Simulation of free surface flows using volume of fluid method and genetic algorithm

N. Soleiman Beygi, H. Hakimzadeh and M. R. Chenaglou

ABSTRACT

In this paper, details of a numerical model development for simulation of fluid flows with moving free surface are presented. The unsteady incompressible Navier–Stokes equations on a fixed grid system are used to obtain velocity and pressure values in the computational domain and volume of fluid (VOF) method is used to determine free surface location. In order to reduce numerical smearing and increase accuracy of numerical modeling of fluid flow with moving free surface, a new free surface-tracking method is proposed. The proposed method is a combination of genetic algorithm and free surface tracking method based on donor and acceptor scheme. The specification of the new combinational method can be summarized in determining orientation vector and plane constant to represent the free surface orientation in each cell. The proposed algorithm can be easily used in any unstructured grids. In this method, the fluid flow equations are explicitly discretized with the finite volume method and the projection method is used to determine the velocity and pressure magnitude in computational domain. Validity of the new solution algorithm is demonstrated through its application to the dam break and the bore motion examples.

Key words | finite volume method, free surface flow, genetic algorithm, Navier–Stokes equations, orientation vector, projection method, volume of fluid method

N. Soleiman Beygi
H. Hakimzadeh (corresponding author)
M. R. Chenaglou
Faculty of Civil Engineering,
Sahand University of Technology,
P. O. Box 51335-1996,
Sahand New Town,
Tabriz,
Iran
E-mail: *hakimzadeh@sut.ac.ir*

NOMENCLATURE

F	volume fluxes
$\vec{F}_{st}$	surface tension force vector
f	marker function
f_{S_j}	fractional volume in the jth face
g	acceleration of gravity
$\bar{\bar{I}}$	unit tensor
H	height of the channel
h	wave height
h_0	initial water depth in channel
L	length of the channel
$\hat{n}$	unit normal vector
$\vec{n}$	unit normal vector to free surface
u	fluid velocity
P	pressure
P_0	atmospheric pressure
S	boundaries of volume cell
t	time
$\vec{V}$	velocity vector
v_0	initial water velocity
v_s	wave speed
x	point on plane

Greek letters

Δt	time increment
$\Delta x, \Delta y$	grid spacing in x and y direction
$\bar{\mu}$	dynamic viscosity
$\bar{\bar{\tau}}$	viscous stress tensor
α	plane constant
$\bar{\rho}$	density
$\forall$	volume of cell
σ	coefficient of surface tension

doi: 10.2166/hydro.2014.152

κ mean curvature
δ_s delta function
$\bar{\nu}$ kinematics viscosity

Subscripts

I cell number
j face number
i, j, k ith, jth and kth computational cell in x, $y$and z directions
n, t normal and tangential unit vector of the free surface
x, y derivative with respect to x and y direction

Superscripts

$*$ intermediate value
$\sim$ convolved
n nth time step
x, y component in x and y direction

INTRODUCTION

Special attention has been paid to the study of incompressible viscous flow with moving free surface due to its important applications in water engineering, environmental engineering and many other areas of interest, but to date such flows have no analytical solution except for the one-dimensional special cases. Experimental models and prior experience had been the main tools for studying such flows until the mid-1960s, when numerical solutions became available. Nowadays, with the rapid increase in computer power and development of numerical models, study on such flows with numerical methods has widely been considered. In numerical methods, the location of free surface flow should be determined at the present time step and should be corrected for the next time step for a given flow field while solving fluid flow equations. In order to trace the moving free surface, different methods have been proposed by researchers. In general, all proposed methods can be classified into two main categories, that is, moving grid method and fix grid method (Gerlach *et al.* 2006). In the moving grid, also known as the Lagrangian method, grid points are always located in the fluid and move with it. Each computational cell contains the same fluid elements. In this method, free surface boundary conditions are directly applied on exact material interfaces and at every step of the calculation, free surface position is then specified explicitly (Kim & Lee 2003). It should be noted that the Lagrangian method can be used for certain problems of fluid flow with moving free surface. If large deformations at free surface fluid flow occur or cells have a large deformation, numerical inaccuracy may occur in the solution of the flow field. Further, the number of grid points in computational domain changes and thus it requires remeshing or rezoning process of the entire computational domain, which complicates this method (Meier *et al.* 2002). To solve the problems of moving grid method, researchers have suggested using other methods such as Lagrangian–Eulerian or Eulerian methods (Hirt *et al.* 1974). In the Eulerian method or fixed grid method, mesh is treated as a fixed reference frame through which the fluid moves. The initially generated mesh is used throughout the entire computation domain and remains unchanged until the end of calculations, thus no geometric difficulty arises. The Eulerian method can overcome disadvantages of the Lagrangian method and reduce computational efforts compared to the moving grid method (Ubbink & Issa 1999; Tryggvason *et al.* 2001). However, the fixed grid method has some shortcomings in determining free-surface fluid flow, for instance, the position of free surface fluid flow is not specified explicitly and various methods should be used to determine position of free surface flow (Udaykumar *et al.* 1997). More importantly, the grid points are fixed at their initial positions and the free surface is located between some of them. Therefore, the free surface is liable to lose intrinsic nature of sharp discontinuity and thus schemes are required to avoid the numerical smearing (Gueyffier *et al.* 1999). The volume tracking method is the known numerical technique that has a potential of dealing with these problems (Kim & Lee 2003). In the fixed grid method, the free surface location can be determined with explicit and implicit methods like the marker-and-cell (MAC) method, where marker particles are used to identify each fluid, and the volume of fluid (VOF) method, where a marker function is used, are the best known examples

(Unverdi & Tryggvason 1992). The advantage of the VOF method is its simplicity compared to the other proposed methods. Some methods are so complex and difficult that they cannot easily be used, even in two-dimensional problems. Like the MAC method, the VOF method does not require redistribution in any computational cycle. More importantly, no special conditions are required in reformation of free surface fluid inside cells (Gueyffier *et al.* 1999). In the volume of fluid method, a marker function f is defined which represents the volume of reference fluid within each cell. The value of f within the cells fully accumulated by reference fluid is equal to one, within the cells empty of reference fluid is equal to zero and in surface cells varies between zero and one. The value of marker function inside each cell can be easily calculated at the beginning of the computational cycle by solving the transport equation, but no approximation of the interface location is known. In other words, special treatments for interface reconstruction are required. In some research studies, the interface is reconstructed by the straightforward SLIC method (Simple Line Interface Calculation) whereas for the others, various PLIC methods (Piecewise Linear Interface Calculation) are used (Noh & Woodward 1976; Young 1982). The second method is more accurate than the first one and it is widely used in many studies of fluid flow (Gueyffier *et al.* 1999). It should be noted that if the VOF/PLIC method is used, then some points of great importance for determining the location of such surfaces arise, including how to calculate the orientation vector and the plane constant to represent the free surface fluid flow in each cell, since they can affect results and may even lead to numerical instability. Accordingly, in this article, we introduce and test a novel algorithm for calculating orientation vector and plane constant for modeling of fluid flow with moving free surface. This new algorithm is based on a combination of genetic algorithm (GA) and free surface tracking, which reconstructs the interface in each cell without any additional efforts. Unlike previous algorithms, we never need to consider a smoothed marker function. Instead, we use a GA for calculating orientation vector and plane constant for positioning the free surface flow. The model is then verified for two test cases of moving free surface fluid flow, selected from the literature: (1) dam break (Cruchaga *et al.* 2007); (2) bore motion in a channel (Bonet & Look 1999). The numerical model results of the tests demonstrate the great ability and accuracy of the coupling algorithm to simulate free surface fluid flow.

GOVERNING EQUATIONS

For the numerical model study, two fluids (liquid and gas) are considered as a single continuum obeying the same set of governing equations, with the different fluids being identified locally by a volume fraction field. The present paper deals with homogeneous, incompressible and viscous fluids for which the governing equations are discretized based on the finite volume method. In general, the governing equations are written in integral forms as follows (Neofyton 2005):

$$\oint_S \vec{V}.\overrightarrow{dS} = 0 \tag{1}$$

$$\frac{\partial}{\partial t}\int_{\forall}\bar{\rho}\vec{V}d\forall + \oint_S \bar{\rho}\vec{V}\left(\vec{V}.\overrightarrow{dS}\right) = \int_{\forall}\bar{\rho}\vec{g}d\forall - \oint_S P\bar{\bar{I}}.\overrightarrow{dS} + \oint_S \bar{\bar{\tau}}.\overrightarrow{dS} + \int_{\forall}\vec{F}_{st}d\forall \tag{2}$$

where $\vec{V}$ is the velocity vector, S the boundaries of volume $\forall$, t the time, $\bar{\rho}$ the fluid density, P the pressure, $\bar{\bar{I}}$ unit tensor, $\bar{\bar{\tau}}$ the stress tensor and g is the acceleration of gravity. In Equation (2), Reynolds stresses are calculated from $\bar{\bar{\tau}} = \bar{\rho}.\bar{\nu}(\nabla\overrightarrow{V} + (\nabla\overrightarrow{V})^T)$, in which $\bar{\nu}$ is the eddy viscosity of fluid flow and is determined of two layers mixing length model. $\vec{F}_{st}$ is the surface tension force which is written as volume force in the momentum equation and is only considered in surface cells. Then, to calculate surface tension force, the following equation is proposed (Brackbill *et al.* 1992):

$$\vec{F}_{st} = \sigma \cdot \kappa \cdot \delta_s \cdot \overrightarrow{n} \tag{3}$$

where σ is the coefficient of surface tension, κ is the mean curvature, δ_s is a delta function and $\overrightarrow{n}$ is the unit vector perpendicular to free surface fluid flow in surface cells. To

calculate the mean curvature of free surface fluid, equation $\kappa = -\nabla.\vec{n}$ may be used (Brackbill *et al.* 1992). Another equation that should be resolved for modeling fluid flow with moving free surface is the advection equation for scalar quantity. Then the advection equation can be written as follows:

$$\frac{\partial f}{\partial t} + \vec{V}.\nabla f = 0 \tag{4}$$

in which f is the marker function that equals one within the cells completely filled with the desired fluid; zero within the cells empty of the desired fluid and between zero and one within the other cells (i.e. surface cells). In discretization process, the derivatives of the marker function are replaced with finite difference approximations. The problem is that these approximations for derivatives of a discontinuous function are highly inaccurate. This leads to serious errors in the computation of both the normal and the curvature. Therefore, the discretization of Equation (4) requires special treatment. It should be noted that $\bar{\rho}$ and $\bar{\nu}$ can be assumed constant for each fluid, but in surface cells where two fluids with different specifications are located within the cell, appropriate values must be determined for these quantities from the following equations:

$$\bar{\rho} = \rho_1 f + \rho_2(1-f) \tag{5}$$

$$\bar{\nu} = \nu_1 f + \nu_2(1-f) \tag{6}$$

Here, ρ_1, ρ_2 and ν_1, ν_2 represent the densities and eddy viscosities of the two fluid flows (liquid and gas), respectively.

Boundary conditions

The boundary conditions can be classified as open, closed, surface, wall and bed. These boundary conditions must be precisely defined for numerical modeling. For an ordinary open boundary, all variables, including velocity and pressure, are usually unknown and experience is needed to specify physically realistic values for engineering application (Jian *et al.* 1998). Also, on the outlet flow boundary, the values of the flow parameters are unknown. One of the best approximations prescribes no change for the values of the flow parameters normal to the boundary; in other words, equation $\partial\varphi/\partial n = 0$ can be used for all variables in the outlet flow boundary (Ghiassi 1995). Such a definition for outlet boundary does not seem very accurate, but with this definition, numerical instability will not be created in the model, and the effect of outlet boundary on the results of numerical modeling can be reduced by placing the fluid outlet boundary away from the studied area (Hakimzadeh 1997). In wall and bed, two different boundary conditions can be applied in the numerical models: no slip and free slip. In the first case, the velocity is set to zero in all nodes defining the rigid wall. The free slip condition imposes that the velocity in the normal direction be zero and the derivative of the tangential velocity with respect to the normal direction is also zero (Hakimzadeh 1997). In free surface flow, which is the contact location of different fluids, continuity of tangential and normal stresses must be applied for velocity and pressure variables (Kleefsman *et al.* 2005). This condition represents that in the absence of external stresses exerted on free surface fluid flow, the normal and tangential stress values should be zero, that is,

$$-P + 2\bar{\mu}\frac{\partial u_n}{\partial n} = -P_0 + \sigma\kappa \tag{7}$$

$$\mu\left(\frac{\partial u_n}{\partial t} + \frac{\partial u_t}{\partial n}\right) = 0 \tag{8}$$

Here, u is the fluid velocity in the x, y and z axes and n, t are the normal and tangential unit vector of the free surface, respectively. P_0 is the atmospheric pressure.

Numerical model

In the present study, the finite volume method is used to discretize free surface fluid flow equations. Finite volume is widely used for discretizing differential equations since it can easily be used in complex geometric conditions and is simple compared to the other methods (Jessee & Fiveland 1996). To solve the Navier–Stokes equations numerically, the computational domain is covered with a fixed Cartesian

grid. For the present numerical model, in order to discretize the physical domain, the staggered structured mesh is used. In this grid, scalar quantities are defined in cell centers and vector quantities are defined on cell faces (Harlow & Welch 1965). Figure 1 shows a typical sample of staggered structured mesh.

After the mesh distribution in the physical domain, the fluid flow equations are discretized on all grid cells. In general, discretized forms of various terms in Equation (2) can be summarized as follows:

$$\frac{\partial}{\partial t}\int_{\forall_I}\bar{\rho}\vec{V}d\forall = \bar{\rho}_I.\forall_I.\frac{\vec{V}_I^{n+1}-\vec{V}_I^{n}}{\Delta t} \tag{9}$$

where $\vec{V}_I^{n+1}$ and $\vec{V}_I^{n}$ are velocity vectors in cell I at time levels $n+1$ and n, respectively. It should be noted that in the above definition, cell I would change position and size depending on the momentum equation. The advective terms of the momentum equations may be discretized as follows:

$$\oint_S \bar{\rho}\vec{V}\left(\vec{V}.\overrightarrow{dS}\right) = \sum \bar{\rho}_j\vec{V}_j\left(\vec{V}_j.\hat{n}\right)S_j \tag{10}$$

where $\hat{n}$ is the outward unit normal on the jth face section of cell I. For the current study, the volume forces and stress terms at the right hand side of the equation are discretized as follows:

$$\oint_S \bar{\bar{\tau}}.\overrightarrow{dS} = \sum\left(\mu.S.\nabla.\vec{V}.\hat{n}\right)_j \tag{11}$$

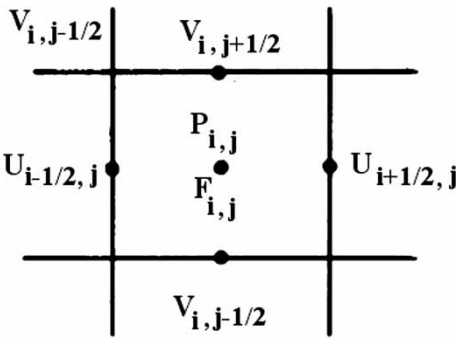

Figure 1 | Field variable value placement around a computational cell.

$$\int_{\forall_I}\bar{\rho}.\vec{g}d\forall = (\bar{\rho}.\vec{g}.\forall)_I \tag{12}$$

Then, by discretizing the momentum and continuity equations in all cells, it is necessary to use iterative methods or fractional step methods to extract fluid flow variables. For this study, fractional step or projection method is used to determine the variables. Thus, at the first step of calculation, the discretized Navier–Stokes equations are discretized and solved regardless of the pressure term in the total computational domain (Hirsch 1990)

$$\frac{\bar{\rho}^{n+1}\vec{V}^{*}-\bar{\rho}^{n}\vec{V}^{n}}{\Delta t}\forall = \left(\int_{\forall}\bar{\rho}\vec{g}d\forall + \oint_S \bar{\bar{\tau}}.\overrightarrow{dS} - \oint_S \bar{\rho}\vec{V}\left(\vec{V}.\overrightarrow{dS}\right)\right)^{n} \tag{13}$$

The velocity values, $\vec{V}^{*}$ extracted from Equation (13), do not satisfy the continuity equation and the above values should be corrected. The correction of the above values is possible by adding the pressure term to the above equation, that is,

$$\frac{\bar{\rho}^{n+1}\vec{V}^{n+1}-\bar{\rho}^{n+1}\vec{V}^{*}}{\Delta t}\forall = \left(-\oint_S P\bar{\bar{I}}.\overrightarrow{dS}\right)^{n+1} \tag{14}$$

$$\sum_S \vec{V}^{n+1}.\vec{S} = 0 \tag{15}$$

The final value of velocity field is obtained from Equation (14). The pressure is calculated in such a way that the velocity field at level time $n+1$ satisfies the divergence free condition (Equation (15)). By taking the divergence of the above equation, pressure correction equation or Poisson equation can be extracted

$$\nabla P.\overrightarrow{S} = \frac{\bar{\rho}^{n+1}}{\Delta t}\nabla.\vec{V}^{*} \tag{16}$$

After calculating pressure from the above equation, velocities at the new time step can be calculated from Equation (14).

FREE SURFACE TRACKING SCHEME

By solving the continuity and momentum equations on staggered structured mesh, velocities and pressure will be extracted at different points of computational domain, in which velocities are placed on cell faces and pressure is displayed in the grid cell centers. With such information, it is possible to solve the advection equation for marker function to determine the amount of the desired fluid volume within all cells. However, the scalar quantity of the desired fluid volume is defined for the cell and cannot be used as a concentrated quantity in the center or face of cells. This means that, unlike the momentum or continuity equations, another scheme is required for solving the advection equation for marker function (Kim & Lee 2003). In order to overcome this problem, modern VOF/PLIC methods employ geometrical techniques to approximate the interface location within all cells. Therefore, in this study, a geometrical technique is used for solving the advection equation. Details of the applied method in this article are explained below.

Advection of volume fraction

In order to transport marker function f with minimal smearing in the VOF method, the advection equation should be discretized in such a way that the interface sharpness is preserved as the interface is advected. Using a simple algebraic method to solve the advection equation will lead to diffusion of the interface (Rider & Kothe 1998). Therefore, in this study a number of geometrical techniques have been employed for approximating the interface and solving the advection equation. Integration of the advection equation in a conservative form over each cell will lead to the following equation:

$$\int_{\forall_I} \left(\frac{\partial f}{\partial t} + \vec{V}.\nabla f \right) d\forall = 0 \tag{17}$$

Then, applying the divergence theorem and integrating over time, Equation (17) is reduced to:

$$f_I^{n+1} = f_I^n + \frac{\Delta t}{\forall_I} \left(-\sum_j \left(f_j \vec{V}.\hat{n} \right) S_j \right) \tag{18}$$

In an alternative approach, the volume advection equation is integrated in time using a split method, that is, Equation (17) must be integrated twice in two dimensions (three times in three dimensions) (Rider & Kothe 1998). In two dimensions, volume fractions f^n are advected in the first sweep to f' according to:

$$f'_I = \frac{f_I^n.\forall_{i,j} - F_{i+1/2,j} + F_{i-1/2,j}}{\forall'_I} \tag{19}$$

$$\forall'_I = \forall_I - \left(u^n_{i+1/2,j}.S_{i+1/2,j} - u^n_{i-1/2,j}.S_{i-1/2,j} \right).\Delta t \tag{20}$$

and in the second sweep, f' fields are advanced to their final values f^{n+1}:

$$f_I^{n+1} = \frac{f'_I.\forall' - F_{i,j+1/2} + F_{i,j-1/2}}{\forall_I} \tag{21}$$

In the above equations, F refers to the volume fluxes passing through the surfaces of the cell I during the time step. To compute the volume flux F of the desired fluid that passes across a cell face during a time step, the total volume flux across the face is first determined. Then, the portion of the volume of the donor cell from which the volume is leaving is specified (Rudman 1998). Figure 2 shows how to calculate input and output fluxes across the face.

Calculating normal and constant of the plane

As mentioned above, in the VOF/PLIC method the interface is approximated by planes in interfacial cells that are defined by the following equation:

$$\vec{n}.x = \alpha \tag{22}$$

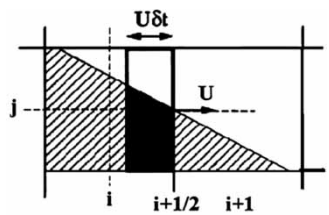

Figure 2 | Calculating volume fluxes across the face.

where x is a point on the plane and α is the plane constant. To define a plane, the normal and constant must be determined. To calculate the normal unit vector from the values of the fluid volume within the cells, the following equation may be used (Young 1984; Kothe *et al*. 1996; Bussman 2000):

$$\vec{n} = \frac{\nabla f}{|\nabla f|} \tag{23}$$

It should be noted that using f directly to calculate normal vector leads to poor results in terms of the magnitude of error and the order of convergence. For these reasons, a new method was selected here to determine the normal vector. After calculating the normal vector, the plane constant must be calculated. To determine the plane constant in a cell, the volume of the polyhedron truncated by the free surface should be equal to the volume fraction calculated from the advection equation for scalar quantity. In other words, the following equation must be satisfied:

$$Vol(\alpha) = f_I \times \forall_I \tag{24}$$

The above equation can be solved with different techniques. In many cases, to solve this equation and determine the plane constant coefficient, iterative methods are used (Rider & Kothe 1998; Scardovelli & Zaleski 2000). The main advantage of an iterative method is generality, since it can be used for any polyhedral shaped cell. Also, the iterative algorithms are simple, robust, efficient and easily understood.

NEW FREE SURFACE TRACKING SCHEME

The original VOF method has two main drawbacks. The first is that poor results can appear in free surface when f directly is used for calculating the normal vector. Therefore, to ameliorate this problems, f is convolved using a smoothing kernel, resulting in a volume fraction field $\tilde{f}$, that does not change abruptly across the interface. If such relations are used, the complexity of fluid flow problems will increase and importantly, such a relationship cannot be easily extracted for a variety of cells. The other drawback is the local gain or loss of water due to rounding the VOF function when $f > 1$ or $f < 1$. Therefore, to solve the advection equation and to determine the location of free surface inside the cells with minimal numerical smearing, a new algorithm is proposed. This algorithm is a combination of previous equations with a GA that calculates the normal unit vector components and the plane constant without any application of other iterative or analytical methods. A brief description of this algorithm is outlined here.

Description of GA

The genetic algorithm is based on principles inspired from the genetic and evolution processes observed in natural systems (Goldberg 1989). This algorithm has recently found extensive applications in solving global optimization searching problems (Poloni *et al*. 2000). In comparison to the conventional searching algorithms, GA has the following characteristics: (a) GA works directly with the discrete points coded by finite-length strings (chromosomes), not the real parameters themselves; (b) GA considers a group of points (called a population size) in the search space in every iteration, not a single point; (c) GA uses fitness function information instead of derivatives or other auxiliary knowledge; and (d) GA uses probabilistic transition rules instead of deterministic rules. Generally, a simple GA consists of three basic genetic operators (Krishnkumar & Goldberg 1992; Adeli & Cheng 1993): (a) selection, (b) crossover and (c) mutation.

Selection

Selection is simply a process to decide which strings should survive and how many copies of them should be produced in the mating pool. The decision is made by comparing the fitness of each string with the average fitness of the population. The fitness is an indicator of the survival potential and reproductive capability of the string in the subsequent generation. For an optimization problem, the fitness is the objective function or a combination of objective function and constraints. This step ensures that the overall quality of the population increases from one generation to the next.

Crossover

Crossover is a recombined operator for two high-fitness strings (parents) to produce two offspring by mixing and matching their desirable qualities through a random process. For the current study, the uniform crossover method is adopted. The procedure is to select a pair of strings from the mating pool at random, then, a mark is selected by random. Finally, two new strings are generated by swapping all characters correspond to the position of the mark where the bit is '1'. Although the crossover is done by random selection, it is not the same as a random search through the search space. Since it is based on the selection process, it is an effective means of exchanging information and combining portions of high-fitness solutions.

Mutation

Mutation is often introduced to guard against premature convergence. Generally, over a period of several generations, the gene pool tends to become more and more homogeneous. The purpose of mutation is to introduce occasional perturbations to the variables to maintain the diversity in the population. Mutation occurs with a small probability in the GA to reflect the small rate of mutation existing in the real world. It is realized by performing bit inversion on some randomly selected bit positions of offspring bit strings (Goldberg 1989).

However, as seen above, the GA includes five fundamental parameters. (a) Population size, which influences the amount of search points in every generation. The larger population size in the GA will increase the efficiency of searching, but it will be time consuming. (b) Crossover probability, which influences the efficiency of exchanging information. (c) Mutation probability, which occurs with a small probability in the GA. A large mutation probability in GA will eliminate the result of selection and crossover, which let GAs become a random search. (d) Chromosome length, which influences the resolution of the searching results. The GA with longer chromosome length will have the higher resolution, but it will increase the search space. (e) Generations, which influences the searching time and searching result. The GA with larger search space and less population size needs more generations for a global optimum.

New free surface tracking scheme

For the current study, a new method is proposed for determining the location of free surface within cells through the combination of the GA with the original VOF. In the proposed method, the initial values for the parameters must be estimated. To extract initial values for normal vector components, Equation (23) is used. With the initial values calculated, an approximate location for free surface fluid flow will be extracted. The initial estimated values will be used to find the exact solution of the equations by GA. The estimated values are only considered to increase the speed of solution and to reduce iterative steps which are modified by GA at the next steps. Samples of the free surface positions in 2D and 3D problems are shown in Figure 3.

As shown in Figure 3, the free surface divides each cell into two sections in which each section contains a certain amount of fluid with different physical characteristics, which in solving fluid flow equations is necessary to determine fluid volumes and their positions relative to each other with sufficient accuracy. In addition, for a 2D cell, four different cases of free surface of location can be imagined by rotating the cell around their axes. Then, all cases of free surface for 2D problems are shown in Figure 3. It can be inferred that by crossing the fluid separator line from a cell node, a new position of the free surface is described. A new algorithm based on the above template

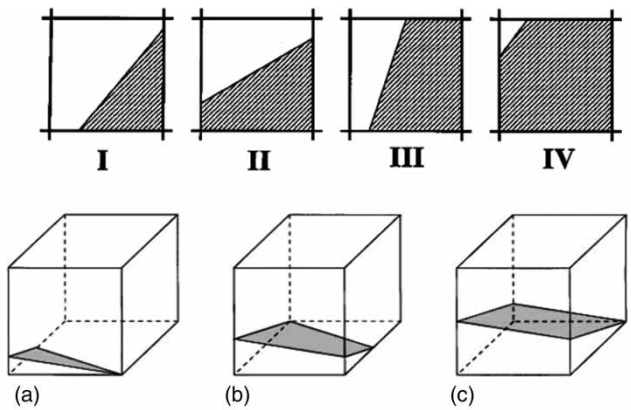

Figure 3 | Samples of the free surface position.

attempts to determine the position of the fluid interface. The proposed algorithm does not need to determine the different positions of the free surface within the cell since it will directly determine the position of free surface. In using the new method, it is only necessary that a search space range be defined for the desired parameters at the beginning of calculations, so that the GA attempts to generate random numbers and determine the exact solution within the considered range. Therefore, in this method two major subjects should be discussed to identify the exact surface location. One is how to define the search space for the unit normal vector components within cells and the other is how to define the search range for the plane constant to define the plane equation. The method for defining the search space for the normal vector is shown in Figure 4. In this figure only case 3-I is discussed. All of the other cases can be determined similar to this case. In the first stage, for the given cell, the initial position of the free surface fluid is extracted from the relationship mentioned in the previous sections. For example, it is assumed that the free surface in the mentioned cell is located in the position of Figure 4(a).

To define the search space for the unit normal vector, the initial estimated values from previous equations should be changed so that the fluid volume inside the cell is equal to the value calculated by the marker function. Most importantly, the changed surface should always be located within the estimated position. As mentioned earlier, crossing of the separator line in 2D or plane in 3D from the cell nodes will define a new position for the free surface. Therefore, the beginning and end of the mentioned position will be calculated from the maximum changes of the initial estimated free surface position. For example, Figures 4(b) and 4(c) show the minimum and maximum search space range for the free surface estimated in Figure 4(a). Then, in Figure 4, the fluid volumes in all cells are equal, and only the values of the unit vector components differ. Thus, the search space range for the numerical implementation is selected from the minimum and maximum calculated values. Also, it should be noted that within the defined range for free surface fluid flow, it is possible for GA to specify different positions in which fluid volumes equal to the value extracted from the marker function, but not the actual free surface. Therefore, to restrict the GA's search and to select the optimal position of free surface location, using information on adjacent cells is the only solution that can direct the GA toward the optimal solution. Using information on adjacent cells will lead to continuity of elements' free surface, besides limiting the algorithm's search space. More importantly, it will lead to increase numerical stability on free surface fluid flow. Thus, the fractional VOF is also defined that must be applied for surfaces of the given cell. Figure 5 shows how to calculate the fractional VOF for cell surface.

In this research, in order to define the fractional VOF for cell surface, the following equation is proposed:

$$f_{S_j} = \frac{S_j^{\text{wet}}}{S_j} \tag{25}$$

where f_{S_j} is the fractional VOF in the jth face of the cell, S_j^{wet} is the area of the jth face which is covered by the desired fluid and S_j is the total area of the jth face. These conditions will be applied for confining the search space with GA. Another parameter that must be determined for calculating the interface location free surface fluid flow is plane or line constant. Then, like the other unknown parameters, GA is used to determine the plane or line constant. Thus by defining the search space, an optimal solution will be extracted for the desired parameter.

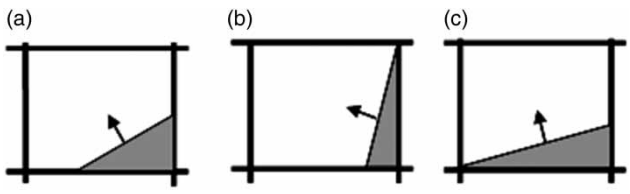

Figure 4 | Description of the search space for normal vector.

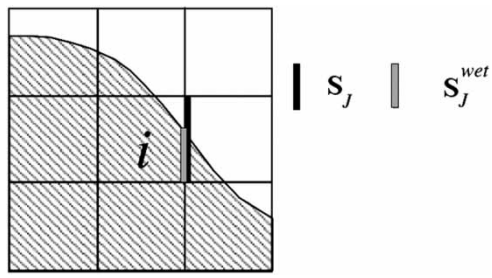

Figure 5 | Definition of fractional VOF for cell surface.

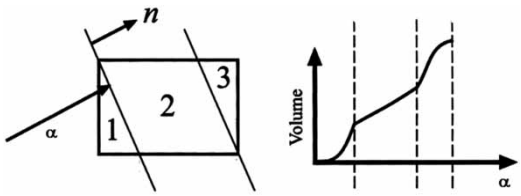

Figure 6 | Relationship between the plane constant with truncation volume.

The relationship between plane constant and fractional VOF within cell is shown in Figure 6. This figure represents the nonlinear relationship between fractional VOF and plane constant. In these cases, the best option to determine the unknown parameter is iterative methods. This is why the GA is used to determine parameters in the current study. To determine the search range of GA, the estimated initial surface should sweep the whole space of cell so that the minimum and maximum values of plane constant are extracted from Equation (22). The algorithm will then attempt to determine the optimum solution in this range.

Solution procedure

Using the free surface-tracking scheme and the solution method of Navier–Stokes equations described in the previous sections, the solution procedure can be summarized as follows (see Figure 7).

1. Define the problem. For a given geometry:
   - specify material properties and parameters;
   - generate a mesh system;

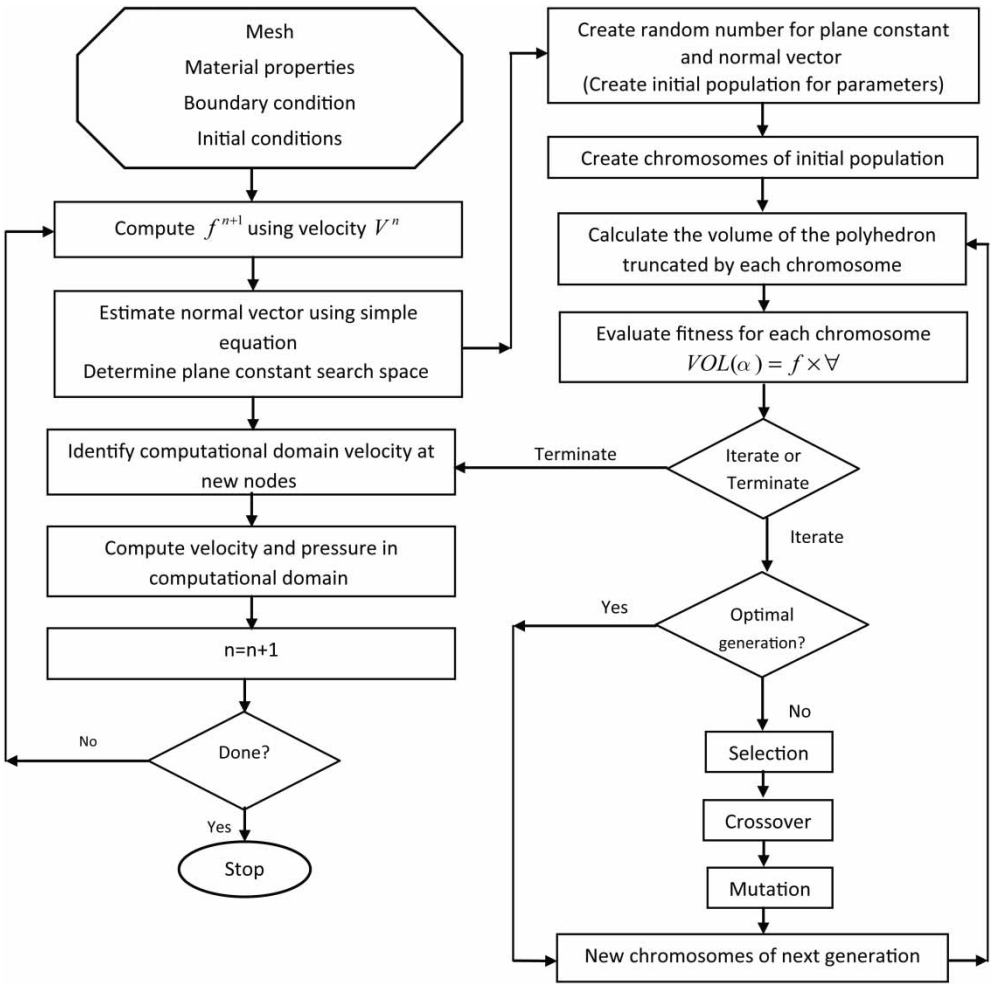

Figure 7 | Solution procedure for the fluid flow with moving free surface.

- obtain information on mesh geometer;
- specify boundary and initial conditions.

2. For a given velocity field, obtain f at the new time step.
3. With the new f, determine the location of the free surface:
   - specify search space for the plane constant and interface normal vector; in the following steps, GA creates random numbers for the plane constant and normal vector in the defined search space;
   - GA evaluates random numbers in objective function and selects the best numbers to generate the next population;
   - GA uses the selection, mutation and crossover techniques to generate optimal new numbers;
   - repeat steps evaluating fitness and produce random numbers until Equation (24) is satisfied.
4. Identify the computational domain. In the following step, specify the velocity and pressure magnitudes on the new computational domain.
5. Repeat steps 2–4 until the prescribed final time is reached.

Thus, emphasis is placed on the new free surface-tracking method that can trace the free surface with minimal numerical smearing. Then, the proposed free surface-tracking scheme became much more efficient when combined with the projection method in solving Navier–Stokes equations.

RESULTS AND DISCUSSION

In order to evaluate the accuracy of the new proposed method, the problems of dam break and bore movement on water surface have been tested. The mentioned problems have been used as bases for evaluating numerical model by many researchers. In these two examples, the experimental and analytical information are available for comparison with the numerical model results. These examples offer a substantial challenge to any free surface fluid flow method. Also, for the dam break and bore motion on water surface problems, experimental data and numerical simulated results (Martin & Moyce 1952; Wang 1992; Bonet & Look 1999; Cruchaga *et al*. 2007) are available.

Dam break problem

As a first example, the dam break problem is considered. This is a good test problem since it has simple initial geometry and the experimental and numerical data are available for this case study. More importantly, the appearance of both vertical and horizontal positions of free surface and water front provides a check on the capability of the proposed numerical model to treat free surfaces. The definition of the problem is shown in Figure 8.

The water column in hydrostatic equilibrium is confined between a fixed wall and the gate. At the beginning of calculations, $t = 0$, the gate is completely removed and the water column starts to collapse under the influence of gravity, with the water wave propagating to the right. The fluid properties and other conditions in this study are those reported in the literature (Cruchaga *et al*. 2007). Density and viscosity for water and air were to be set 1,000 kg/m^3, 1×10^{-3} Pa.s and 1 kg/m^3, 1×10^{-5} Pa.s, respectively and the gravitational acceleration was to be set to 9.81 m/s^2. The time step used in this simulation was 0.001 s and the computational field was discretized with uniform cells of $\Delta x = 0.01$ m. Free surface profiles at various times are shown in Figure 9.

The position of water front along the bottom and the residual water height along the left vertical wall with respect to the elapsed time are shown in Figure 10. The experimental data and the other researchers' numerical results are also included in this Figure. For convenience, dimensionless time and length are defined as $t^* = t\sqrt{g/H}$, $y^* = y/H$ and $x^* = x/L$, respectively. The comparisons in Figure 10 show that the numerical results of the current study have good

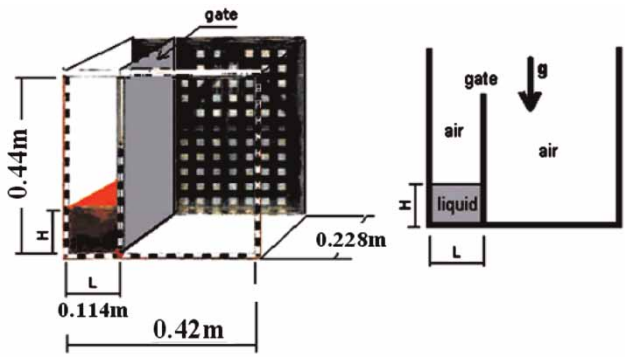

Figure 8 | Dam break problem.

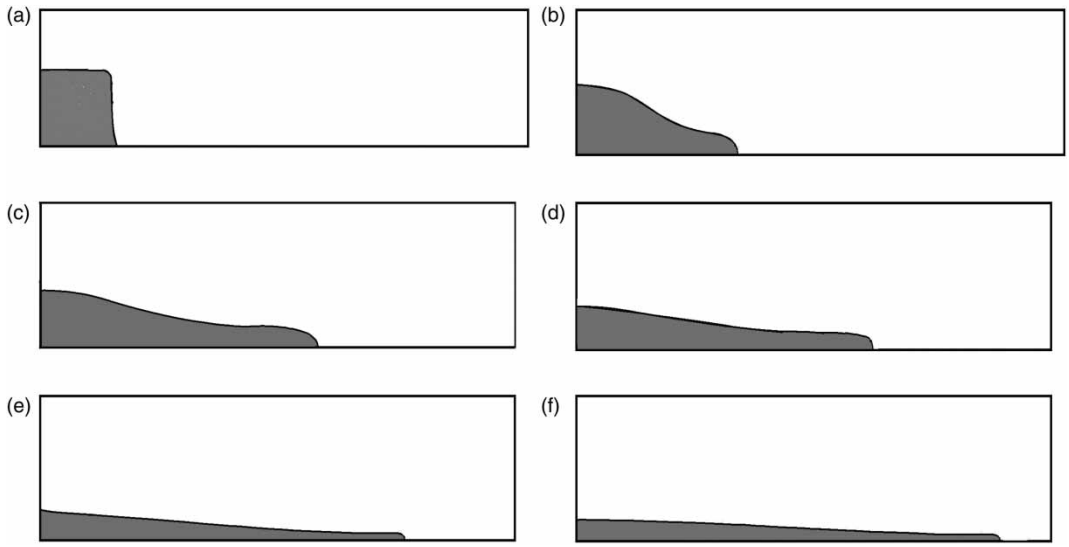

Figure 9 | Free surface profiles for the dam break problem at various times: (a) 0.02 s, (b) 0.1 s, (c) 0.15 s, (d) 0.2 s, (e) 0.25 s, (f) 0.3 s.

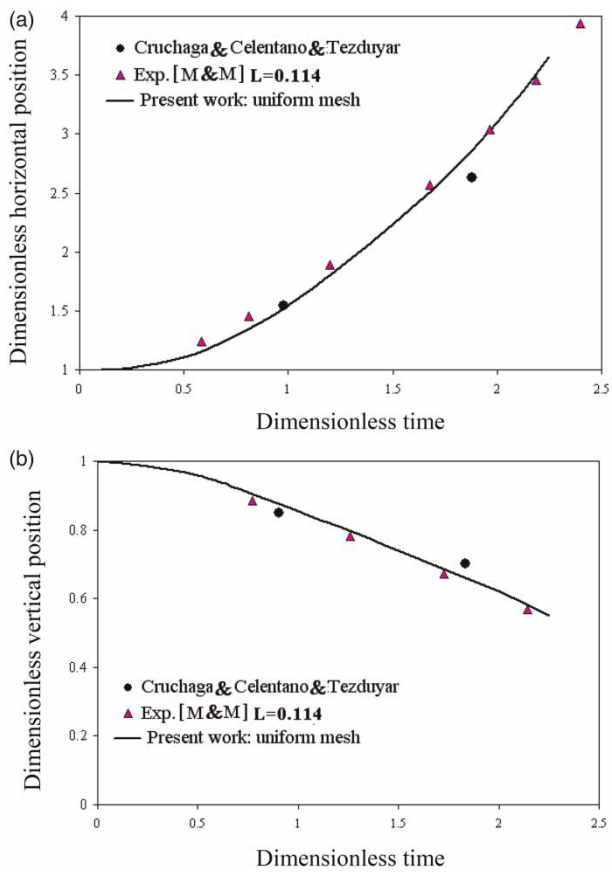

Figure 10 | (a) Position of the water waves front and (b) height of the residual water column as a function of time.

agreement with the experimental data and the proposed numerical model has a good performance for large deformation free surface flows. Also, comparisons indicate that the GA has a good influence on the accuracy of the present numerical results.

Furthermore, in order to show the advantage of the coupling scheme, a convergence test based on the numerical error history was done. As shown in Figure 11, the model with coupling scheme converges with fewer number of iterations compared with the model without coupling scheme. Coupling scheme was described above under 'New free surface tracking scheme'.

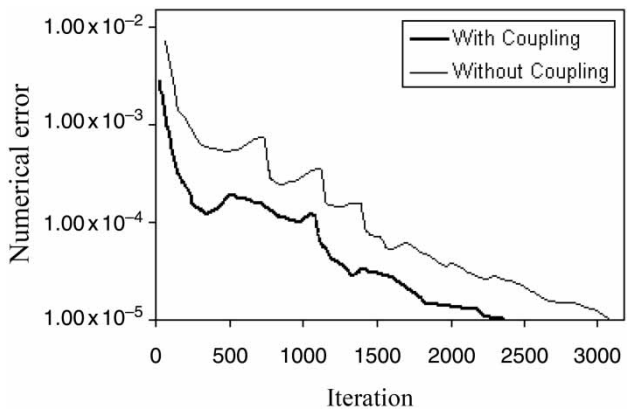

Figure 11 | Numerical error history at dam break problem.

Bore movement on water surface problem

As a second example, the bore movement on water surface is studied to evaluate the numerical model in dealing with highly unsteady problems. The mentioned example was also published in the literature (Bonet & Look 1999; Cruchaga *et al.* 2005). The layout of the idealized problem is shown in Figure 12.

A horizontal layer of water with initial depth of h_0 is pushed back to the fixed wall with a velocity v_0 applied at the other end. This generates a wave near the wall that runs away from the wall with a wave height of h and wave speed of v_s. If h_0 and v_0 are known, h and v_s can be analytically computed by the following equation (Cruchaga *et al.* 2005):

$$v_s = -v_0 + \left(0.5 \times gh \times \left(1 + \frac{h}{h_0}\right)\right)^{0.5}$$
$$\frac{h}{h_0} = \frac{v_s + v_0}{v_s} \tag{26}$$

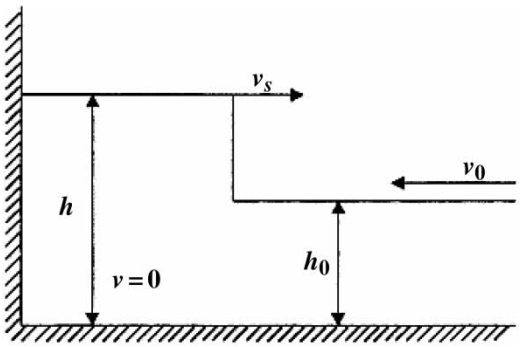

Figure 12 | Bore problem.

In order to model this phenomenon with the proposed numerical method, the initial wave height and velocity are selected 0.1 m and 0.291 m/sec, respectively (Cruchaga *et al.* 2005). Slip conditions are assumed at the solid boundaries, and the pressure is set to zero at the top of the computational domain. Density and viscosity of water and air are set to be 1,000 kg/m^3, 1×10^{-3} Pa.s, and 1 kg/m^3, 1×10^{-5} Pa.s, respectively. The length of the channel is considered to be L = 0.75 m which is discretized along the channel with uniform cells of $\Delta x = 0.01$ m. The positions of free surface at different times, obtained from the proposed numerical model are shown in Figure 13. It can be seen that the water level increases at the left wall. Once the bore is fully developed, the wave height reaches a constant level. Table 1 presents a comparison of numerical model results with the analytical solution, smooth particle hydrodynamic (SPH) method and the other research methods.

The interface profiles and results shown in Figure 13 and Table 1 are slightly different from those reported (Bonet & Look 1999), mainly due to the different numerical methods used. Nevertheless, as shown in Table 1, the results are in reasonably good agreement with the reported ones (Bonet & Look 1999).

As before, in order to present the advantage of the coupling scheme, again a convergence test based on the numerical error history was carried out for this model test. As shown in Figure 14, the model with coupling scheme converges with fewer number of iterations compared with the model without coupling scheme.

Figure 13 | Interface positions at various instants.

Table 1 | Mean values of the wave height and speed

	Analytical solution	SPH method (Bonet & Look 1999)	Other numerical method (Cruchaga *et al.* 2005)	Present method
Wave height (m)	0.132	0.130	0.134	0.133
Wave speed (m/s)	0.928	0.910	0.900	0.905

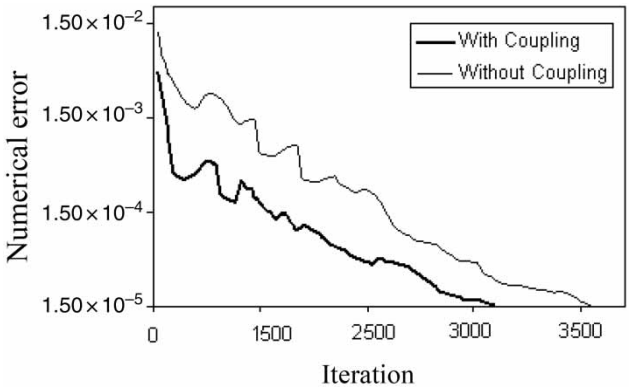

Figure 14 | Numerical error history at bore movement problem.

CONCLUSIONS

In this paper, a new computing method has been proposed based on using GA and the original VOF method for simulation of the time-dependent, viscous, incompressible flow with moving free surface. The proposed technique has more simplicity and efficiency compared to the other numerical models for studying a wide range of free surface problems. The new combinational method is characterized in determining orientation vector and plane constant. These parameters mainly give information on the free surface orientation and VOF in a cell. For this study, by using a simple relationship of the previous studies, the combinational method creates a space search for unknown parameters and attempts to determine the optimum solution in the defined search space. In order to increase the speed of solution, using information on adjacent cells is necessary. By the techniques used in the numerical model, free surface could be traced with minimal numerical effort. Validity of the present method has been shown in simulation of the dam break and bore movement on water surface problems. The results have also demonstrated the efficiency and accuracy of the model. The proposed method can be easily implemented in similar problems outlined in the current study.

REFERENCES

Adeli, H. & Cheng, N. T. 1993 Integrated genetic algorithm for optimization of space structures. *J. Aerosp. Eng.* **6** (4), 315–328.

Bonet, J. & Look, T. 1999 Variational and momentum preservation aspects of smooth particle hydrodynamic formulations. *J. Comp. Meth. Appl. Mech. Eng.* **180**, 97–115.

Brackbill, J. U., Kothe, D. B. & Zemach, C. 1992 A continuum method for modeling surface tension. *J. Comput. Phys.* **100**, 335–350.

Bussman, M. 2000 A Three Dimensional Model of an Impacting Droplet. PhD Thesis. University of Toronto, Canada.

Cruchaga, M. A., Celentano, D. G. & Tezduyar, T. E. 2005 Moving-interface computations with the edge tracked interface locator technique. *Int. J. Numer. Meth. Fluids* **47**, 451–469.

Cruchaga, M. A., Celentano, D. G. & Tezduyar, T. E. 2007 Collapse of a liquid column: numerical simulation and experimental validation. *J. Comput. Mech.* **39**, 453–476.

Gerlach, D., Tomar, G., Biswas, G. & Durst, F. 2006 Comparison of volume-of-fluid methods for surface tension-dominant two phase flows. *Int. J. Heat Mass Trans.* **49**, 740–754.

Ghiassi, R. 1995 Three Dimensional Coastal Flow Modeling using the Finite Volume Method. PhD Thesis. University of Bradford, UK.

Goldberg, D. E. 1989 *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Company Inc., USA.

Gueyffier, D., Li, J., Nadim, A., Scardovelli, R. & Zaleski, S. 1999 Volume of fluid interface tracking with smoothed surface stress methods for three dimensional flows. *J. Comput. Phys.* **152**, 423–456.

Hakimzadeh, H. 1997 Turbulence Modeling of Tidal Currents in Rectangular Harbors. PhD Thesis. University of Bradford, UK.

Harlow, F. H. & Welch, J. E. 1965 Numerical calculation of time dependent viscous incompressible flow of fluid with free surface. *Int. J. Numer. Meth. Fluids* **12**, 2182–2189.

Hirsch, C. 1990 *Numerical Computation of Internal and External Flows*. John Wiley & Sons, Chichester, UK.

Hirt, C. W., Amsden, A. A. & Cook, J. L. 1974 An arbitrary Lagrangian–Eulerian computing method for all flow speeds. *J. Comput. Phys.* **14**, 227–253.

Jessee, J. P. & Fiveland, W. A. 1996 A cell vertex algorithm for the incompressible Navier–Stokes equations on non-orthogonal grids. *Int. J. Numer. Meth. Fluids.* **23**, 271–293.

Jian, Y., Mccorquodale, J. A. & Barron, R. M. 1998 A three dimensional hydrodynamic model in curvilinear co-ordinates with collocated grid. *Int. J. Numer. Meth. Fluids.* **28**, 1109–1134.

Kim, M. S. & Lee, W. I. 2003 A new VOF-based numerical scheme for the simulation of fluid flow with free surface. Part I: new free surface-tracking algorithm and its verification. *Int. J. Numer. Meth. Fluids.* **42**, 765–790.

Kleefsman, K. M. T., Fekken, G., Veldman, A. E. P., Iwanowski, B. & Buchner, B. 2005 A volume of fluid based simulation method for wave impact problems. *J. Comput. Phys.* **1206**, 363–393.

Kothe, D. B., Rider, W. J., Mosso, S. J., Brock, J. S. & Hochstein, J. I. 1996 Volume tracking of interfaces having surface tension in two and tree dimensions. *AIAA*, 96–0859.

Krishnkumar, K. & Goldberg, D. E. 1992 Control system optimization using genetic algorithms. *J. Guidance, Control Dynam.* **15**, 735–740.

Martin, J. C. & Moyce, W. J. 1952 An experimental study of the collapse of liquid on a rigid horizontal plane. *Philos. Trans. R. Soc. Lond. A. Math. Phys. Eng. Sci.* **244**, 312–324.

Meier, M., Yadigarouglu, G. & Smith, B. L. 2002 A novel technique for including surface tension in PLIC-VOF methods. *Eur. J. Mech. B Fluids.* **21** (1), 61–73.

Neofyton, P. 2005 A third orders upwind finite volume method for generalised Newtonian fluid flows. *Adv. Eng. Softw.* **36**, 664–680.

Noh, W. F. & Woodward, P. 1976 SLIC: (simple line interface calculation). In: *Fifth International Conference on Fluid Dynamics* (A. I. Vande & P. J. Zandbergen, eds). Lecture in Physics, 59, pp. 330–340.

Poloni, C., Giurgevich, A., Onesti, L. & Pediroda, V. 2000 Hybridization of multi-objective genetic algorithms, a neural network and a classical optimizer for a complex design problem in fluid dynamics. *Comput. Meth. Appl. Mech. Eng.* **186**, 403–420.

Rider, W. J. & Kothe, D. B. 1998 Reconstructing volume tracking. *J. Comput. Phys.* **141**, 112–152.

Rudman, M. 1998 A volume tracking method for incompressible multifluid flows with large density variations. *Int. J. Numer. Meth. Fluids.* **28**, 357–378.

Scardovelli, R. & Zaleski, S. 2000 Analytical relations connecting linear interfaces and volume fractions in rectangular grids. *J. Comput. Phys.* **164**, 228–237.

Tryggvason, G., Bunner, B., Esmaeeli, A., Juric, D., Al-Rawahi, N., Tauber, W., Han, J., Nas, S. & Jan, Y. J. 2001 A front-tracking method for the computations of multiphase flow. *J. Comput. Phys.* **169**, 708–759.

Ubbink, O. & Issa, R. I. 1999 A method for capturing sharp fluid interfaces on arbitrary meshes. *J. Comput. Phys.* **153**, 26–50.

Udaykumar, H. S., Kan, H. C., Shyy, W. & Tran-Son-Tay, R. 1997 Multiphase dynamics in arbitrary geometries on fixed Cartesian grids. *J. Comput. Phys.* **137**, 366–405.

Unverdi, S. O. & Tryggvason, G. 1992 A front-tracking method for viscous, incompressible, multi-fluid flow. *J. Comput. Phys.* **100**, 25–37.

Young, D. L. 1982 Time-dependent multi-material flow with large fluid distortion. In: *Numerical Methods for Fluid Dynamics* (K. W. Morton & M. J. Baines, eds). Academic Press, New York, pp. 273–285.

Young, D. L. 1984 An interface tracking method for a 3d Eulerian hydrodynamics code. Technical Report, 44/92/35/, Awre.

Wang, S. P. 1992 Numerical Simulation and Experimental of Incompressible Viscous with Moving Free Surface. PhD Thesis. Cornell University, Ithaca, NY.

First received 22 September 2012; accepted in revised form 18 November 2013. Available online 3 March 2014