Turkish J. Eng. Env. Sci.
31 (2007) , 311 – 321.

Total Volume Conservation in Simulation of Unsteady Free-Surface Flows

Ender DEMİREL
Eskişehir Osmangazi University, Civil Engineering Department, Eskişehir-TURKEY
e-mail: edemirel@ogu.edu.tr
İsmail AYDIN
Middle East Technical University, Civil Engineering Department, Ankara-TURKEY

Received 26.04.2007

Abstract

Free surface tracking methods used in the numerical simulation of unsteady free surface flows may introduce sources or sinks resulting in changes in total fluid volume in the computational domain. A computational model is developed for incompressible, 2-dimensional, unsteady free surface flows to investigate the conditions of total volume conservation. The model is based on finite volume discretization of the Navier-Stokes equations coupling momentum and mass conservation. Free surface position is tracked using a depth-integrated continuity equation. Possible free surface cell configurations and a solution procedure for continuity are described. A flux-corrected transport method is applied to the free surface solution to maintain numerical stability and eliminate unphysical surface oscillations. The discretization scheme and the computer code are validated in lid-driven cavity flow. Liquid sloshing in a partially filled rectangular tank and dam-break flows are simulated. Numerical solutions preserving total volume are presented. Computed free surface profiles are verified by experimental data.

Key words: Free-surface flow, Unsteady flow, Dam-break, Sloshing tank, Finite volume method

Introduction

Multi-dimensional free-surface flow computations in large domains such as earthquake excited fluid-structure interactions in dam reservoirs, tsunami generation in deep water, water wave generations due to land slides, and dam-break simulations are becoming common tools for risk analysis in engineering practice. In order to simulate the fluid flow phenomena with satisfactory resolution in space and time domains, the Navier-Stokes equations are solved numerically using a suitable discretization technique such as finite difference, finite volume, or finite element methods. The finite volume method is widely used in computational fluid dynamics (CFD) applications due to the advantages of integrated equations to enforce conservation of physical quantities in arbitrary control volumes.

Free surface simulation became a popular research area with the rapid development of computing power. There are several methods to track the free surface position such as Marker-And-Cell (MAC), Kinematic Boundary Condition (KBC), Depth-Integrated Continuity equation (DIC), Volume of Fluid (VOF), moving grid techniques, and level set methods.

The MAC method described by Harlow and Welch (1965) was the first to track the free surface on a discretized domain. This method uses massless marker particles, which are used to indicate the fluid configuration showing which region is occupied by fluid and which region is empty. The marker particles are moved to new positions using local fluid velocities. Chan and Street (1970) developed the SUMMAC method, which is a modification of the MAC method, using an interpolation technique for

free surface velocity instead of solving the continuity equation on the free surface.

VOF was first introduced by Hirt and Nichols (1981) and it can be applied to multi-fluid problems. It can handle complex physical flows such as breaking waves and splashing. In the VOF method, a transport equation of a volume fraction parameter F is introduced with values between zero and one to describe the position of the free surface. Computational cells are labeled according to the F value in the cell.

The free surface position is defined by the local height function in the KBC method. The no-mass-flux condition on the free surface is satisfied based on the concept that all fluid particles on the free surface always remain on the surface. DIC is obtained by integrating an incompressible continuity equation in a liquid column from the bottom to the free surface. This method also uses the height function, similar to KBC. DIC and KBC formulations are valid if water depth is a single valued function of horizontal coordinates in each computational cell. Free surface formations like bubbles, drops, and breaking waves cannot be handled due to the single-valued depth function. However, these methods are convenient for large domain computations since they are easy to implement, computationally efficient, and require minimum memory storage.

Exact volume conservation is not guarantied in the above methods and it can be difficult to preserve the total fluid volume in the computational domain for long simulation durations. Free surface tracking methods used in the simulation of free surface flows may produce sources or sinks during the computations, causing the total fluid volume to change over time. Such observations have been reported by Wang et al. (2007) and Kleefsman et al. (2005).

The authors of this paper have also faced difficulties in preserving total fluid volume in the computational domain in an attempt to compute water surface deformations in dam reservoirs due to earthquake accelerations. After some numerical tests it was recognized that there may be 2 sources of error causing variations in total fluid volume in the computational domain. One source of error is due to inappropriate boundary conditions applied at the far field of a large domain extending to infinity. The second source of error may originate from the free surface tracking algorithms. The motivation behind the present study was to investigate the conditions of satisfying exact volume conservation in a numerical simulation of incompressible free surface flows so that the total fluid volume in the computational domain remains the same throughout the simulation time. The research done in relation to free surface algorithms is presented in this paper. The main aim of the study is to develop a computer code for unsteady free surface flows suitable for large spatial domains such as dam reservoirs.

Governing Equations

Mass conservation is expressed as volume conservation for incompressible flows considered in this study. The equations of motion for 2-dimensional incompressible flows in a vertical plane are given as

$$a_x + \frac{\partial u}{\partial t} + \overrightarrow{\nabla} \cdot \left(u \overrightarrow{V} \right) = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \nu \nabla^2 u \qquad (1)$$

$$a_z + \frac{\partial w}{\partial t} + \overrightarrow{\nabla} \cdot \left(w \overrightarrow{V} \right) = -g - \frac{1}{\rho} \frac{\partial p}{\partial z} + \nu \nabla^2 w \qquad (2)$$

$$\overrightarrow{\nabla} \cdot \overrightarrow{V} = 0 \qquad (3)$$

where x and z are coordinate axes in horizontal and vertical directions, respectively, a_x and a_z are ground accelerations, u and w are velocity components, $\overrightarrow{V}$ is the velocity vector, p is pressure, t is time, g is gravitational acceleration, ν is kinematic viscosity, ρ is fluid density, and $\overrightarrow{\nabla}$ is the del operator. Ground accelerations are included to represent earthquake excitations or accelerations due to shaking of experimental tanks. The computational domain is assumed to move with the ground.

Mathematical Formulation

To obtain finite volume formulation of free surface flow, Eqs. (1), (2), and (3) are integrated applying Gauss divergence theorem.

$$\frac{\partial}{\partial t} \int\limits_{cv} u d\forall + \int\limits_{cs} u \overrightarrow{V} \cdot \overrightarrow{dA} = -\frac{1}{\rho} \int\limits_{cv} \frac{\partial p}{\partial x} d\forall + \nu \int\limits_{cs} \overrightarrow{\nabla} u \cdot \overrightarrow{dA} - \int\limits_{cv} a_x d\forall \qquad (4)$$

$$\frac{\partial}{\partial t} \int\limits_{cv} w d\forall + \int\limits_{cs} w \overrightarrow{V} \cdot \overrightarrow{dA} = -\frac{1}{\rho} \int\limits_{cv} \frac{\partial p}{\partial z} d\forall + \nu \int\limits_{cs} \overrightarrow{\nabla} w \cdot \overrightarrow{dA} - \int\limits_{cv} (g + a_z) d\forall \qquad (5)$$

$$\int_{cs} \overrightarrow{V} \cdot \overrightarrow{dA} = 0. \tag{6}$$

The integrated equations are discretized on a staggered grid arrangement shown in Figure 1.

$$u_{i,j}^{n+1} = F_{i,j}^{n} + \Delta t\{[(p_{i,j}^{n+1} - p_{i+1,j}^{n+1})/\rho]/\Delta x_{i+1/2} - a_x\} \tag{7}$$

$$w_{i,j}^{n+1} = G_{i,j}^{n} + \Delta t\{[(p_{i,j}^{n+1} - p_{i,j+1}^{n+1})/\rho]/\Delta z_{j+1/2} - (g + a_z)\} \tag{8}$$

$$F_{i,j}^{n} = u_{i,j}^{n} + \Delta t\{[\nu(Difu)_{i,j}^{n} - (Conu)_{i,j}^{n}]/(\Delta x_{i+1/2}\,\Delta z_j)\} \tag{9}$$

$$G_{i,j}^{n} = w_{i,j}^{n} + \Delta t\{[\nu(Difw)_{i,j}^{n} - (Conw)_{i,j}^{n}]/(\Delta x_i\,\Delta z_{j+1/2})\} \tag{10}$$

$$(u_{i,j}^{n+1} - u_{i-1,j}^{n+1})/\Delta x_i + (w_{i,j}^{n+1} - w_{i,j-1}^{n+1})/\Delta z_j = 0. \tag{11}$$

where Δt is the time step, Δx and Δz are mesh sizes, and *Dif* and *Con* represent diffusive and convective fluxes, respectively. To utilize the advantage of the staggered grid system, convenient control volumes are selected for each equation as shown in Figure 1. First order derivatives in diffusive fluxes are discretized using second order polynomial approximation on a variable mesh. Convective fluxes are evaluated by first order upwind (FOU) and by QUICK approximation using a 3-point upstream-weighted quadratic interpolation for cell face values. The code can switch between FOU and QUICK, depending on user preferences. Detailed descriptions on temporal and spatial discretizations can be found in Li and Baldacchino (1995).

Pressure solution is obtained from the Poisson equation for pressure. The discretized form of the Poisson equation for pressure is obtained by substituting Eqs. (7) and (8) into (11).

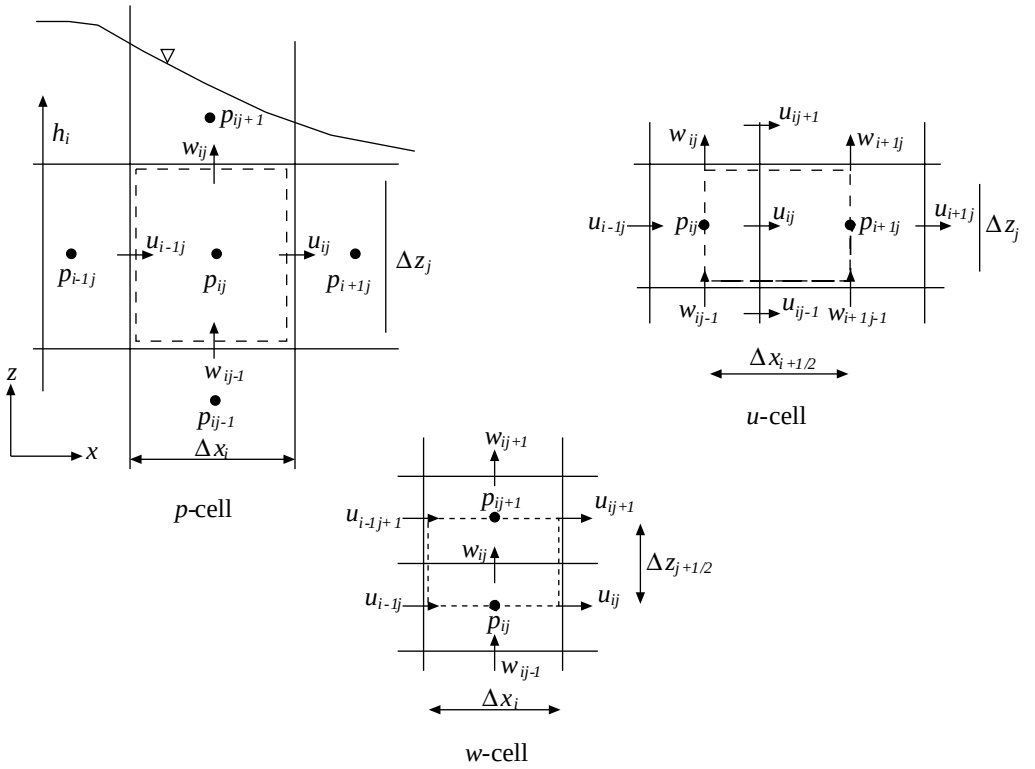

Figure 1. Computational grid and control volumes.

$$[(p_{i,j}^{n+1} - p_{i+1,j}^{n+1})/\Delta x_{i+1/2} - (p_{i-1,j}^{n+1} - p_{i,j}^{n+1})/\Delta x_{i-1/2}]/\Delta x_i + [(p_{i,j}^{n+1} - p_{i,j+1}^{n+1})/\Delta z_{j+1/2} - (p_{i,j-1}^{n+1} - p_{i,j}^{n+1})/\Delta z_{j-1/2}]/\Delta z_j = -\left[\frac{F_{i,j}^n - F_{i-1,j}^n}{\Delta x_i} + \frac{G_{i,j}^n - G_{i,j-1}^n}{\Delta z_j}\right]\frac{\rho}{\Delta t} \quad (12)$$

The momentum equations (7) and (8) and the pressure Poisson equation (12) are solved by sequential iterations. A computer code in FORTRAN language is developed by the authors to perform the computations.

Free Surface Tracking

KBC for the free surface assumes that all fluid particles on the free surface will remain on the free surface, implying no mass flux across the free surface boundary. Mathematically it is formulated as

$$\frac{\partial h}{\partial t} + u_s \frac{\partial h}{\partial x} = w_s \quad (13)$$

where u_s and w_s are horizontal and vertical velocities on the free surface and h is the water depth defined as a single-valued function at the center of a liquid column. The KBC is simple and effective to track the free surface position. The space centered and forward-in-time difference approximation of Eq. (13) is unstable because of a negative truncation error. This truncation error is compensated for by adding a positive diffusion term. The complete finite difference form of Eq. (13) is given by Nichols and Hirt (1973). At the beginning of this study KBC was used to trace the free surface variations. However, it was found that the total fluid volume in the computational domain was continuously changing as the computations proceeded.

As an alternate approach, the integral continuity equation is used to replace Eq. (13). Equation (3) is integrated from the bottom ($z = 0$) to the free surface ($z = h$) to obtain the DIC equation:

$$\frac{\partial h}{\partial t} + \frac{\partial}{\partial x}\int_0^h u dz = w_0 \quad (14)$$

where w_0 is the velocity of the bottom boundary set to zero throughout this study. Equation (14) can be discretized as

$$h_i^{n+1} = h_i^n - \Delta t\left[(q_{i+1/2}^{n+1} - q_{i-1/2}^{n+1})/\Delta x_i\right] \quad (15)$$

Volume fluxes on cell faces are evaluated from

$$q_{i+1/2}^{n+1} = \left[\int_0^h u dz\right]_{i+1/2} = \sum_{j=1}^{j_s} u_{i,j}^{n+1}\Delta z_j \quad (16)$$

$$q_{i-1/2}^{n+1} = \left[\int_0^h u dz\right]_{i-1/2} = \sum_{j=1}^{j_s} u_{i-1,j}^{n+1}\Delta z_j \quad (17)$$

where j_s is the index of the surface cell at the i^{th} column. Fluxes on the cell faces are computed directly by using velocities on the cell faces taking advantage of the staggered grid arrangement.

In the numerical solution, location of the free surface must be followed by an appropriate algorithm identifying the computational cells as fluid cell (F) when it is completely filled by fluid, empty cell (E) when there is no fluid, surface cell (S) when the cell is partially filled, and boundary cell (B) on the solid boundaries (Figure 2). When applying any numerical procedure on a computational cell, 4 neighbors, namely, east, west, north, and south contiguous cells, must be identified. Integration and interpolation practices depend on the type of neighboring cells. A standard solution procedure is applicable for F cells when the neighbors are also composed of F or S cells. Special procedures are required to obtain velocity and pressure when E cells appear as neighbors. A common approach used to calculate the velocity components at the interfaces of E and F cells is extrapolation of the velocity components from the closest available velocities obtained from momentum solutions. A detailed description of the extrapolation procedure can be found in Armenio (1997).

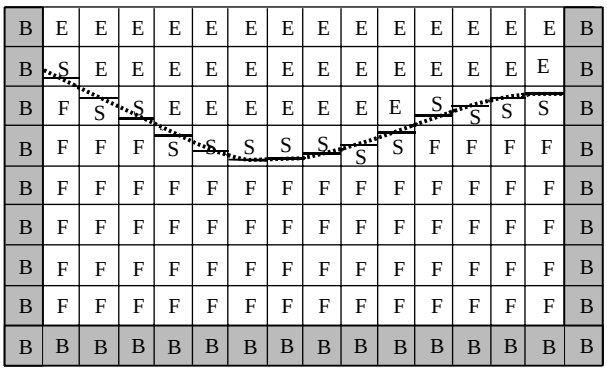

Figure 2. Computational cells and labeling.

The pressure equation is solved in F cells, the horizontal momentum equation is solved at F-F, F-S, and S-S interfaces, and the vertical momentum equation is solved at F-F and F-S interfaces. Pressure on the free surface in S cells is computed from the free surface stress conditions given by Batchelor (1967). A non-zero pressure just on the free surface is evaluated from

$$p_{fs} = 2\mu\, \partial w/\partial z \tag{18}$$

where μ is the dynamic viscosity. Then, the pressure at the centre of the surface cell is calculated by linear interpolation

$$p_s = \eta p_{fs} + (1-\eta) p_F \tag{19}$$

where $\eta = h/d$. A detailed description of free surface boundary conditions can be found in Chen et al. (1995), Kleefsman (2005), Griebel (1997), and Tome et al. (2001).

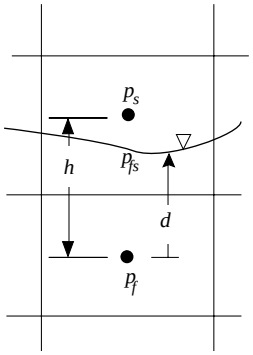

Figure 3. Pressure interpolation in a surface cell.

Numerical Solution

The momentum equations, pressure equation, and free surface equation are solved by sequential iterations using an explicit procedure. In order to ensure computational stability of the numerical algorithm, a combined stability condition is imposed on the time step based on the convection and diffusion processes (Chan and Street, 1970).

$$\Delta t = CFL \min\left\{ \frac{\Delta x_{i+1/2}}{|u_{i,j}|}, \frac{\Delta z_{j+1/2}}{|w_{i,j}|}\ ,\ \ \frac{1}{2\nu}\frac{\Delta x_i^2 \Delta z_j^2}{(\Delta x_i^2+\Delta z_j^2)}, \frac{2\Delta x_i \Delta z_j}{c(\Delta x_i+\Delta z_j)} \right\} \tag{20}$$

where CFL is the Courant-Friedrichs-Lewy number, which is fixed as 0.5 throughout this study, and c is the surface wave celerity. The pressure equation (12) is solved by the Point Successive-Over-Relaxation (PSOR) method.

In some cases, sharp deformations on the free surface may occur and this may lead to stability problems. One of the most effective techniques to reduce stability problems is the Flux Corrected Transport (FCT) method, which was originally developed by Boris and Book (1973) for the treatment of shock waves in compressible flow. This technique diffuses the solution throughout the computational domain and then anti-diffuses the solution using a flux-limiter.

Surface elevations are computed from the free surface equation (Eq. (15)) and the FCT method is applied to surface elevations throughout the domain. The diffusive flux is computed as

$$d_{i+1/2}^{n+1} = \lambda(h_{i+1}^{n+1} - h_i^{n+1}) \tag{21a}$$

$$d_{i-1/2}^{n+1} = \lambda(h_i^{n+1} - h_{i-1}^{n+1}) \tag{21b}$$

where λ is the diffusion coefficient computed from the equation suggested by Book et al. (1975):

$$\lambda = CFL/(1+2CFL)^2 \tag{21c}$$

and the diffused solution is obtained as

$$h_i^{n+1} = h_i^{n+1} + d_{i+1/2}^{n+1} - d_{i-1/2}^{n+1} \tag{22}$$

Then the diffused solution is corrected to eliminate excess diffusion using flux limiters defined by Yost and Rao (1999):

$$h_i^{n+1} = h_i^{n+1} + L_{i+1/2}^{n+1} + L_{i-1/2}^{n+1} \tag{23}$$

where

$$L_{i+1/2}^{n+1} = Sgn(\Delta h_{i+1/2}^{n+1})$$

$$\max\left\{0, \min\left[\begin{array}{l} Sgn(\Delta h_{i+1/2}^{n+1})(h_i^{n+1} - h_{i-1}^{n+1}), \\ Sgn(\Delta h_{i+1/2}^{n+1})(h_{i+2}^{n+1} - h_{i+1}^{n+1}), \\ \lambda\left|\Delta h_{i+1/2}^{n+1}\right| \end{array}\right]\right\} \tag{24a}$$

$$L_{i-1/2}^{n+1} = Sgn(\Delta h_{i-1/2}^{n+1})$$

$$\max\left\{0, \min\left[\begin{array}{l} Sgn(\Delta h_{i-1/2}^{n+1})(h_i^{n+1} - h_{i-1}^{n+1}), \\ Sgn(\Delta h_{i-1/2}^{n+1})(h_{i-1}^{n+1} - h_{i-2}^{n+1}), \\ \lambda \left|\Delta h_{i-1/2}^{n+1}\right| \end{array}\right]\right\} \tag{24b}$$

and

$$\Delta h_{i+1/2}^{n+1} = h_{i+1}^{n+1} - h_i^{n+1} \tag{24c}$$

$$\Delta h_{i-1/2}^{n+1} = h_i^{n+1} - h_{i-1}^{n+1} \tag{24d}$$

The FCT correction is effective to maintain the stability of computations only when the free surface has steep slopes as in the case of a dam break solution. When the free surface has mild slopes FCT has no influence on the solution. Equation (15) is solved for each liquid column in the domain. Numerical fluxes are evaluated at the side faces of the liquid column and integrated from the bottom to the free surface using Eqs. (16) and (17).

Lid-driven cavity flow

The computational algorithm is first tested in a lid-driven square cavity problem. The purpose of the cavity solution is to verify the momentum and pressure solution procedures in an internal flow without a free surface. The convective fluxes are computed by QUICK and also by First Order Upwind (FOU) to exhibit the improvement provided by QUICK. The stream lines for $R_e = 400$ and comparison of u-velocity distribution along the vertical line through $x = L/2$ are shown in Figure 4. Velocity distributions obtained by QUICK and FOU are given together with the results reported by Ghia et al. (1982) for comparison. The numerical solution is obtained on a 128×128 uniform grid as in the solution given by Ghia et al. Although QUICK causes some increase in CPU time, it is superior to FOU by producing more accurate solutions for the same grid resolution. The same accuracy may be achieved by FOU using a more refined mesh system.

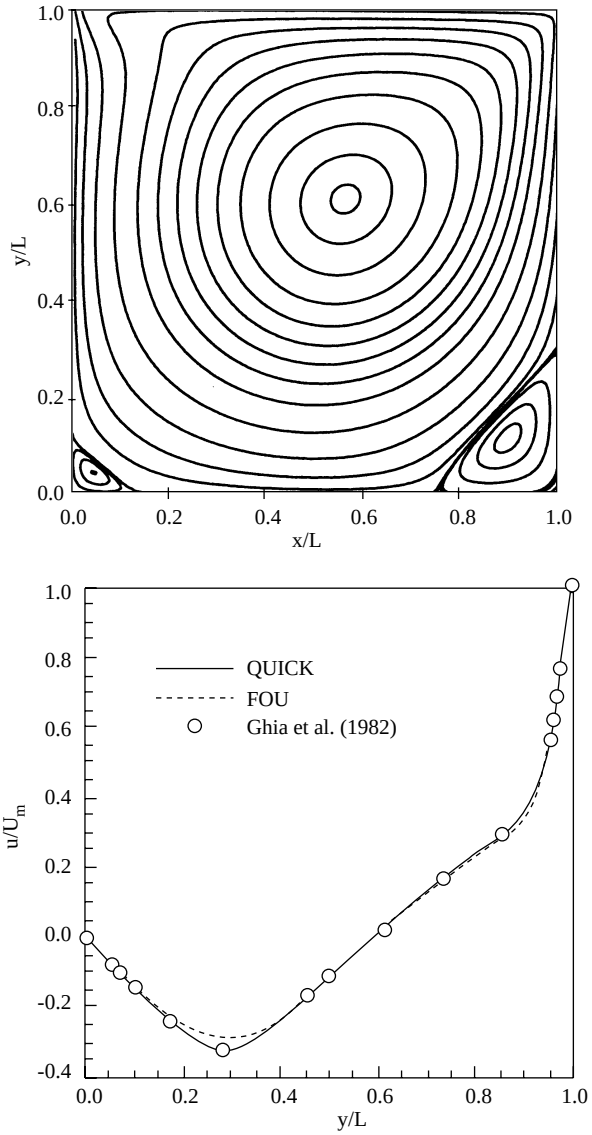

Figure 4. Cavity flow, a) stream lines at $R_e = 400$, b) Velocity profile along vertical at $x = L/2$.

Liquid sloshing in an oscillating rectangular tank

Liquid sloshing inside a partially filled rectangular tank is one of the well known test cases for unsteady free surface flows. The tank (Figure 5) is oscillated along the horizontal axis by a sinusoidal excitation. Horizontal displacement α of the container is given by

$$\alpha = A\sin(2\pi\, t/T) \tag{25}$$

where A is the amplitude of horizontal displacement and T is the period of oscillation. Equation (25) is

differentiated twice to obtain the horizontal acceleration. The parameters of oscillation selected are the same as those in the experiments reported by Okamoto and Kawahara (1990) ($A = 0.93$ cm, $T = 1.183$ s, $b = 1$ m, $H = 0.5$ m) to compare the free surface computations with the experimental data. This test case was defined such that the frequency of oscillation was equal to the natural frequency of the tank to observe possible resonating free surface oscillations.

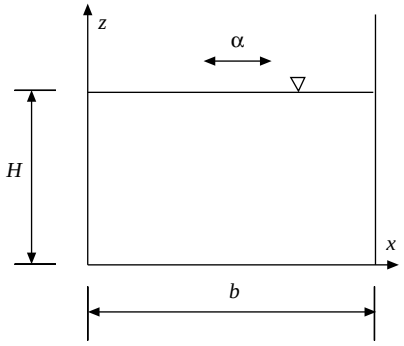

Figure 5. Definition of geometric parameters in an oscillating tank.

Different grid distributions are used to investigate the mesh size effect on the free surface computations. Grid clustering in the vertical direction is applied around the free surface to reduce possible truncation errors due to partially filled surface cells. Three non-uniform grid arrangements, 30×45, 50×63, and 80×92, are used to obtain the free surface profiles at $t = 3.55$ s. Free surface profiles obtained by KBC and DIC compared to experimental data of Okamoto and Kawahara (1990) are shown in Figure 6. KBC gives smoother surface profiles but volume conservation is lost when coarser grids are used. DIC solutions exhibit small oscillations but volume conservation is satisfied exactly at all time steps. DIC solutions give much better fit to experimental data even with coarse grid distributions.

Free surface profiles computed by DIC are compared to measurements given by Okamoto and Kawahara (1990) at different time levels of tank oscillation (Figure 7). The spatial and temporal agreement between the computations and the experiment clearly indicates the accuracy achieved by computational developments and the adequacy of the grid arrangement (80×92) selected in the solution.

Another test for volume conservation was performed by observing the diminishing of surface deformations when the tank oscillation is stopped. The tank is oscillated for the first 14 s and then stopped. Computations are continued with no excitation until the kinetic energy of the oscillating fluid volume is dissipated totally and the free surface becomes nearly horizontal. Water level at the left wall is shown as a function of simulation time in Figure 8. During the first 15 s, the surface waves are amplified, reaching a maximum wave height of approximately 0.4 m, and then start to decrease when the tank oscillations are stopped. It takes about 200 s for the surface waves to be reduced to negligible amplitude. It is observed that the steady-state water depth at the end of computations is the same as the initial water depth, indicating no leaks or sources of fluid volume after 1000 s of simulation.

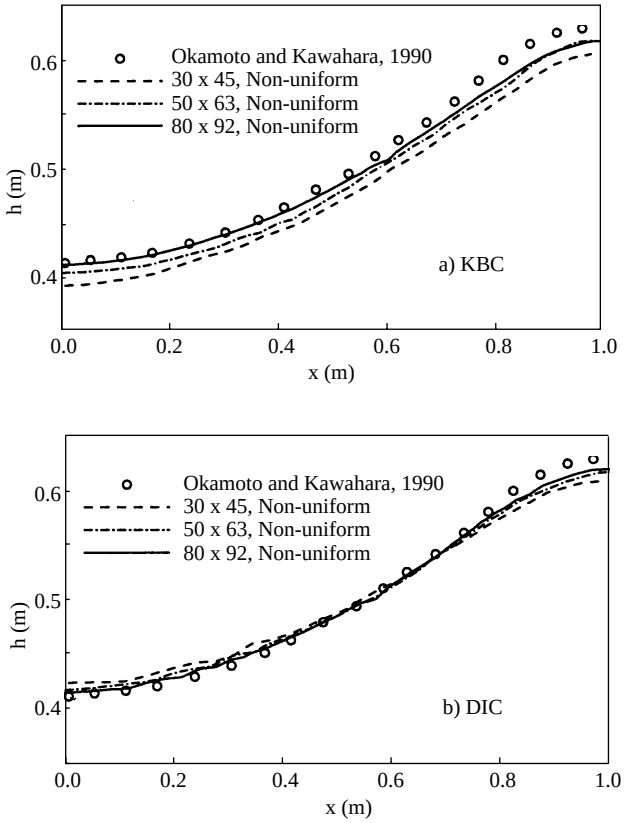

Figure 6. Free surface profiles at $t = 3.55$ s using KBC and DIC.

Dam-break problem

An idealized dam-break problem is described by a rectangular column of water confined between 2 vertical walls (Figure 9). Initially the fluid is at rest. Motion starts when the right wall is removed suddenly at $t = 0$ and water flows to the right over a

dry bed. The depth and width of the water column are chosen as $H = 0.057$ m, the same as in Martin and Moyce (1952), to compare the computational results to the experimental data provided. The numerical solution is obtained using 30 × 20, 50 × 30, and 70 × 40 uniform grid arrangements. Dimensionless water levels $h^+ = h/H$ on the left wall ($x = 0$ m) are plotted as a function of dimensionless time $t^+ = t(g/H)^{1/2}$ and presented in Figure 10 in comparison to the experimental data given by Martin and Moyce (1952). At the beginning of the motion, the right boundary of the water column is vertical and then slope starts to decrease and finally becomes horizontal. Computation of the free surface with very steep surface slopes is the most difficult test case for the present algorithm. Therefore, much error is expected at the early stages of motion.

Computational results for different grid resolutions are very close to each other except for the 30 × 20 grid. There is a discrepancy of less than 3% in depth between computations with the 30 × 20 grid and the other grids. The solution using the 30 × 20 grid underestimates the water level by less than 3% on the left wall while overestimating the water level on the right wall at the same amount such that the total fluid volume is conserved. For the grid resolutions 50 × 30 and higher, all numerical solutions become identical.

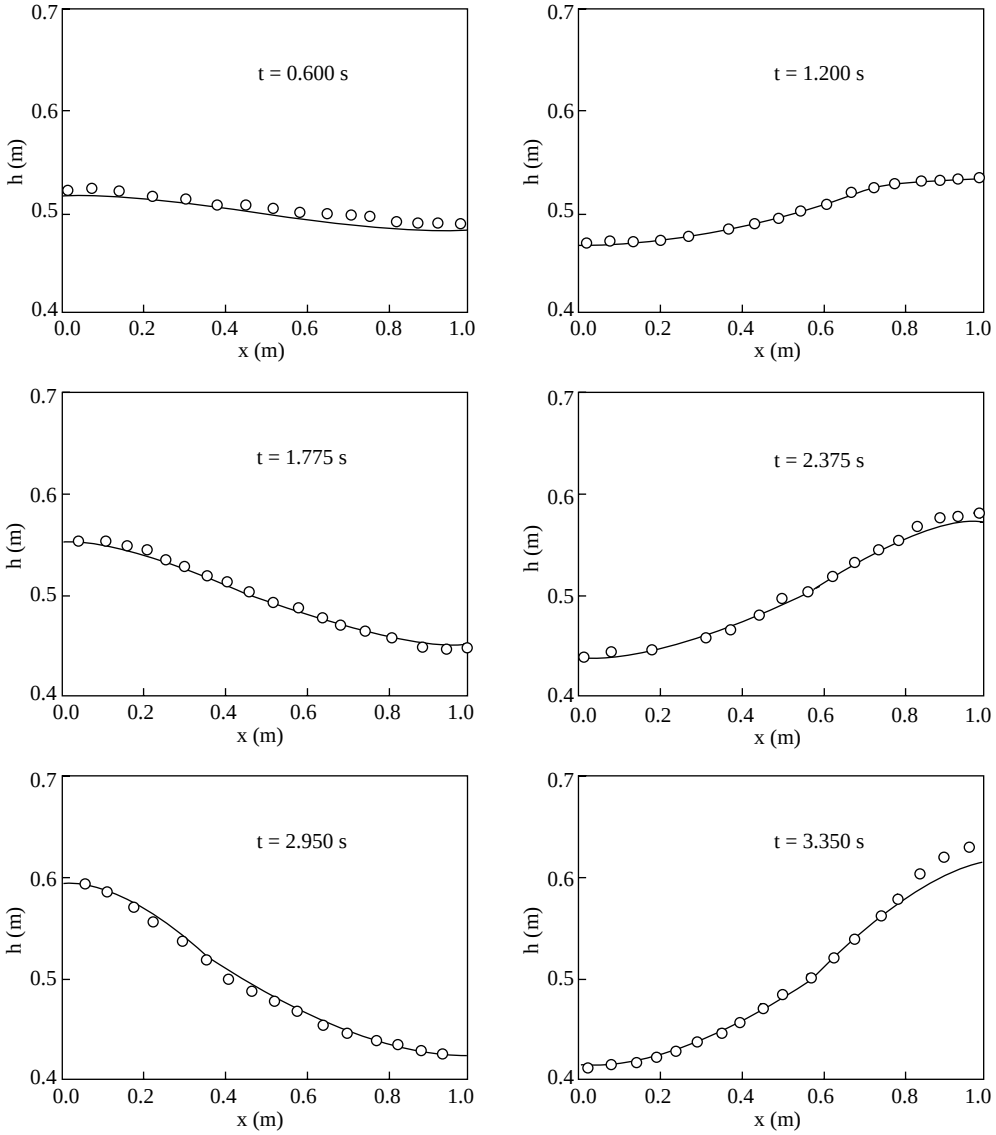

Figure 7. Comparison of free surface profiles in an oscillating tank with experimental data.

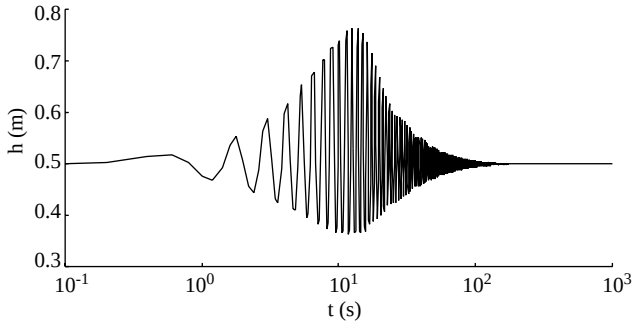

Figure 8. Water level at the left wall of the oscillating tank as a function of time.

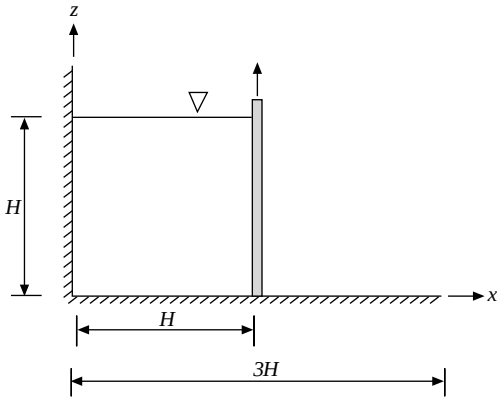

Figure 9. Description of the dam-break problem.

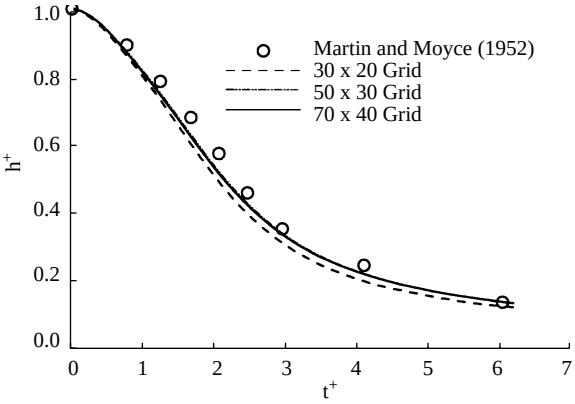

Figure 10. Water level on the left wall in the dam-break problem for different grid resolutions.

The dimensions of the dam-break experimental test case are so small that the accuracy of the numerical solution for large domains, keeping the number of grid points constant, cannot be directly illustrated. However, similar solutions obtained for larger domain dimensions can be shown on the same graph. Dam-break solutions for $H = 1$, 10, 100, and 1000 m obtained using the same number of grid points (70×40) in the computational domain are given together in Figure 11. Computational parameters $\Delta x, \Delta z$, and $\Delta t$are different for each case. Variation in dimensionless water level on the left wall over dimensionless time is conformal for all reservoir dimensions. This indicates that the numerical solution obtained with different computational parameters converges to the same result. There is no difference in the computational errors depending on the dimensions of the computational domain. It should also be noted that conformal plots of dimensionless water levels indicate similar solutions for different reservoir dimensions. This is simply due to the fact that the viscous effects on water level on the left wall are negligible and therefore the numerical solution becomes equivalent to a potential solution producing overlapping water levels for different reservoir heights.

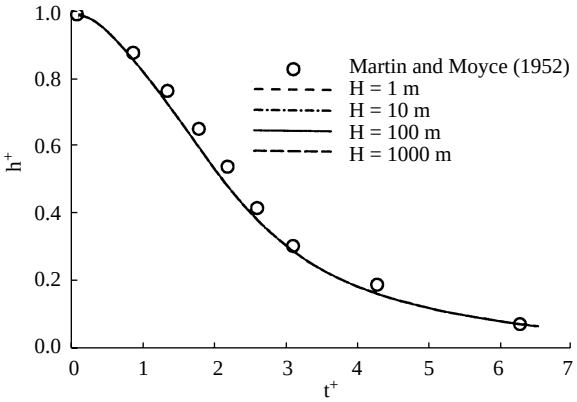

Figure 11. Water level on the left wall in the dam-break problem for different reservoir dimensions.

The KBC method applied to the test cases considered in this study failed to satisfy total volume conservation. Variation in total fluid volume was small and slow. Since unsteady flow simulation durations are usually short, small changes in volume may not be noticed if not checked. Actually, it may not cause significant errors, depending on the physical quantities predicted from the numerical simulation. However, if the pressure field is to be computed and if there is any possibility of compressibility effects on the development of the pressure field, then the smallest unphysical change in fluid volume would prevent prediction of correct pressure variations.

Conclusions

A computational model for simulating 2-dimensional unsteady free surface flows is developed. The model

solves Navier-Stokes equations using a finite volume technique. The free surface tracking is accomplished by DIC and KBC methods alternately, to investigate total volume conservativeness of the formulations. The model has been tested in lid-driven cavity, sloshing in an oscillating tank, and dam-break flows. The following conclusions are drawn from this study.

1. Use of the KBC method to track the free surface results in variations in total fluid volume and therefore fails to satisfy overall mass conservation. One possible reason for failure can be the requirement of artificial diffusion to smooth the surface oscillations. Since the KBC method cannot enforce volume conservation directly, any erroneous water level violating volume conservation is not corrected in the solution procedure.

2. The DIC method produces free surface profiles satisfying mass conservation exactly. The FCT technique removes unphysical surface oscillations in the DIC solution when a sufficient grid resolution is provided. Numerical procedures for KBC and DIC are similar. The additional computation required in DIC is the integration of net flux for each water column in the computational grid.

3. Use of FOU or QUICK for convection terms has no effect on volume conservation in the DIC solution.

4. The algorithm developed here can deal with steep surface slopes, such as dam-break, without computational problems. Accuracy in predicting local water depths may deteriorate due to extrapolations in the free surface computation when a coarse grid is used. However, total volume conservation is always satisfied.

5. Total volume conserving property and computational accuracy are independent of the dimensions of the computational domain. This feature of the model makes it suitable for computations in large spatial domains.

Note: This paper is a part of the PhD study by Ender Demirel in the Civil Engineering Department of Eskişehir Osmangazi University.

Nomenclature

A	amplitude of oscillation
a_x, a_z	acceleration components of ground motion
b	tank width
c	wave speed
d	distance between instantaneous water depth and fluid cell
F, G	tentative velocity fields
g	acceleration of gravity
H	initial water depth in the tank
h	instantaneous water depth
p	fluid pressure
p_{fs}	fluid pressure at surface cell
p_s	interpolated fluid pressure at surface cell
q	flux at cell face
T	period of oscillation
t	time
u, w	velocity components of fluid
u_s, w_s	surface velocities
α	displacement of the tank
$\vec{\nabla}$	del operator
Δt	computational time step
Δx, Δz	mesh size
λ	diffusion coefficient
μ	dynamic viscosity
ν	kinematic viscosity
ρ	fluid density
$\forall$	volume

References

Armenio, V. "An Improved MAC Method (SIMAC) for Unsteady High-Reynolds Free Surface Flows", International Journal for Numerical Methods in Fluids, 24, 185-214, 1997.

Batchelor, G.K., "An Introduction to Fluid Dynamics", Cambridge University Press: Cambridge, 1967.

Book, D.L., Boris, J.P. and Hain, K., "Flux-Corrected Transport II: Generalization of the Method", Journal of Computational Physics, 18, 248-278, 1975.

Boris, J.P. and Book, L.D., "Flux-Corrected Transport I: SHASTA, A Fluid Transport Algorithm That Works", Journal of Computational Physics, 11, 38-69, 1973.

Chan, R.K.-C. and Street, R.L., "A Computer Study of Finite-Amplitude Water Waves", Journal of Computational Physics, 6, 68-94, 1970.

Chen, S., Johnson, D.B. and Raad, P.E., "Velocity Boundary Conditions for the Simulation of Free Surface Fluid Flow", Journal of Computational Physics, 116, 262-276, 1995.

Ghia, U., Ghia, K.N. and Shin, C.T., "High-Re Solutions for Incompressible Flow Using the Navier-Stokes Equations and a Multigrid Method", Journal of Computational Physics, 48, 387-411, 1982.

Griebel, M., Dornseifer, T. and Neunhoeffer, T., "Numerical Simulation in Fluid Dynamics: A Practical Introduction", SIAM Publications: Philadelphia, 1997.

Harlow, F.H. and Welch, J.E., "Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid with Free Surface", The Physics of Fluids, 8, 2182-2189, 1965.

Hirt, C.W. and Nichols, B.D., "Volume of Fluid (VOF) Method for the Dynamics of Free Boundaries", Journal of Computational Physics 39, 201-225, 1981.

Kleefsman, T., "Water Impact Loading in Offshore Structures", PhD Thesis, University of Groningen, The Netherlands, 2005.

Kleefsman, K.M.T., Fekken, G., Veldman, A.E.P., Iwanowski, B. and Buchner, B., "A Volume-of-Fluid Based Simulation Method for Wave Impact Problems", Journal of Computational Physics, 206, 363-393, 2005.

Li, Y. and Baldacchino, L., "Implementation of Some Higher-Order Convection Schemes on Non-Uniform Grids", International Journal for Numerical Methods in Fluids, 21, 1201-1220, 1995.

Martin, J.C. and Moyce, W.J., "Part IV: An Experimental Study of the Collapse of Liquid Columns on a Rigid Horizontal Plane", Philosophical Transactions of the Royal Society of London, Series A. Mathematical, Physical and Engineering Sciences, 244, 312-324, 1952.

Nichols, B.D. and Hirt, C.W., "Calculating Three-Dimensional Free Surface Flow in the Vicinity of Submerged and Exposed Structures", Journal of Computational Physics, 12, 234-246, 1973.

Okamoto, T. and Kawahara, M., "Two-Dimensional Sloshing Analysis by Lagrangian Finite Element Method", International Journal for Numerical Methods in Fluids, 11, 453-477, 1990.

Tome, M.F., Filho, A.C., Cuminato, J.A., Mangiavacchi, N. and McKee, S., "GENSMAC3D: A Numerical Method for Solving Unsteady Three-Dimensional Free Surface Flows", International Journal for Numerical Methods in Fluids, 37, 747-796, 2001.

Wang, H.W., Huang, C.J. and Wu, J., "Simulation of a 3D Numerical Viscous Wave Tank", Journal of Engineering Mechanics, 133, 761-772, 2007.

Yost, S.A. and Rao, P.M.S.V., "Flux-Corrected Transport Technique for Open Channel Flow", International Journal for Numerical Methods in Fluids, 29, 951-973, 1999.