University of Thi-Qar Journal Vol.10 No.3 SEP 2015
Web Site: https://jutq.utq.edu.iq/index.php/main   Email: journal@jutq.utq.edu.iq

A Computational Fluid Dynamics Model for Transient Two-Dimensional Free Surface Flows

https://doi.org/10.32792/utq/utj/vol10/3/9

Hussam A. Khalaf [1] , Mohanned S. Ali [2] , and Hussein H. Mosa [1]

[1] University of Thi-Qar, Research and Development Department

[2] University of Thi-Qar,, College of Engineering

Abstract

This work presents a numerical technique for simulating free surface with and without bottom obstacles. The numerical method solves the governing equations using the finite difference method on a staggered grid and uses numerical computations for the analysis of Dam-Break Flow using two-dimensional flow equations in a vertical plane. The numerical model uses the general approach of the simplified marker and cell method combined with the volume of fluid approach for the surface tracking.

The variable models were based on a computational fluid dynamics technique, called the SOLA-VOF scheme, which possesses the capability of treating transient fluid flow problems with evolution of free boundaries.

الملخص

هذا العملِ يُقدّمُ تكنيك عددي لمحاكاة تدفقِ السطح الحرِ بوجود وعدم وجود الحواجز السفلية. الطريقةُ العدديةُ حلت المعادلاتُ الحَاكِمةُ باستخدامَ طريقةَ الفروق المحدودةِ للشبكة التراكبية وطبقت الحساباتَ العدديةَ لتحليلِ تدفقِ إنهيار السدِّ، ثنائي البعد في مستوي عمودي. استعملُ النموذجُ العدديُ، الاسلوب العامّ لطريقة العلامةِ والخلية المُبَسَّطةِ المندمجة او المتركبة مع اسلوب او طريقة حجم السائل للتتبع السطحي.

الموديلات اعتمدت على اساس التكنيك الحسابي لديناميكا السوائل، المسمى SOLA-VOF ، الذي يَمتلكُ قابليةَ مُعَالَجَة مشاكلِ التدفقِ السائلةِ العابرةِ مع تطورِ الحدودِ الحرة.

Introduction

One of the important subjects that have been considered by fluid mechanics researchers is studying the interfacial flows. Theoretical studies of two phase viscous flows involving free surface instabilities or very strong interface tearing and stretching are difficult to perform. Numerical methods have begun to be used to simulate the flow dynamics of the problem. The numerical methods can be divided into two groups depending on the type of grids used: moving grid or fixed grid, Renardy Y. and M. Renardy (2002) and Gerlach, D. *et al.* (2006). Two important approaches of fixed-grid methods, namely the volume-of-fluid and the level-set approaches, are among the most commonly used methods. The volume-of-fluid method, Renardy Y. and M. Renardy (2002), tracks the volume of each fluid in all cells containing portions of the interface, rather than the interface itself. The VOF method solves a non-diffusive solution of the advection equation, by a geometrically based calculation technique of the void fraction fluxes at the cell faces based on the reconstructed interface.

The objectives of the present study are to use the VOF based numerical method that we have recently developed for calculating interface of the two fluids. The major incentive for using the VOF method is that the types of problems that can be solved involve highly complex free surface flows. Reasonable accuracy is attainable and yet the method is relatively simply implemented. The basic algorithm is available in a two-fluid code called SOLA-VOF and part of our work has been devoted to modifying this algorithm to adapt it to the problems of interest, which involve transient free surface flows with two immiscible fluids.

The purpose in the present study was to investigate the dam-break flow numerically with and without bottom obstacles in the tank.

Governing Equations

The mass, momentum and energy conservation equations that govern fluid flow can be expressed in vector calculus notation, as it follows:

a) Mass conservation equation (continuity) for incompressible fluids, Zohdi T. (2007):

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \vec{v} = 0 \qquad \ldots\ldots\ldots (1)$$

and if □ is assumed to be constant then

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\xi u}{x} = 0 \qquad \ldots\ldots\ldots (2)$$

b) The momentum conservation equation (Navier–Stokes) can be expressed as Zohdi T. (2007):

$$\rho \left\{ \frac{\partial \vec{v}}{\partial t} + \vec{v} \cdot \nabla \vec{v} \right\} = -\nabla P + \mu \nabla^2 \vec{v} + \rho \vec{g} \qquad \ldots\ldots\ldots (3)$$

and if fluid viscosity and density were assumed constant, in x-direction Eq. (3) can be simplified to

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + g_x + \gamma\left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \xi\left(\frac{1}{x}\frac{\partial u}{\partial x} - \frac{u}{x^2}\right)\right] \qquad \ldots\ldots\ldots (4)$$

and in y-direction

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial y} + g_y + \gamma\left[\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\xi}{x}\frac{\partial v}{\partial x}\right] \qquad \ldots\ldots\ldots (5)$$

Two dimensional flow is assumed in either plane geometry (x,y) or cylindrical geometry (r,z). The parameter value $\xi = 0$ is the setting for the plane case, while $\xi = 1$ is the setting for cylindrical coordinates (i.e., axisymmetric flow).

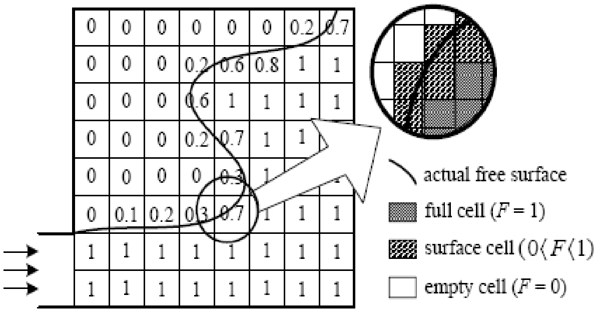
Volume of Fluid Method (VOF)

Many reconstruction algorithms use the fraction of cell volume occupied by one of the fluids as the marker quantity. If this fraction is 0 for a given cell, the fluid does not occupy the cell and there is no interface in that cell. Conversely, if die fraction is 1, the cell is completely occupied by the fluid and again there is no interface present. An interface is to be constructed only if the fraction is between 0 and 1. Since there is only one piece of information regarding the interface per cell available, certain arbitrariness is reconstructing the shape of the interface has to be allowed. The accuracy with which the reconstructed interfaces approximate the real interface is difficult to judge and different types of distortions are possible (Ban and Ashurst, 1984). The VOF method of Hirt, C. W. and Nichols, B. D., (1981) defined a function F(x,y.t) that is equal to unity at any point occupied by fluid and zero elsewhere. When averaged over the cells of a computational mesh, the average value of F in a cell is equal to the fractional volume of the cell occupied by fluid. In particular, a unit value of F corresponds to a cell full of fluid, whereas a zero value indicates that the cell contains no fluid. Cells with F values between zero and one contain a tree surface (Fig.1). The VOF method requires only one storage for each mesh cell, which is consistent with the storage requirements for all other dependent variables.

In addition to defining which cells contain a boundary, the F function can be used to define where fluid is located in a boundary cell. The normal direction to the boundary lies in the direction in which the value of F changes most rapidly.

The time dependence of F is governed by the equation,

$$\frac{\partial F}{\partial t}+u\frac{\partial F}{\partial x}+v\frac{\partial F}{\partial y}=0 \qquad \dots\dots\dots \ (6)$$



0	0	0	0	0	0	0.2	0.7
0	0	0	0.2	0.6	0.8	1	1
0	0	0	0.6	1	1	1	1
0	0	0	0.2	0.7	1	1	[illegible]
0	0	0	0	0.3	1	[illegible]	1
0	0.1	0.2	0.3	0.7	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1

Figure 1: Numerical methods for volume-of-fluid method

Computational Treatment

The computational treatment of the governing equations is documented based on the SOLA-VOF algorithm. We use an Eulerian mesh of rectangular cells having variable sizes, δx_i for the *i*th column and δy_j for the *j*th row, as shown in Fig. 2. The governing and boundary equations are discretized by using a finite difference scheme on a staggered grid.

The computational mesh is a two-dimensional orthogonal mesh representing either Cartesian or cylindrical coordinates. For simplicity in the form of the equations, the cylindrical coordinate system is modified to use x=r, y=rmax Θ and z=z as the mapping between the coordinate systems with the terms in the continuity equation, the Navier–Stokes equation, and the F-convection equation modified accordingly.

Solution is accomplished on a "staggered" grid where scalar quantities, such as the "fullness" function, F, and the pressure, p, are located at the computational cell centers and vector components such as the velocity components u and v are located on the cell faces as shown in Fig. 3 for a two-dimensional mesh.

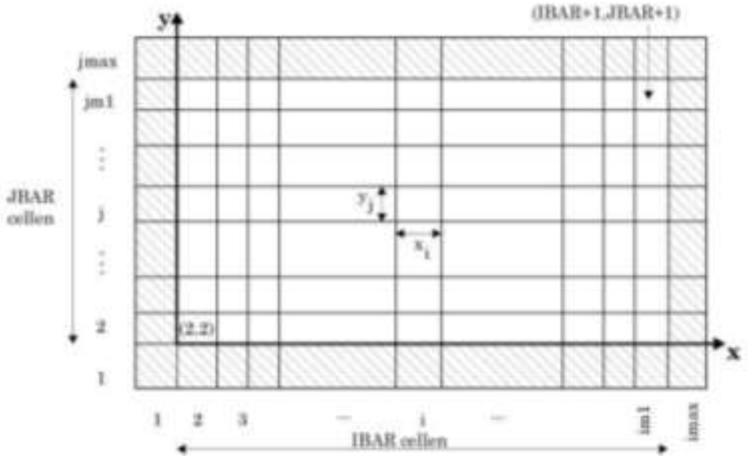


Figure 2: A finite difference mesh with variable rectangular cells.

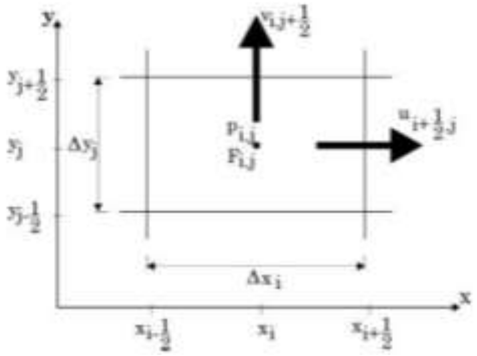


Figure 3: Diagram of a two-dimensional computational cell.

Finite-difference approximations to momentum conservation equations

A standard finite-difference approximation to Eqs. (4) and (5) in MAC-type methods is expressed as:

$$u^{n+1}_{i+1/2,j} = u^{n}_{i+1/2,j} + \frac{\Delta t}{\rho}\left(\frac{\partial p}{\partial x}\right)^{n+1} + \Delta t\left(-FUX - FUY + g_x + VISX\right)^n \qquad \ldots\ldots\ldots (7)$$

$$v^{n+1}_{i,j+1/2} = v^{n}_{i,j+1/2} + \frac{\Delta t}{\rho}\left(\frac{\partial p}{\partial y}\right)^{n+1} + \Delta t\left(-FVX - FVY + g_y + VISY\right)^n \qquad \ldots\ldots\ldots (8)$$

Following Hirt, C. W. and Nichols, B. D., (1981), the convective terms in Eqs. (7) and (8) are discretized using a combination of first-order donor-cell and centered-difference approximations. The expression for FUX is then given by

$$FUX = \frac{u^n_{i+1/2,j}}{\Delta x_{ax}}\left[\Delta x_{i+1} DUL + \Delta x_i DUR + \alpha\,\mathrm{sgn}\left(u^n_{i+1/2,j}\right)\left(\Delta x_{i+1} DUL - \Delta x_i DUR\right)\right] \qquad \ldots\ldots\ldots (9)$$

where

$$DUR = \left(u^n_{i+3/2,j} - u^n_{i+1/2,j}\right)/\,\Delta x_{i+1}\ ,\ DUL = \left(u^n_{i+1/2,j} - u^n_{i-1/2,j}\right)/\,\Delta x_i$$

$$\Delta x_{\alpha x} = \Delta x_i + \Delta x_{i+1} + \alpha\,\mathrm{sgn}\left(u^n_{i+1/2,j}\right)\left(\Delta x_{i+1} - \Delta x_i\right)$$

When α=0, expression (9) reduces to the second- order accurate centered-difference approximation. For α=1, the first-order donor-cell form is recovered. The expressions for FUY is given by

$$FUY = \frac{v^n_{i+1/2,j}}{\Delta y_{ax}}\left[\Delta y_{j+1/2}DUB + \Delta y_{j-1/2}DUF + \alpha\,\mathrm{sgn}\left(v^n_{u_{i+1/2,j}}\right)\left(\Delta y_{j+1/2}DUB - \Delta y_{j-1/2}DUF\right)\right]$$

$$\ldots\ \ldots\ \ldots\ (10)$$

where

$$DUF = \left(u^n_{i+1/2,j+1} - u^n_{i+1/2,j}\right)/\,\Delta y_{j+1/2}\ ,\ DUB = \left(u^n_{u_{i+1/2,j}} - u^n_{u_{i+1/2,j-1}}\right)/\,\Delta y_{j-1/2}$$

$$\Delta y_{\alpha x} = \Delta y_{j+1/2} + \Delta y_{j-1/2} + \alpha\,\mathrm{sgn}\left(v^n_{i+1/2,j}\right)\left(\Delta y_{j+1/2} - \Delta y_{j-1/2}\right),$$

Likewise, the approximations for the convective accelerations in the y-

$$FVX = \frac{u^n_{i,j+1/2}}{\Delta x_{ay}}\left[\Delta x_{i+1/2}DVL + \Delta x_{i-1/2}DVR + \alpha\,\mathrm{sgn}\left(u^n_{i,j+1/2}\right)\left(\Delta x_{i+1/2}DVL - \Delta x_{i-1/2}DVR\right)\right] \ldots (11)$$

$$FVY = \frac{v^n_{i,j+1/2}}{\Delta y_{ay}}\left[\Delta y_{j+1}DVB + \Delta y_j DVF + \alpha\,\mathrm{sgn}\left(v^n_{i,j+1/2}\right)\left(\Delta y_{j+1}DVB - \Delta y_j DVF\right)\right] \quad \ldots\ \ldots\ \ldots\ (12)$$

where

$$DVR = \left(v^n_{i+1,j+1/2} - v^n_{i,j+1/2}\right)/\,\Delta x_{i+1/2}\ ,\ DVL = \left(v^n_{i,j+1/2} - v^n_{i-1,j+1/2}\right)/\,\Delta x_{i-1/2}$$

$$DVF = \left(v^n_{i,j+3/2} - v^n_{i,j+1/2}\right)/\,\Delta y_{j+1}\ ,\ DVB = \left(v^n_{i,j+1/2} - v^n_{i,j-1/2}\right)/\,\Delta y_j$$

$$\Delta x_{\alpha y} = \Delta x_{i+1/2} + \Delta x_{i-1/2} + \alpha\,\mathrm{sgn}\left(u^n_{i,j+1/2}\right)\left(\Delta x_{i+1/2} - \Delta x_{i-1/2}\right)$$

$$\Delta y_{\alpha y} = \Delta y_{j+1} + \Delta y_j + \alpha\,\mathrm{sgn}\left(v^n_{i,j+1/2}\right)\left(\Delta y_{j+1} - \Delta y_j\right)$$

Thus, according to Lemos (1994), the following discretization of the viscous term in Eqs. (7) and (8) was adopted:

$$VISX = \left[v + (1-\alpha)\frac{\Delta t}{2}\left(u^n_{i+1/2,j}\right)^2\right]\left[\frac{DUR - DUL}{\Delta x_{i+1/2}}\right]$$

$$\left[v + (1-\alpha)\frac{\Delta t}{2}\left(u^n_{i+1/2,j}\right)^2\right]\left[\frac{DUF - DUB}{\Delta y_j}\right] \quad \ldots\ \ldots\ \ldots\ (13)$$

$$VISY = \left[v + (1-\alpha)\frac{\Delta t}{2}\left(v^n_{i,j+1/2}\right)^2\right]\left[\frac{DVR - DVL}{\Delta x_i}\right]$$

$$\left[v + (1-\alpha)\frac{\Delta t}{2}\left(v^n_{i,j+1/2}\right)^2\right]\left[\frac{DVF - DVB}{\Delta y_{j+1/2}}\right] \quad \ldots\ \ldots\ \ldots\ (14)$$

Finite-difference approximations to continuity equation

Velocities computed from Eqs. (7) and (8) in general will not satisfy the continuity equation because p^{n+1} is not available. To satisfy the continuity equation and to determine the correct pressure, values of pressures and velocities must be adjusted in each cell that is occupied by fluid. In a full cell, pressure is changed in such a way that the divergence $D_{i,j}$ left by the first step is driven to zero, in a free-surface cell, the cell pressure may be determined in such a way that a linear interpolation between the pressure in the surface and adjacent full cell yields the wanted value p_s (usually zero) at the free-surface location. In both cases, the velocities located on the sides of the cell are simultaneously adjusted, in response to the pressure change in the cell.

The pressure in a full cell is split into an old time-level component and a correction such that

$$\Delta p_{i,j} = p^{n+1}_{i,j} - p^{n}_{i,j} \qquad \dots\dots\dots (15)$$

Then we can work out the pressure derivatives:

$$\left(\frac{\partial p}{\partial x}\right)^{n+1} = \frac{\partial}{\partial x}\left(p^{n+1} - p^{n} + p^{n}\right) = \frac{\partial \Delta p}{\partial x} + \left(\frac{\partial p}{\partial x}\right)^{n}$$

$$\left(\frac{\partial p}{\partial y}\right)^{n+1} = \frac{\partial}{\partial y}\left(p^{n+1} - p^{n} + p^{n}\right) = \frac{\partial \Delta p}{\partial y} + \left(\frac{\partial p}{\partial y}\right)^{n}$$

Substituting these in the momentum equations gives:

$$u^{n+1}_{i+1/2,j} = u^{n}_{i+1/2,j} + \frac{\Delta t}{\rho}\frac{\partial \Delta p}{\partial x} + \Delta t\left(\frac{1}{\rho}\frac{\partial p}{\partial x} - FUX - FUY + g_x + VISX\right)^{n} \qquad \dots\dots\dots (16)$$

$$v^{n+1}_{i,j+1/2} = v^{n}_{i,j+1/2} + \frac{\Delta t}{\rho}\frac{\partial \Delta p}{\partial y} + \Delta t\left(\frac{1}{\rho}\frac{\partial p}{\partial y} - FVX - FVY + g_y + VISY\right)^{n} \qquad \dots\dots\dots (17)$$

Let us define the following quantities:

$$D^{n} = \left[\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right]^{n}$$

$$Q^{n}_{x} = \left[\frac{1}{\rho}\frac{\partial p}{\partial x} - FUX - FUY + g_x + VISX\right]^{n}, \quad Q^{n}_{y} = \left[\frac{1}{\rho}\frac{\partial p}{\partial y} - FVX - FVY + g_y + VISY\right]^{n}$$

Then the momentum equations are written as:

$$u^{n+1}_{i+1/2,j} = u^{n}_{i+1/2,j} - \frac{\Delta t}{\rho}\frac{\partial \Delta p}{\partial x} + Q^{n}_{x} \qquad \dots\dots\dots (18)$$

$$v^{n+1}_{i,j+1/2} = v^{n}_{i,j+1/2} + \frac{\Delta t}{\rho}\frac{\partial \Delta p}{\partial y} + Q^{n}_{y} \qquad \dots\dots\dots (19)$$

The iterative method starts with calculating a first estimate of the velocities with a fully explicit guess ($\Delta p^{(1)} = 0$):

$$u^{(1)}_{i+1/2,j} = u^{n}_{i+1/2,j} + Q^{n}_{x} \qquad \dots\dots\dots (20)$$

$$v^{(1)}_{i,j+1/2} = v^{n}_{i,j+1/2} + Q^{n}_{y} \qquad \dots\dots\dots (21)$$

For an improved guess the pressure correction $\Delta p^{(2)}$ should be included such that

$$u^{(2)}_{i+1/2,j} = u^{(1)}_{i+1/2,j} - \frac{\Delta t}{\rho}\left(\frac{\partial \Delta p}{\partial x}\right)^{(2)} \qquad \dots\dots\dots (22)$$

$$v^{(2)}_{i,j+1/2} = v^{(1)}_{i,j+1/2} - \frac{\Delta t}{\rho}\left(\frac{\partial \Delta p}{\partial y}\right)^{(2)} \qquad \dots\dots\dots (23)$$

Substitute these into the continuity equation. Then, a form of Poisson equation results:

$$\nabla^{2}(\Delta p)^{(2)} = \frac{\rho D^{(1)}}{\Delta t} \qquad \dots\dots\dots (24)$$

This equation can be solved for $\Delta p^{(2)}$. For this purpose we used finite differences taking into consideration the variable mesh to obtain:

$$D^{(1)}_{i,j} = \frac{\Delta t}{\rho}\left[\frac{1}{\Delta x_i}\left(\frac{\Delta p^{(2)}_{i+1,j} - \Delta p^{(2)}_{i,j}}{\Delta x_{i+1/2,j}} - \frac{\Delta p^{(2)}_{i,j} - \Delta p^{(2)}_{i-1,j}}{\Delta x_{i-1/2,j}}\right) + \frac{1}{\Delta y_j}\left(\frac{\Delta p^{(2)}_{i,j+1} - \Delta p^{(2)}_{i,j}}{\Delta y_{i,j+1/2}} - \frac{\Delta p^{(2)}_{i,j} - \Delta p^{(2)}_{i,j-1}}{\Delta y_{i,j-1/2}}\right)\right]$$

The pressure correction $\Delta p^{(2)}$ is now computed from the requirement $D_{i,j}^{(2)}=0$ such that

$$\Delta p_{i,j}^{(2)}=-\frac{S}{(\partial S/\partial p)} \qquad \dots\dots\dots (25)$$

with $S=D_{i,j}^{(1)}=\dfrac{\Theta_{i+1/2,j}u_{i+1/2,j}^{(1)}-\Theta_{i-1/2,j}u_{i-1/2,j}^{(1)}}{\Theta_{i,j}\Delta x_i}+\dfrac{\Theta_{i,j+1/2}v_{i,j+1/2}^{(1)}-\Theta_{i,j-1/2}v_{i,j-1/2}^{(1)}}{\Theta_{i,j}\Delta y_j}$

$$\frac{\partial S}{\partial p}=\frac{\Delta t}{\rho\Theta_{i,j}}\left[\frac{1}{\Delta x_i}\left(\frac{\Theta_{i+1/2,j}}{\Delta x_{i+1/2}}+\frac{\Theta_{i-1/2,j}}{\Delta x_{i-1/2}}\right)+\frac{1}{\Delta y_i}\left(\frac{\Theta_{i,j+1/2}}{\Delta y_{j+1/2}}+\frac{\Theta_{i,j-1/2}}{\Delta y_{j-1/2}}\right)\right]$$

Convergence of Eq. (25), which is a variant of the Newton-Raphson relaxation technique, can be accelerated if $D_{i,j}^{(1)}$ is multiplied by an over-relaxation factor ω such that 1 ≤ω≤ 2 . An optimum value of ω is often equal to 1.7; an unstable iteration results if ω exceeds 2.

In order to deal with a free surface, we need an additional procedure because the location of the free surface is unknown a priori. The procedure adopted herein is described in brief as follows (refer to Fig. 4). The surface cell pressure $p_{i,j}$ may be determined by a linear interpolation (or extrapolation) between the surface pressure, p_s, and pressure, p_N, inside the fluid. Namely,

$$p_{i,j}^{n+1}=(1-\xi)p_N+\xi p_s \qquad \dots\dots\dots (26)$$

where ζ= d_c/d is the ratio of the distance between the cell centers to the distance between the free surface and the center of the neighbor interpolation cell. When the surface tension effect is neglected, p_s can be set zero. Equation (25) can be used to compute the pressure correction for a surface cell, provided S is replaced by

$$S=(1-\xi)p_N+\xi p_s-p_{i,j} \qquad \dots\dots\dots (27)$$

After the pressure correction is found from Eq. (25), neighbor velocities are updated using Eqs. (22) and (23). The pressure correction is always computed using the most up-to-date velocities.

$$p_{i,j}^{(2)}=p_{i,j}^{n}+\Delta p_{i,j}^{(2)} \qquad \dots\dots\dots (28)$$

$$u_{i\pm1/2,j}^{(2)}=u_{i\pm1/2,j}^{(1)}\pm\frac{\Delta t}{\rho\Delta x_{i\pm1/2,j}}\Delta p_{i,j}^{(2)} \qquad \dots\dots\dots (29)$$

$$v_{i,j\pm1/2}^{(2)}=v_{i,j\pm1/2}^{(1)}\pm\frac{\Delta t}{\rho\Delta y_{j\pm1/2,j}}\Delta p_{i,j}^{(2)} \qquad \dots\dots\dots (30)$$

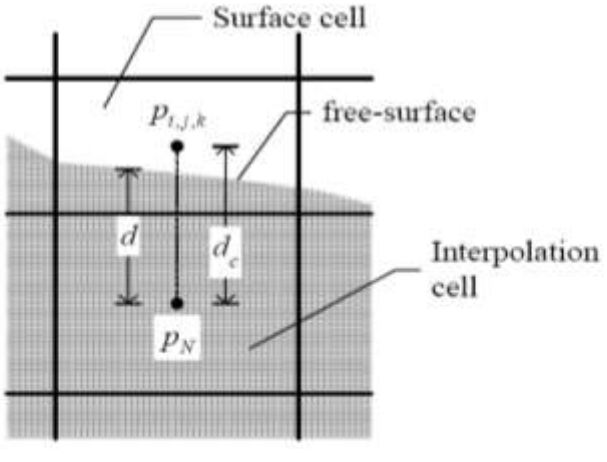


Figure 4: Sketch for pressure interpolation procedure

In summary, the procedure of pressure iteration can be stated as follows (refer to Fig 5). The correction of pressure is calculated from either Eq. (25) for full cells or from Eq. (27) for free surface cells. The corrected pressure is then obtained from Eq. (28). The velocities compatible with the new corrected pressure are obtained from Eqs. (29) and (30). This process is done iteratively until the $D_{i,j}$ term becomes sufficiently small such that the velocity field is in required accuracy.

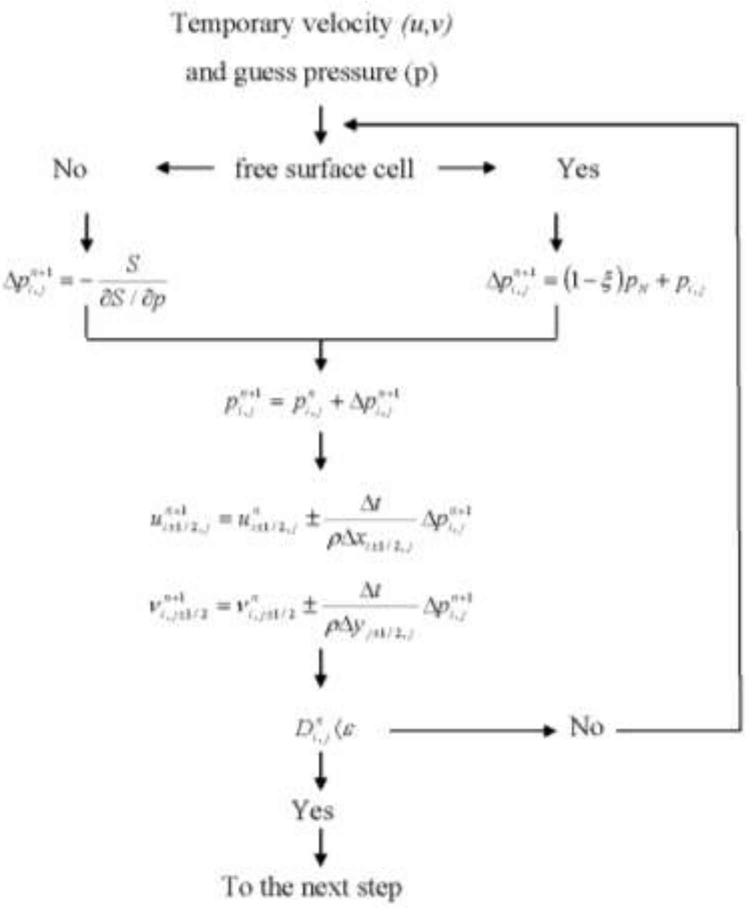

Temporary velocity (u,v)
and guess pressure (p)
No
free surface cell
Yes
$\Delta p_{i,j}^{n+1} = -\frac{S}{\partial S / \partial p}$
$\Delta p_{i,j}^{n+1} = (1-\xi) p_N + p_{i,j}$
$p_{i,j}^{n+1} = p_{i,j}^{n} + \Delta p_{i,j}^{n+1}$
$u_{i\pm 1/2,j}^{n+1} = u_{i\pm 1/2,j}^{n} \pm \frac{\Delta t}{\rho \Delta x_{i\pm 1/2,j}} \Delta p_{i,j}^{n+1}$
$v_{i,j\pm 1/2}^{n+1} = v_{i,j\pm 1/2}^{n} \pm \frac{\Delta t}{\rho \Delta y_{j\pm 1/2,j}} \Delta p_{i,j}^{n+1}$
$D_{i,j}^{n} \langle \varepsilon$
No
Yes
To the next step

Figure 5: Flowchart of pressure-velocity iteration

Approximations for volume of fluid function

The divergence equation for F (eq. 6) can be finite differenced for the advection term in the x-direction in terms of an upstream donor (*d*) cell at (*id*, *j*) and a downstream acceptor (*a*) cell at (*ia*,*j*) (if $u_{i,j}^{n+1} \succ 0$), *idm* = *i* −1, *id* = *i*, *ia* = *i* +1, otherwise, *ia* = *i*, *id* = *i* +1, *idm* = *i* + 2 :

$$\widehat{F}_{id,j} = F_{id,j}^{n} - \frac{\Delta t \Delta (Fu)}{\Delta x_{id}} \quad \dots\dots\dots (31)$$

$$\widehat{F}_{ia,j} = F_{ia,j}^{n} - \frac{\Delta t \Delta (Fu)}{\Delta x_{ia}} \quad \dots\dots\dots (32)$$

Here the amount of *F* fluxed across the cell face in Δ*t* is:

$$\Delta t \Delta (Fu) = \min\left(F_{iad,j} \left| u_{i+1/2,j}^{n+1} \Delta t \right| + CF_{x}, F_{id,j} \Delta x_{id} \right) \quad \dots\dots\dots (33)$$

with the correction factor:

$$CF_{x} = \max\left\lfloor \left(F_{idm,j} - F_{iad,j} \right) \left| u_{i+1/2,j}^{n+1} \Delta t \right| - \left(F_{idm,j} - F_{id,j} \right) \Delta x_{id}, 0 \right\rfloor \quad \dots\dots\dots (34)$$

In these expressions, superscript *a* denotes the acceptor cell, subscript *d* denotes the donor cell, and double subscript *ad* corresponds to either *a* or *d* depending on the surface orientation. Double subscript *dm* denotes the upstream of the donor cell and Δx_{id} stands for the width of the donor cell. The operator *min* in Eq. (33) prevents the fluxing of more *F* from the donor cell than it has to give, while the operator *max* in Eq. (34) accounts for an additional *F* flux, CF_x, if the amount of void (1-*F*) to be fluxed exceeds the amount of void available in the donor cell. Fig. 6 provides a pictorial explanation of Eq. (33), where the shaded region represents the amount of fluid in each cell and the striped region represents the amount of fluid to be fluxed. The donor and acceptor cells are defined in Fig. 6*a* for fluxing across a vertical cell face.

Following Nichols, B. D., and Hirt, C. W., (1981), the rules for choosing *ad* = *a* or *ad* = *d* are the following. When *ad* = *d*, the flux is an ordinary donor-cell value, $F = F_d|V|$, in which the *F* value in the donor cell is used to define the fractional area of the cell face fluxing, as shown in Fig. 6*b*. When *ad* = *a*, the value of *F* in the acceptor cell is used to define the fractional area of the cell face across which fluid is flowing. In case (*c*) of Fig. 6, all the fluid in the donor cell is fluxed because everything lying between the dashed line and the flux boundary moves into the acceptor cell. In case (*d*) of Fig. 6, more fluid than the amount $F = F_a|V|$, must be fluxed, the extra fluid between the dashed line and the flux boundary is equal to the CF_x value in Eq. (33).

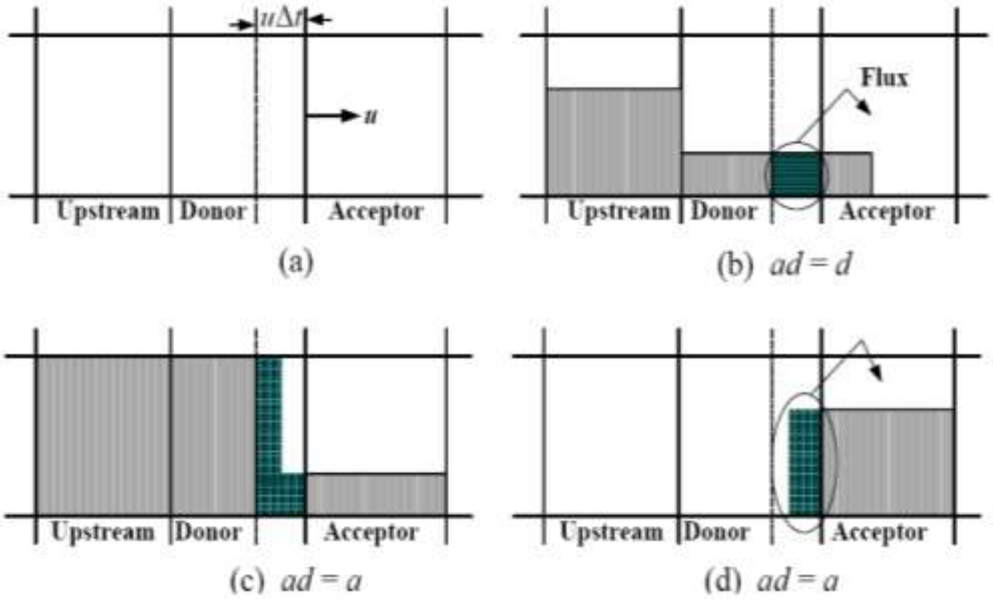


Figure 6: Examples of free-surface shapes used in the advection of *F*

Mesh boundary conditions

At the mesh boundaries, several types of boundary conditions may be imposed using the layer of the fictitious cells that surround the mesh. The built-in types

of boundary conditions are as follows: free-slip, no-slip, continuative and constant pressure boundary conditions. Consider a left boundary, for instance, as illustrated in Fig. 8. If this is a rigid free-slip wall, then the normal velocity and the gradient of the tangential velocity are both set zero. Namely,

$$u_{1,j}^{n+1} = 0 \ ; \ v_{1,j}^{n+1} = v_{2,j}^{n+1} ;$$

$$F_{1,j}^{n+1} = F_{2,j}^{n+1} ; \ p_{1,j}^{n+1} = p_{2,j}^{n+1} \qquad \ldots\ldots\ldots (35)$$

If the left boundary is a rigid no-slip wall, then both of the normal and tangential velocities are set to zero, Fig.8. That is to say,

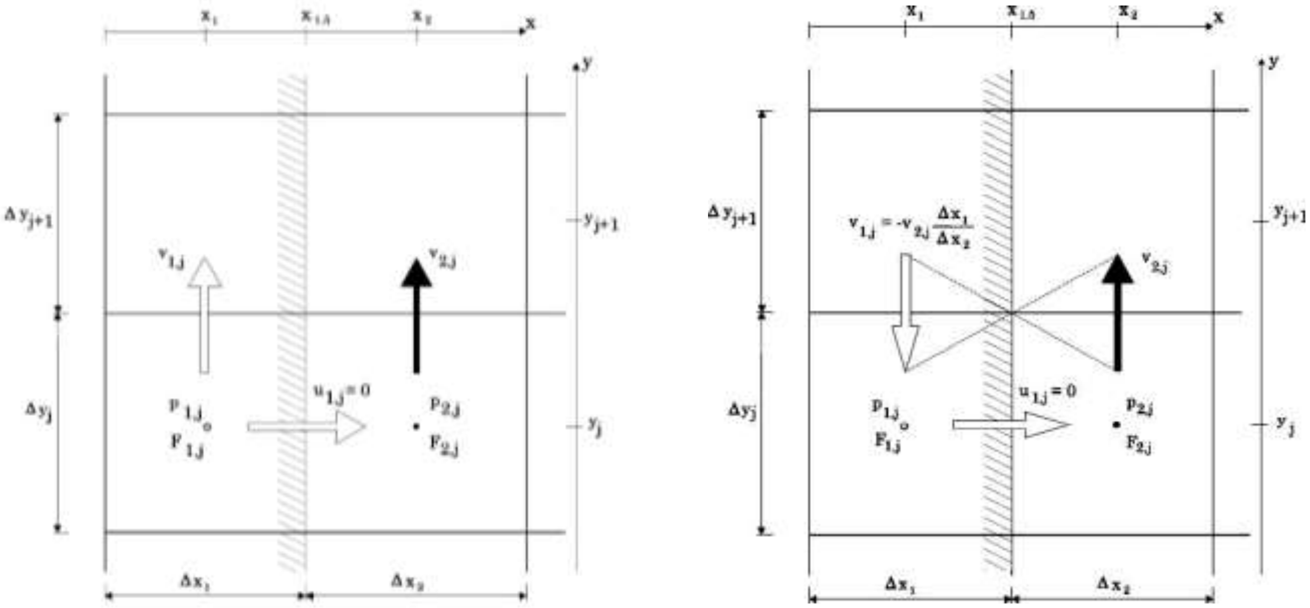
$$u_{1,j}^{n+1} = 0 \ ; \ v_{1,j}^{n+1} = -\frac{v_{2,j}^{n+1}\Delta x_1}{\Delta x_2} \ ;$$

$$F_{1,j}^{n+1} = F_{2,j}^{n+1} ; \ p_{1,j}^{n+1} = p_{2,j}^{n+1} \qquad \dots\dots\dots (36)$$

For a continuative or outflow boundary, a prescription is needed so that the fluid may flow out of the domain computation. The continuative boundary conditions imposed at the left wall are expressed as

$$u_{1,j}^{n+1} = u_{2,j}^{n+1} ; \ v_{1,j}^{n+1} = v_{2,j}^{n+1} ;$$

$$F_{1,j}^{n+1} = F_{2,j}^{n+1} ; \ p_{1,j}^{n+1} = p_{2,j}^{n+1} \qquad \dots\dots\dots (37)$$



(a) free-slip rigid wall (b) no-slip rigid wall

Figure 8: Principle sketch of the "free-slip rigid wall and no-slip rigid wall" boundary condition for calculating the variables u, v, p and F cell (i, j) (indicated by white arrow and white ball).

Free-Surface Construction

For the accurate application of boundary conditions, the free-surface of the fluid has to be constructed at each time level. In the Volume-of-Fluid (VOF) technique, it is assumed that the boundary can be approximated by a straight line cutting through the cell. By first determining the slope of this line, it can then be moved across the cell to a position that intersects the known amount of Fvolume in the cell.

To determine the boundary slope, it must be recognized that the boundary can be represented either as a single-valued function $Y(x)$ or as $X(y)$, depending on its orientation. To accomplish this in a cell (i,j), eight cells surrounding it should be considered, i.e., the computational molecule consists of nine adjacent cells (3x3) as shown in Fig. 9. A good approximation to $Y(x)$ is

$$Y_i = F(i, j-1)\delta y_j + F(i, j)\delta y_j + F(i, j+1)\delta y_{j+1} \qquad \dots\dots\dots (38)$$

where $Y = 0$ is taken as the bottom edge of the *j-1* row of the molecule. Y_{i+l} and Y_{i-l} are similarly obtained. Then

$$(\frac{dY}{dx})_i = \frac{1}{\delta\overline{x}_{i-1} + \delta\overline{x}_i}\left\{\frac{\delta\overline{x}_{i-1}}{\delta x_i}\left(Y_{i-1} + Y_i\right) + \frac{\delta\overline{x}_i}{\delta x_{i-1}}\left(Y_i + Y_{i-1}\right)\right\} \qquad \dots\dots\dots (39)$$

where $\delta\overline{x}_i = \left(\delta x_i + \delta x_{i+1}\right)/2$. Similarly, $X(y)$ may be approximated by

$$X_j = F\left(i-1, j\right)\delta x_{i-1} + F\left(i, j\right)\delta x_i + F\left(i+1, j\right)\delta x_{i+1} \qquad \dots\dots\dots (40)$$

where $X = 0$ is taken as the left edge of the i-1 column of the molecule. Having evaluated X_{j+1} and X_{j-1} similarly, we can write

$$(\frac{dX}{dy})_j = \frac{1}{\delta\overline{y}_j + \delta\overline{y}_{j-1}}\left\{\frac{\delta\overline{y}_{j-1}}{\delta\overline{y}_j}\left(X_{j+1} - X_j\right) + \frac{\delta\overline{y}_j}{\delta\overline{y}_{j-1}}\left(X_j - X_{j-1}\right)\right\} \qquad \dots\dots\dots (41)$$

where $\delta\overline{y}_j = \left(\delta y_j + \delta y_{j+1}\right)/2$

The free-surface orientation in an (i,j) cell can be determined by comparing the absolute values of the above two derivatives. The derivative with the smallest magnitude gives the best approximation to the slope of the free-surface because the corresponding Y or X approximation is more accurate in that case. Therefore, if $\backslash dY/dx\backslash$ is smaller than $\backslash dX/dy\backslash$, the boundary is more nearly horizontal than vertical; otherwise it is more nearly vertical.

The values of the above derivatives may also be used to find out where the fluid exists in a free-surface cell. Suppose $\backslash dY/dx\backslash$ is smallest so the interface is more horizontal than vertical. If dX/dy is negative, F fluid lies below the boundary, and cell (i,j-1) is used as the neighboring interpolation cell for the surface cell in the free-surface pressure difference equation, Eq. (27). Had dX/dy been positive, cell *(i,j+1)* would be chosen for the neighboring interpolation cell because fluid would then be above the boundary.

Once the boundary slope and the side occupied by fluid have been determined, a line can be constructed in the cell with the correct amount of F volume lying on the F fluid side. This line is used as an approximation to the actual boundary and provides the information necessary to calculate ξ for the application of the free-surface pressure difference equation, Eq. (27).

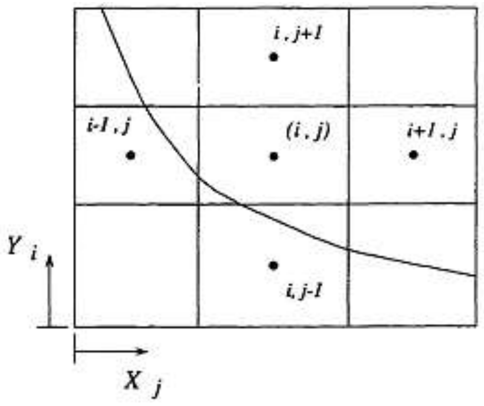


Figure 9: Computational molecule to construct free-surface in $(i.j)$ cell

Free-Surface Boundaries

There are three sets of boundary conditions at the free surface: surface tension effects, liquid-solid contact line conditions, and specifications of velocities at and outside the interface.

<u>Surface Tension:</u> Two essential steps are needed to include surface tension effects in a calculation. First, it is necessary to compute a local curvature, *J*, in each free-surface cell using

$Y(x)$ or $X(y)$ definitions from Eqs. (38) and (40), so that a surface tension pressure, $\boldsymbol{P_s}$**,** can be evaluated. Once the curvature $\boldsymbol{J}$ is found, this surface tension pressure is calculate and given by

$$P_s = J\gamma \qquad \text{… … … (42)}$$

Second, it is necessary to impose this surface force on all interfaces based on the pressure relation for the free-surface ceils, Eq. (27).

In the first step, $\boldsymbol{J}$ is given by

$$J = J_{xy} + J_{cyl} = \frac{1}{R_{xy}} + \frac{1}{R_{cyl}} \qquad \text{… … … (43)}$$

where R_{xy} is the principal radius of curvature in the x-y plane and R_{cyl} is the principal radius of curvature associated with the azimuthal direction.

To evaluate J_{xy} we use the following equation if the interface is mostly horizontal

$$J_{xy} = \frac{d}{dx}\left[\frac{dY/dx}{\sqrt{1+(dY/dx)^2}}\right] \qquad \text{… … … (44)}$$

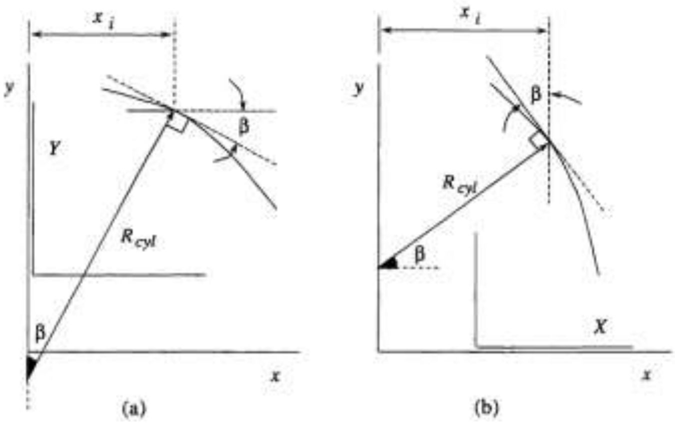


Figure 10: Cylinderical curvature, J_{cyl}.

When the interface is mostly vertical, the roles of x and y must be reversed with $X(y)$ replacing $Y(x)$.

To evaluate J_{cyl}, first we calculate the angle β which is the angle between the interface tangent and the positive x axis when the interface is horizontal. If the interface is vertical, β is the angle it makes with the positive y axis. For the near horizontal case, as shown in Fig. 10(a), then we may write

$$\beta = \arctan(|dY/dx|) \qquad \text{… … … (45)}$$

$$J_{cyl} = \frac{1}{R_{cyl}} = \frac{\sin\beta}{x_i} \qquad \text{… … … (46)}$$

For the near vertical case, Fig. 10(b), we have

$$\beta = \arctan(|dX/dy|) \qquad \text{… … … (47)}$$

$$J_{cyl} = \frac{1}{R_{cyl}} = \frac{\cos\beta}{x_i} \qquad \text{… … … (48)}$$

<u>Liquid-Solid Contact Line:</u> The liquid-solid contact line needs special attention. We incorporate the concept of contact angle, θ, in the free surface boundary condition, Eq. (42). θ is the

angle between the solid wall and the fluid interface that includes the fluid; θ is assumed to be known. To impose this angle condition we adjust the $Y(x)$ and $X(y)$ values at the wall. For example, consider the boundary cell *(i,j)* as shown in Figs. 11(a) and 11(b) with fluid assumed to be below the boundary. The surface tension pressure, P_s, will be calculated in the same manner as outlined before based on Eq. (42), except that the surface is assumed to make an angle, θ, with the wall by adjusting the $i+1$ value of $Y(x)$ and the j+1 value of $X(y)$ as

$$Y_{i+1} = Y_i + \frac{(\delta x_i + \delta x_{i+1})/2}{\tan\theta} \qquad \ldots\ \ldots\ \ldots\ (49)$$

and

$$X_{j+1} = X_i - \frac{(\delta y_i + \delta y_{i+1})/2}{\tan\theta} \qquad \ldots\ \ldots\ \ldots\ (50)$$

Similar adjustments are made to the appropriate variables for all other orientations of walls and interfaces. The above equations are used when calculating the mean curvature, J, of the liquid meniscus near the solid wall as was explained previously. It should be mentioned that the contact angle might have different values for different solid walls of the computational domain. In addition, the contact angle may be a dynamic contact angle that varies with time in which case it has to be updated after each time step.

Interface Velocities Specification: Velocities immediately outside a free surface must be specified at each time step because their values are needed in the finite-difference approximation for points outside the surface. Velocities are set on every cell boundary between a surface cell and an empty cell. If the surface cell has only one neighboring empty cell, the boundary velocity is set to insure the vanishing of equation, the velocity divergence defined in Eq. (2). When there are two or more empty cell neighbors, the individual contributions to the divergence, $1/2\partial(xu)/\partial x$ and $\partial v/\partial y$, are separately set to zero. In addition, we have to set the velocities on boundaries between empty cells adjacent to a surface cell. This is accomplished by setting zero values for $\partial u/\partial y$ and $\partial v/\partial x$. i.e., no tangential stresses at the free-surface boundaries.

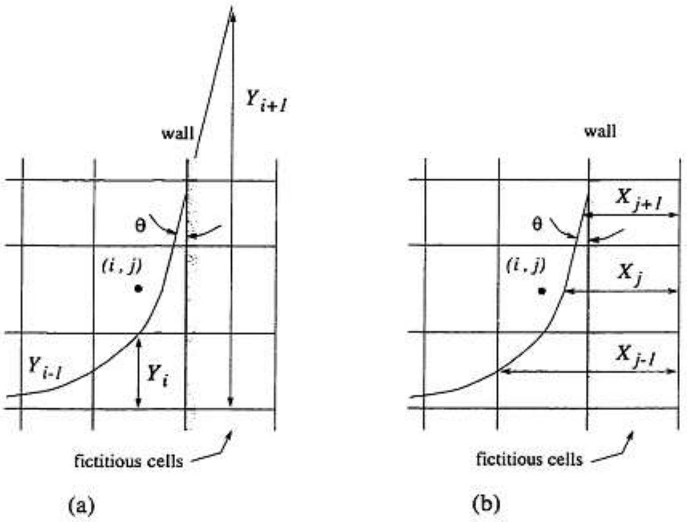


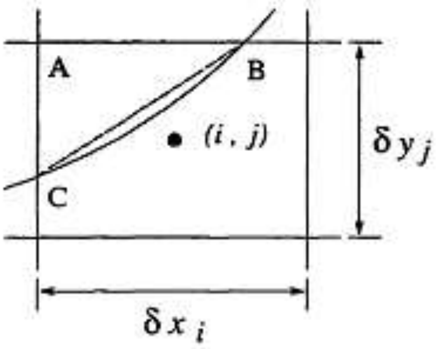
Figure 11: Liquid-solid contact line condition.

Initial Conditions

At the beginning of the first step of the computation, the initial values of volume-of-fluid function, $F_{i,j}^{n=0}$, and the initial velocity distributions within the fluid must be given. When setting the values of $F_{i,j}^{n=0}$ for free-surface cells, first we need to find where a cell is cut by the fluid free-surface. Assuming a straight line for the cut through the cell, then the area of the cell occupied by the fluid can be calculated as shown in Fig. 12. Now for the cells for which $F_{i,j}^{n=0}\rangle 0$ the x-direction velocity, $u_{i,j}^{n=0}$, is set to zero, and

$$\upsilon_{i,j}^{n=0} = -V_0 \qquad \text{……… (51)}$$

where V_0 is the droplet impact velocity.



$$F_{i,j}^{n=0} = \frac{\overline{AB}\cdot\overline{AC}}{2\delta x_i \delta y_j}$$

Figure 12: Calculation of the initial F values for free-surface cells.

Numerical Stability

Numerical calculations often have computed quantities that develop large, high-frequency oscillations in space, time, or both. This behavior is usually referred to as a numerical instability; especially if the physical problem being studied is know not to have unstable solutions. When the physical problem does have unstable solutions and if the calculated results exhibit significant variations over distances comparable to a cell width or over times comparable to the time increment, the accuracy of the results cannot be relied on. To prevent this type of numerical instability or inaccuracy certain restrictions must be observed in defining the mesh increments ∂x_i and ∂y_j, the time increment ∂t, and the upstream differencing parameter α.

The mesh increments should be chosen small enough to resolve the expected spatial variations in all dependent variables. When this is impossible because of limitations imposed by computing time or memory requirements, special care must be exercised in interpreting calculation results. For example, when computing the flow in a large chamber, it is usually impossible to resolve thin boundary layers along the confining walls. In many applications, however, the presence of thin boundary layers is unimportant and free-slip boundary conditions can be justified as a good approximation.

Once a mesh has been chosen, the choice of the time increment necessary for stability is governed by several restrictions. First, material cannot move through more than one cell in one time step because the difference equations assume fluxes only between adjacent cells. Therefore, the time increment must satisfy the inequality,

$$\partial t \prec \min\left\{\frac{\partial x_i}{\left|u_{i,j}\right|}, \frac{\partial y_j}{\left|v_{i,j}\right|}\right\} \qquad \text{……… (52)}$$

where the minimum is with respect to every cell in the mesh. Typically, ∂t is chosen equal to one-fourth to one-third of the minimum cell transit time. Second, when a nonzero value of kinematic viscosity is used, momentum must not diffuse more than approximately one cell in one time step. A linear stability analysis shows that this limitation implies

$$\nu\partial t \prec \frac{1}{2}\frac{\partial x_i^2 \partial y_j^2}{(\partial x_i^2 + \partial y_j^2)} \qquad \dots\dots\dots (53)$$

Third, when surface tension is included there must also be a limit on ∂t to prevent capillary waves from traveling more than one cell width in one time step. A rough estimate for this stability condition is

$$\sigma\partial t^2 \prec \frac{\rho_m \partial x_m^3}{4(1+CYL)} \qquad \dots\dots\dots (54)$$

where ρ_m is the minimum of ρ_F and ρ_C, ∂x_m is the minimum cell size in the mesh (either ∂x or ∂y), and CYL is 1.0 for cylindrical geometry and 0.0 for the Cartesian case. Finally, with ∂t chosen to satisfy the above inequalities, the last parameter needed to insure numerical stability is α. The proper choice for α is

$$1 \geq \alpha \succ \max\left\{\left|\frac{u_{i,j}\partial t}{\partial x_i}\right|, \left|\frac{v_{i,j}\partial t}{\partial y_j}\right|\right\} \qquad \dots\dots\dots (55)$$

As a rule of thumb, an α approximately 1.2 to 1.5 times larger than the right-hand member of the last inequality is a good choice. If α is too large, an unnecessary amount of numerical smoothing (diffusion-like truncation errors) may be introduced.

1. The Broken Dam Test

It is desired to analyze the motion of water within a basin. Initially, the basin contains a dam, and the water is confined by the dam as shown. At the start of the analysis, the dam is removed and the water flows into the rest of the basin, Figure (13).

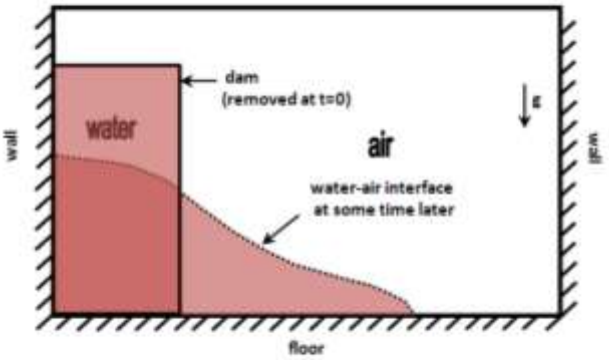
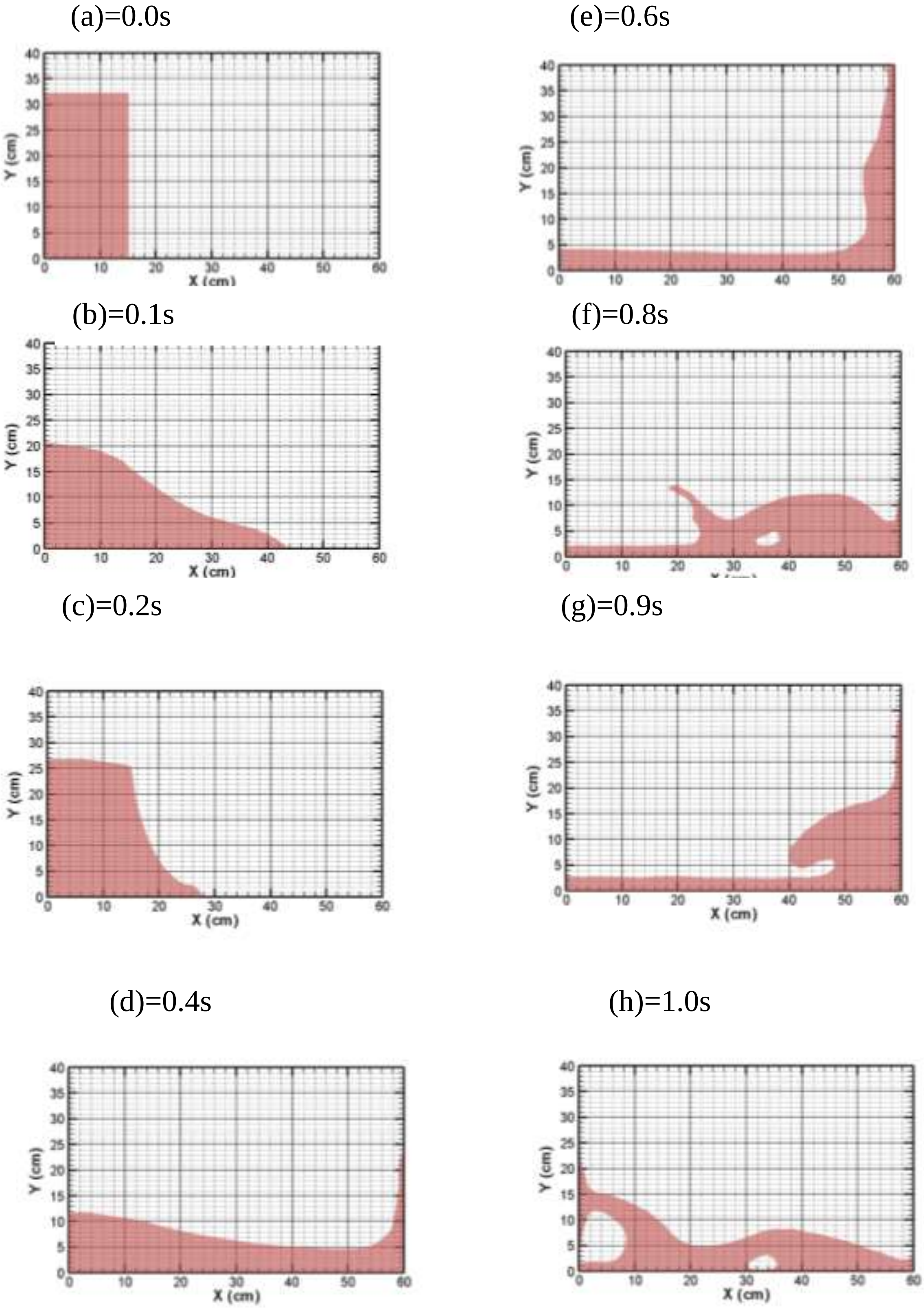


Figure 13: Dam break physical setup



(a)=0.0s

(e)=0.6s

(b)=0.1s

(f)=0.8s

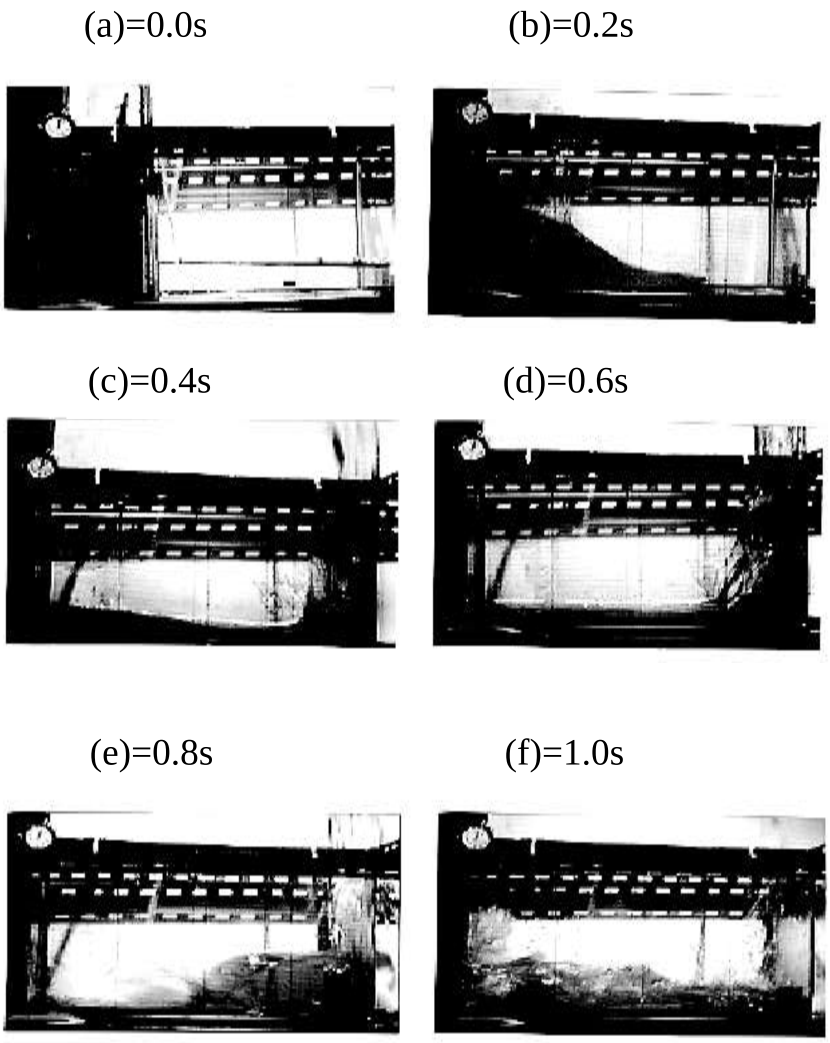
(c)=0.2s

(g)=0.9s

(d)=0.4s

(h)=1.0s

Figure 14: Time evolution of the dam break flow. (a)=0.0s, (b)=0.1s, (c)=.0.2s, (d)=0.4s, (e)=0.6s, (f)=0.8s, (g)=0.9s, (h)=1.0s



(a)=0.0s

(b)=0.2s

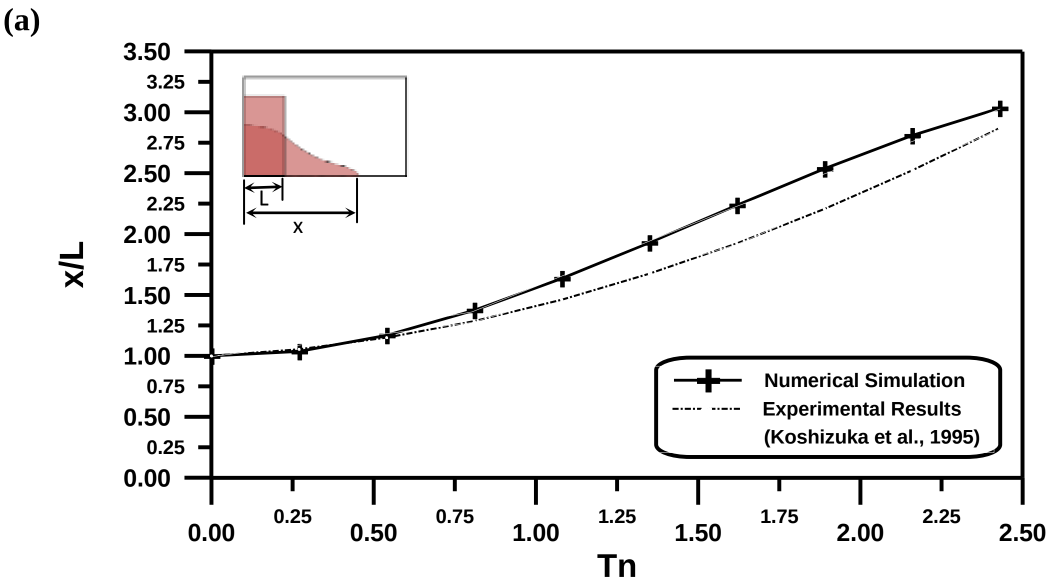
(c)=0.4s

(d)=0.6s

(e)=0.8s

(f)=1.0s

Figure15: Experimental results of a collapsing water column (Koshizuka *et al*. (1995))



(b)

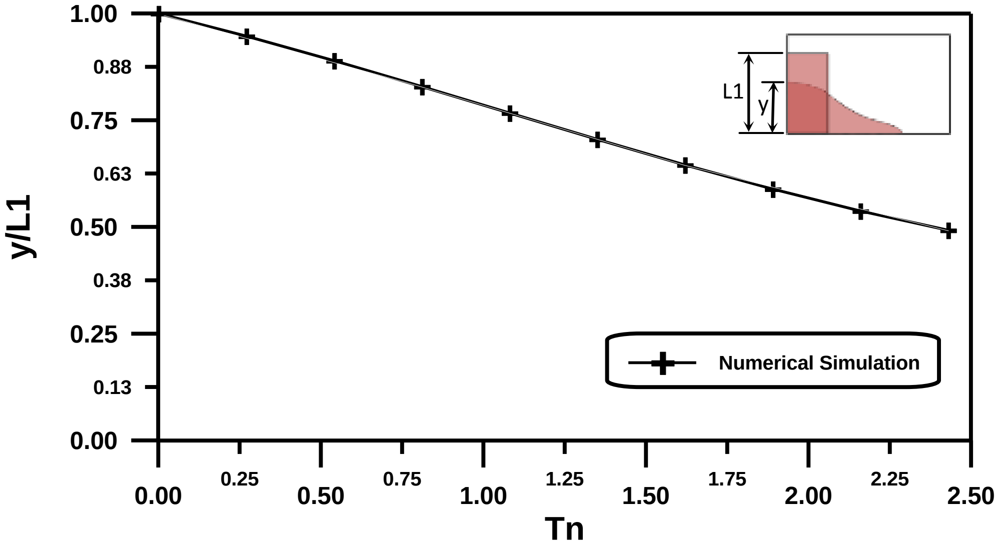


Figure 16 (a) and (b): Predicted and measured time histories of waterfront location and water column height in dam-break problem.

Numerical Results and Discussion

The results displayed in Fig. 14 are part of a movie sequence obtained from the results of the run of the dam break flow from time t=0.0s to t=1.0s. At time t=0.0s, the water column was allowed to flow. A relatively high velocity and shallow water depth flew in the x-direction quickly formed (e.g. t=0.2s). As time progressed, the flow impacted on the vertical wall at the opposite side of the tank. An upward water jet was suddenly formed that rose until gravity overcame the upward momentum (before at t=0.6s). At this moment, the jet became thicker and the flow started to reverse. We note that the water leaves the tank when it is time =0.6s. The momentum carries part of the fluid along the vertical wall up to the height of the square cavity.

Due to the oncoming flow, an adverse momentum gradient was created that resulted in an overturning wave (t= 0.6s and t=0.8s). This wave formation continued until the wave tip reconnected with the incident shallow water flow that now had less forward momentum. A sudden rise in pressure occurred at the reconnection point that was of the same order of magnitude as the pressure on the impact wall. This was due to the existence of high relative momentum between the fluid at the wave tip and the free-surface just before the attachment (note at time t=0.8s). At this stage, the flow became complicated as several big and small pockets of entrained air were created due to the first and subsequent impacts on the free-surface.

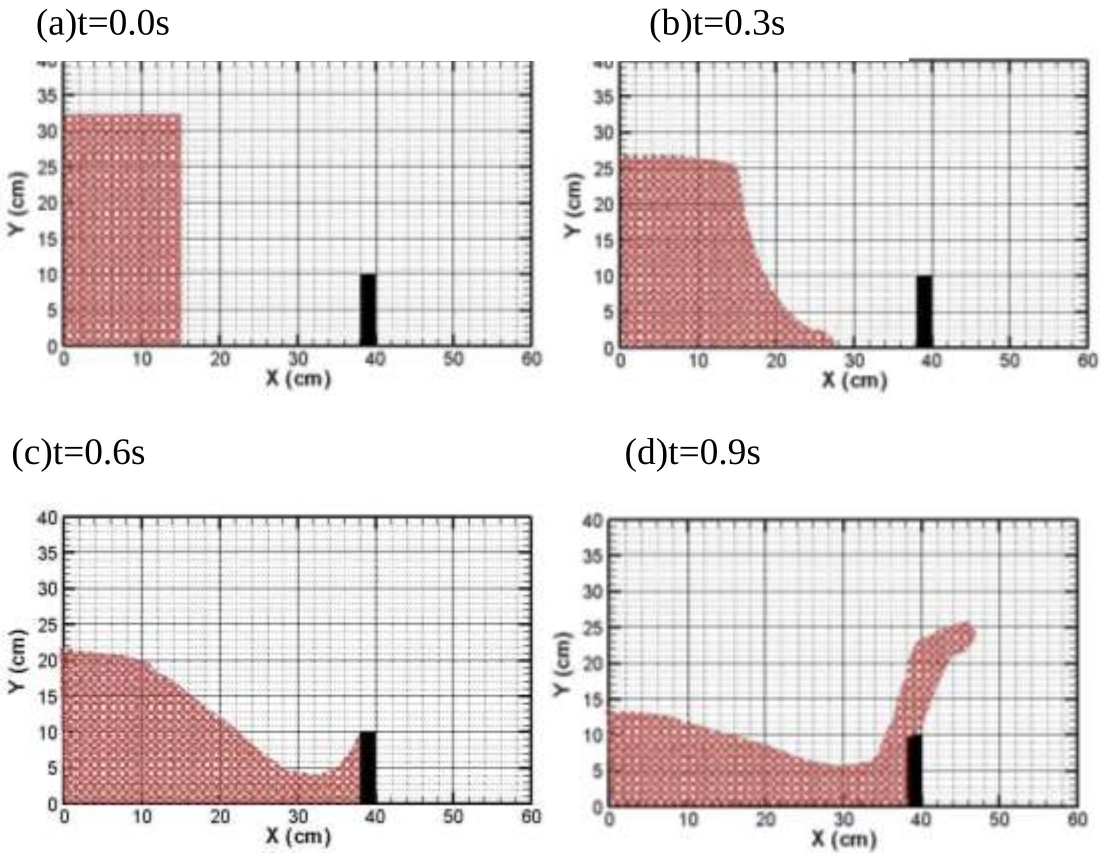
Gravity (i.e. g=9.81 m/s^2) caused the water column on the left of the reservoir to seek the lowest possible level of potential energy (Figs 2a-h). Thus, the water column would collapse and eventually come to rest. The initial stages of the flow were dominated by inertia forces with viscous effects increasing as the water came to rest.

The height of the collapsing water column decreased rapidly as time increased (Fig. 2a). The leading edge of the collapsing water column increased as time increased (Fig. 2c). At time t=0.2s; approximate 75% of the base is covered with water, while at time t=0.4s; the horizontal interface makes a small angle with the base of the tank and the water starts to leave the domain at the top right corner. At times t=0.6s and t=0.8s; the horizontal interface is almost parallel to the base of the tank and the water against the right wall starts to fall back under the influence of gravity. At times t=0.8s and t=0.9s; the backward moving wave has folded over and a small amount of air is trapped.

A returning wave is then formed as the fluid momentum is lost and gravity effects takes over forcing the fluid along the vertical right wall to move back down. Finally at time t=1.0s; the tongue of the backward moving wave impinges upon the left wall, trapping a small air bubble in middle and large bubble attached to the left wall .

Results presented and available from the experiment and computing results by other researchers show the same tendency in references (Koshizuka *et al*. (1995), Fig. 15, Koshizuka *et al*. (2000), Zima, P. (2005) , Kocaman S. (2007) and Denys *et al*. (2009) and M. Dimitrios (2009)) Comparisons with our numerical results, see Fig. 14 and Fig. 15. The results from the simulation were in good agreement with the experimental results for Koshizuka *et al*. (1995).

The time histories of the waterfront location and water column height are shown in Fig. 16. The computing and experimental results are shown on the normalized time ($T_n = t\sqrt{2g/L}$) background.


(a)t=0.0s

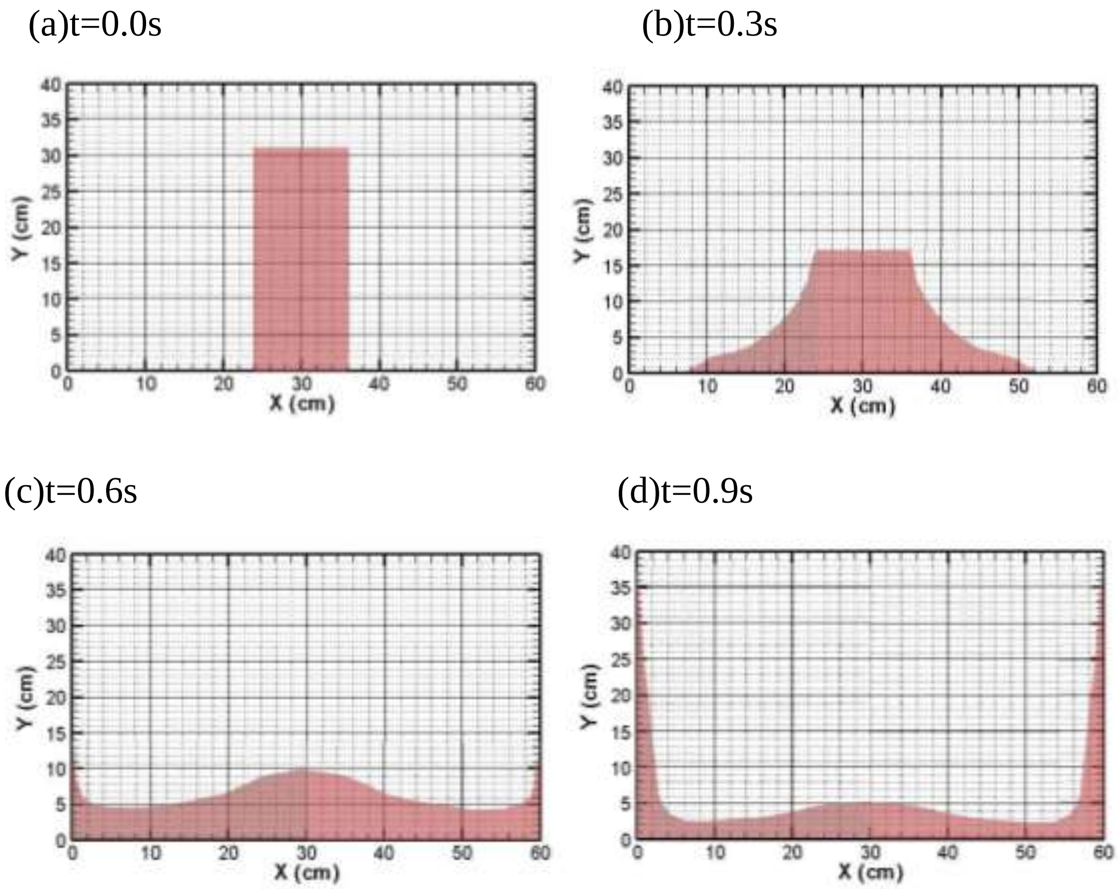
(b)t=0.3s

(c)t=0.6s

(d)t=0.9s

Figure 17: Time evolution of the dam break flow with obstacle. (a)=0.0s, (b)=0.3s, (c)=.0.6s, (d)=0.9s

Fig. 17 shows time evolution of the dam break flow with obstacle in the one side.
When the plate is lifted, a very complex flow pattern developed due to the presence of the obstacle, the water flows rapidly into the tank and reaches the obstacle after approximately 6 s (Fig. 17c). There, the water reflects against the wall, a bore forms and begins to travel in the upstream direction, back towards upstream direction while the other part moves up the bump. Velocities become minimum just behind the bump and velocities of the negative wave front travelling towards upstream direction increase. Also recirculation zones can appear near the obstacle.



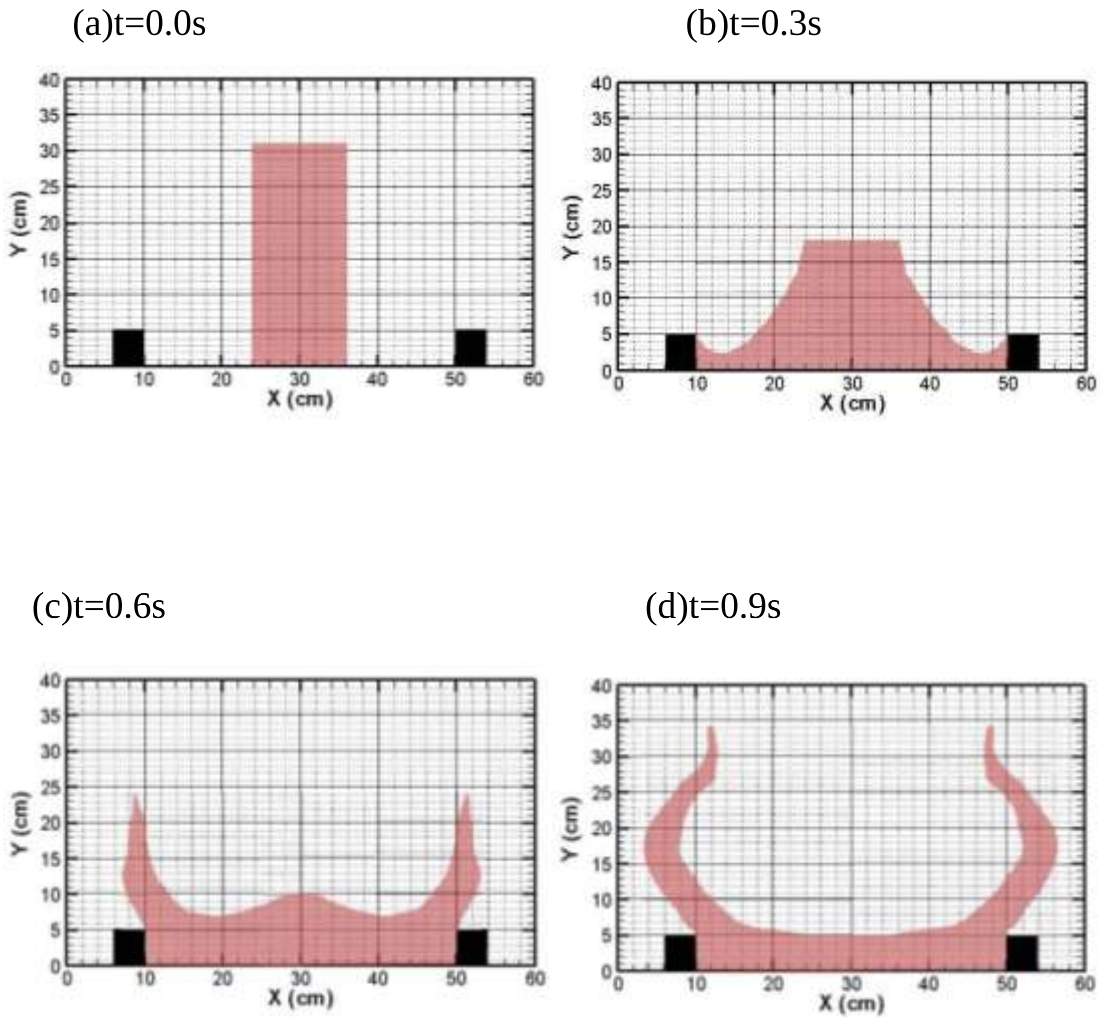
(a)t=0.0s

(b)t=0.3s

(c)t=0.6s

(d)t=0.9s

Figure 18: Time evolution of the dam break flow, water column in the center of the tank. (a)=0.0s, (b)=0.3s, (c)=.0.6s, (d)=0.9s



(a)t=0.0s

(b)t=0.3s

(c)t=0.6s

(d)t=0.9s

Figure 19: Time evolution of the dam break flow, water column in the center of the tank and with obstacles in the two sides. (a)=0.0s, (b)=0.3s, (c)=.0.6s, (d)=0.9s

Fig. 18 and 19 show also time evolution of the dam break flow without and with obstacles in the two sides and water column in the center of the tank. When the plate is lifted from two sides, also can be shown a very complex flow pattern developed due to the presence of the obstacle, the water flows rapidly into the tank and reaches to end of the tank in 6s (Fig. 18c) and reaches to the obstacle after approximately 3s (Fig. 19b). Without obstacle, the direction of the flow will be to the ends of the tank and stabilize the flow of water quickly while with two obstacles, can be seed the same results obtained previously (Fig.17), water reflects against the wall, and a bore forms and begins to travel in the upstream direction, back towards upstream direction while the other part moves up the bump and rise to the top.

Conclusion

The Volume of Fluid (VOF) technique is a convenient and powerful tool for modelling fluid flows which contain a free surface. Under the VOF method, fluid location is recorded using a volume of fluid function. It is particularly useful because it uses a minimum of stored information, treats intersecting free boundaries automatically. The VOF technique was described in detail as it has been used to follow free surfaces in an incompressible hydrodynamics code.

References

Denys, D. and M. Dimitrios (2009). On the relevance of the dam break problem in the context of nonlinear shallow water equations. AIMS' Journals Volume X, Number 0X, XX 200X

Ding, L., C. Shu and N. Zhao (2009). Numerical Simulation of Dam Break by Adaptive Stencil Diffuse Interface Method, Modern Physics Letters B, 23, 293-296.

Gerlach, D., G. Tomar, G. Biswas and F. Durst (2006). Comparison of volume of fluid methods for surface tension-dominant two-phase flows. International Journal of Heat and Mass Transfer, 49, pp. 740–754.

Isobe, M. (2001). A renew of numerical models for wave trans-formation in shallow water. Proc. of International Work-shop on Advanced Design of maritime Structures in the 21st century, Port and Harbour research Institute: pp. 200-205.

Isobe, M., S. Takahashi, S. P. Yu, T. Sakakiyama, K. Fujima, K. Kawasaki, Q. Jiang, M. Akiyama and H. Oyama (1999). Interim report on development of numerical wave flume for maritime structure design. Proceeding of Civil Engineering in the Ocean. J.S.C.E. Vol. 15: pp. 321-326.

Iwata, K., R. C. Kawasaki and D. Kim (1996). Braking limit, breaking and post breaking wave deformation due to submerged structures. In: Proceedings 25th Conference on Coastal Engineering, Orlando, USA Vol. 3: pp. 2338-2351.

Hatice O.C. and Selahattin K. (2011). Dam-Break Flow the Presence of Obstacle: Experiment and CFD Simukation. Engineering Applications of Computational Fluid Mechanics Vol. 5, No. 4, pp. 541–552.

Kocaman S. (2007). Experimental and Theoretical Investigation of Dam Break Problem. PhD Thesis, Institute of Natural and Applies Science, Uuniversity of Çukurova.

Koshizuka S. and Y. Oka (2000). Moving particle semi-implicit method: Fully Lagrangian analysis of incompressible flows. *In*: Proceedings of the European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS), Barcelona, Spain. 11-14 September 2000.

Koshizuka, S., H. Tamako and Y. Oka (1995). A particle method for incompressible viscous flow with fluid fragmentation. Computational Fluid Dynamics JOURNAL, Vol. 4(1), p. 29-46.

Lemos, C. M. (1992). Wave breaking, a numerical study, Lecture Notes in Engineering No. 71. Springer-Verlag, Berlin. Germany.

Mckibben J. F. (1993). A Computational Fluid Dynamics Model for Transient Three-Dimensional Free Surface Flows. PhD Thesis, Institute of Paper Science and Technology Atlanta, Georgia.

Nichols, B. D., C. W. Hirt and R. S. Hotchkiss (1981). SOLA-VOF: A solution algorithm for transient fluid flow with multiple free boundaries. Los Alamos, CA, USA, Los Alamos Scientific Report, Report LA-8355.

Pilliod J. E. and E. G. Puckett (1997). Second-order accurate volume of fluid algorithms for tracking material interfaces. Lawrence Berkley Lab. Tech. Report, No.LBNL-0744.

Renardy Y. and M. Renardy (2002). PROST: a parabolic reconstruction of surface tension for the volume-of-fluid method. J. Comp. Phys. 183, pp. 400–421.

Torrey, M. D., L. D. Cloutman, R. C. Mjolsness and C. W. Hirt (1985). NASA-VOF2D: a computer program for incompressible flows with free surfaces, Los Alamos, NM, USA, Los Alamos Scientific Report. Report LA-10612-MS.

Torrey, M. D., Mjolsness, R. C., and L. R. Stein (1987). NASA-VOF3D: A Three-Dimensional Computer Program for Incompressible Flows with Free Surfaces. Los Alamos National Laboratory Report LA-11009-MS.

Zima, P. (2005). The numerical simulation of two-dimensional vertical incompressible viscous flow. Water management and hydraulic engineering: Ninth International Symposium on Water Management and Hydraulic Engineering, pp. 455-462

Zohdi T. (2007). An Introduction to Modeling and Simulation of Particulate Flows. Society for Industrial and Applied mathematical (SIAM), ISBN 0898716276, Philadelphia.