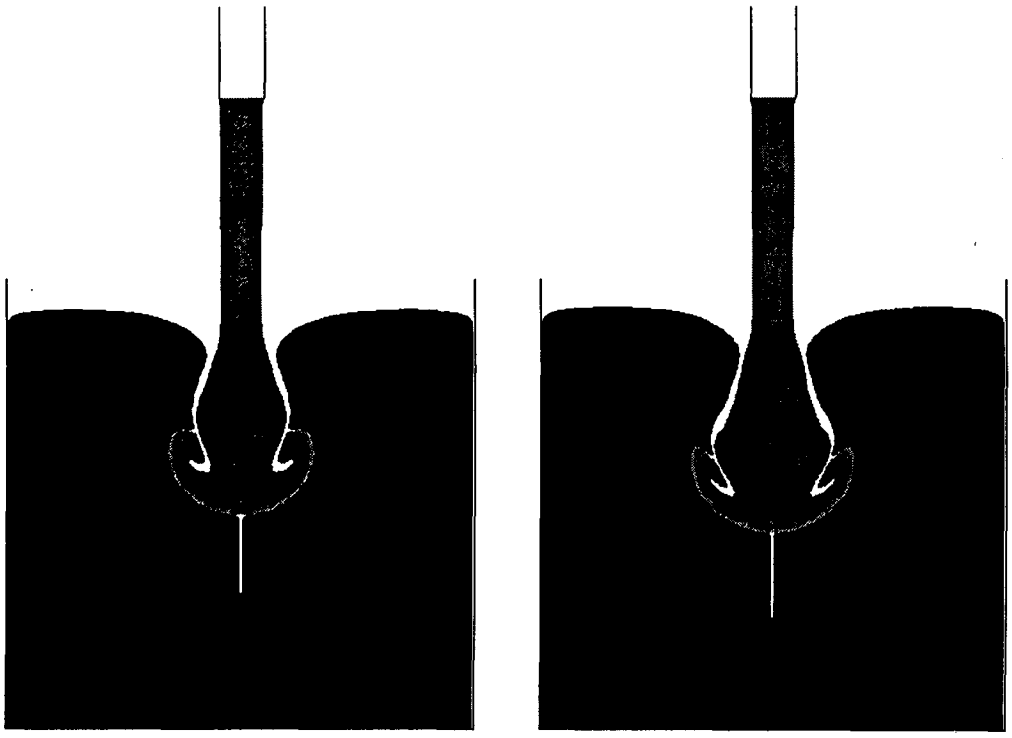


Fig. 8. Continued.

6.2 Simulation of Splashing Drop

To demonstrate the accuracy of the free surface approximations presented in Section 4.2 we present a calculation which simulates the "splashing of a drop of fluid". We consider a pool of 22 cm $\times$ 7 cm of fluid and a spherical drop of 3.2 cm of diameter falling under the force of gravity into the pool. The flow is assumed to be axisymmetric. The cell size was $\delta r = \delta z = 0.2$ cm. Gravity was set to $g = -981$ cms^{-2} and viscosity was taken to be $\nu = 1$ cm^{2}s^{-1}. The scaling parameters were set to $U = L = 1$ (physical coordinates were employed in this run). The drop was set at a distance of 3 cm above the pool with an initial velocity of - 1 cms^{-2} and let to fall under gravity. To visualize the flow produced by the drop, the fluid in the pool and in the drop are set to different colors. The results showing the flow configuration (with a mirror image to the left) at different times are displayed in figure 10.

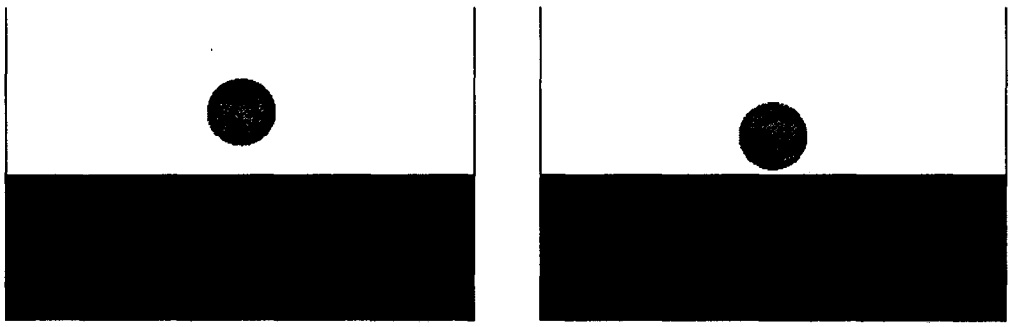


Fig. 10. The splashing of a drop of fluid: flow configuration at times $t = 0.006$, 0.048, 0.078, 0.102, 0.126, 0.144, 0.156, 0.168, 0.180, 0.186, 0.192, 0.198, 0.204 and 0.222 s, respectively.

Fig. 10. Continued.

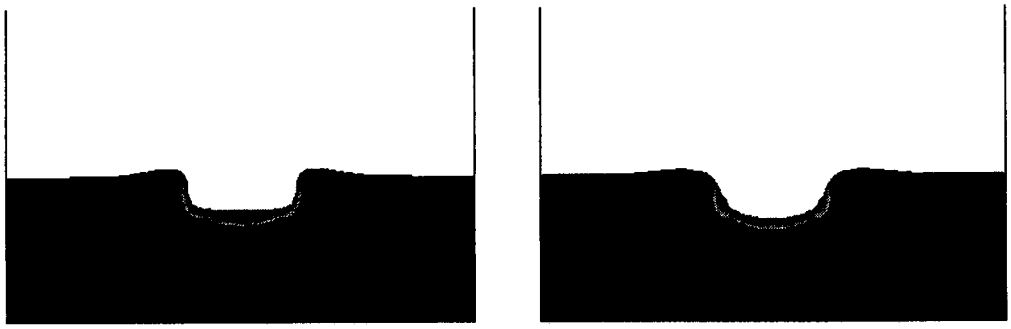


Fig. 10. Continued.

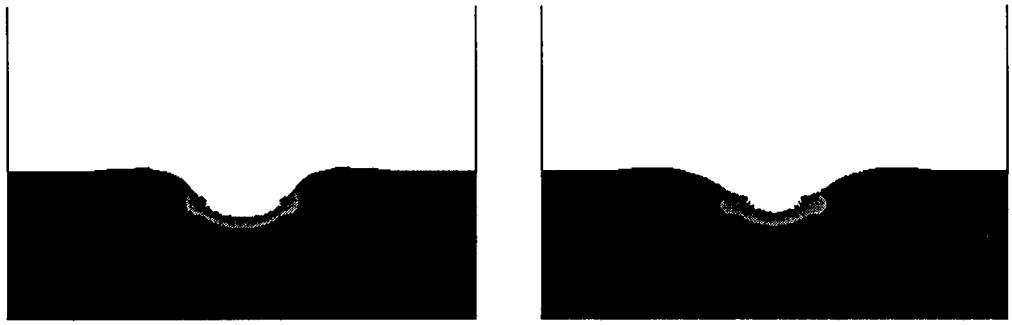


Fig. 10. Continued.

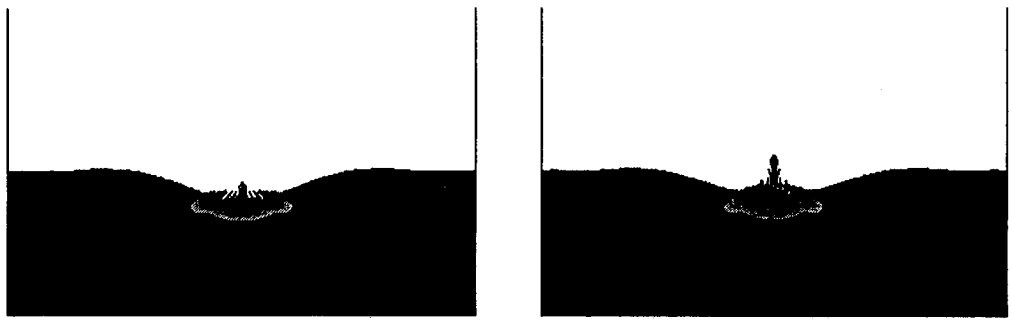


Fig. 10. Continued.

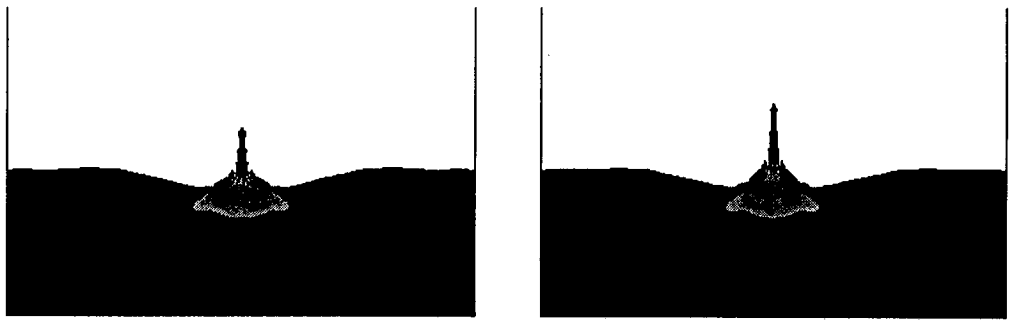


Fig. 10. Continued.

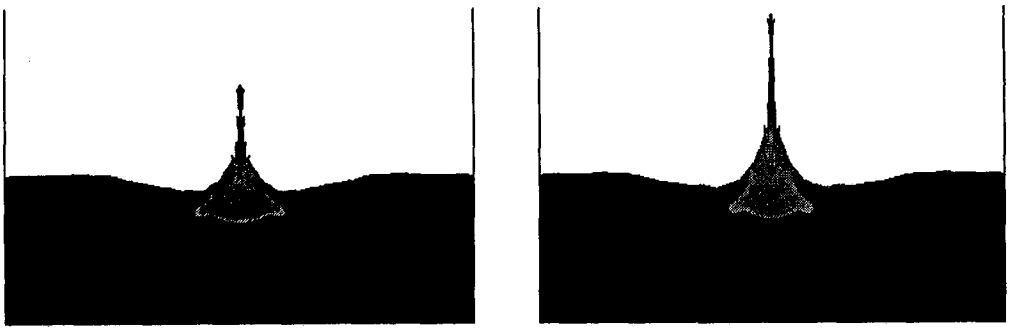


Fig. 10. Continued.

6.3 Simulaltion of Cavity Filling

To futher illustrate the applicability of this new technique implemented in GENS-MAC being able to handle complicated flow regimes we have applied the code to simulate the cavity filling of a "circular tub". This is a common problem encountered in the food industry and there is interest in predicting the flow behaviour during the filling stage. For circular tubs, a circular nozzle from which the fluid is injected into the tub is employed (see Fig. 11a). The flow is assumed to be axisymmetric and the flow domain is taken to be as shown in Fig. 11b.

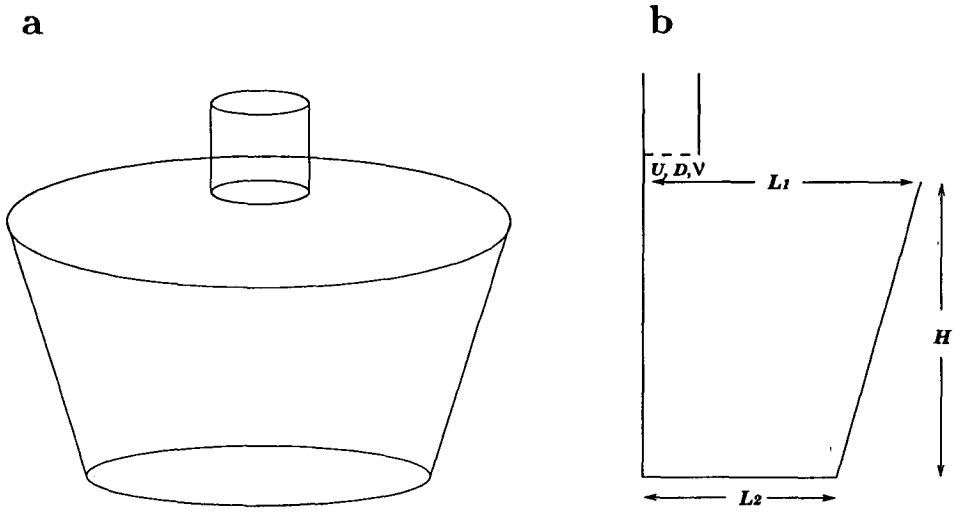


Fig. 11. The filling of a circular tub.

To simulate the filling of the tub we have run the code with the following input data

$$U = 1.0 \text{ ms}^{-1} \text{ (nozzle velocity)}, \; D = 16 \text{ mm} \text{ (nozzle diameter)}, \; \nu = 0.01 \text{ m}^2\text{s}$$

This gives a Reynolds number of $Re = UD/\nu = 1.6$ and $1/F_r^2 = 0.01$. The flow domain was defined by setting (see Fig. 9b) $L_1 = 5$ cm, $L_2 = 6$ cm and $H = 6$ cm. The nozzle is set at a distance of 1 cm above the tub. A mesh spacing of $\delta r =$

$\delta z = 1$ mm was employed, giving a mesh size of 50×80 cells. Thermal effects were neglected. Usually the fluid is non-Newtonian; the extension of this technique to non-Newtonian fluids (eg. power-law, Cross model) is under development. Figure 11 displays several snapshots of the flow configuration (with a mirror image to the left to give a complete picture) at different times.

To demonstrate the code can cope with complex geometries we have simulated the filling of the container shown in Fig. 12a. The data employed for this run were

$U = 1.0\ \mathrm{ms}^{-1}$ (nozzle velocity) , $D = 12$ mm (nozzle diameter) , $\nu = 0.001\ \mathrm{m}^2\mathrm{s}$

The results are shown in figure 12.

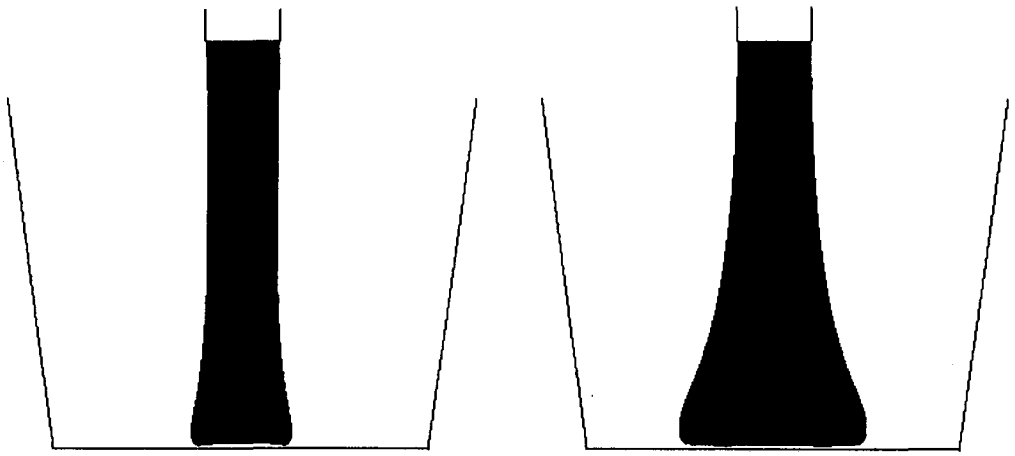


Fig. 11. The filling of a circular tub: Fluid plots at different times.

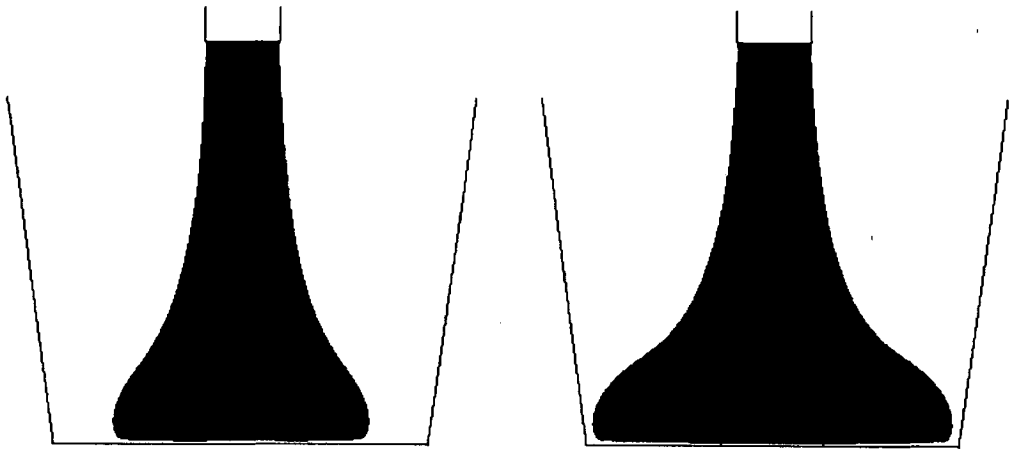


Fig. 11. Continued.

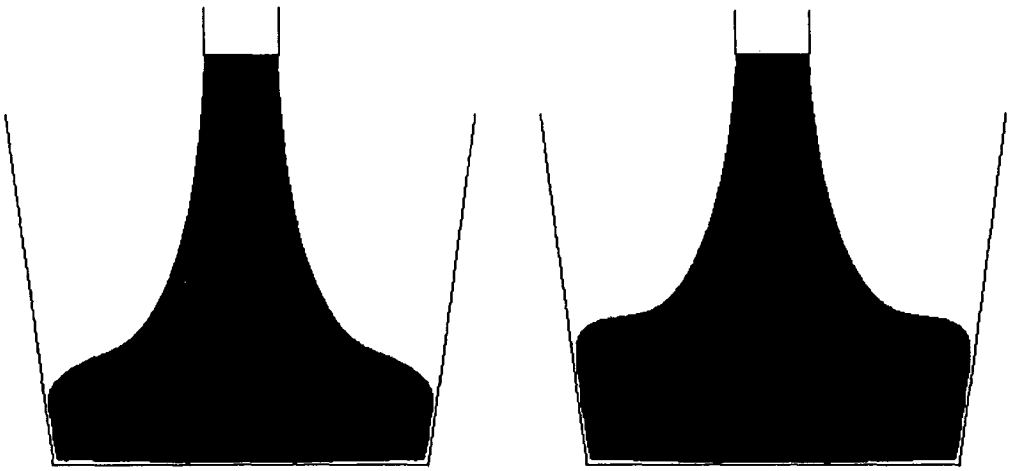


Fig. 11. Continued.

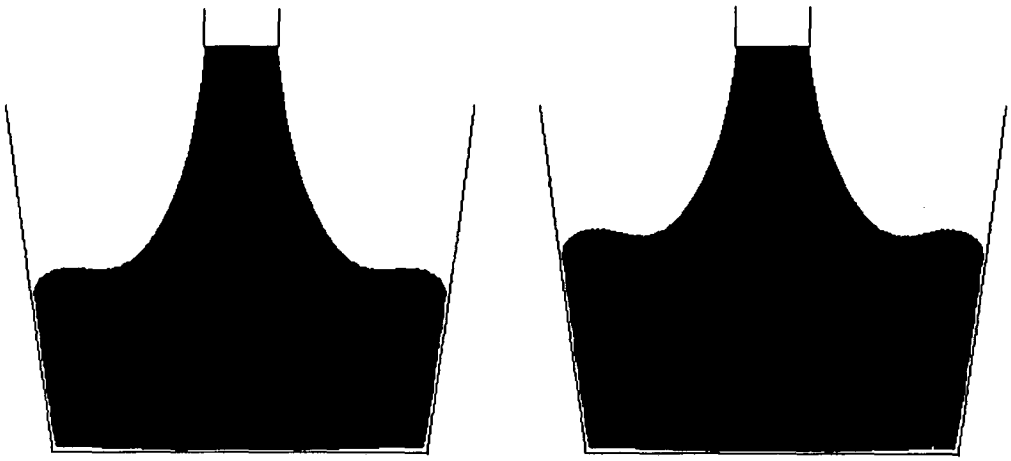


Fig. 11. Continued.

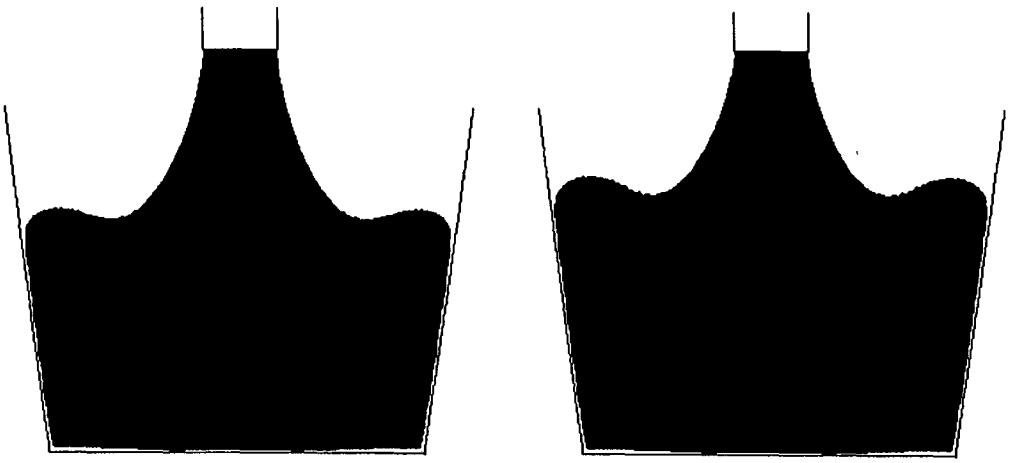


Fig. 11. Continued.

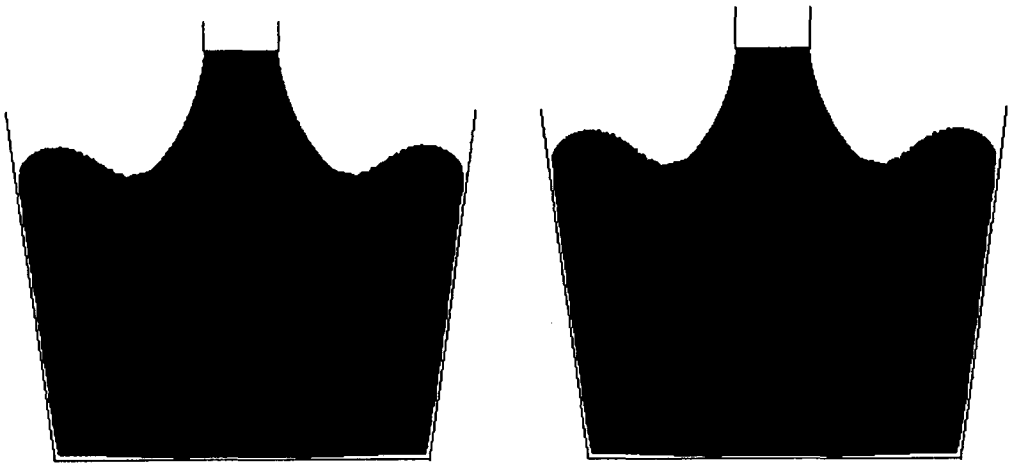


Fig. 11. Continued.

a

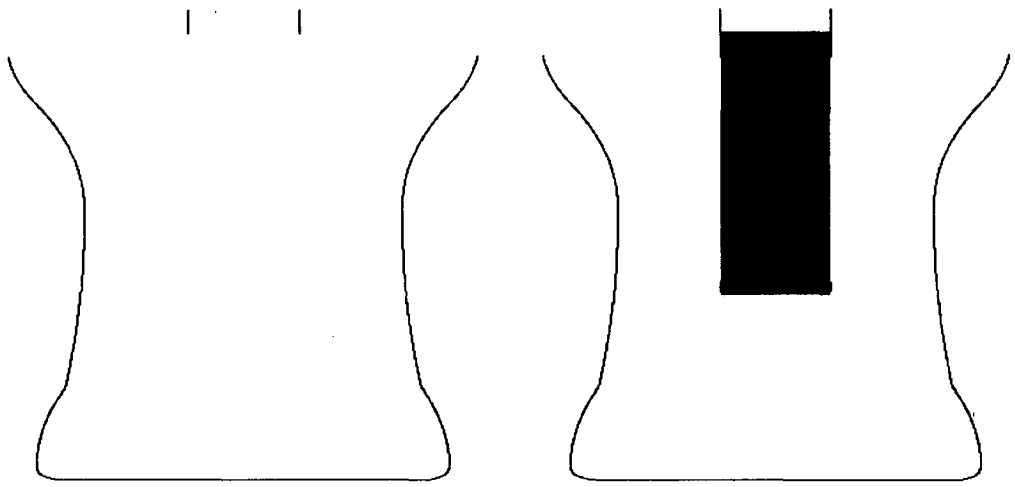


Fig. 12. The filling of a complex shaped container: Fluid plots at different times.

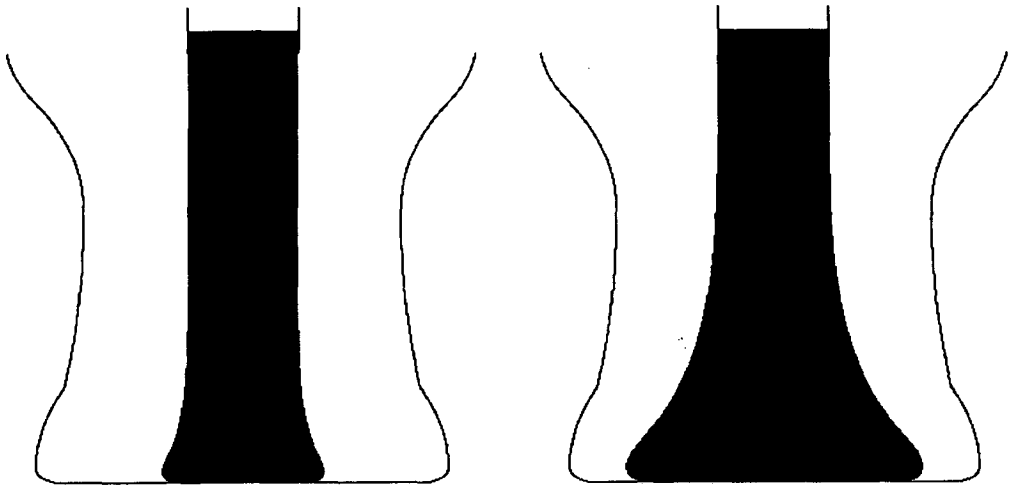


Fig. 12. Continued.

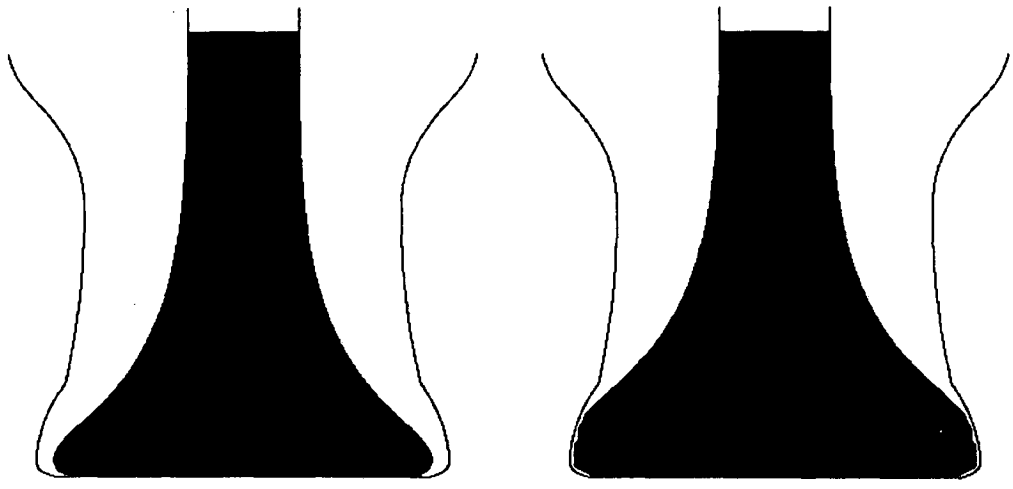


Fig. 12. Continued.

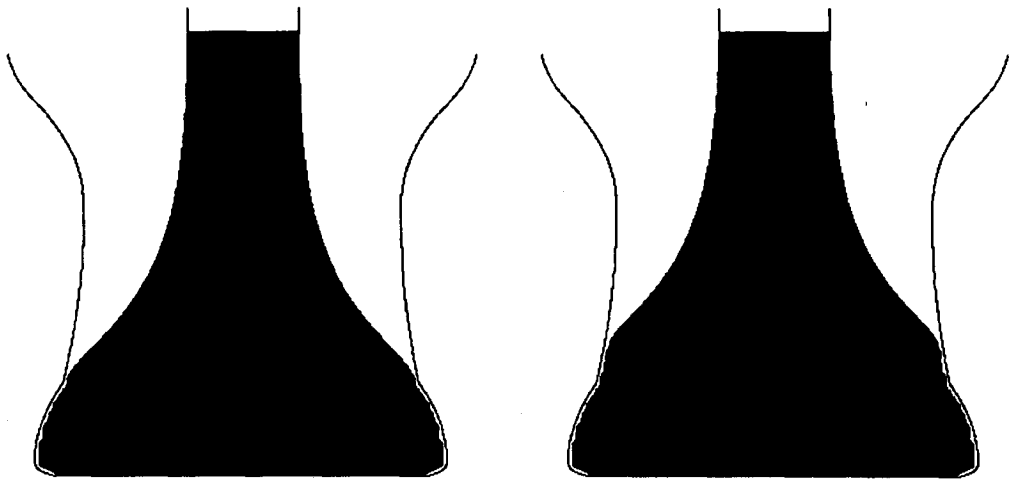


Fig. 12. Continued.

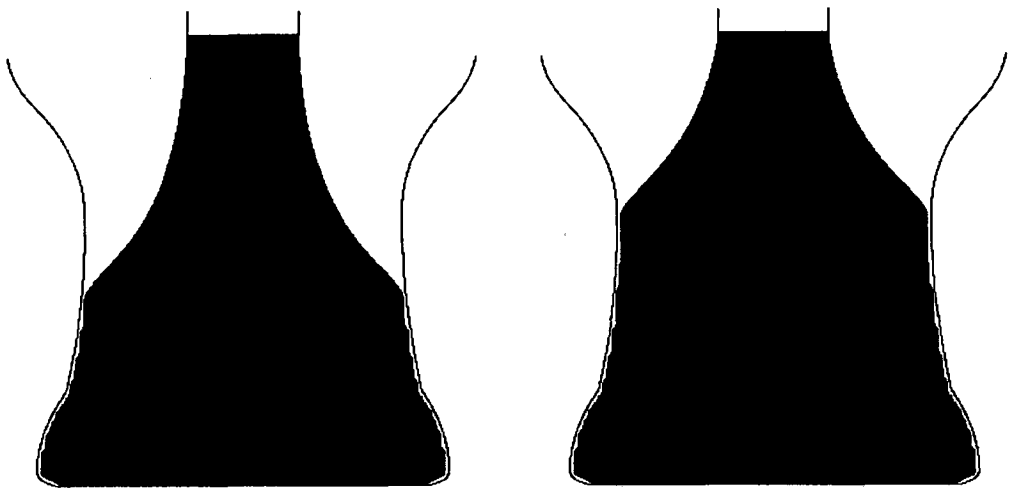


Fig. 12. Continued.

6.4 Three-dimensional Visualization of Axisymmetric Flows

The simulation data employed by the VisFreeFlow software for generating two-dimensional visualizations of fluid flow (see Sections 6.1, 6.2. and 6.3) have been used by the visualization package, VTK - The Visualization Toolkit - [11], to generate the three-dimensional visualizations presented in this Section. VTK provides a portable and extensible set of data formats and visualization algorithms to operate on data. The package can be tailored for user specific applications by building an adequate interface. Its design is Object-Oriented and it can be programmed using the languages C++ or Tcl/Tk.

By taking advantage of the axisymmetric nature of the free surface flows under study, 3D visualizations were generated through a rotational extrusion of the 2D data available. The 2D fluid flow simulation data were converted from their original format into a suitable VTK data format through a C program [12]. The VTK primitives used to describe the container, the ejector and the fluid flow at different time steps were polylines. The polylines describing the 2D profiles of the container, the ejector and the fluid itself at each instant of time were generated, as illustrated in figure 13. The profiles for both the splashing drop and the cavity filling simulations at different times were rotationally extruded around the z axis, generating the 3D visualizations shown in figures 14 and 15 respectively. Each picture displays the fluid flow configuration at a different time. The whole set of images can be recorded as a movie file for animation of the flow simulation. Details on the data conversion, file formats and visualization programs can be found in [12].

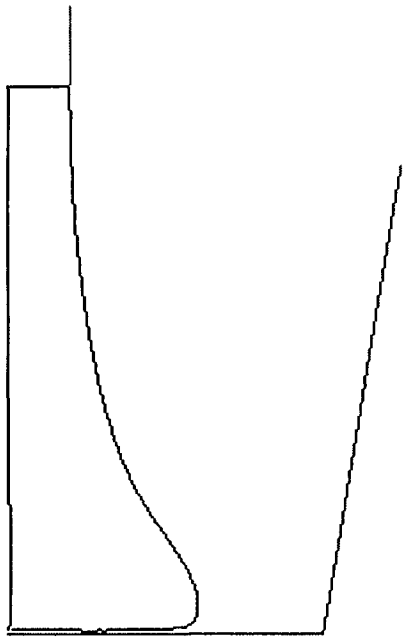


Fig. 13. Example of profiles for the cavity filling simulation.

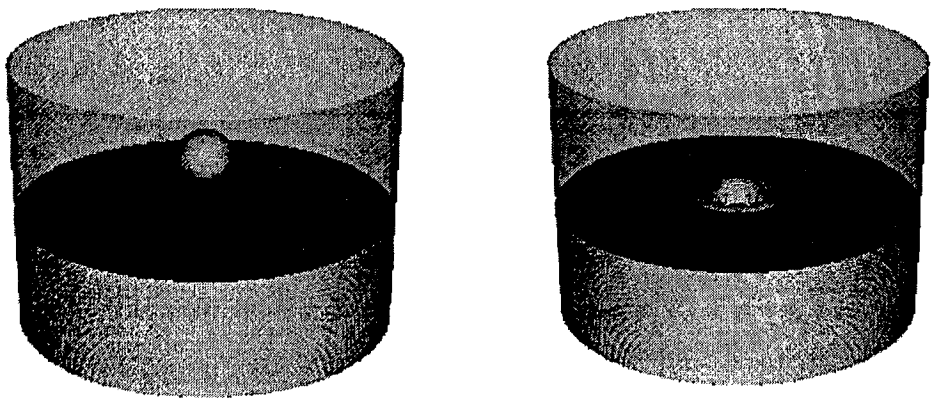


Fig. 14. Three-dimensional visualization of the splashing drop simulation.

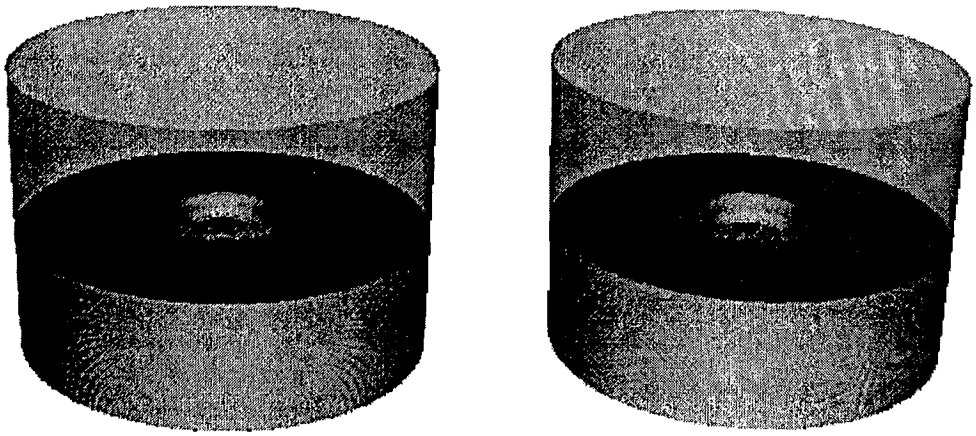


Fig. 14. Continued.

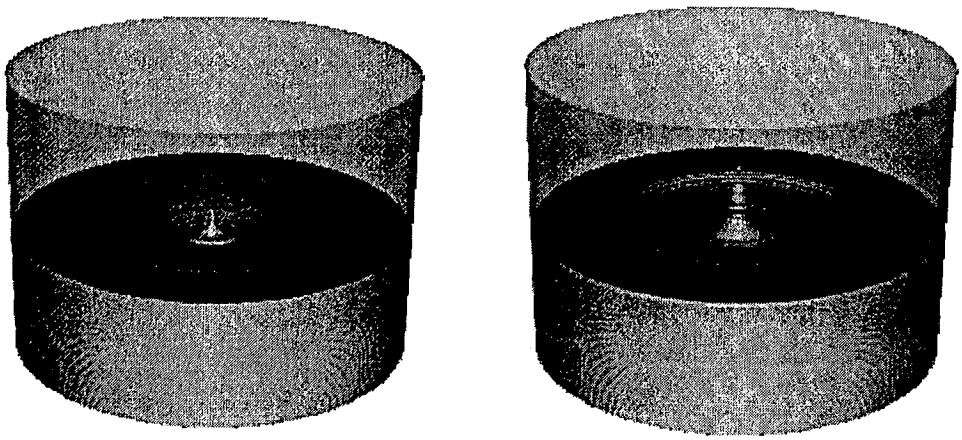


Fig. 14. Continued.

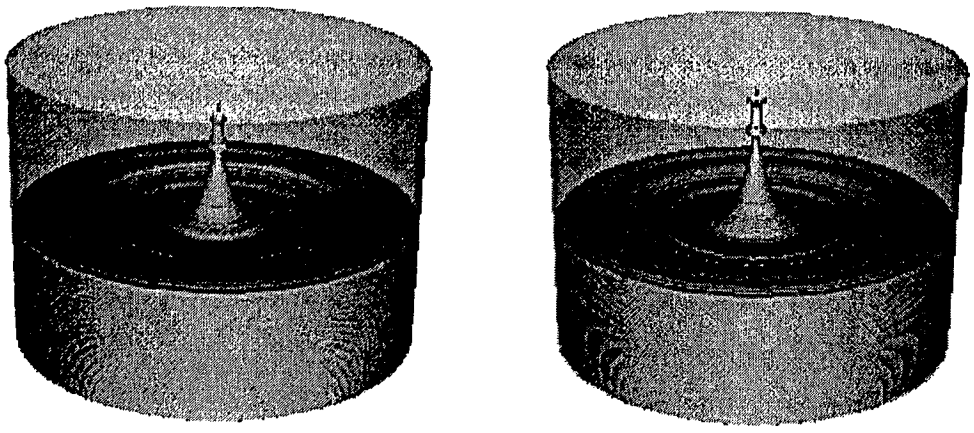


Fig. 14. Continued.

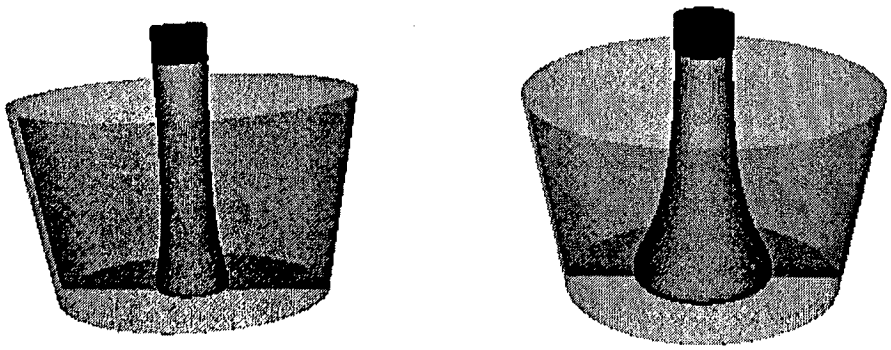


Fig. 15. Three-dimensional visualization of the circular tub simulation.

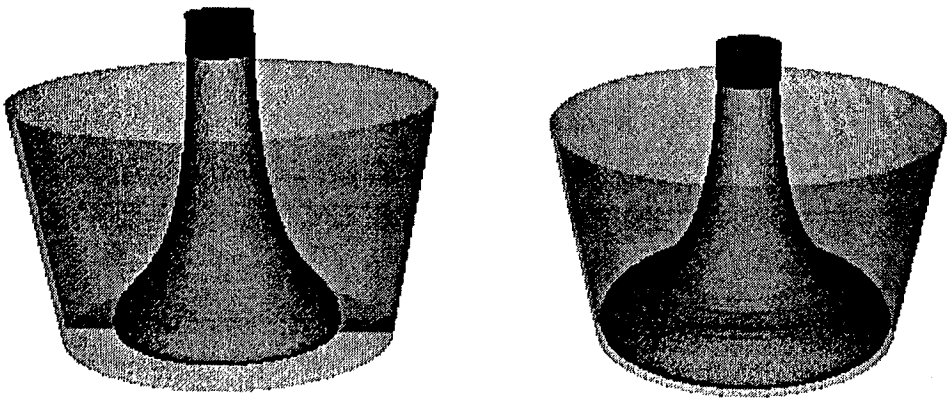


Fig. 15. Continued.

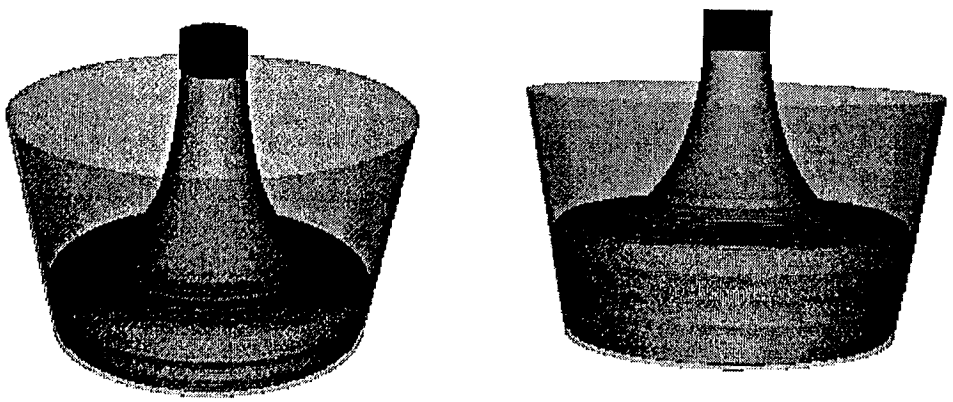


Fig. 15. Continued.

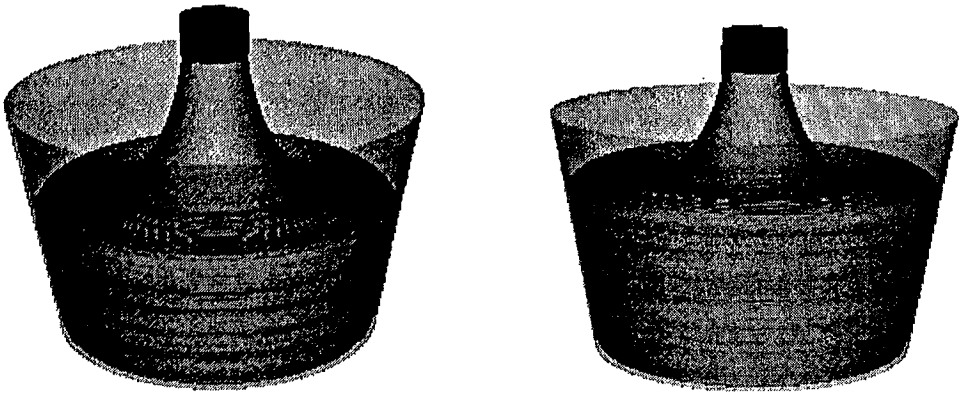


Fig. 15. Continued.

ACKNOWLEDGMENTS

We thank Dr. Saulo M. Barros of Department of Applied Mathematics of IME (Institute of Mathematics and Statistics) of University of São Paulo for useful discussions on solving the Poisson equation using cylindrical coordinates. We would also like to thank the staff of the lcad–USP de São Carlos (high performance computing laboratory) for their support to this project. This work has been funded by CNPq – Conselho Nacional de Pesquisa and FAPESP – Fundação de Amparo a Pesquisa do Estado de São Paulo.

References

[1] Harlow, F. and Welsh, J. E., 1965, "Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid with a Free Surface", *Physics of Fluids*, Vol. 8, pp. 2182-2193.

[2] Viecelli, J. A., 1971, "A Computing Method for Incompressible Flows Bounded by Moving Walls", *Journal of Computational Physics*, Vol. 8, pp. 119-143

[3] Amsden,A. A. and Harlow, F. H., 1971, "The SMAC Method: A Numerical Technique for Calculating Incompressible Fluid Flow", Los Alamos Scientific Lab., Report LA-4370, Los Alamos, New Mexico.

[4] Miyata, H., 1986, "Finite Difference Simulation of Breaking Waves", *Journal of Computational Physics*, Vol. 65, pp. 179-214.

[5] Hirt, C. W. and Shannon, J. P., 1968, "Free-Surface Stress Conditions for Incompressible-Flow Calculations", *Journal of Computational Physics*, Vol. 2, pp. 403-411.

[6] Tome, M. F., 1993, GENSMAC: A Multiple Free Surface Fluid Flow Solver, Ph.D. Thesis, University of Strathclyde, Glasgow, U.K.

[7] Tome, M. F. and McKee, S., 1994, "GENSMAC: A Computational Marker-and-Cell Method for Free Surface Flows in General Domains", *Journal of Computational Physics*, Vol. 110, pp. 171-186.

[8] Tome, M. F., Duffy, B. and McKee, S., 1996, "A Numerical Technique for Solving Unsteady non-Newtonian Free Surface Flows", *Journal of Non-Newtonian Fluid Mechanics*, Vol. 62, pp. 9-34

[9] Bachelor, G. K., 1967, An Introduction to Fluid Dynamics,(Cambridge University Press, Cambridge).

[10] Taylor, G., 1967, Film Notes for Low-Reynolds Number Flows, Cambridge University.

[11] W. Schroeder, K. Martin and B. Lorensen, 1996, The Visualization Toolkit - An Object-Oriented Approach to 3D Graphics, Prentice-Hall, New Jersey.

[12] R. Minghim, M. C. F. Oliveira, 1996, Three-dimensional Visualization of Free Surface Flows - Internal Report - ICMSC - USP

Address for correspondence:

M. F. Tome
Department of Computer Sciences
ICMSC - USP - P. O. Box 668
13560-970 São Carlos - SP
Brazil

NOTAS DO ICMSC

SÉRIE COMPUTAÇÃO

029/96 TOMÉ, M.F.; CASTELO FILHO, A.; CUMINATO, J.A.; McKEE, S. - GENSMAC3D: Implementation of the Navier-Stokes equations and Boun dary conditions for 3D free surface flows.

028/96 MARTINS, TERESA B.F.; GHIRALDELO, CLAUDETE M.; OLIVEIRA JR., O.N. - Readability formulas applied to textbooks in brazilian portuguese.

027/96 ALUISIO, SANDRA M.; OLIVEIRA, MARIA C..F. DE; NETO, ALVARO GARCIA; MASIERO, PAULO C.; OLIVEIRA JR., OSVALDO N. - Writing tools and a software architecture to assist writing in a foreing language.

026/96 ALUISIO, SANDRA M.; OLIVEIRA JR., OSVALDO N. - A detailed schematic structure of research paper introductions: an application in support-writing tools.

025/96 NUNES, M.G.V.; HASEGAWA,R.; KAWAMOTO,S.; OLIVEIRA, M.C.F. DE; TURINE, M.A.S.; GHIRALDELO, C. M.; OLIVEIRA JR., O.N.; RIOLFI, C.R.; SIKANSKI, N.S.; MARTINS, T.B. - Style and grammar checkers for brazilian portuguese.

024/96 FORTES, R.P.M. - Uma ferramenta orientada a links para avaliação de hiperdocumentos.

023/96 CANSIAN, A.M.; MOREIRA, E.S.; MAURO, R.B.; MORISHITA, F.T.; CARVALHO, A.C.P.L.F. - Um sistema adaptativo de detecção de instrusão em redes de computadores.

022/96 BRIGANTE,W.J.; MOREIRA, E.S. - Utilização de monitores OLTP no gerenciamento de ambientes de manufatura integrada voltados à produção discreta/

021/95 BEZERRA, L.A.F.; SANTANA, R.H.C.; SANTANA, M.J. - Sistema auxiliar de arquivos baseado em disco WORM para ambientar computacional distribuído.

020/95 NUNES, M.G.V.; HASEGAWA, R., - PROTEMA: intelligent tutoring systems for mathematics.