Transaction B: Mechanical Engineering
Vol. 16, No. 1, pp. 64–74

SCIENTIA IRANICA

Generalized Curvilinear Coordinate Interface Tracking in the Computational Domain

A.H. Nikseresht[1,*], M.M. Alishahi[2] and H. Emdad[2]

Abstract. *Volume Of Fluid (VOF) is one of the most powerful methods to resolve free surface flows. In this study, a new algorithm is developed in a curvilinear coordinate system, which implements an implicit pressure based method (SIMPLE) with a staggered grid and a Lagrangian propagation of the interface, using the VOF method in the computational domain. Based on this algorithm, a computer code is generated and two test cases of dam-breaking problems, both in curvilinear and Cartesian grid systems, are examined and, then, two applications of this method, including flow through a curved gate under a dam and the impact problem of a circular cylinder, are presented. The results show good agreement with experimental and other computational results.*

Keywords: *Free-surface; Volume-of-fluid; Front tracking; Curvilinear coordinate system; Dam-breaking*

INTRODUCTION

Incompressible viscous flow with a moving free surface occurs in many instances, both in industry and in nature, such as environmental engineering, die-casting, injection molding processes, marine sciences and many others. The available numerical methods for solving such problems can be classified into moving and fixed grid approaches. The moving grid approach is typically confined to special applications, due to limitations in the rezoning technique [1-3]. In this regard, the fixed grid approach seems to be a more viable method whenever a general motion of free surface flow is considered [4].

Among the existing fixed grid approaches, Harlow and Welch [5] proposed the well-known marker and cell method (MAC) that labels fluid particles with markers. Nakayama and Mori [6] improved the MAC method to preclude the possibility of producing an unphysical liquid front advancement. In the MAC method, the region occupied by the fluid is tracked by the location of the markers in the course of the fluid motion. Such a method defines the fluid region rather than the free surface and, thus, requires large computer storage and additional computational time to move all the fluid markers to new locations, especially when a three-dimensional problem is encountered [7]. Furthermore, a finite volume far from the free surface might be unrealistically overfilled or partially filled with markers, due to numerical errors. Hirt and Nichols [8] introduced the Volume Of Fluid method (VOF) for incompressible flow with a moving free surface.

In the VOF method, the interface describes, implicitly, the data structure that represents the interface, which is the fraction, C, of each cell that is filled with a reference phase, say, phase 1. The scalar field, C, is often referred to as the color function. The magnitude of C in the cells cut by the free surface is between 0 and 1($0 < C < 1$) and, away from it, is either zero or one. The data, C, are given at the beginning of a computational cycle but no approximation of the interface position is known apriori. The method is implicit, since one needs to "invert" data C to find the approximate interface position. In other words, an algorithm for interface reconstruction is needed. Typically, one can reconstruct the interface by the straightforward Simple Line Interface Calculation (SLIC) method [9], Flux Corrected Transport method (FCT) [10] or by various "Piecewise Linear Interface Calculation" (PLIC) methods [11,12]. The second and third methods give much better results than the first method.

1. *Department of Mechanical Engineering, Shiraz University of Technology, P.O. Box 71555-313, Shiraz, Iran.*
2. *Department of Mechanical Engineering, Shiraz University, Shiraz, Iran.*
*. *Corresponding author. E-mail: nikser@sutech.ac.ir*

Received 23 October 2006; received in revised form 4 August 2007; accepted 24 September 2007

It should be mentioned that, in earlier studies, the VOF method was applied in the Cartesian coordinate. Some recent works that implement the VOF method in curvilinear coordinates are presented by Kothe et al., [13] who work mainly in the physical domain. Their methods have some difficulties and cannot be easily used, along with operator split techniques.

In the present study, the finite volume method in the physical domain is used to solve the Navier-Stokes equations. However, the free surface equation is transferred to the computational domain and the free surface is resolved in that plane. The two steps of propagation and reconstruction are carried out in this domain. In the present work, the PLIC for the interface reconstruction method is used. The main advantages of the proposed technique, i.e. the PLIC-VOF method in the computational domain over the earlier method of PLIC, in the physical domain, are as follows:

1. Operator splitting in the computational domain is more easily carried out than in the physical domain. In the computational domain, splitting can be done in each coordinate direction, separately.
2. The method of interface tracking (PLIC-VOF) in the computational domain needs less computer resources, because just one flux computation on each face is required compared to two-face computation in the curvilinear physical domain.
3. The extension of the PLIC method to 3-D in the computational domain is straightforward.

Incompressible Navier-Stokes equations are discretized using the finite volume method, based on the Patankar pressure correction algorithm (SIMPLE) [14]. Hence, the appropriate numerical algorithm to solve the Navier Stokes equations, for a two-phase with a high-density difference incompressible flow in the curvilinear coordinate, is introduced. The performance of the proposed numerical procedure is examined through the solution of two well-documented dam-breaking examples in Cartesian and curvilinear coordinates and comparison of the results with the experiment is also presented.

GOVERNING EQUATIONS

A single set of governing equations, covering both liquid and the surrounding air, for incompressible flows, can be written in the following non-dimensionalized form [12]:

$$\rho^* \left(\frac{\partial u}{\partial \tau} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} \right) = -\frac{\partial p}{\partial x} + \frac{\mu^*}{\mathrm{Re}}\nabla^2 u, \quad (1)$$

$$\rho^* \left(\frac{\partial v}{\partial \tau} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} \right) = -\frac{\partial p}{\partial y} - \rho^* \frac{1}{\mathrm{Fr}^2} + \frac{\mu^*}{\mathrm{Re}}\nabla^2 v, \quad (2)$$

$$\nabla.\vec{u} = 0, \quad (3)$$

where L is an arbitrary characteristic length, $\mathrm{Fr} = \frac{U_\infty}{\sqrt{gL}}$ is the Froude no. and density ρ^* and viscosity μ^* are unity in the liquid region, which jump to another constant in the air region, i.e., $\rho^* = \left\{ \begin{array}{ll} 1 & \text{in liquid} \\ \frac{\rho_a}{\rho_L} & \text{in air} \end{array} \right\}$, $\mu^* = \left\{ \begin{array}{ll} 1 & \text{in liquid} \\ \frac{\mu_a}{\mu_L} & \text{in air} \end{array} \right\}$. This jump happens in a few cells across the interface, as explained in the following. At the interface, a kinematic, as well as a dynamic condition, should be applied. Note that the dynamic condition, i.e., continuity of pressure at the interface, is automatically implemented via solving N-S equations through the interface. The kinematic condition, which states that the interface is convected with the fluid, can be expressed in terms of C, as follows:

$$\partial_t C + u.\nabla C = 0. \quad (4)$$

μ^* and ρ^* at any cells (denoted by ij) can be computed using a simple volume average over the cell:

$$\rho^*_{ij} = C_{ij}\rho^*_L + (1 - C_{ij})\rho^*_a, \quad (5)$$

$$\mu^*_{ij} = C_{ij}\mu^*_L + (1 - C_{ij})\mu^*_a. \quad (6)$$

DISCRETIZATION OF EQUATIONS

The general transport equation for dependent variable ϕ is written as:

$$\frac{\partial(\rho\phi)}{\partial t} + \nabla.(\rho u\phi) - \nabla.(\Gamma\nabla\phi) + B = 0, \quad (7)$$

where u, ρ and Γ are the velocity, density and diffusion coefficient, respectively and B is a source term, which generally depends on ϕ. Regarding the diffusion term in Equation 7, in general, the curvilinear coordinate system will be decomposed into two parts. One part is similar to the discretized diffusion term in an orthogonal curvilinear system and the other part contains mixed derivatives. Therefore, the numerical schemes that are mainly prepared for flux calculations in orthogonal coordinate systems, such as the QUICK or the Power-law scheme [14], can be equally applied in general curvilinear coordinates. The cross derivative terms are discretized with an alternative approximation that is introduced in [15].

Discretization of the Momentum Equations

A staggered grid arrangement is adopted, in which the pressure computation at the geometric center of the control volume and the tangential velocity components, U^{ξ_i}, lie at the midpoints of the respective control volume surfaces and have the following expression [15]:

$$U^{\xi_i} = \frac{S^i.u}{|S^i|\cos\alpha_i}, \tag{8}$$

where S^i are the surface area vectors and α_i are the angles between the surface area vectors and the tangential vectors, e_i. The unit tangent vectors, e_i, are calculated at the centers of the control volume surfaces and are locally parallel to the coordinate lines, ξ_i. In order to discretize the momentum equations, auxiliary discretizations for the Cartesian velocity components are considered [16]. The resulting equation can be written as follows:

$$a_P U_P^{\xi_1} = a_E U'^{\xi_1}_E + a_W U'^{\xi_1}_W + a_N U'^{\xi_1}_N + a_S U'^{\xi_1}_S + b'_{U^{\xi_1}}, \tag{9}$$

where:

$$U'^{\xi_1}_E = a_{11}u_E + a_{12}v_E, \tag{10}$$

$$U'^{\xi_1}_W = a_{11}u_W + a_{12}v_W, \tag{11}$$

$$U'^{\xi_1}_N = a_{11}u_N + a_{12}v_N, \tag{12}$$

$$U'^{\xi_1}_S = a_{11}u_S + a_{12}v_S, \tag{13}$$

$$b'_{U^{\xi_1}} = a_{11}b_u + a_{12}b_v, \tag{14}$$

with:

$$a_{11} = \left(\frac{S^1_x}{|S^1|\cos\alpha_1}\right)_P,$$

and:

$$a_{12} = \left(\frac{S^1_y}{|S^1|\cos\alpha_1}\right)_P.$$

In Equations 9 to 14, the primed velocities are recognized as the velocity components parallel to $U_P^{\xi_1}$ at the neighboring points. It is seen that the primed velocity components, such as $U'^{\xi_1}_E$, are combinations of U^{ξ_1} and U^{ξ_2} and that a field solution, in terms of these velocities, cannot be obtained. This difficulty can be easily removed by introducing the "actual" neighbors (e.g., $U_E^{\xi_1}$) in the discretization equation as follows:

$$\begin{aligned} a_P U_P^{\xi_1} &= a_E U_E^{\xi_1} + a_W U_W^{\xi_1} + a_N U_N^{\xi_1} + a_S U_S^{\xi_1} + b'_{U^{\xi_1}} \\ &+ a_E(U'^{\xi_1}_E - U_E^{\xi_1}) + a_W(U'^{\xi_1}_W - U_W^{\xi_2}) \\ &+ a_N(U'^{\xi_1}_N - U_N^{\xi_1}) + a_S(U'^{\xi_1}_S - U_S^{\xi_1}). \end{aligned} \tag{15}$$

In Equation 15, terms such as $a_E(U'^{\xi_1}_E - U_E^{\xi_1})$ and similar terms in the source term, $b'_{U^{\xi_1}}$, represent the effect of curvature. They are equivalent to the discretized source terms, which would be resulted from tensor analysis. The discretized form of the momentum equations, with part of the pressure difference term written explicitly, can be expressed as:

$$a_P U_P^{\xi_1} = \sum a_{nb} U_{nb}^{\xi_1} + A^{\xi_1}(P_w - P_e) + b_{U^{\xi_1}}, \tag{16}$$

$$a_P U_P^{\xi_2} = \sum a_{nb} U_{nb}^{\xi_2} + A^{\xi_2}(P_s - P_n) + b_{U^{\xi_2}}, \tag{17}$$

where nb denotes the neighbor, $b_{U^{\xi_1}}$ and $b_{U^{\xi_2}}$ include all explicit terms and:

$$A^{\xi_i} = a_{i1}\left(\frac{\partial y}{\partial \xi_2}\right) + a_{i2}\left(\frac{\partial x}{\partial \xi_2}\right), \qquad (i = 1, 2).$$

INTERFACE TRACKING

In the PLIC, at each time step, given the volume fraction of one of the two fluids in each computational cell and an estimate of the normal vector to the interface, a planar surface is constructed within the cell having the same normal. This planar surface also divides the cell into two parts, each of which contains the proper volume of one of the two fluids. This planar interface is then propagated by the flow, and the resulting volume fluxes of each fluid into neighboring cells are determined. The updated values of the volume fraction are found throughout the domain and the numerical simulation can proceed to the next time step. The next three subsections describe the procedure for estimating the normal vector, the construction of the interface in each cell and the propagation of the interface by the flow. It should be noted that all of these three steps are carried out in the computational domain, (ξ_1, ξ_2). For this purpose, at first, the volume fraction equation:

$$\frac{\partial C}{\partial t} + \vec{u} \bullet \nabla C = \frac{\partial C}{\partial t} + u\frac{\partial C}{\partial x} + v\frac{\partial C}{\partial y} = 0.0, \tag{18}$$

should be transformed to the computational domain as follows:

$$\nabla C = \frac{1}{\text{Vol}}\left(S^1\frac{\partial C}{\partial \xi_1} + S^2\frac{\partial C}{\partial \xi_2}\right), \tag{19}$$

where Vol is a volume of each cell.

$$\vec{u} \bullet \nabla C = \frac{1}{\text{Vol}}\left(\vec{u} \bullet S^1\frac{\partial C}{\partial \xi_1} + \vec{u} \bullet S^2\frac{\partial C}{\partial \xi_2}\right).$$

From Equation 8, we have $\vec{u} \bullet S^i = U^{\xi_i}\left|S^i\right|\cos\alpha_i$ then:

$$\begin{aligned} \frac{\partial C}{\partial t} + \vec{u} \bullet \nabla C = 0.0 \Rightarrow \frac{\partial C}{\partial t} &+ \frac{(U^{\xi_1}\left|S^1\right|\cos\alpha_1)}{\text{Vol}}\frac{\partial C}{\partial \xi_1} \\ &+ \frac{(U^{\xi_2}\left|S^2\right|\cos\alpha_2)}{\text{Vol}}\frac{\partial C}{\partial \xi_2} = 0.0. \end{aligned} \tag{20}$$

Now, if $U = \frac{(U^{\xi_1}\left|S^1\right|\cos\alpha_1)}{\text{Vol}}$ and $V = \frac{(U^{\xi_2}\left|S^2\right|\cos\alpha_2)}{\text{Vol}}$, then, Equation 20 has a similar form to its counterpart in the Cartesian coordinate:

$$\frac{\partial C}{\partial t} + U\frac{\partial C}{\partial \xi_1} + V\frac{\partial C}{\partial \xi_2} = 0.0. \tag{21}$$

Normal Estimation

Reconstruction is based on the idea that a normal vector, $\vec{m}$, together with the fractional volume, C, determines a unique line interface cutting the cell. In the first part of the reconstruction, a normal direction to the interface is estimated using a finite-difference formula. The normal vector in the ξ_1, ξ_2 domain is defined as:

$$\vec{m} = \nabla C. \tag{22}$$

At first, a cell corner value of the normal vector, $\vec{m}$, is computed at $(i+1/2,\, j+1/2)$ by:

$$m_{\xi_1,i+1/2,j+1/2} = \frac{1}{2h}(C_{i+1,j} - C_{i,j} + C_{i+1,j+1} - C_{i,j+1}), \tag{23}$$

$$m_{\xi_2,i+1/2,j+1/2} = \frac{1}{2h}(C_{i,j+1} - C_{i,j} + C_{i+1,j+1} - C_{i+1,j}), \tag{24}$$

where $h = \Delta\xi_1 = \Delta\xi_2 = 1$. Then, the required cell centered values are computed from the cell corner values by averaging:

$$m_{ij} = \frac{1}{4}(m_{i+1/2,j-1/2} + m_{i-1/2,j-1/2} + m_{i+1/2,j+1/2} + m_{i-1/2,j+1/2}). \tag{25}$$

Connecting Fractional Volume and Interface Position

In the second part of the reconstruction, an interface, which divides the computational cell into two parts containing the proper volume of each fluid, must be found. This is achieved by deriving an explicit expression, which relates the "cut" volume to parameter α, which completely defines the interface. The problem can be stated as follows.

Given a rectangular (or square) cell of sides c_1 and c_2 in the $(\xi_1,\, \xi_2)$ plane, as depicted in Figure 1, and a straight line (such as EH) with normal vector $\vec{m}$, one must find the area of the region (ABFGDA). To obtain an expression for this area, let one suppose that the components, m_1 and m_2, of the normal are both positive. The most general equation for the straight line in the $(\xi_1,\, \xi_2)$ plane with normal $\vec{m}$ is:

$$m_1\xi_1 + m_2\xi_2 = \alpha. \tag{26}$$

This equation is similar to its Cartesian counterparts if $(x_1,\, x_2)$ is substituted for $(\xi_1,\, \xi_2)$. Thus, the same formulation for the area computation would be used in terms of $(\xi_1,\, \xi_2)$ [10,11].

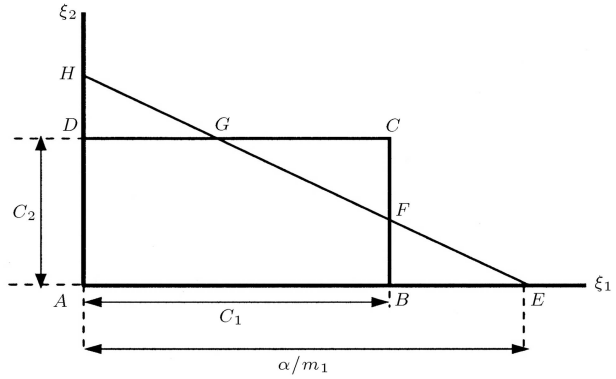


Figure 1. The "cut area" refers to the region within the rectangle ABCD, which also lies below straight line EH, having normal m and parameter α.

Lagrangian Propagation of the Interface Segments

Once the interface has been reconstructed, its motion, by the underlying flow field, must be modeled by a suitable advection algorithm. This can be achieved by either an Eulerian or a Lagrangian scheme. The Lagrangian approach to the propagation of the interface can be best described by considering the way in which the given interface, Equation 26, is convected by the flow. For this purpose, rewrite Equation 26 with superscript (n) attached to all the variables,

$$m_1^{(n)}\xi_1^{(n)} + m_2^{(n)}\xi_2^{(n)} = \alpha^{(n)}, \tag{27}$$

and think of this as the equation for the interface in a given cell at the initial time, t_n. The Lagrangian advection of this interface, by the flow, as time proceeds to $t_{n+1} = t_n + \tau$, will modify it to a new form, which must be calculated. Since, in practice, the time stepping is performed separately in each spatial direction through operator splitting, the advection of the interface along any spatial coordinate, say ξ_1, will be described here.

To make the description simpler, let one suppose that the left face of the cell is located at $\xi_1 = 0$, and the right face at $\xi_1 = h = c_1$. Also, denote the ξ_1 components of the velocity on the faces by U_0 and U_h, as indicated in Formula 21. These are taken to be constant over the entire face to which they are assigned. The formulation in the general coordinate system will be the same as those of Cartesian coordinates [10-12] and only the x_1 and x_2 should be exchanged with ξ_1 and ξ_2. To illustrate the method, the procedure is sketched in Figure 2. The shaded region represents the volume lost by the original cell and gained by the downwind cell, which can be calculated from parallel piped AEFB. With this procedure, the volume fraction field is updated at time t_{n+1}. This Lagrangian method is stable and satisfies the physical constraint on the

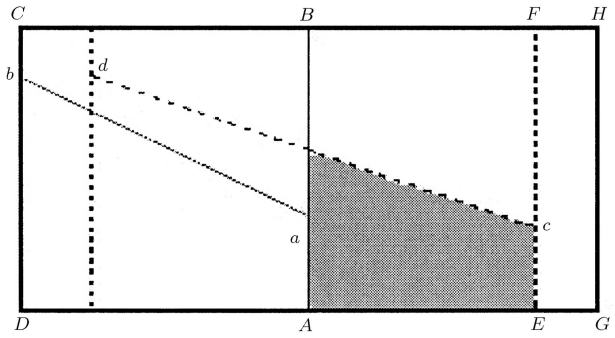


Figure 2. A schematic illustration of the Lagrangian propagation of the interface.

volume fraction, $0 \leq C \leq 1$, when the CFL condition, $(\max|u|)\tau/h < 1/2$, is satisfied. The programming of the Lagrangian method is considerably simplified by the fractional-step strategy, which can be easily applied in the computational domain.

NUMERICAL PROCEDURE

The algorithm follows these steps:

a) Initialize the flow field variables and, then, the numerical procedure in one time step is as follows.

b) Propagate the volume fraction for the new time step, based on the velocity from the previous time step and update phase averaged quantities by the following sub steps:

   1- Normal estimation;

   2- Reconstructing the interface;

   3- Propagating the interface;

   4- Compute new values of C and other averaged quantities.

c) Use a SIMPLE algorithm to solve the flow field governing equations.

d) Repeat b-c.

RESULTS AND DISCUSSION

Dam-Breaking Flow

To examine the performance of the present numerical procedure, the results of two cases of dam-breaking problems are compared with the experimental and other numerical results. To show the robustness and versatility of the code, a random curvilinear grid, as well as a Cartesian grid, is used in the computation of the mentioned problems and results are compared with each other and experiments. Water and air are adopted as the media of the flow. The height and width of the water column of the two cases are (2.25 in, 2.25 in) and (4.5 in, 2.25 in). Corresponding Reynolds numbers, in terms of the height of the liquid region, are 43,129 and 121,986, respectively. In most free surface flows, the grid near the free surface should be fine, but in applications such as dam-breaking, since the flow sweeps through a great part of the domain at all possible inclinations, it is better to use fine grids in the entire flow field. Thus, for simplicity, a uniform Cartesian grid and a random curvilinear grid system are used. For sure, a random general grid is not the proper choice for any fluid flow problem. However, it provides a critical test case for the correctness and robustness of the curvilinear coordinate algorithm when it produces the same results as those of a Cartesian grid. In the case of a (2.25 in, 2.25 in) problem, three grids, namely, 81×41, 101×51 and 201×101 grid points, are employed on the dimensionless domain of $0 \leq x \leq 4$ and $0 \leq y \leq 2$, which means Cartesian grid sizes of $\Delta x = \Delta y = 0.05$ and 0.04, respectively. The random curvilinear grid sizes of 101×51 for the first case and 101×41 for the second case are presented in Figure 3. Also, the conditions applied on each boundary are depicted in Figure 3.

Figure 4 shows the resulting water front, $x_f(\tau)$, of the present study on various grids, for the case of a square water column, $H = W = 2.25$, in both grid systems. The results show no significant difference. The available experimental data [17] and the existing numerical results, such as the standard

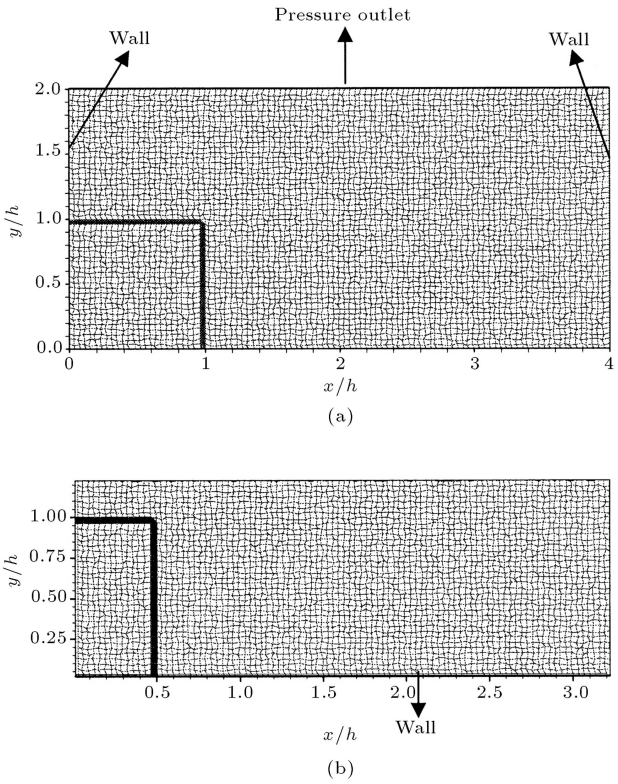


Figure 3. Curvilinear grids of two cases of dam-breaking flows. (a) Case 1 with grids of (101*51) and (b) Case 2 with grids of (101*41).

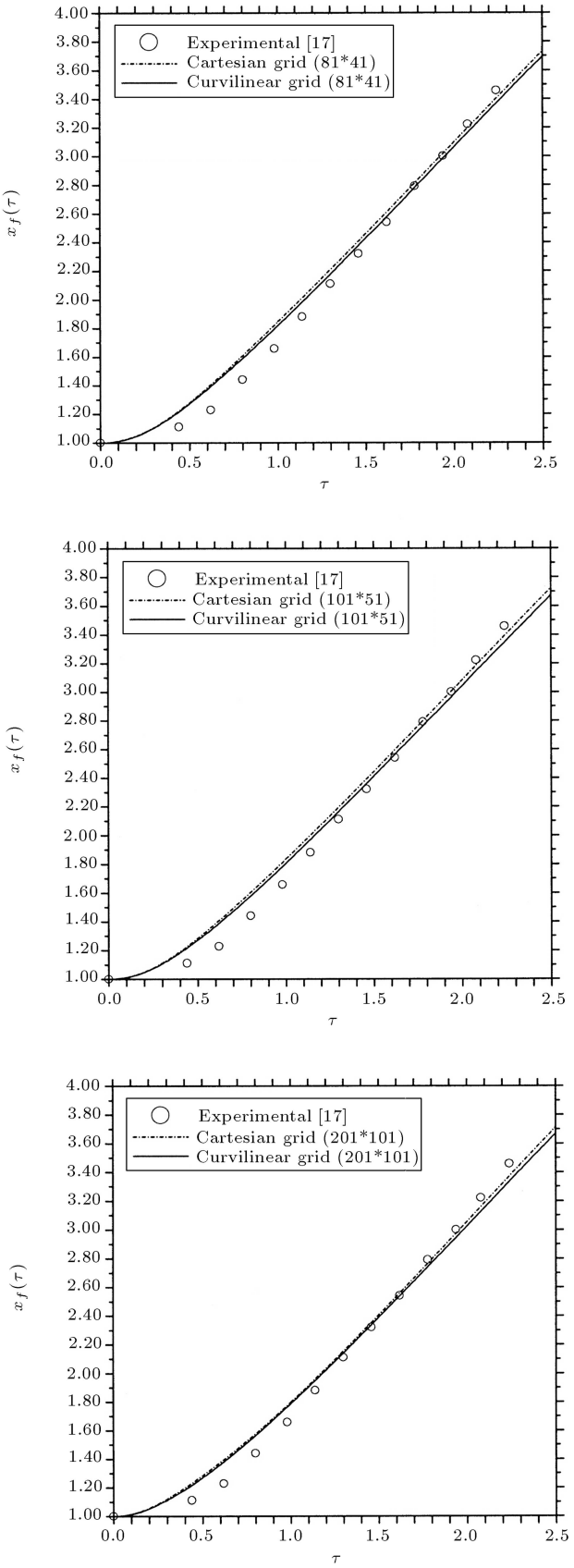


Figure 4. Comparison of waterfront $x_f(\tau)$ among the present results with three grid systems and experimental data at various grid points for the case of (2.5 in * 2.5 in).

MAC method [5] and the modified MAC method [6] are also plotted in Figure 5. As shown in Figure 5, the MAC method overpredicts the experimental results, but the present work shows much better agreement with the experiment.

Considering that the Power-law scheme is of first order accuracy, the discretization accuracy is also of the first order. Therefore, the accuracy of the wave front computation in the present study is of order Δx.

Figures 6 and 7 show the isobar and velocity vector of the first case, (2.5 in * 2.5 in), at various

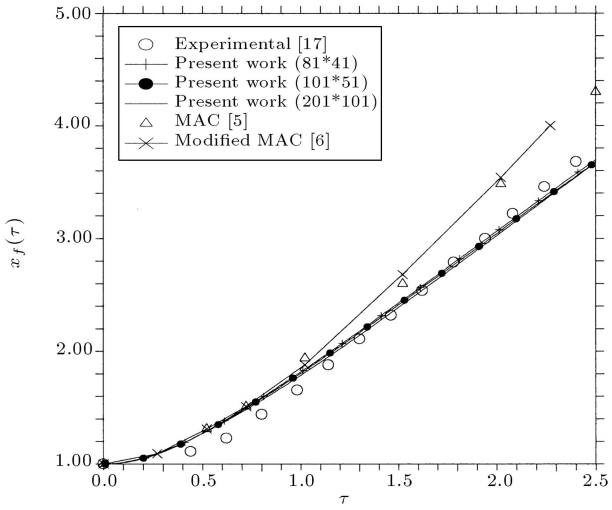
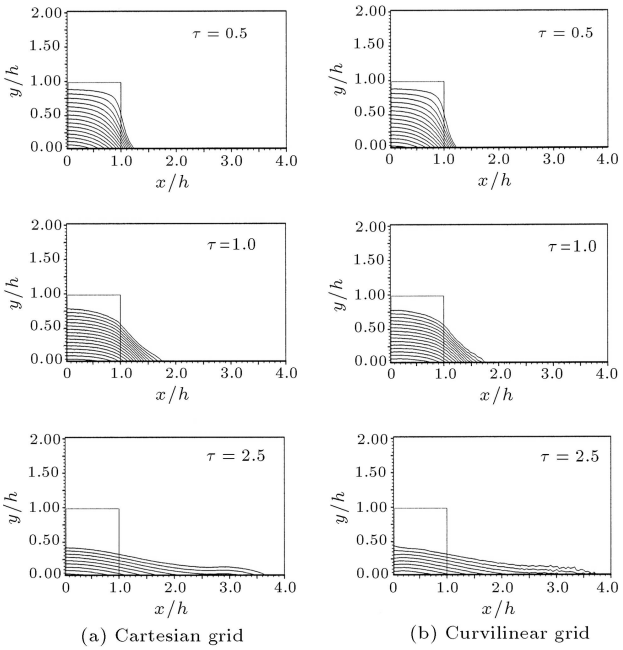


Figure 5. Comparison of waterfront $x_f(\tau)$ from the present results, experimental data and other existing numerical results for the case of (2.5 in *2.5 in).



(a) Cartesian grid (b) Curvilinear grid

Figure 6. Isobars with increment of $\Delta p = 0.05$ at various times for the case of (2.5 in * 2.5 in).

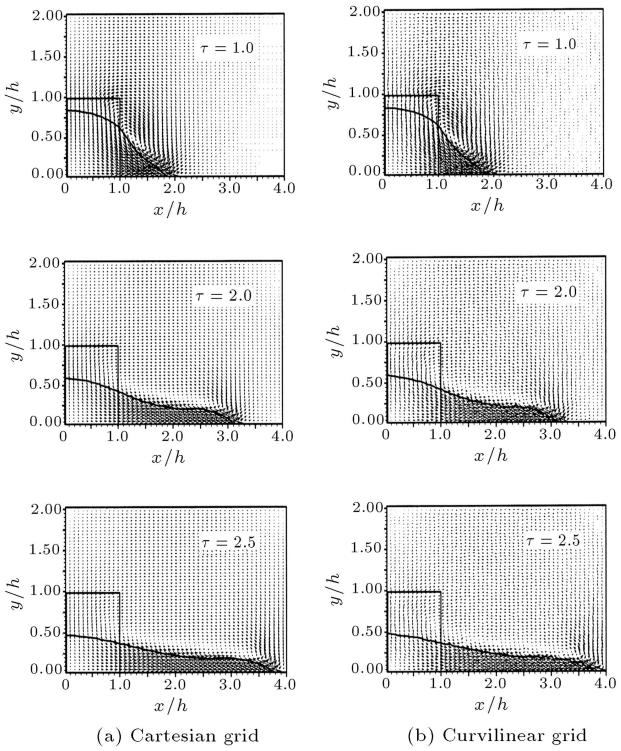


Figure 7. Velocity vectors at various times for the case of (2.5 in * 2.5in).

time steps and compares the results of two systems of grid. The shape and position of isobars, including the waterfront, is almost the same in both grid systems. However, there are small wiggles in the right most regions of isobars, especially at later times. These might be due to an anomaly of randomly generated grids (Figure 3). It is obvious that the pressure is essentially near zero in the air region and this can be attributed to the negligible density of the air, as compared to the water. It is interesting to note, from velocity vectors in Figure 7, that the liquid motion induces a vortex in a layer of air adjacent to the free surface. This finding is consistent with physical reasoning.

In the second case of (4.25 in * 2.25 in), two grids, namely 101×41 and 201×81 grid points, are employed on the dimensionless domain of $0 \leq x \leq 10/3$ and $0 \leq y \leq 4/3$ in both Cartesian and curvilinear systems. Figure 8 presents a comparison of the present results of the Cartesian and curvilinear grids and shows good agreement between the results of these two grid systems. Figure 9 shows the comparison of a waterfront from different sources, including the results of the present work, the previous VOF code of Hirt and Nicholls [8] and the modified MAC method. It is obvious from Figure 9 that the Hirt and Nichols code [8] overpredicts the experimental data. The MAC results are more comparable with the experimental data in Figure 9 than with those of Figure 5, however, the

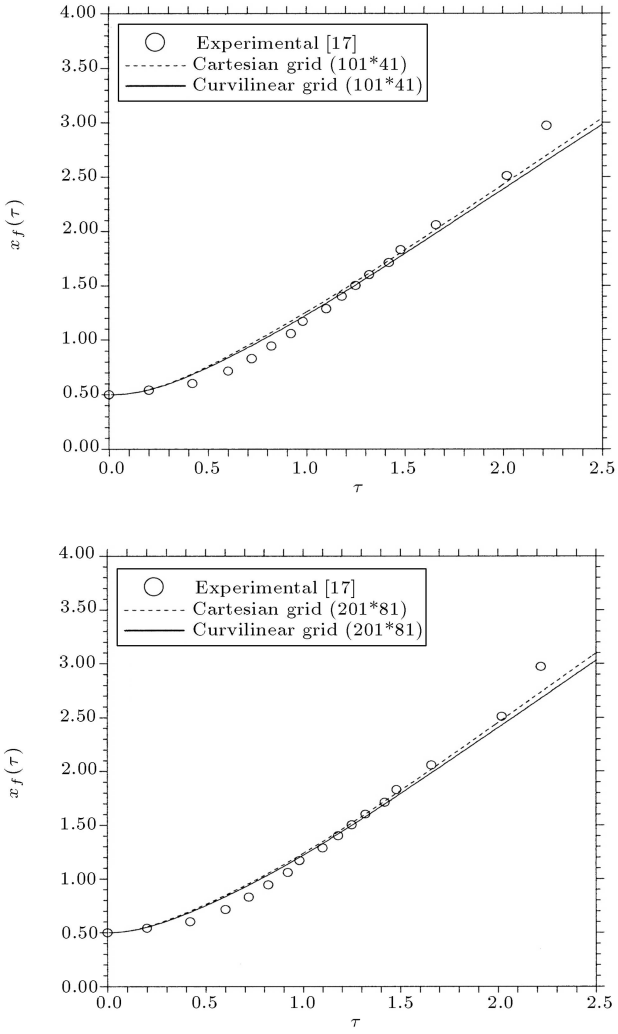


Figure 8. Comparison of waterfront $x_f(\tau)$ among the present results with two grid systems and experimental data at various grid points for the case of (2.5 in * 2.5 in).

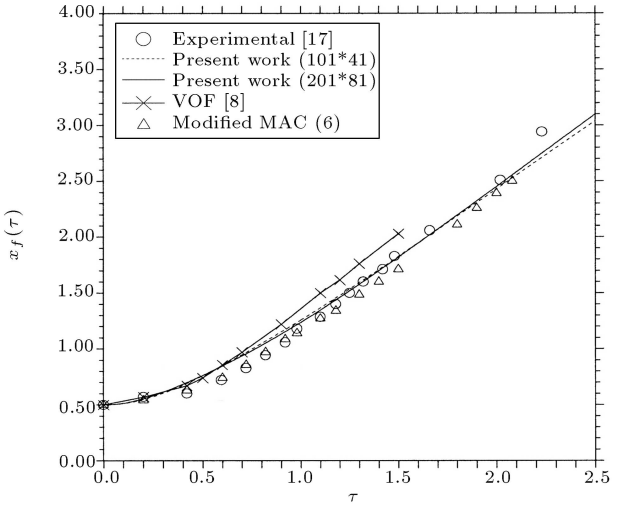


Figure 9. Comparison of waterfront $x_f(\tau)$ among the present results and experimental data and other existing numerical results for the case of (4.5 in * 2.5 in).

consistency of the present results with experiments are almost the same in both cases.

Flow Under Sluice Gate

To present another example, which may be of some practical interest, consider a model of an arc gate with a dimensionless radius of 0.69, with the center at (1.67, 0.81), under a dam of the dimensionless height of 1.4, as shown in Figure 10. This figure shows the dimensionless geometry, different conditions on each boundary and the grid system. This example obviously shows the benefits of using curvilinear grids and the simplifications provided by the computational domain approach. Figure 11 presents the pressure distribution of the flow at various times. It can be seen that the free surface contracts around the exit, i.e. vena contracta, and at later times ($\tau = 1.5$, $\tau = 2.0$), it re-expands as physically expected. In Figure 12, the velocity vectors of flow at various times are shown and it is obvious that, after the flow discharges with high velocity from the gate at some distance downstream, the fluid velocity decreases and, thus, the elevation of the water front increases accordingly. It is interesting that the wiggles

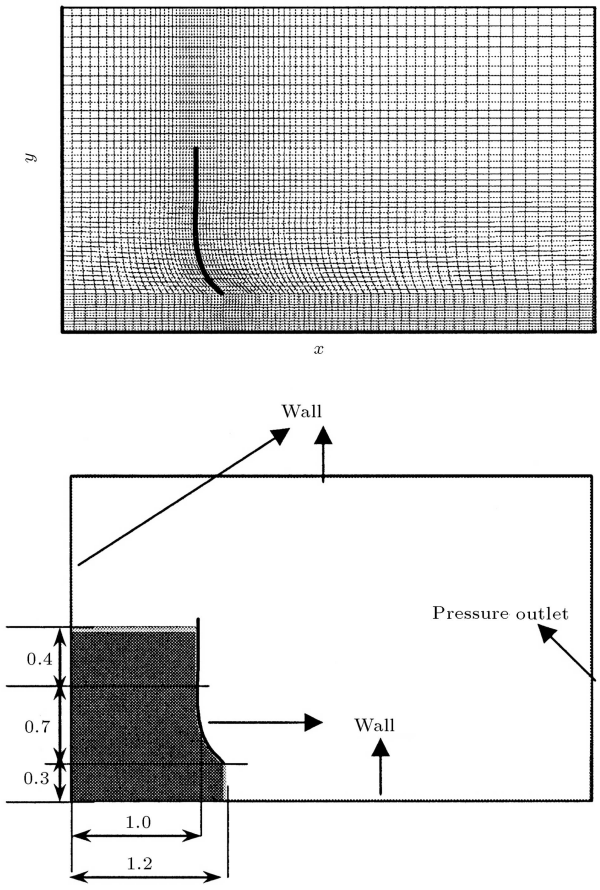


Figure 10. Mesh system and dimensionless geometry of gate.

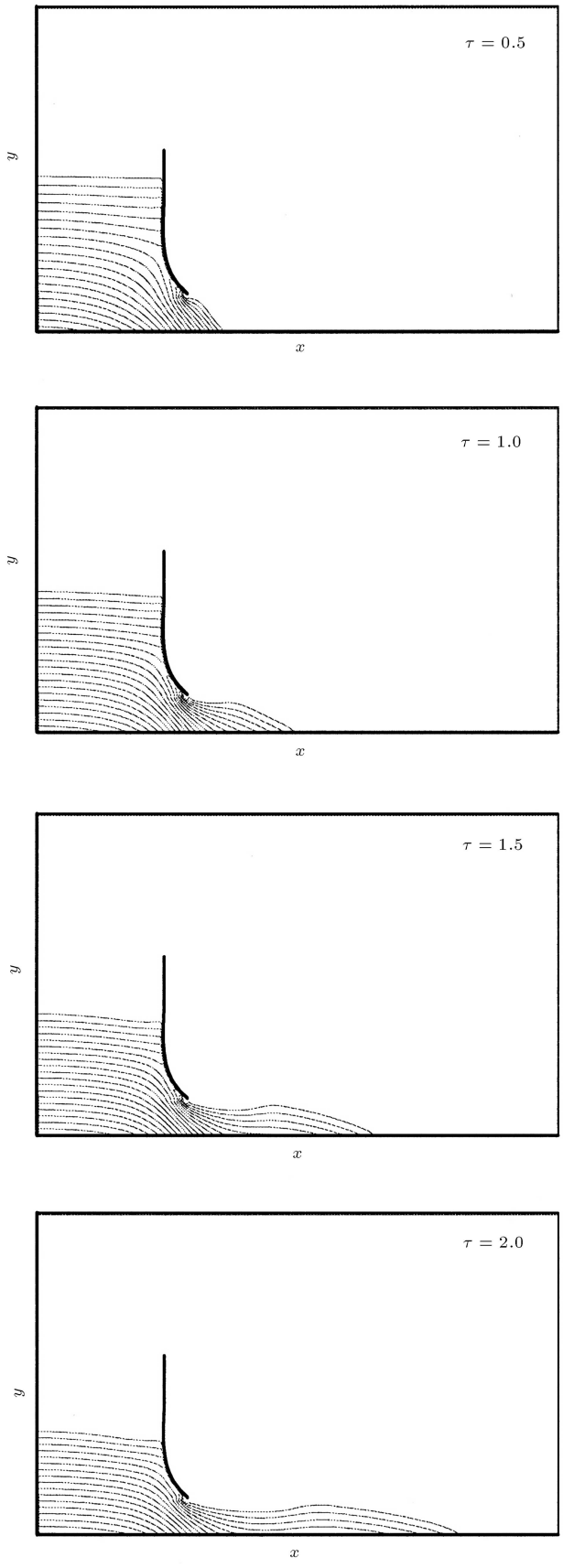


Figure 11. Isobars with increment of $\Delta p = 0.05$ at various times for flow out of opening gate.

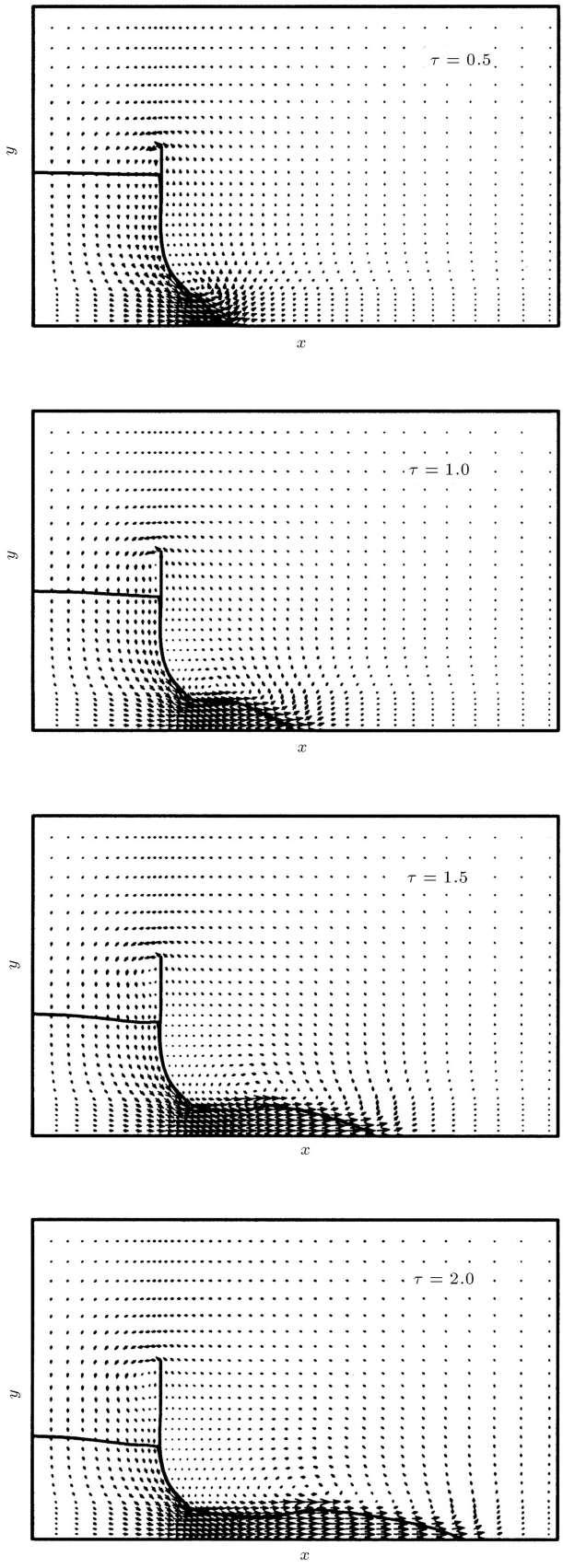


Figure 12. velocity vectors at various times for flow out of opening gate.

in Figures 6 and 7 are absent in this case, indicating that these errors originate from the randomly generated grids in Figure 3.

Water Impact of Circular Cylinder

Another applicable example is the water impact of circular cylinders, which has been studied extensively to clarify wave impact on members of offshore structures in a splash zone. The circular cylinder used in this calculation has a radius of 5.5 m. It is dropped with a constant velocity of 10 m/s. A grid system, with elements of (180 × 250) and the schemes of Power-Law and QUICK, were also used. In the present computational scheme, in order to avoid unsteady calculation and grid regeneration, a circular cylinder, as a rigid body, is fixed to the grid system and, instead of moving the body downward, the free surface is forced to move upward. These conditions can be realized by imposing a continuous inflow of the fluid from the lower boundary of the physical domain and the moving side walls. The pressure on the upper boundary is also fixed. The numerical results of this study, together with the published experimental, analytical and other numerical results, are shown in Figure 13. In this figure, the non-dimensional impact force, (f), namely the slamming coefficient, $(C_S = \frac{f}{1/2\rho V^2 D})$, is plotted as a function of the non-dimensional immersion height. As shown in this figure, the theoretical model of Von-Karman indicates an initial slamming coefficient, $C_{So} = \pi$ [18], whereas that of Wagner and Fabula gives $C_{So} = 2\pi$ [19,20]. The experimental coefficient, C_{So}, exhibits a considerable degree of scatter, ranging from 3.5 to 6.5 [21]. Based on experimental data, Campbell [21] proposed an empirical formula for the

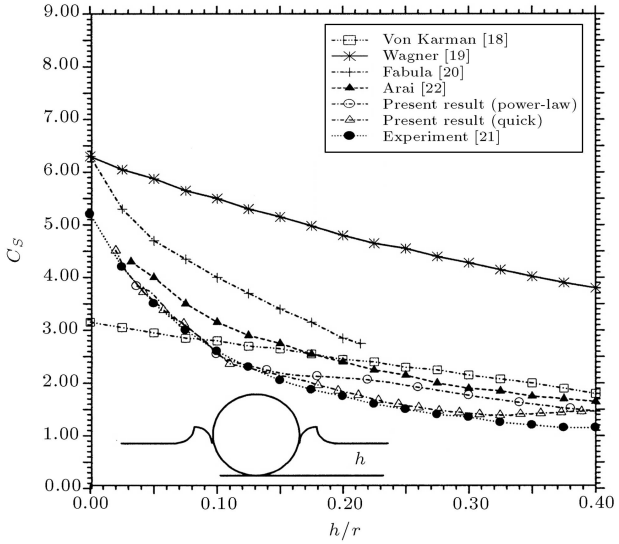


Figure 13. Slamming coefficients of circular cylinder ($r = 5.5$ m, $V = 10$ m/s) versus immersion height.

slamming coefficient, as shown in this figure. The results of a finite difference computation of Arai [22], also included in Figure 13, is based on inviscid flow. The present work results, of both Power-Law and QUICK schemes, are shown and the agreement with experimental data is good, especially for QUICK schemes at h/r> 0.15.

With regard to the above examples, it can be expressed that the new method of PLIC-VOF interface tracking in the computational domain is robust and accurate enough to handle critical test cases, such as random grids, and produce acceptable results for applied and non-rectangular geometries.

CONCLUSION

A single set of dimensionless equations is derived to handle both liquid and air phases in viscous incompressible free surface flows. The momentum equations are solved using the SIMPLE method in a staggered grid. The Lagrangian approach in the computational domain is also applied in the context of a VOF method, to resolve the free surface effects. It is indicated that, with this new implementation of the PLIC-VOF method in the computational domain, it is an easy task to split the propagation algorithm, because the same algorithm as in the Cartesian grid is applied in the computational grid approach. The results of two cases of dam-breaking problems are presented, which show good agreement with experimental data and other numerical methods. The results for a random curvilinear grid and a Cartesian grid show the same agreement with experimental results. The application of a flow of water under a curved gate and a water impact problem, using curvilinear coordinates, are also presented. It can be concluded that the present method and the code are robust, produce results of good quality and, also, can be easily implemented.

REFERENCES

1. Frits, M.J. and Bories, J.P. "The Lagrangian solution of transient problems in hydrodynamics using a triangular mesh", *J. Comput. Phys.*, **31**, pp. 173-215 (1979).
2. Fyfe, D.E., Oran, E.S. and Fritts, M.J. "Surface tension and viscosity with Lagrangian hydrodynamics on triangular mesh", *J. Comput. Phys.*, **76**, pp. 349-384 (1988).
3. Lewis, R.W., Navti, S.E. and Taylor, C. "A mixed Lagrangian Eulerian approach to modeling fluid flow during mould filling", *Numer. Meth. Fluids*, **25**, pp. 931-952 (1997).
4. Chen, C.W., Li, C.R., Han, T.H., Shei, C.T., Hwang, W.S. and Houng, C.M. "Numerical simulation of filling pattern for an industrial die casting and its comparison with the defects distribution of an actual casting", *Trans. Am. Foundrymen's Soc.*, **104**, pp. 139-146 (1994).
5. Harlow, F.H. and Welch, J.E. "Numerical calculation of time dependent viscous incompressible flow of fluid with free surface", *Phys. Fluids*, **8**, pp. 2182-2189 (1965).
6. Nakayama, T. and Mori, M. "An Eulerian finite element method for time dependent free surface problems in hydrodynamics", *J. Numer. Meth. Fluids*, **22**, pp. 175-194 (1996).
7. Harlow, F.H., Amsden, A.A. and Nix, J.R. "Relativistic fluid dynamics calculations with the particle-in-cell technique", *J. Comput. Phys.*, **20**, pp. 119-129 (1976).
8. Hirt, C.W. and Nicholls, B.D. "Volume of fluid (VOF) method for dynamics of free boundaries", *J. Comput. Phys.*, pp. 39-201 (1981).
9. Noh, W.F. and Woodward, P. "SLIC (Simple Line Interface Calculation)", *Proceedings of the Fifth International Conference on Fluid Dynamics*, A.I. Vande Vooren and P.J. Zandbergen, Eds., Lecture Notes in Physics, **59**, Springer-Verlag, Berlin, p. 330 (1976).
10. Rudman, M. "Volume-tracking methods for interfacial flow calculations", *International Journal for Numerical Methods in Fluids*, **24**, pp. 671-691 (1997).
11. Gueyffier, D., Li, J., Nadim, A., Scardovelli, R. and Zaleski, S. "Volume-of-fluid interface tracking with smoothed surface stress methods for three-dimensional flows", *Journal of Computational Physics*, **152**, pp. 423-456 (1999).
12. Nikseresht, A.H., Alishahi, M.M. and Emdad, H. "Volume-of-fluid interface tracking with Lagrangian propagation for incompressible free surface flows", *Journal of Science and Technology*, **12**(2), pp. 131-140 (Spring 2005).
13. Kothe, D.B., Rider, W.J., Mosso, S.I., Brock, J.S. and Hochstein, J.I. "Volume tracking of interface having surface tension in two and three dimensions", *AIAA*, 96-0859 (1996).
14. Patankar, S.V. "Numerical heat transfer and fluid flow", *Hemisphere*, Washington, DC, USA (1980).
15. He, P. and Salcudean, M. "A numerical method for 3D viscous incompressible flows using non-orthogonal grids", *International Journal For Numerical Methods in Fluids*, **18**, pp. 449-469 (1994).
16. Mousavi Mirkalaee, S.M. "Numerical study of two dimensional laminar incompressible flow around hovercraft", MSc Thesis, Department of Mechanical Engineering, Shiraz University (Oct. 2001).
17. Martin, J.C. and Moyee, W.J. "An experimental study of the collapse of liquid columns on a rigid horizontal plane", *Philos. Trans. Roy. Soc. London*, **244A**, pp. 312-324 (1952).
18. Von Karman, T., *The Impact on Seaplane Float During Landing*, NACA TN321 (1929).

19. Wagner, H., *Landing of Seaplanes*, NACA TM 622 (1931).

20. Fabula, A. "Ellipse fitting approximation of two-dimensional normal symmetric impact of rigid bodies on water", *Fifth Midwestern Conference on Fluid Mechanics*, University of Michigan Press (1957).

21. Campbell, I.M.C. and Weynberg, P.A. "Measurement of parameters affecting slamming", *Wolfson Unit for Marine Technology and Industrial Aerodynamics*, Univ. of Southampton, Report No. 440 (1980).

22. Makoto, A., Liang-Yee, C. and Yoshiyuki, I. "A computing method for the analysis of water impact of arbitrary shaped bodies", *Journal of The Society of Naval Architects of Japan*, **176**, pp. 233-240 (1994).