INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2010; **63**:651–680
Published online 1 June 2009 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.2090

Numerical simulation of free-surface flow using the level-set method with global mass correction

Yali Zhang*,†, Qingping Zou and Deborah Greaves

PRIMaRE Peninsular Research Institute for Marine Renewable Energy, Centre for Coastal Dynamics and Engineering, School of Engineering, University of Plymouth, Devon PL4 8AA, U.K.

SUMMARY

A new numerical method that couples the incompressible Navier–Stokes equations with the global mass correction level-set method for simulating fluid problems with free surfaces and interfaces is presented in this paper. The finite volume method is used to discretize Navier–Stokes equations with the two-step projection method on a staggered Cartesian grid. The free-surface flow problem is solved on a fixed grid in which the free surface is captured by the zero level set. Mass conservation is improved significantly by applying a global mass correction scheme, in a novel combination with third-order essentially non-oscillatory schemes and a five stage Runge–Kutta method, to accomplish advection and re-distancing of the level-set function. The coupled solver is applied to simulate interface change and flow field in four benchmark test cases: (1) shear flow; (2) dam break; (3) travelling and reflection of solitary wave and (4) solitary wave over a submerged object. The computational results are in excellent agreement with theoretical predictions, experimental data and previous numerical simulations using a RANS-VOF method. The simulations reveal some interesting free-surface phenomena such as the free-surface vortices, air entrapment and wave deformation over a submerged object. Copyright © 2009 John Wiley & Sons, Ltd.

Received 13 February 2009; Revised 17 April 2009; Accepted 17 April 2009

KEY WORDS: level-set method; global mass correction; mass conservation; free-surface flow; incompressible fluid; finite volume method

1. INTRODUCTION

Efficient and accurate computation of incompressible free-surface flow problems has enormous value in numerous scientific and industrial applications [1]. Applications include marine and coastal

*Correspondence to: Yali Zhang, PRIMaRE Peninsular Research Institute for Marine Renewable Energy, Centre for Coastal Dynamics and Engineering, School of Engineering, University of Plymouth, Devon PL4 8AA, U.K.
†E-mail: yali.zhang@plymouth.ac.uk

Contract/grant sponsor: South West of England Regional Development Agency
Contract/grant sponsor: Natural Environmental Research Council; contract/grant number: NE/E002129/1

fluid–structure interactions, wave sloshing in tanks, wave loading and run-up, crystal growth, image processing and so on.

Strategies for dealing with flows containing the free interface fall into two main categories. These are interface-tracking methods [2–6] and interface capturing methods [1, 7–14]. The interface-tracking methods include boundary integral method [2, 3], particle- tracking method [15] and front-tracking method [4–6], in which the free surface is explicitly identified and tracked by predefined markers. Generally the governing equations are solved within the single fluid and at free-surface grid cells. An alternative to interface tracking is interface capturing, which includes volume of fluid (VOF) methods [7, 11, 12] and level-set methods [1, 8, 9, 13, 14]. They differ from interface tracking in that the solution may be obtained in both air and water with the fluid properties changing at the interface. The free surface is implicitly captured by a contour of a certain volume fraction function.

The boundary integral approach is efficient as only the interface separating liquid and gas is discretized, but the approach assumes potential flow in each fluid and thus neglects viscous effects, such as vortex shedding and separation. Furthermore, complex grid movement must be implemented in order to continue the computation from one time step to the next [2, 3]. Particle-tracking methods, such as smoothed particle hydrodynamics, SPH [15], are powerful in that they eliminate the need for a mesh and so complex mesh movement issues. However, they can be computationally expensive, especially in 3D, as they require a large number of closely spaced particles to represent the fluid motion. In the front-tracking method for free-surface problems used extensively by Glimm [5], Tryggvason *et al*. [6], markers distributed evenly on the interface are used to track the interface. During a simulation, markers drift along the moving interface and may become bunched or stretched in some areas. In these cases, markers are added, deleted or even redistributed if necessary for the sake of regularity. This difficulty in handling changes in interface topology is an inherent weakness of the front-tracking method.

The VOF method employs a volume fraction function, which is the fraction of volume of the computational cell filled with the reference fluid. From the volume fraction, a reconstruction of the interface is made and the interface is implicitly propagated by updating the volume fraction function. Interface reconstruction from the volume fraction function is required to determine the weighted density and viscosity for the computational cells and to compute the volume flux necessary for the convective terms in the evolution equation. The choice of method to reconstruct the interface geometry distinguishes members of the VOF method, e.g. simple line interface calculation (SLIC) method by Noh and Woodward [11], piecewise linear interface calculation (PLIC) by Youngs [12], SOLV-VOF by Hirt and Nichols [16] and parabolic reconstruction of surface tension (PROST) by Renardy and Renardy [17]. VOF methods are conservative and can deal with topological changes of the interface. However, a high order of accuracy is hard to achieve due to the discontinuity of the volume fraction function. Also the properties of the interface such as normal and curvature are difficult to calculate accurately.

The level-set method was proposed by Osher and Sethian [13] for curvature-driven motions. Reviews of the level-set method are given by Osher and Fedkiw [18] and Sethian [19]. Application of the method to incompressible multiphase flows is extended by Sussman *et al.* [20] and Chang *et al.* [14]. The level-set function is defined as the signed shortest normal distance from the interface in a fixed grid system. To distinguish between the two fluids, a positive sign is associated with one fluid and a negative sign is attached to the other. Thus, the interface is represented by the zero level set. Fluid properties can be calculated from the level-set function through a smoothed Heaviside function. The evolution of the level-set function is governed by a standard advection equation, i.e.

DOI: 10.1002/fld

the level-set equation. The level-set method is capable of computing geometric properties of highly complicated boundaries naturally and can be extended to 3D easily. However, one of the common difficulties is how to keep the interface thickness finite and to conserve mass. Discretization of the level-set equation can lead to significant numerical dissipation and cause a mass loss. Thus, the level-set function is solved numerically using higher-order advection schemes. In the evolution of the level-set function, a steep gradient of the distance function in areas of high curvature is formed and the distance function drifts away from its property. This again causes mass conservation problems. To alleviate such problems, the redistancing procedure introduced by Sussman *et al.* [20] is performed. The detailed redistancing procedure is given by Sussman and Fatemi [21]. It improves mass conservation significantly. However, it still does not eliminate entirely the mass imbalance problem of the level-set method.

There have been several attempts to improve mass conservation of level-set methods. Enright *et al.* [22] devised the particle-level-set method. The motions of massless particles seeded around the interface are tracked. As the particles move with the same velocity as the level-set function, they should not cross the zero level set unless numerical inaccuracies arise. With such an additional constraint accounted for in the evolution of the level-set function, very good mass conservation is achieved. Bourlioux [23] and Sussman and Puckett [24] coupled the level-set method with the VOF method to use the strength of the VOF method in conserving mass. In the coupled level-set/VOF approach, while the volume fraction function is used for mass conservation, the level-set function is used for better approximation of geometrical quantities. Chang *et al.* [14] proposed a re-initialization procedure to preserve mass during the evolution of the level-set function. The idea is to conserve the total mass in the domain by solving a perturbed Hamilton–Jacobi equation to a steady state. Although the total mass in the domain is well preserved, the mass conservation of each individual fluid is not clear. Lakehal *et al.* [25] applied global area or volume correction technique for mass correction. Mass conservation is deduced with the area or volume of the reference fluid conserved. A correction through an iterative procedure is added to the level-set function so that area or volume of the reference fluid represented by the new corrected level-set function is equal to its initial area or volume. In all these cases, the original simplicity of level-set methods is partly lost.

In this paper, a new approach is taken in which the global mass correction method [9] is coupled together with the level-set and re-distancing operations, resulting in excellent conservation of mass of the reference fluid (selected from one of the fluids). This in turn conserves the mass of other fluids automatically. It is worth pointing out that the methodology differs significantly from the free-surface method of Yap *et al.* [9]. First, a dual time stepping instead of simple explicit time step is used to solve the global mass correction equation to speed up the convergence. Second, a third-order essentially non-oscillatory scheme (ENO), instead of the second-order curve linear advection method (CLAM) used by Yap *et al.* [9], is used to model convection in the level-set equation. Also a five stage Runge–Kutta time integration method rather than a semi-implicit scheme with deferred correction for the level-set equation is applied to better achieve total variation stability [26]. Finally, two-step projection instead of SIMPLER is used to resolve the velocity–pressure coupling. This eliminates the need to solve two Poisson equations, resulting in an efficient computation. Thus, a numerical method that couples 2D incompressible Navier–Stokes equations with the level-set method in a Cartesian coordinate system is presented here for study of free-surface flows. The paper is organized as follows. Section 2 presents the mathematical model for the Navier–Stokes equations, smoothing method of the two phase flow system, level-set equation, redistancing for the level-set and global mass correction. Section 3 describes the numerical methods for the

DOI: 10.1002/fld

Navier–Stokes solver and level-set solver. In Section 4, results obtained from application of the coupled solver to four bench mark test cases are presented: (1) shear flow; (2) dam break; (3) solitary waves; (4) solitary waves with submerged objects. Finally, conclusions are given in Section 5.

2. MATHEMATICAL MODEL

Governing equations for an incompressible fluid flow are the mass conservation equation and the Navier–Stokes momentum conservation equations written as

$$\frac{\partial u_j}{\partial x_j}=0 \tag{1}$$

and

$$\frac{\partial u_i}{\partial t}+\frac{\partial (u_i u_j)}{\partial x_j}=-\frac{1}{\rho}\frac{\partial p}{\partial x_i}+\frac{1}{\rho}\frac{\partial \tau_{ij}}{\partial x_j}+\frac{1}{\rho}f_i \tag{2}$$

where subscript $i=1, 2$ denotes the 2D geometrical descriptions and Cartesian tensor notation is used. u_j, p and x_j are the velocities, pressure and spatial coordinates. f_i represents the body force. τ_{ij} is the viscous term given by

$$\tau_{ij}=\mu\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right) \tag{3}$$

ρ, μ are the density and viscosity appropriate for the phase occupying the particular spatial location at a given instance of time. For immiscible liquids, density and viscosity are constant along particle paths and they are advected by fluid velocity; therefore, they satisfy

$$\frac{\partial \rho}{\partial t}+u_j\frac{\partial \rho}{\partial x_j}=0 \tag{4}$$

and

$$\frac{\partial \mu}{\partial t}+u_j\frac{\partial \mu}{\partial x_j}=0 \tag{5}$$

Owing to the existence of steep gradients in density and viscosity across the free surface, excessive numerical diffusion is experienced when computing viscous flows [27] if Equations (4) and (5) are solved. Instead, the level-set method is used to capture the interface between the two phases. The level-set function ϕ is defined as the signed distance function from the interface. Henceforth, the phases with positive ϕ and negative ϕ will be denoted as phase 1 (liquid) and phase 2 (gas) respectively for ease of presentation. The evolution of the level-set function is governed by

$$\frac{\partial \phi}{\partial t}+u_j\frac{\partial \phi}{\partial x_j}=0 \tag{6}$$

Since the density and viscosity at the interface are discontinuous across the interface, either excessive numerical diffusion or problems with oscillations around the jump are expected if there

DOI: 10.1002/fld

is no special treatment used at the front. To avoid these problems, an interface grid that explicitly marks the position of the interface is introduced. Also to avoid introducing disturbances of length scale equal to the mesh by having the properties jump abruptly from one grid point to the next, the interface is not kept sharp but given a small thickness ε of the order of the mesh size [28]. In this transition zone the fluid properties change smoothly from the value on one side of the interface to the value on the other side. This artificial interface thickness is a function of the mesh size used, only, and does not change during the calculations. Therefore, no numerical diffusion is present. This transition zone is achieved by defining a smoothed Heaviside function.

$$H(\phi)=\begin{cases}0, & \phi<-\varepsilon\\ \dfrac{\phi+\varepsilon}{2\varepsilon}+\dfrac{1}{2\pi}\sin\left(\dfrac{\pi\phi}{\varepsilon}\right), & -\varepsilon\leqslant\phi\leqslant\varepsilon\\ 1, & \phi>\varepsilon\end{cases} \tag{7}$$

where ε is related to the grid size and is usually taken as a factor of the grid spacing. Using the smoothed Heaviside function, these properties are calculated using

$$\beta=(1-H)\beta_1+H\beta_2 \tag{8}$$

where β can be density, viscosity or another property of interest. Since ϕ is the signed normal distance from the interface, it satisfies

$$\nabla|\phi|=0 \tag{9}$$

When Equation (6) moves the level set $\phi=0$ at the correct velocity, ϕ may become irregular after some period of time [20]. In a complex non-uniform flow field, it is possible for ϕ to develop steep gradients from Equation (6), especially when the interface has a steep slope. As a consequence, it becomes difficult to maintain a finite thickness of the transition zone. The computation of the unit normal and curvature is no longer accurate and a significant loss or gain of mass occurs. Thus, to ensure that ϕ remains a distance function that satisfies Equation (9), redistancing must be performed. This is achieved by solving for a second distance function ϕ' given by the following equation:

$$\frac{\partial\phi'}{\partial\bar{t}}+s(\phi)(|\nabla\phi'|-1)=0 \tag{10}$$

Here, $\bar{t}$ is a pseudotime for the variable ϕ', the initial condition is $\phi'(\mathbf{x},0)=\phi(\mathbf{x})$ and $s(\phi)$ is the smoothed sign function defined as

$$s(\phi)=\frac{\phi}{\sqrt{\phi^2+(|\nabla\phi|\varepsilon)^2}} \tag{11}$$

For incompressible flows, the total mass is conserved in time. However, the numerical discretization of the level-set formulation does not preserve this property in general. Even with the above redistancing process for the level-set function, it has been found that a considerable amount of total mass is lost in time [9, 14]. To overcome this difficulty, the global mass correction coupled

DOI: 10.1002/fld

with the first and second distance functions is used to preserve the total mass in time. Thus, the steady-state solution to a third distance function ϕ'' is obtained using:

$$\frac{\partial \phi''}{\partial t'} = M_{\text{cor}} \tag{12}$$

where t' and M_{cor} are a pseudotime, which can be different from $\bar{t}$, and mass correction factor. The steady-state values of the second distance function ϕ' are used as the initial condition for Equation (12).

There are two scenarios where mass surplus or depletion may occur during the computation process. Mass must be removed or added to ensure the conservation of the reference phase. Thus, the mass correction factor is calculated to ensure the mass conservation.

$$M_{\text{cor}} = \text{sign}(\phi_{\text{ref}}) \frac{M_o - M_t}{M_o} \tag{13}$$

where M_o and M_t are the original total mass and the total mass of the reference phase at time t, respectively. Either fluid can be chosen as the reference fluid. The original mass is the mass calculated at the beginning. The mass at time t increases when there is an injection of the reference phase but decreases when there is removal of the reference phase. Depending on the choice of the reference phase, the mass of the reference can be calculated using

$$M = \begin{cases} \sum \rho_{\text{ref}} H_{\text{ref}} \Delta V, & H_{\text{ref}} = 1 \\ \sum \rho_{\text{ref}} (1 - H_{\text{ref}}) \Delta V, & H_{\text{ref}} = 0 \end{cases} \tag{14}$$

3. NUMERICAL METHOD

3.1. Navier–Stokes solver

A finite volume method is employed to solve the Navier–Stokes equations (1) and (2) on a staggered Cartesian grid. The typical control volume (CV) for p, u, v is shown in Figure 1. A node (grid point) labelled P is located at the centre of the pressure CV. The neighbouring nodes are denoted as W, E, N and S. This CV has four boundaries, denoted by e, w, n and s (with area of A_e, A_w, A_n and A_s, respectively). Scalar variables, such as pressure, density, viscosity and the level-set variable, are stored at the node P. Velocity components are stored at the CV boundaries. While u is staggered half a CV to the left, v is staggered half a CV downward. A staggered grid is adopted to ensure strong pressure–velocity coupling and enforce mass conservation in each CV. The two-step projection method [29] has been used to solve the momentum equations

$$\frac{\tilde{u}_i^{k+1} - u_i^k}{\Delta t} + \frac{\partial (u_i^k u_j^k)}{\partial x_j} = -\frac{1}{\rho} \frac{\partial p^{k+1}}{\partial x_i} + \frac{1}{\rho} \frac{\partial \tau_{ij}^k}{\partial x_j} + \frac{1}{\rho} f_i^k \tag{15}$$

The first step is to introduce an intermediate velocity $\tilde{u}_i$ as

$$\frac{\tilde{u}_i^{k+1} - u_i^k}{\Delta t} + \frac{\partial (u_i^k u_j^k)}{\partial x_j} = \frac{1}{\rho} \frac{\partial \tau_{ij}^k}{\partial x_j} + \frac{1}{\rho} f_i^k \tag{16}$$

DOI: 10.1002/fld

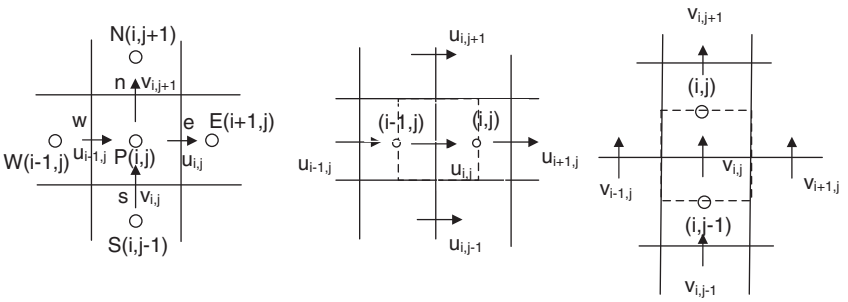


Figure 1. A staggered grid arrangement for the p, u, v control volume.

in which the superscript indicates the time level and Δt is the time step size. The time step size must satisfy the Courant–Friedrichs–Lewy (CFL) condition $\Delta t \leqslant \min_{\Omega} 0.5/(|u|+|v|)$, Ω is the whole computational domain. The intermediate velocity $\tilde{u}_i$ does not include the pressure effect. The second step is to update the intermediate velocity field onto a divergence free plane to obtain the final velocity:

$$\frac{u_i^{k+1}-\tilde{u}_i^{k+1}}{\Delta t}=-\frac{1}{\rho^k}\frac{\partial p^{k+1}}{\partial x_i} \tag{17}$$

$$\frac{\partial u_i^{k+1}}{\partial x_i}=0 \tag{18}$$

Equation (18) is the continuity equation. Taking the divergence of Equation (17) and applying Equation (18) to the resulting equation yields the Poisson pressure equation,

$$\frac{\partial}{\partial x_i}\left(\frac{1}{\rho^k}\frac{\partial p^{k+1}}{\partial x_i}\right)=\frac{1}{\Delta t}\frac{\partial \tilde{u}_i^{k+1}}{\partial x_i} \tag{19}$$

Equation (19) is solved for the pressure using the intermediate velocity. The updated pressure information is then used in Equation (17) to obtain the final velocity field.

Equation (16) can be cast into a general transport equation of the form:

$$\underbrace{\frac{\partial \Phi}{\partial t}}_{\text{Transient}}+\underbrace{\frac{\partial (u\Phi)}{\partial x}+\frac{\partial (v\Phi)}{\partial y}}_{\text{Convection}}=\underbrace{\frac{\partial}{\partial x}\left(\frac{\mu}{\rho}\frac{\partial \Phi}{\partial x}\right)+\frac{\partial}{\partial y}\left(\frac{\mu}{\rho}\frac{\partial \Phi}{\partial y}\right)}_{\text{Diffusion}}+\underbrace{S}_{\text{Source}} \tag{20}$$

where Φ represents the unknown velocities u or v.

Integrating Equation (20) over a CV with the Gauss divergence theorem and rearranging gives

$$\frac{(\Phi_P-\Phi_P^0)\Delta V}{\Delta t}+J_e-J_w+J_n-J_s=S\Delta V \tag{21}$$

DOI: 10.1002/fld

where

$$J_e = \left(u\Phi - \frac{\mu}{\rho}\frac{\partial \Phi}{\partial x} \right)_e A_e \tag{22a}$$

and

$$J_n = \left(v\Phi - \frac{\mu}{\rho}\frac{\partial \Phi}{\partial y} \right)_n A_n \tag{22b}$$

J_w and J_s are fluxes on the west and south faces of the CV. All terms are evaluated at $t+\Delta t$ except Φ_P^0, which is evaluated at t. J terms in Equations (22) are evaluated at the CV surfaces. These J terms need to be related to Φ at the neighbouring nodes. For this purpose, the mixed scheme [30] is employed. For the east face of the CV, e, it is defined as

$$J_e = \left[\lambda_1 (\| F_e, 0 \| \Phi_P - \| -F_{e,0} \| \Phi_E) + (1-\lambda_1) F_e \frac{(\Phi_P + \Phi_E)}{2} \right] - D_e(\Phi_E - \Phi_P) \tag{23}$$

where

$$F_e = u_e A_e \tag{24}$$

$$D_e = \left(\frac{\mu}{\rho} \right)_e \frac{A_e}{\delta x_e} \tag{25}$$

and $\|A, B\|$ is defined as the greater of A and B.

The coefficient λ_1 is the weighting factor between the upwind and central difference method. In practice, λ_1 is generally selected in the range of 0.3–0.5 to produce stable and accurate results. μ_e and ρ_e are the harmonic mean of value at the two adjacent grid points, and J_w, J_n and J_s are defined similarly. For numerical implementation, Equation (21) can be written in a compact form as

$$a_P \Phi_P = a_E \Phi_E + a_W \Phi_W + a_N \Phi_N + a_S \Phi_S + b \tag{26}$$

where

$$a_E = D_e[\lambda_1 + (1-\lambda_1)(1-0.5|Pe_e|)] + \| -F_e, 0 \|$$

$$a_W = D_w[\lambda_1 + (1-\lambda_1)(1-0.5|Pe_w|)] + \| F_w, 0 \|$$

$$a_N = D_n[\lambda_1 + (1-\lambda_1)(1-0.5|Pe_n|)] + \| -F_n, 0 \|$$

$$a_S = D_s[\lambda_1 + (1-\lambda_1)(1-0.5|Pe_s|)] + \| F_s, 0 \|$$

$$a_P = a_E + a_W + a_N + a_S + \Delta V/\Delta t$$

$$b = S\Delta V + \Delta V \Phi_0/\Delta t$$

and the Peclet number, Pe, at CV faces,

$$Pe_e = F_e/D_e, \quad Pe_w = F_w/D_w, \quad Pe_n = F_n/D_n, \quad Pe_s = F_s/D_s \tag{27}$$

DOI: 10.1002/fld

The Poisson pressure equation can be expressed as

$$a_P P_P = a_E P_E + a_W P_W + a_N P_N + a_S P_S + b \tag{28}$$

where

$$a_E = A_e/(\rho^k \Delta x)_e$$
$$a_W = A_w/(\rho^k \Delta x)_w$$
$$a_N = A_n/(\rho^k \Delta y)_n$$
$$a_S = A_s/(\rho^k \Delta y)_s$$
$$a_P = a_E + a_W + a_N + a_S$$

and

$$b = \frac{1}{\Delta t}(\tilde{u}_e^{k+1} A_e - \tilde{u}_w^{k+1} A_w + \tilde{u}_n^{k+1} A_n - \tilde{u}_s^{k+1} A_s) \tag{29}$$

The algebraic equations (26) and (28) are solved using the tridiagonal matrix algorithm [31].

3.2. Free-surface solver

Finite difference and related finite volume schemes are based on interpolations of discrete data using polynomials or other simple functions. It is well known that a wide stencil leads to higher-order accuracy interpolation. Traditional finite difference methods are based on fixed stencil interpolations. However, fixed stencil interpolation of second or higher-order accuracy is necessarily oscillatory near a discontinuity. There are two common ways to eliminate or reduce such spurious oscillations near discontinuities. One way is to add an artificial viscosity [26]. The disadvantage of this approach is that fine tuning of the parameter controlling the size of the artificial viscosity is problem dependent. More recent approaches apply limiters to eliminate the oscillations. Since we are interested in obtaining accurately the convection of the interface position, the non-conservative form of the ENO scheme will be used. Furthermore, the ENO scheme is a higher-order scheme and can have adaptive stencils. The key idea of ENO is to choose the locally 'smoothest' stencil to approximate the numerical fluxes at the cell faces, among several candidates, and hence avoid crossing discontinuities in the interpolation procedure as much as possible. Thus, the numerical viscosity is adjusted adaptively by measuring the local smoothness of the solution to eliminate spurious oscillation near the discontinuity. The ENO scheme was first described in the classical paper of Harten *et al.* [32] for hyperbolic conservation laws and later extended to the efficient implementation of ENO shock capturing schemes applied to the Hamilton–Jacobi equations by Shu and Osher [26]. The higher-order ENO scheme is obtained from the lower-order ENO by making use of a hierarchy of divided differences, and that makes the implementation of high-order ENO schemes rather straightforward. The second level-set equation (redistancing equation) is modelled based on Godunov's method [8], and Equation (10) is advanced with the five stage Runge–Kutta scheme which is total variation stable.

The convective terms are discretized as

$$\int_{\Delta A} u_j \frac{\partial \phi}{\partial x_j} \, \mathrm{d}A = u_{i+1,j} \phi_e A_e - u_{i,j} \phi_w A_w + v_{i,j+1} \phi_n A_n - v_{i,j} \phi_s A_s \tag{30}$$

DOI: 10.1002/fld

The procedure for computing the quantity ϕ_e using the third-order ENO is as follows:

$$\phi_e^3 = \phi_e^2 + \frac{1}{3} m_2 \left(\frac{3(n_2 - i)^2 - 1}{2} \right) \tag{31}$$

where ϕ_e^2 is the second-order ENO expression,

$$\phi_e^2 = \phi_e^1 + \tfrac{1}{2} m_1 (1 + 2(i - n_1)) \tag{32}$$

and ϕ_e^1 is the first-order upwind value,

$$\phi_e^1 = \phi_{n_1} \tag{33}$$

and

$$m_1 = \begin{cases} a_- & \text{if } |a_-| \leqslant |a_+| \\ a_+ & \text{otherwise} \end{cases} \tag{34a}$$

$$m_2 = \begin{cases} b_- & \text{if } |b_-| \leqslant |b_+| \\ b_+ & \text{otherwise} \end{cases} \tag{34b}$$

$$n_1 = \begin{cases} i, & u_e \geqslant 0 \\ i+1 & \text{otherwise} \end{cases} \tag{34c}$$

$$n_2 = \begin{cases} n_1 - 1 & \text{if } |a_-| \leqslant |a_+| \\ n_1 & \text{otherwise} \end{cases} \tag{34d}$$

Here, a_- and a_+ are first-order finite difference expressions and b_- and b_+ are second-order finite differences,

$$\begin{aligned} a_- &= \phi_{n_{1,j}} - \phi_{n_{1-1,j}} \\ a_+ &= \phi_{n_{1+1,j}} - \phi_{n_{1,j}} \end{aligned} \tag{34e}$$

$$\begin{aligned} b_- &= \phi_{n_{2-1,j}} - 2\phi_{n_{2,j}} + \phi_{n_{2+1,j}} \\ b_+ &= \phi_{n_{2,j}} - 2\phi_{n_{2+1,j}} + \phi_{n_{2+2,j}} \end{aligned} \tag{34f}$$

Redistancing should be done to maintain the level-set function as the signed distance from the interface. The computation of $|\nabla\phi|$ in Equation (11) is performed based on Godunov's method [8],

$$|\nabla\phi| = \begin{cases} \sqrt{\max(a_+^2, b_-^2) + \max(c_+^2, d_-^2)} & \text{if } \phi > 0 \\ 0 & \text{if } \phi = 0 \\ \sqrt{\max(a_-^2, b_+^2) + \max(c_-^2, d_+^2)} & \text{if } \phi < 0 \end{cases} \tag{35}$$

DOI: 10.1002/fld

where

$$
\begin{aligned}
a &= \frac{\phi_{i,j}-\phi_{i-1,j}}{\Delta x}+\frac{(x_{i,j}-x_{i-1,j})}{2}l^- \\
b &= \frac{\phi_{i+1,j}-\phi_{i,j}}{\Delta x}+\frac{(x_{i+1,j}-x_{i,j})}{2}l^+ \\
c &= \frac{\phi_{i,j}-\phi_{i,j-1}}{\Delta y}+\frac{(x_{i,j}-x_{i,j-1})}{2}m^- \\
d &= \frac{\phi_{i,j+1}-\phi_{i,j}}{\Delta y}+\frac{(x_{i,j+1}-x_{i,j})}{2}m^+
\end{aligned}
\tag{36}
$$

$$
\begin{aligned}
l^- &= \begin{cases} a_1 & \text{if } |a_1| \leqslant |a_2| \\ a_2 & \text{if } |a_1| > |a_2| \end{cases} \\
l^+ &= \begin{cases} a_2 & \text{if } |a_2| \leqslant |a_3| \\ a_3 & \text{if } |a_2| > |a_3| \end{cases}
\end{aligned}
\tag{37}
$$

The $+$ superscript denotes the positive part and the $-$ superscript denotes the negative part, m^- and m^+ are like l^- and l^+ but for b_1, b_2 and b_3, also a_1, a_2 and a_3 are the central difference approximations of $\partial^2\phi/\partial x^2$ on $(x_{i-2,j}, x_{i-1,j}, x_{i,j})$, $(x_{i-1,j}, x_{i,j}, x_{i+1,j})$, $(x_{i,j}, x_{i+1,j}, x_{i+2,j})$. Finally, b_1, b_2 and b_3 are the central difference approximation of $\partial^2\phi/\partial y^2$ on $(y_{i,j-2}, y_{i,j-1}, y_{i,j})$, $(y_{i,j-1}, y_{i,j}, y_{i,j+1})$, $(y_{i,j}, y_{i,j+1}, y_{i,j+2})$.

Various time schemes were tested with the ENO method and five stage Runge–Kutta was found to be stable and converge most quickly. A five stage Runge–Kutta time integration is used to advance the solution in time from k to $k+1$ as follows [33]:

$$
\begin{aligned}
\phi_P^{(1)} &= \phi_P^{(k)} - \alpha_1 \frac{\Delta t}{\Delta V} R(\phi_P^{(k)}) \\
\phi_P^{(2)} &= \phi_P^{(0)} - \alpha_2 \frac{\Delta t}{\Delta V} R(\phi_P^{(1)}) \\
\phi_P^{(3)} &= \phi_P^{(0)} - \alpha_3 \frac{\Delta t}{\Delta V} R(\phi_P^{(2)}) \\
\phi_P^{(4)} &= \phi_P^{(0)} - \alpha_4 \frac{\Delta t}{\Delta V} R(\phi_P^{(3)}) \\
\phi_P^{(k+1)} &= \phi_P^{(0)} - \alpha_5 \frac{\Delta t}{\Delta V} R(\phi_P^{(4)})
\end{aligned}
\tag{38}
$$

where $R = u_{i+1,j}\phi_e - u_{i,j}\phi_w + v_{i,j+1}\phi_n - v_{i,j}\phi_s$ and the coefficients are

$$
\alpha_1 = \tfrac{1}{4}, \quad \alpha_2 = \tfrac{1}{6}, \quad \alpha_3 = \tfrac{3}{8}, \quad \alpha_4 = \tfrac{1}{2}, \quad \alpha_5 = 1 \tag{39}
$$

Dual time stepping with a five stage Runge–Kutta is used to solve the global mass correction equation to speed up the convergence of the mass correction. A five-stage Runge–Kutta time integration algorithm is used between each physical time step to iterate the numerical solution in

DOI: 10.1002/fld

an artificial time τ until convergence is reached [34]. Therefore, the converged solution from the artificial steady-state equations becomes the time accurate solution at current physical time t.

4. COMPUTATION RESULTS

4.1. Shear flow

In order to assess the capabilities of the interface-tracking part of the numerical scheme, tests are carried out for circular interfaces being advected through a prescribed velocity field. Only the level-set equations and not the Navier–Stokes equations are solved. The following velocity field on the unit square is used as the initial condition:

$$\begin{aligned} u &= \sin^2(\pi x)\sin(2\pi y) \\ v &= -\sin^2(\pi y)\sin(2\pi x) \end{aligned} \tag{40}$$

Numerous researchers, such as Rider and Kothe [35], Ubbink [36], Olsson and Kreiss [37], Greaves [38] and Yue *et al*. [8], used this flow field in the single vortex to assess the integrity and capability of an interface-tracking method in cases where significant interface stretching occurs. A circle with radius 0.15 centred at (0.5, 0.75) in a unit square is used as the initial condition. The interface is first advected forward up to $t=2.0$ s and then the velocities are reversed for the same length of time in order to return to the initial condition. A perfect advection scheme should be able to recover the initial circular interface.

Comparison of shear flow at different grid densities at $t=1.5$ s is shown in Figure 2(a)–(c). There is a filament in the tail at $t=1.5$ s on grid 50×50 and the tail is much longer for the fine grid 200×200. Figure 2(d) shows the mass error for different grid densities. The finer the grid, the smaller the mass error. For the coarse grid, the mass error increases rapidly after a short time of simulation. The mass is conserved very well on the fine grid. The simulations were performed in a PC with a Intel(R) Core TM 2 CPU 6600@2.4 GHz and 3.25 GB RAM running on Windows XP operative system. The calculation times per time step for grid 50×50, 100×100 and 200×200 were 0.19, 0.46 and 1.03 s, respectively.

Computations on a 200×200 grid were performed and the advected interface at $t=0$ s, $t=0.5$ s, $t=1.0$ s, $t=1.5$ s and $t=2.0$ s is shown in Figure 3. Third-order ENO schemes and global mass correction were used for both advection and redistancing. After the flow reverses at $t=2.0$ s (Figure 3(e)), the resulting circular interface at $t=4.0$ s is very close to the initial circular interface (Figure 3(f)).

To quantify the numerical errors, the mass error is defined as

$$\lambda_{\text{error}} = \left| \frac{M(0)-M(t)}{M(0)} \right| \times 100\% \tag{41}$$

where $M(t)=\int_{A_{\text{water}}} \rho H(\phi)\,\mathrm{d}A$ is the mass of the reference phase at the correct time t.

Figure 4 shows the time history of the mass error for the shear flow with the global mass correction and without global mass correction. Mass losses occur where the initial interface is most distorted without the global mass correction method. However, mass is conserved well up to machine accuracy despite the highly shearing flow field and large topological changes of the interface with the global mass correction method.

DOI: 10.1002/fld

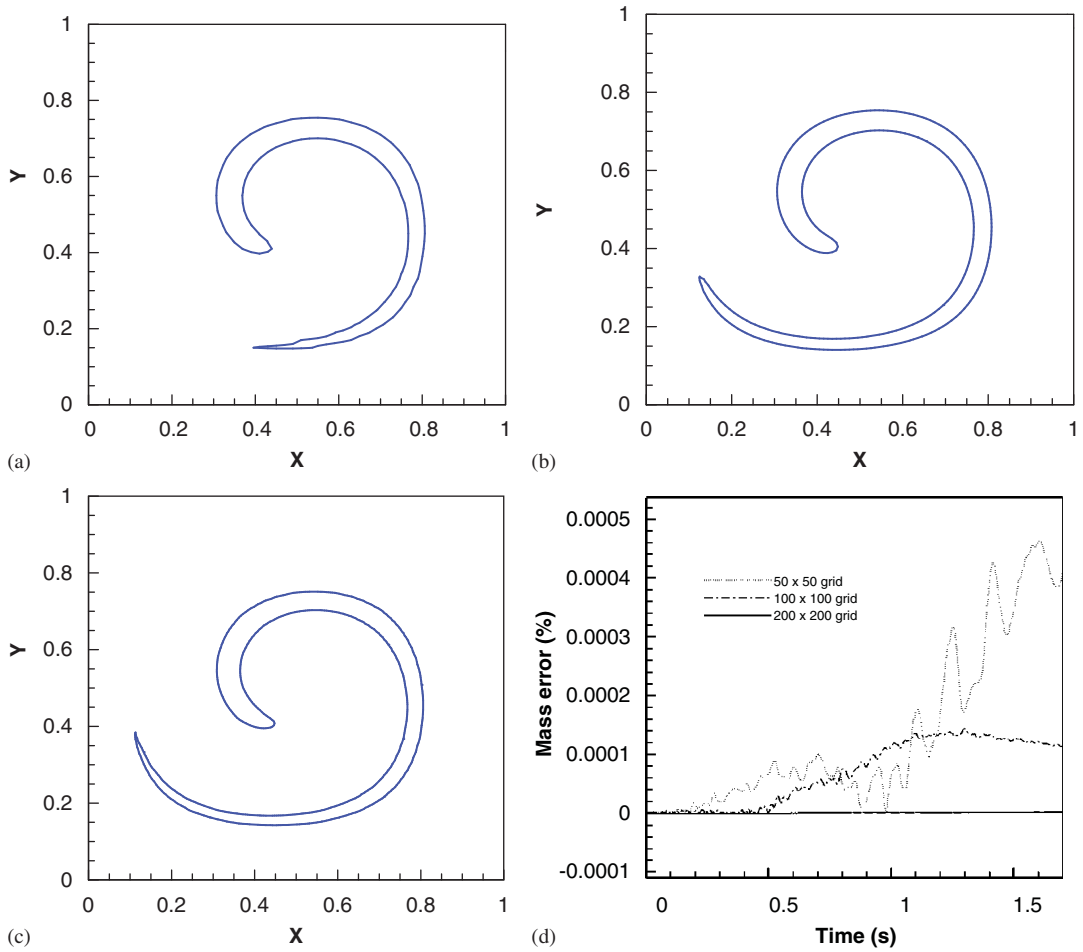


Figure 2. Comparison of shear flow at different grid densities at time $t=1.5\,\mathrm{s}$ with global mass correction: (a) 50×50 grid; (b) 100×100 grid; (c) 200×200 grid; and (d) comparison of mass error for different grid densities.

Comparisons of long time simulations for shear flow between with and without the global mass correction at $t=3.0\,\mathrm{s}$, $4.0\,\mathrm{s}$ and $5\,\mathrm{s}$ are shown in Figure 5. With the dual time stepping global mass correction scheme, the mass can be conserved for long time simulations compared without the global mass correction. The fluid is torn into a filament by the shearing flow and becomes thinner at $t=5\,\mathrm{s}$ with global mass correction while this phenomenon occurs at the earlier time at $t=4\,\mathrm{s}$ without global mass correction. At $t=5\,\mathrm{s}$ without global mass correction, the interface will disappear soon due to the poor mass conservation.

4.2. Dam break

Simulation of the dam break has been investigated numerically by various researchers, such as Ubbink [36], Jeong and Yang [39, 40], Greaves [7], and experimental data exist from Martin and

DOI: 10.1002/fld

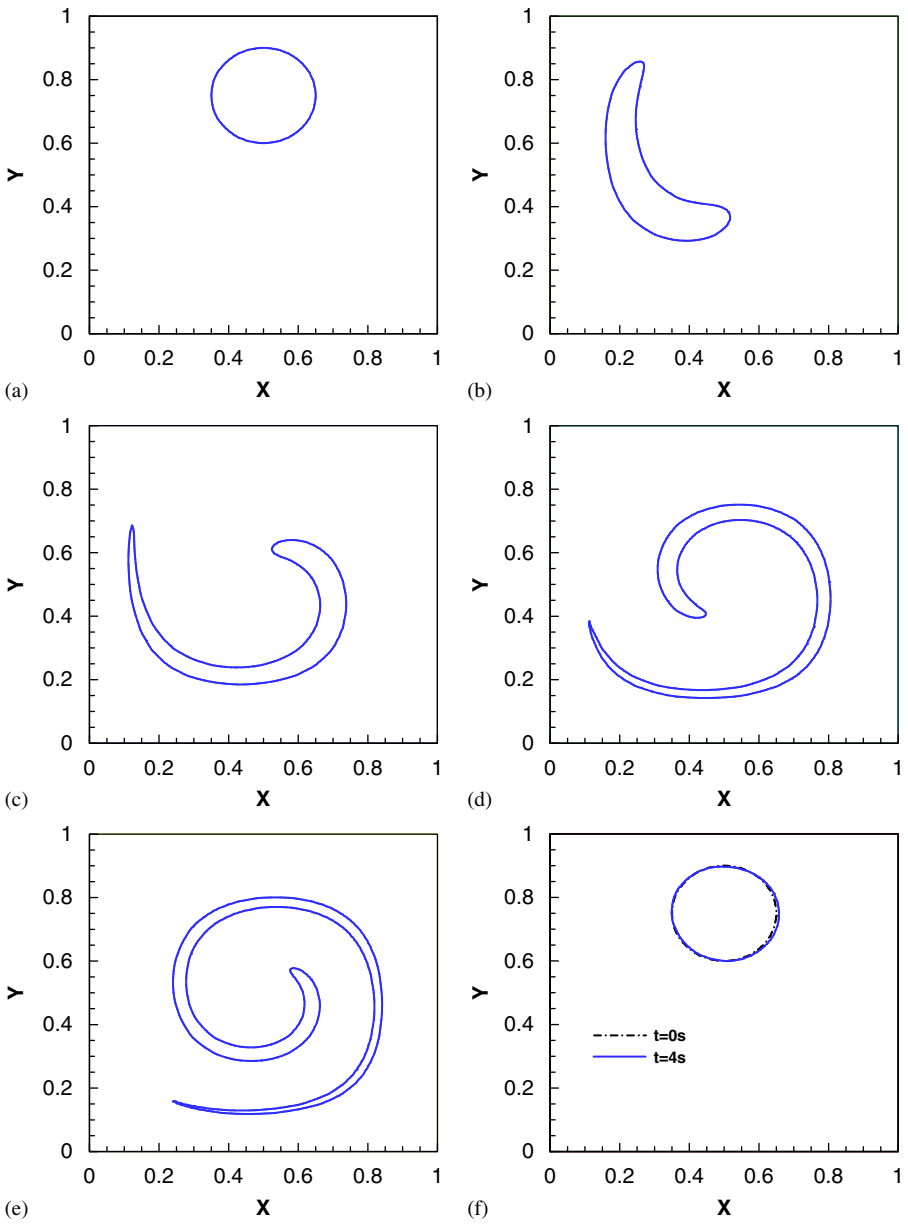


Figure 3. Shear flow at: (a) $t=0$ s; (b) $t=0.5$ s; (c) $t=1.0$ s; (d) $t=1.5$ s; (e) $t=2.0$; and (f) $t=4.0$ s.

Moyce [41]. In the present study, the global mass correction level-set method is combined with the Navier–Stokes flow solver and the simulation is carried out for different grid sizes. The initial conditions for the dam break are shown in Figure 6(a). The still water column is initially in hydrostatic balance. The unit square domain size is $4a \times 4a$ where $a=0.05715$ m. A column of

DOI: 10.1002/fld

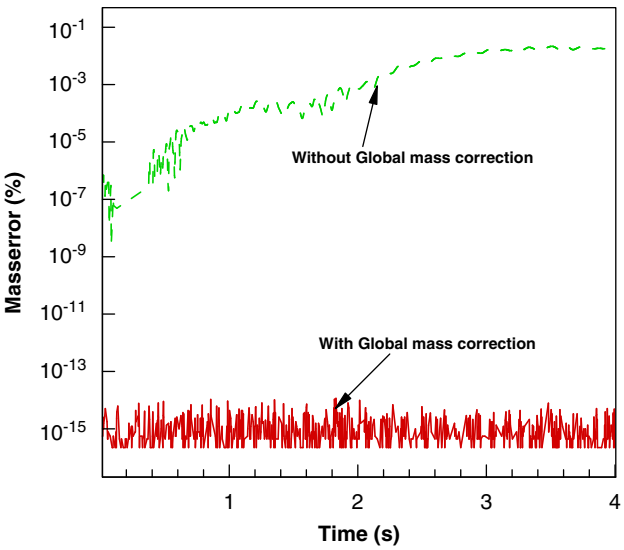


Figure 4. Time history of the mass error for the shear flow.

water a wide and $2a$ high is held in place at $t=0$ with a vertical wall, which is instantaneously removed. When the dam fails, the still water column behind the wall starts to collapse. The water has a dynamic viscosity of $1\times10^{-3}\,\mathrm{kg/m/s}$ and the air $1.8\times10^{-5}\,\mathrm{kg/m/s}$, the density of water is $1000\,\mathrm{kg/m^3}$ and air $1\,\mathrm{kg/m^3}$ and the acceleration due to gravity is taken to be $g=9.81\,\mathrm{m/s^2}$. Initially the velocity everywhere is zero; no-slip boundary conditions are applied on all walls. The thickness of the interface is fixed to be four grid sizes for the uniform grid case.

Snapshots of the water air interface and velocity vectors calculated on a uniform 128×128 grid with time step 0.0004 s are presented at non-dimensional time, $T=0$, 1.71, 3.99, 4.91, 6.45 and 8.15, where $T=t\sqrt{g/a}$, in Figure 6 alongside video images from Koshizuka *et al.* [42]. At $T=1.71$, the water is greatly accelerated and moves rapidly along the bottom wall. Later the water column impacts the right-hand wall and climbs up it. Then the water against the right-hand wall starts to fall back under the influence of gravity. When the surge front falls down from the right wall and plunges into the bottom water, air is entrapped by the water, forming an air bubble. The tongue of the moving water impinges on the left wall and traps an air bubble as well. The velocity vector field shows that a large vortex is formed in the air region in the vicinity of the water surface. The snapshots compare well with photographs of the experiment taken by Koshizuka *et al.* [42] demonstrating that the proposed numerical method can capture the complex phenomena involved in air bubble entrainment, wave breaking and splash up.

Figure 7(a) shows the comparison of the non-dimensional surge front positions $\bar{s}=s/a$ versus the non-dimensional time $T=t\sqrt{2g/a}$ between the present simulations on uniform grid sizes 32×32, 64×64 and 128×128 and Martin and Moyce's [41] experiment. For all these simulations, the predicted surge front positions are in an excellent agreement with the experimental data. Figure 7(b) shows a similar comparison of the non-dimensional height of the water $\bar{h}=h/2a$ versus the non-dimensional time $T=t\sqrt{g/a}$ at the left-hand wall between the simulations and experiment. Good agreement can be seen.

DOI: 10.1002/fld

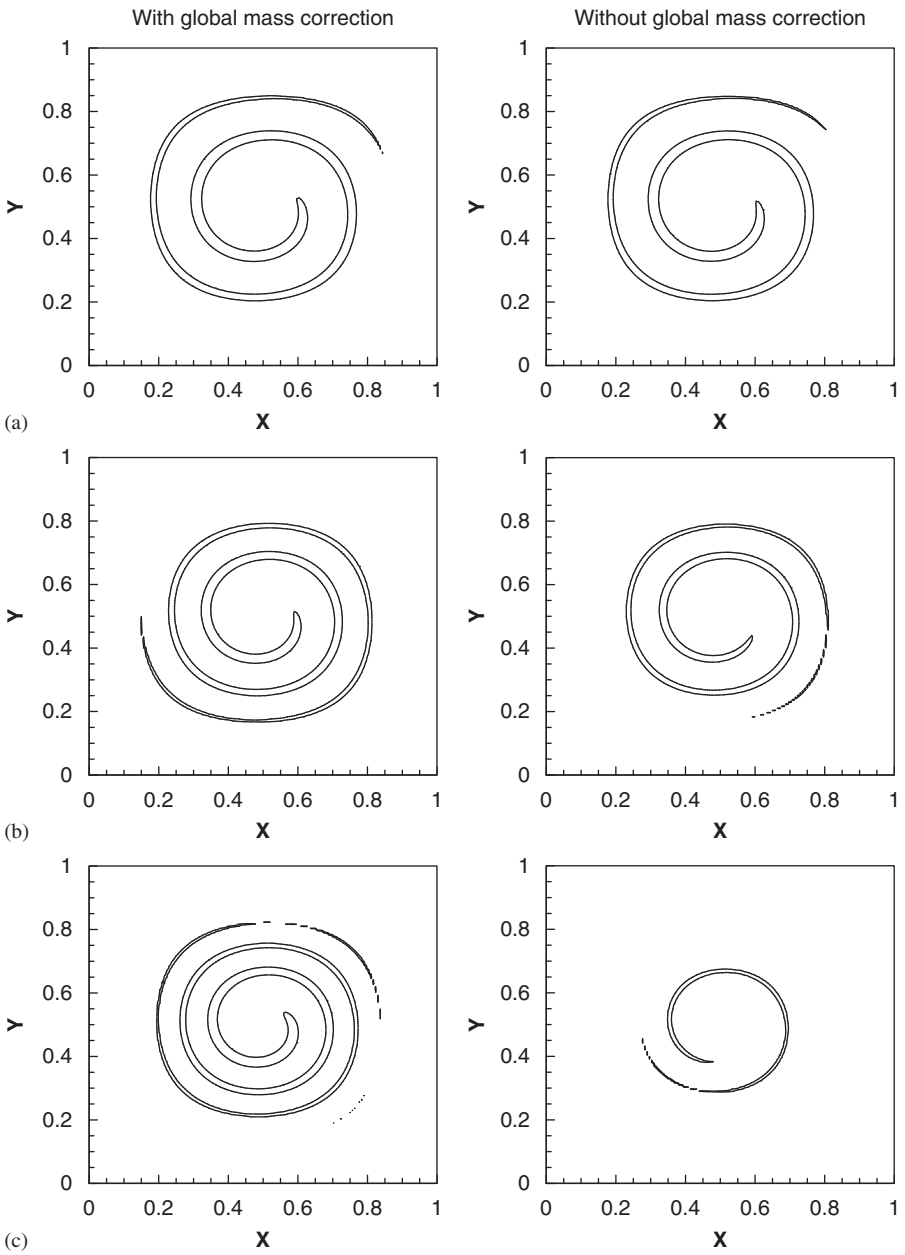
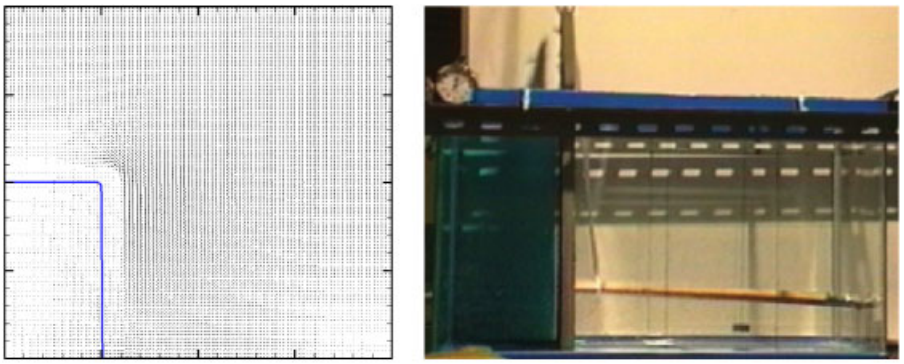
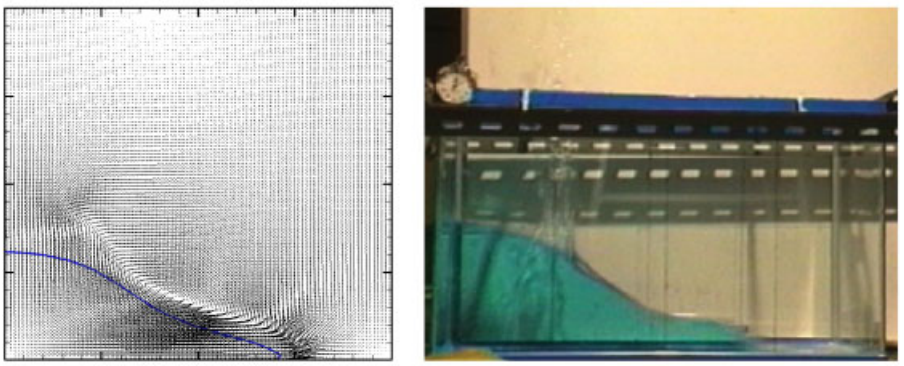


Figure 5. Comparison of shear flow for long time simulations between with and without the global mass correction at different times: (a) $t=3$ s; (b) $t=4$ s; and (c) $t=5$ s. Left column with the global mass correction and right column without.

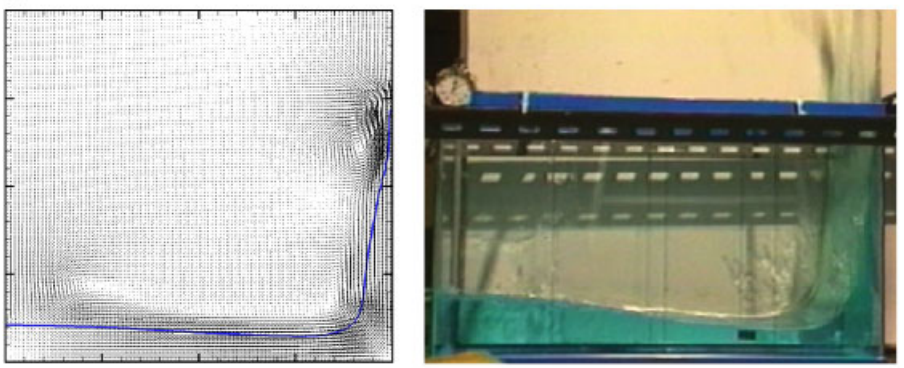
DOI: 10.1002/fld



(a)



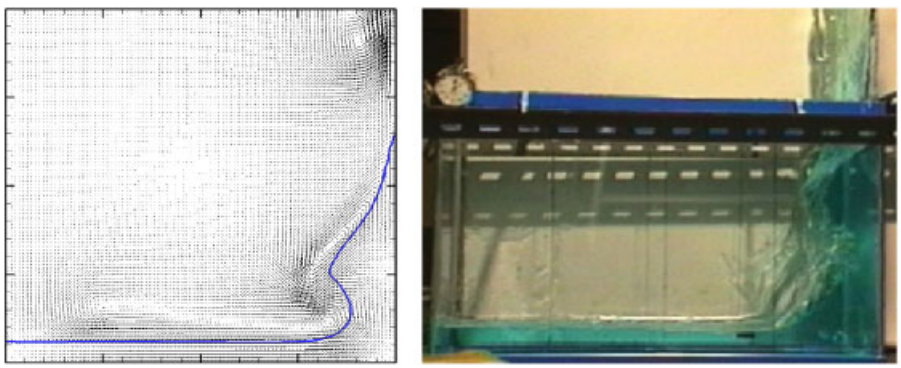
(b)



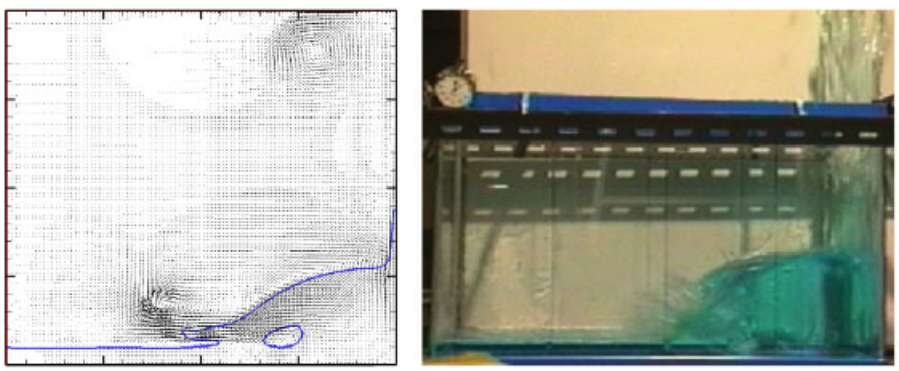
(c)

Figure 6. Dam break interface evolution and velocity vectors for grid 128×128, LHS and video images taken by Koshizuka *et al.* [42]: (a) $T = 0$; (b) $T = 1.71$; (c) $T = 3.99$; (d) $T = 4.91$; (e) $T = 6.45$; and (f) $T = 8.15$.

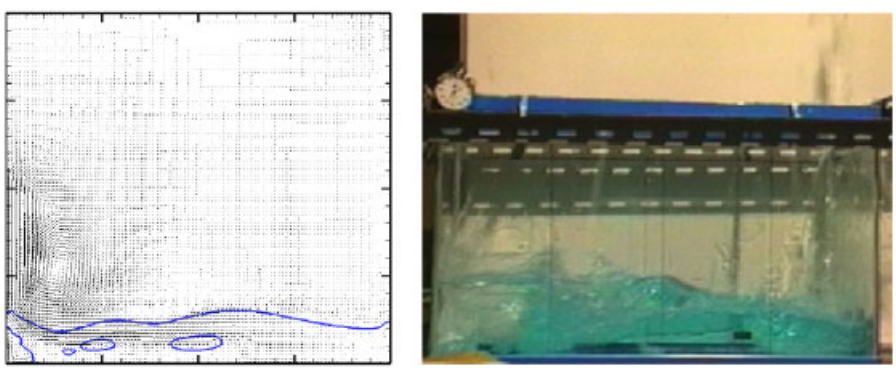
DOI: 10.1002/fld



(d)



(e)



(f)

Figure 6. *Continued*.

Figure 8 shows the mass error versus the time step for the dam break case. The global mass correction reduces the mass error to machine zero, whereas without global mass correction, the mass error is approximately 0.1%. This demonstrates that the present numerical method conserves mass well than without the global mass correction.

DOI: 10.1002/fld

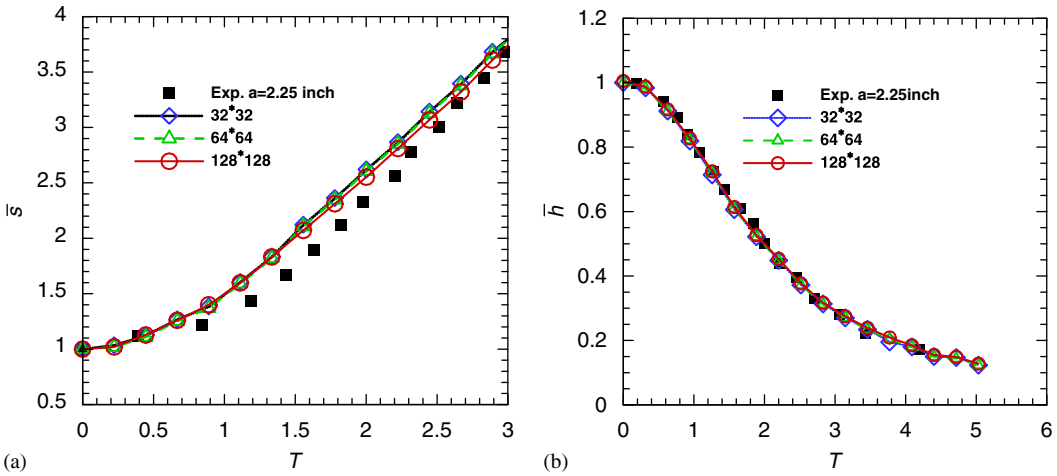


Figure 7. (a) Surge front position and (b) height of water column.

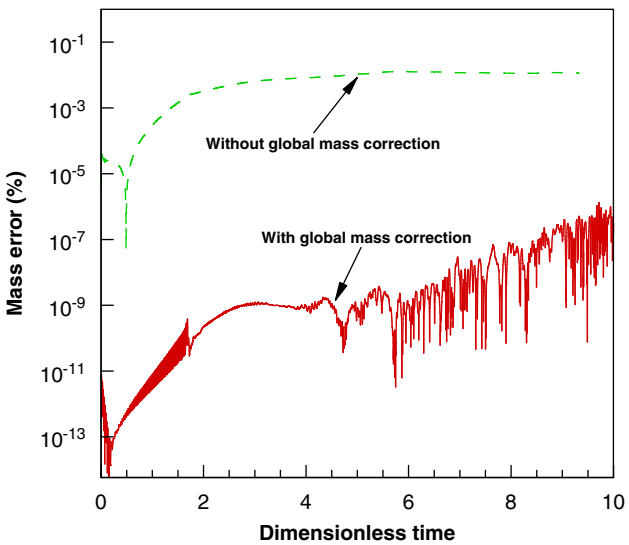


Figure 8. Mass errors of dam break versus time.

4.3. Solitary wave

A classical case used in the validation of the mathematical modelling of free-surface flows or two fluid systems is the propagation of a solitary wave. The level-set method with global mass correction is used to predict the run up on a vertical wall of a travelling solitary wave as shown in Figure 9. h is the depth of still water and the channel size is $20h \times 2h$. S_0, Sc and Srunup represent the amplitude of the solitary wave released from the left vertical wall, the amplitude of the solitary wave at the horizontal centre of the computational domain and wave run up at the right-hand wall

DOI: 10.1002/fld

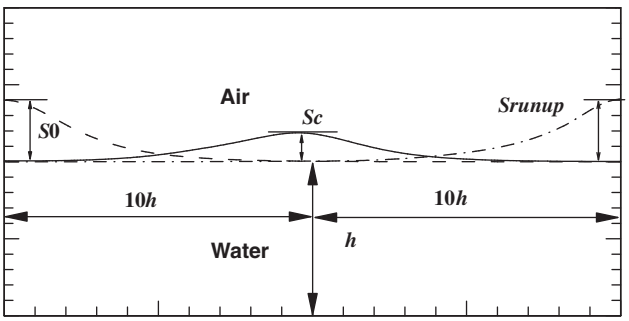


Figure 9. Computational domain of travelling solitary wave simulations.

boundary, respectively. The theoretical wave speed $c_w = \sqrt{gh} = 1\,\mathrm{m/s}$. Non-dimensional time T is defined from $T = t/(h/c_w)$. A grid of 200×150 is used, with uniform spacing in the x and y directions. The thickness of the interface is fixed at four grid sizes. The water and air have the same values of dynamic viscosity and density as used above for the dam break simulation. To generate a solitary wave, Laitone's analytical approximation [43] can be used. An initially still water surface with zero initial velocity and a Boussinesq profile from the left vertical wall, which is in hydrostatic balance, is released:

$$S(x,0) = S_0 \Big/ \cosh^2\left(\frac{\sqrt{3S_0}}{2}x\right) \tag{42}$$

Figure 10(a) shows the travelling train of the solitary wave and the climb at the right vertical wall for the case $S_0/h = 0.4$. The wave speed measured from Figure 10(a) is 1.03, which is very close to the theoretical value. Figure 10(b) shows a typical velocity field for $S_0/h = 0.4$, where, in order to make the figure clear, vectors are plotted at every third node. There is a vortex formed at the wave top.

Figure 11 shows the quantitative comparison for the wave run up at the right-hand wall boundary versus incident wave amplitude. Eight cases with different initial wave amplitudes are simulated and the run up at the right-hand wall boundary recorded. The computational results are compared with the experimental data by Chan and Street [44] and numerical results by Lin *et al.* [45], Yang and Stern [46]. The agreement between computation and experiment is very good for $\mathrm{Sc}/h < 0.3$. Above 0.3, there are some differences between the experimental data and numerical simulations and the experiment exhibits some scatter. However, these results demonstrate that the present numerical method can accurately predict the viscous damping characteristics.

4.4. Solitary wave over a submerged rectangular object

In order to demonstrate the model's ability to capture free-surface evolution in a more complex situation, the simulation of a solitary wave passing over a submerged rectangular object is carried out. A staggered non-uniform or uniform Cartesian grid is used with the block off method [31] to handle the interior bodies, solid boundaries and highly irregular flow domains. The basic idea behind this technique is that the submerged object can be modelled as a special case of two phase flow with infinite viscosity within the object and zero velocity at the nominal boundary.

DOI: 10.1002/fld

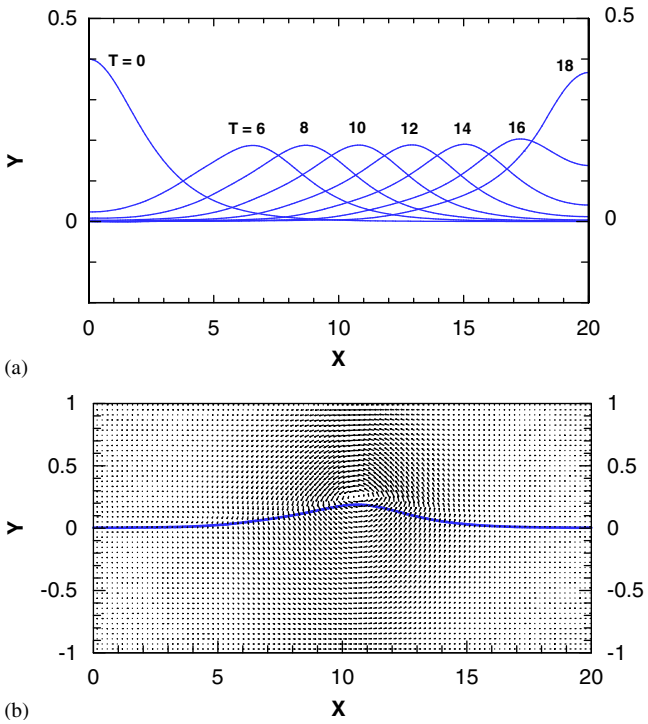


Figure 10. (a) Travelling trains of a solitary wave and (b) a typical velocity field of a solitary wave.

Figure 12 shows the computational domain for the simulation of a solitary wave over a submerged rectangular object. The rectangle is 0.4 m in width and 0.08 m in height and its leading edge is located 2.6 m from the inflow boundary. The computation domain is $-3<x<3.5$ m and $-0.16<y<0.26$ m. A non-uniform grid system is used in the x direction; with minimum grid size 0.001 m being employed in the vicinity of the object. A uniform grid with grid size 0.002 m is used in the y direction. Behind the object from 0 to 0.1 m, six locations from 1 to 6 have been chosen to record the velocity distribution from the bottom to the free surface. The velocity and the free-surface elevation at the inflow boundary are given according to the Boussinesq equation [47] for solitary waves. The incoming solitary wave height is a $0.18h$, where $h=0.16$ is the still-water depth.

Two wave gauges are located in front of and behind the rectangular obstacle. Free-surface elevations are recorded in the present numerical simulation and then compared with the experimental data and numerical simulation in [48] in Figure 13. It demonstrates that the present numerical model can predict the free-surface profiles well.

More detailed wave motion and vortex generation can be seen in snapshots of the water air interface contours and velocity vectors. Water air interface contours calculated on a non-uniform grid with time step 0.0005s are presented at times, $t=1.75$, 2.7, 3.55 and 4.15 s shown in Figure 14. Velocity vectors at time steps $t=2.7$, 2.9, 3.1 and 3.3 s are shown in Figure 15. Before the solitary wave crest reaches the submerged object, the free surface maintains the solitary wave profile very well as shown in Figure 14(a) due to the unidirectional flow above the submerged object. As it

DOI: 10.1002/fld

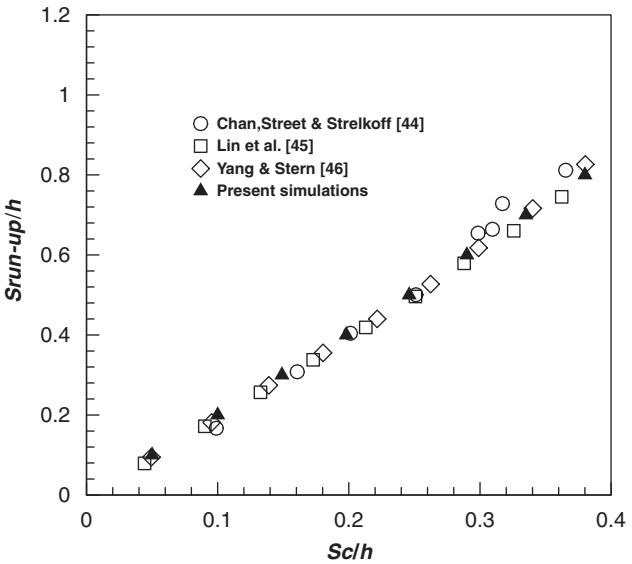


Figure 11. Wave run-up versus incident wave amplitude.

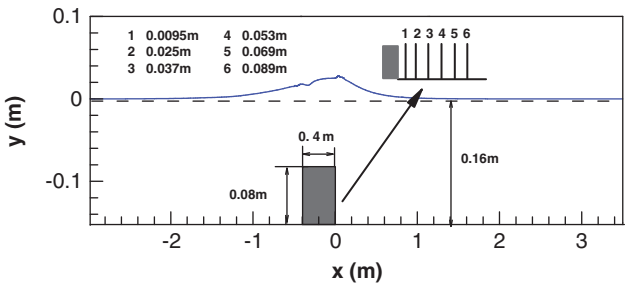


Figure 12. Computation domain for solitary wave over a submerged object. Behind the object from 0–0.1 m, six locations from 1 to 6 have been chosen to record the velocity distribution from the bottom to the free surface.

passes the submerged object, the wave becomes steepened with a larger wave height and short wavelength due to the shoaling effect as shown in Figure 14(b) and a vortex is generated at the weatherside of the rectangular object as shown in Figure 15(a). Owing to the wave crest being directly above the vortex, the vortex is confined by the object and it is stretched in the downstream direction as shown in Figure 15(b). The vortex at the weatherside of the crest becomes larger during this wave crest passing the rectangular object as shown in Figure 15(c). Also flow separation occurs and a vortex starts to form behind the submerged object. From Figure 15(c) to (d) the eye of the vortex drifts slowly downstream. After the solitary wave has passed the submerged object, the wave reduces amplitude with a trailing tail and a secondary wave is generated and propagated to the leading edge of the object as seen in Figure 14(c)–(d). Overall the numerical model describes the detailed evolution of the wave deformation and vortex generation well.

DOI: 10.1002/fld

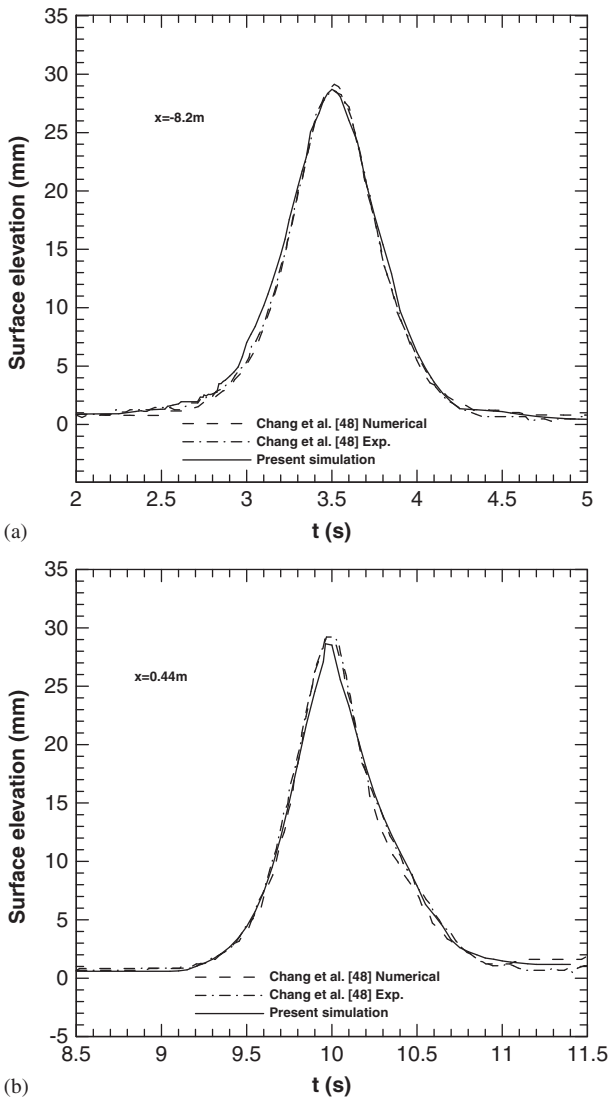


Figure 13. Comparison of free-surface profiles before and after the object.

The velocities in the leeside of the submerged object are investigated in detail to show that the model is able to predict the flow field near the object accurately. Figures 16 and 17 show the velocity components in the horizontal and vertical directions, respectively, at six locations behind the object and the corresponding free-surface position at time 3.09 s. It is observed that the present numerical results are in good agreement with the experiment and numerical results by Chang *et al.* [48]. There is a slight difference in horizontal velocity u on the bottom wall at the location 3–6 as shown in Figure 16. The differences observed at locations 1 and 2 coincide with a large velocity gradient, as shown in Figures 16 and 17, and a finer grid may be necessary to resolve the

DOI: 10.1002/fld

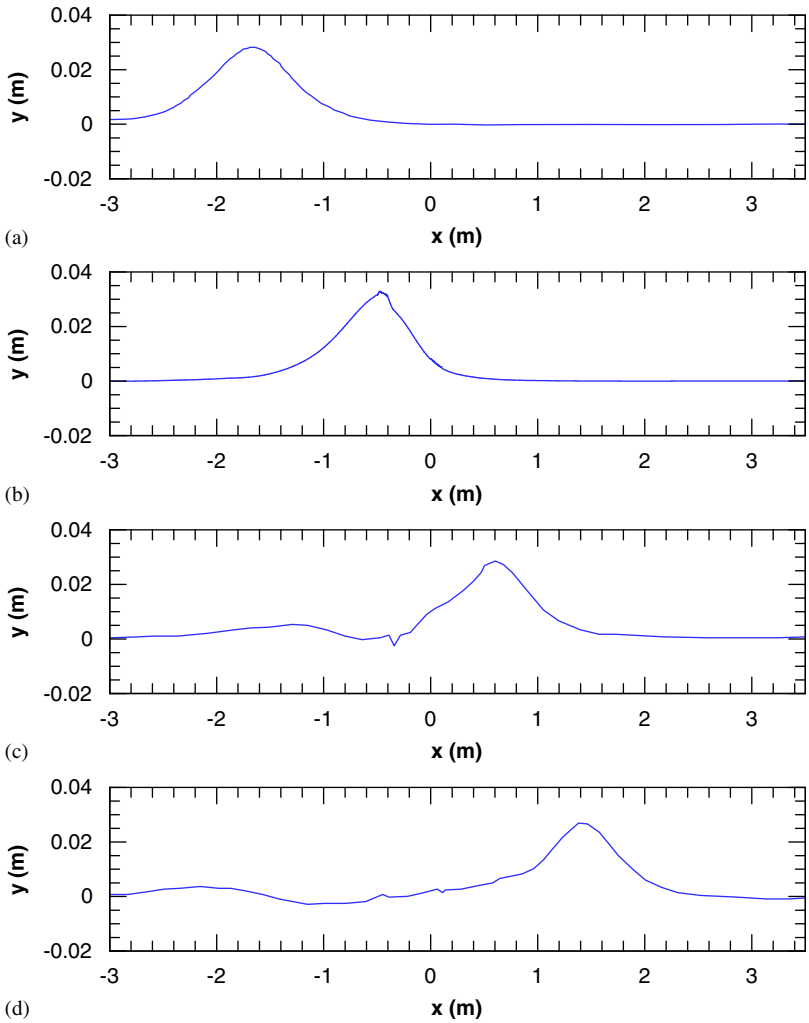


Figure 14. Interface evolution of solitary wave over a submerged rectangular object for a non-uniform grid: (a) $t=1.75$ s; (b) $t=2.7$ s; (c) $t=3.55$ s; and (d) $t=4.15$ s.

flow field further. In addition, Chang's results [48] appear to agree better in this region and they used RANS-VOF with k–ε turbulent model while our model does not include turbulence.

5. CONCLUSION AND FUTURE WORK

This paper presents a new method for coupling the incompressible Navier–Stokes equations with the level-set method. The finite volume method is used to discretize Navier–Stokes equations with the two-step projection method on a staggered Cartesian grid. The free-surface flow problem is

DOI: 10.1002/fld

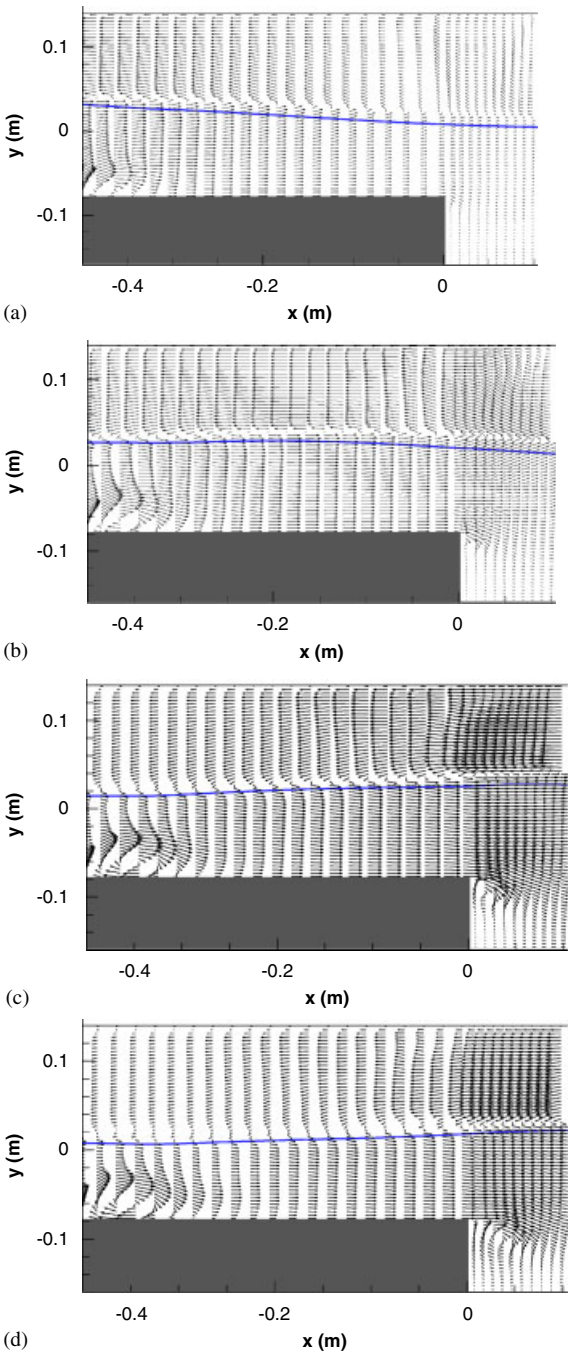


Figure 15. Vector of solitary wave over a submerged rectangular object for a non-uniform grid: (a) $t = 2.7$ s; (b) $t = 2.9$ s; (c) $t = 3.1$ s; and (d) $t = 3.3$ s.

DOI: 10.1002/fld

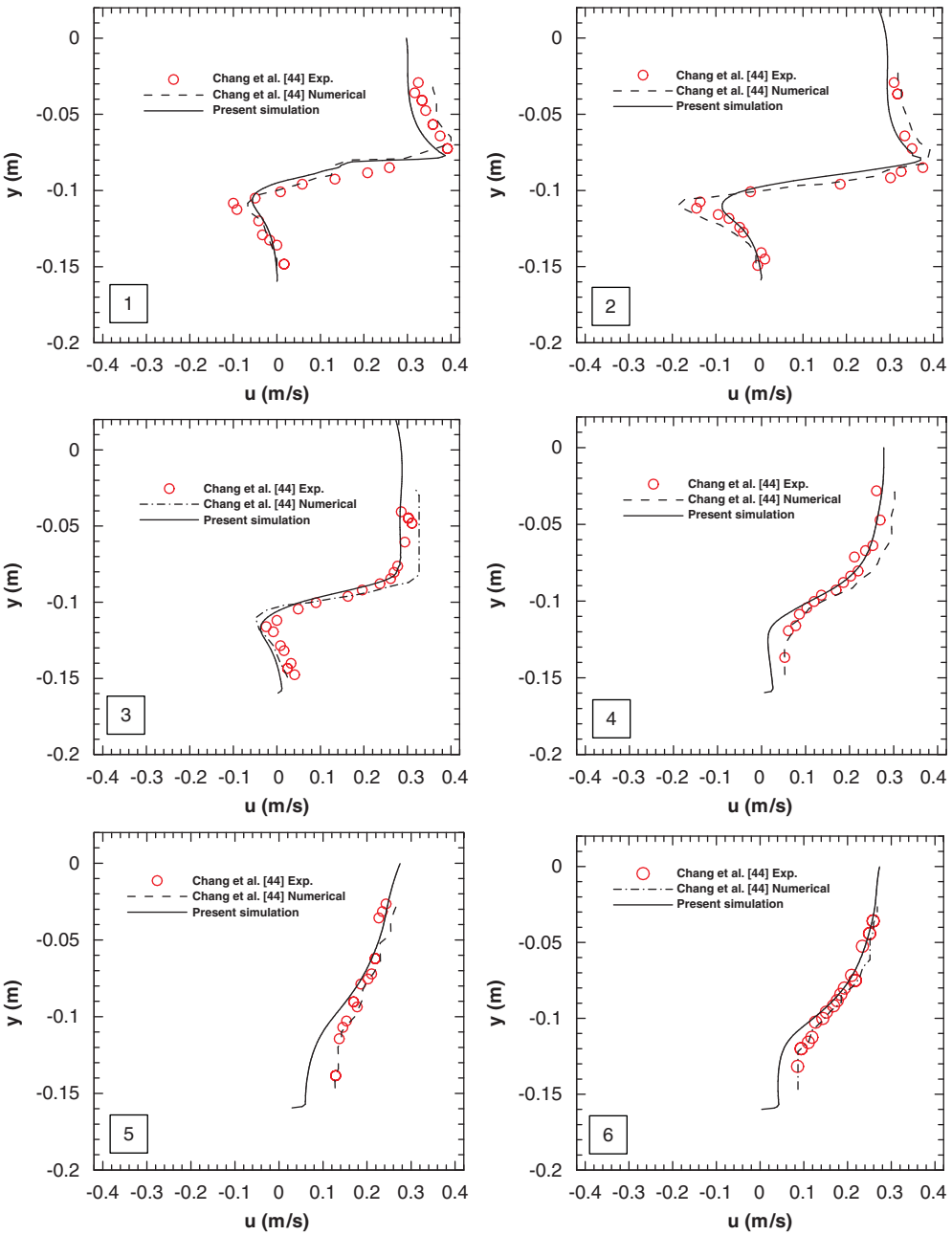


Figure 16. The comparison of horizontal velocity u at six locations in the wake behind the object velocity fields between the experimental measurements, previous numerical results and presented numerical results at $t = 3.09$ s.

DOI: 10.1002/fld

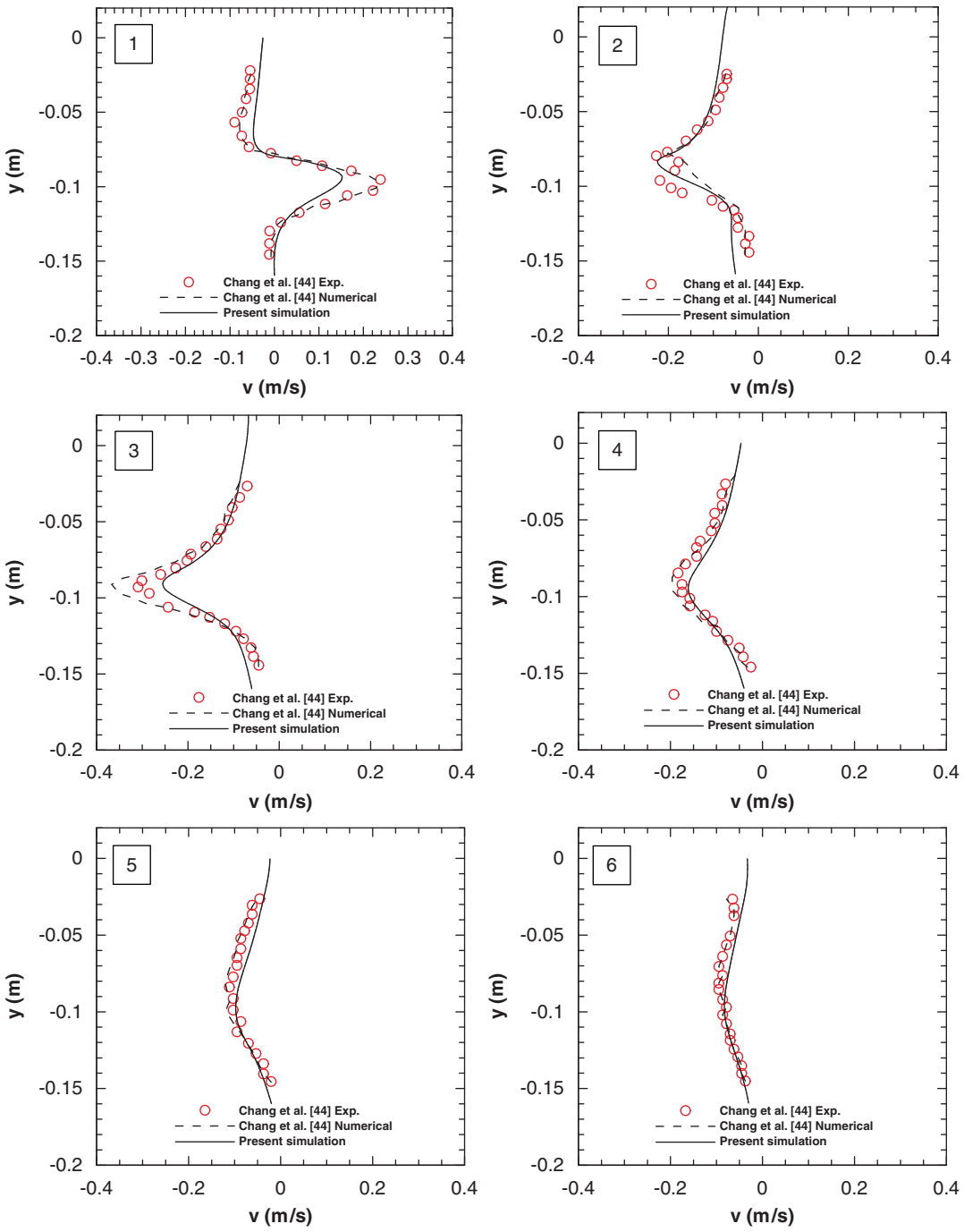


Figure 17. The comparison of vertical velocity v at six locations in the wake behind the object between the experimental measurements, previous numerical results and presented numerical results at $t = 3.09$ s.

DOI: 10.1002/fld

solved on a fixed grid in which the free surface is captured by the zero level set. Mass conservation is improved significantly by applying a global mass correction scheme, in a novel combination with third-order ENOs and a five stage Runge–Kutta method, to accomplish advection and re-distancing of the level-set function. The coupled system is applied to benchmark tests, such as simulation of shear flow, dam break and a 2D solitary wave, and also used to simulate a solitary wave over a submerged object. The computational results are in excellent agreement with theoretical predictions, experimental data and previous numerical simulations using a RANS-VOF method. It demonstrates that the mass error can be reduced to machine zero with the new modified global mass correction and the interface can be maintained very well for long time simulations with the global mass correction despite the highly distorted flow fields. The complex interface changes such as air entrapment, wave deformation and evolution, large vortices in the solitary wave and vortices generation in the leeside and weatherside of the object are revealed well and in detailed manner.

By conducting numerical experiments, the method developed in the study is shown to be a promising tool for simulating free-surface flow. Furthermore, the current model is a two phase model and can treat air entrapment well near the free surface. It conserves mass and may be readily extended to three dimensions since the initial Eulerian grid remains the same throughout the simulation. Also it can be extended to model the compressibility for the air entrapped in sufficiently small cavities required by practical applications such as the importance of dispersed bubbles compressibility in the case of waves impacting against breakwaters. The easiest way to account for the compressibility in this two phase model is to allow the density of the air to vary in time. This approach can give reliable results, but faces some computational challenges connected with the coupling of the two different fluids [49].

ACKNOWLEDGEMENTS

The authors thank Dr Xin Lu for helpful discussions on numerical schemes. The authors also thank Prof. Dominic Reeve, Dr Alison Hunt-Raby, Dr David Graham and Dr Phil James for useful comments. The authors also gratefully acknowledge the support of the South West of England Regional Development Agency in funding this project through PRIMaRE. The second author also would like to acknowledge the support of the Natural Environmental Research Council (Grant No. NE/E002129/1) during this project.

REFERENCES

1. Sussman M, Smith KM, Hussaini MY, Ohta M, Zhi RW. A sharp interface method for incompressible two phase flows. *Journal of Computational Physics* 2007; **221**:469–505.
2. Best J. The formation of toroidal bubbles upon the collapse of transient cavities. *Journal of Fluid Mechanics* 1993; **251**:79–107.
3. Zhang YL, Yeo K, Khoo BC, Wang C. 3rd jet impact and toroidal bubbles. *Journal of Computational Physics* 2001; **166**:336–360.
4. Juric D, Tryggvason G. Computations of boiling flows. *International Journal of Multiphase Flow* 1998; **24**(3): 387–410.
5. Glimm J, Grove JW, Li XL, Shyue KM, Zeng Y, Zhang Q. Three-dimensional front-tracking. *SIAM Journal on Scientific Computing* 1998; **19**(3):702–723.
6. Tryggvason G, Bunner B, Esmaeeli A, Juric D, Al-Rawahi N, Tauber W, Han J, Nas S, Jan YJ. A front-tracking method for the computations of multiphase flow. *Journal of Computational Physics* 2001; **169**(2):708–759.
7. Greaves DM. Simulation of viscous water column collapse using adapting hierarchical grids. *International Journal for Numerical Methods in Fluids* 2005; **50**:693–711.
8. Yue WS, Lin CL, Patel VC. Numerical simulation of unsteady multidimensional free surface motions by level set method. *International Journal for Numerical Methods in Engineering* 2003; **42**:853–884.

DOI: 10.1002/fld

9. Yap YF, Chai JC, Wong TN, Toh KC, Zhang HY. A global mass correction scheme for the level set method. *Numerical Heat Transfer*, *Part B* 2006; **50**:455–472.
10. Szymczak WG, Rogers J, Solomon J, Berger AE. A numerical algorithm for hydrodynamic free boundary problems. *Journal of Computational Physics* 1993; **106**:319–336.
11. Noh WF, Woodward P. *SLIC* (*Simple Line Interface Calculation*), Lecture Notes in Physics, vol. 59. Springer: New York, 1976; 330–340.
12. Youngs DL. Time-dependent multi-material flow with large fluid distortion. In *Numerical Methods for Fluid Dynamics*, Morton KW, Baines MJ (eds). Academic: New York, 1982; 273–285.
13. Osher S, Sethian JA. Fronts propagating with curvature-dependent speed algorithms based on Hamilton–Jacobi formulations. *Journal of Computational Physics* 1988; **79**:12–49.
14. Chang YC, Hou TY, Merriman B, Osher S. A level set formulation of Eulerian interface capturing methods for incompressible fluid flows. *Journal of Computational Physics* 1996; **124**:449–464.
15. Monaghan JJ. Simulating free surface flows with SPH. *Journal of Computational Physics* 1994; **65**:179–214.
16. Hirt CW, Nichols BD. Volume of fluid (VOF) methods for the dynamics of free boundaries. *Journal of Computational Physics* 1981; **39**:201–225.
17. Renardy Y, Renardy M. PROST—a parabolic reconstruction of surface tension for the volume-of-fluid method. *Journal of Computational Physics* 2002; **183**:400–421.
18. Osher S, Fedkiw RP. Level-set methods: an overview and some recent results. *Journal of Computational Physics* 2001; **169**:463–502.
19. Sethian JA. *Level Set Methods and Fast Marching Methods*: *Evolving Interfaces in Computational Geometry*, *Fluid Mechanics*, *Computer Vision*, *and Materials Science*. Cambridge University Press: Cambridge, U.K., 1999.
20. Sussman M, Smereka P, Osher S. A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational Physics* 1994; **114**:146–159.
21. Sussman M, Fatemi E. An efficient, interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow. *SIAM Journal on Scientific Computing* 1999; **20**(4):1165–1191.
22. Enright D, Fedkiw R, Ferziger J, Mitchell I. A hybrid particle level set method for improved interface capturing. *Journal of Computational Physics* 2002; **183**(1):83–116.
23. Bourlioux A. A coupled level-set volume-of-fluid for tracking material interfaces. *Sixth International Symposium on Computational Fluid Dynamics*, Lake Tahoe, NV, 4–8 September 1995.
24. Sussman M, Puckett EG. A coupled level set and volume-of-fluid method for computing 3D and axisymmetric incompressible two-phase flows. *Journal of Computational Physics* 2000; **162**(2):301–337.
25. Lakehal D, Meier M, Fulgosi M. Interface tracking towards the direct simulation of heat and mass transfer in multiphase flows. *International Journal of Heat and Fluid Flow* 2002; **23**:242–257.
26. Shu CW, Osher S. Efficient implementation of essentially non-oscillatory shock capturing schemes, II. *Journal of Computational Physics* 1989; **83**:32–78.
27. Ferziger JH, Peric M. *Computational Methods for Fluid Dynamics* (3rd edn). Springer: New York, 2002.
28. Unverdi OS, Tryggvason G. A front-tracking method for viscous, incompressible, multifluid flows. *Journal of Computational Physics* 1992; **100**:25–37.
29. Chorin AJ. Numerical solution of the Navier–Stokes equations. *Mathematics of Computation* 1968; **22**:745–762.
30. Liu Philip LF, Lin PZ. A numerical model for breaking waves: the volume of fluid method. *Journal of Fluid Mechanics* 1998; **359**:239–264.
31. Patankar SV. *Numerical Heat Transfer and Fluid Flow*. Hemisphere/Mcgraw-Hill: Washington, 1980.
32. Harten A, Engquist B, Osher S, Chakravarthy S. Uniformly high-order accurate essentially non-oscillatory schemes, III. *Journal of Computational Physics* 1987; **71**:231–303.
33. Lv X, Zhao Y. An efficient parallel/unstructured-multigrid preconditioned implicit method for simulating 3D unsteady compressible flows with moving objects. *Journal of Computational Physics* 2006; **215**(2):661–690.
34. Jameson A. Time dependent calculations using multigrid, with applications to unsteady flows past airfoils and wings. *AIAA Paper* 1991; 1596.
35. Rider J, Kothe DB. Stretching and tearing interface tracking methods. *Twelfth AIAA Computational Fluid Dynamics Conference*, San Diego, 20 June 1995. Paper Number AIAA951717.
36. Ubbink O. Numerical prediction of two fluid systems with sharp interfaces. *Ph.D. Thesis*, Imperial College of Science, Technology and Medicine, London, 1997.
37. Olsson E, Kreiss G. A conservative level set method for two phase flow. *Journal of Computational Physics* 2005; **210**:225–246.
38. Greaves DM. A quadtree adaptive method for simulating fluid flows with moving interfaces. *Journal of Computational Physics* 2004; **194**:35–56.

DOI: 10.1002/fld

39. Jeong JH, Yang DY. Finite element analysis of transient fluid flow with free surface using VOF (volume of fluid) method and adaptive grid. *International Journal for Numerical Methods in Fluids* 1998; **26**:1127–1154.
40. Jeong JH, Yang DY. Three-dimensional finite element analysis of transient fluid flow with free surface using marker surface method and adaptive grid refinement. *International Journal for Numerical Methods in Fluids* 1999; **29**:657–684.
41. Martin JC, Moyce WJ. An experimental study of the collapse of liquid columns on a rigid horizontal plane. *Philosophical Transactions of the Royal Society of London* 1952; **A244**:312–324.
42. Koshizuka S, Tamako H, Oka Y. A particle method for incompressible viscous flow with fluid fragmentation. *Computational Fluid Dynamics Journal* 1995; **113**:134–147.
43. Ramaswamy B. Numerical simulation of unsteady viscous free surface flow. *Journal of Computational Physics* 1990; **90**:396–430.
44. Chan RK, Street RL. A computer study of finite amplitude water waves. *Journal of Computational Physics* 1970; **6**:68–94.
45. Lin CL, Lee H, Lee T, Weber L. A level set characteristic Galerkin finite element method for free surface flows. *International Journal for Numerical Methods in Fluids* 2005; **49**:521–547.
46. Yang J, Stern F. Large-Eddy simulation of breaking waves using embedded boundary/level set method. *AIAA* 2007; 1455.
47. Lee JJ, Skjelbreia JE, Raichlen F. Measurement of velocities in solitary waves. *Journal of Waterway*, *Port*, *Coastal and Ocean Division* (ASCE) 1982; **108**:200–218.
48. Chang KA, Hsu TJ, Philip Liu LFK. Vortex generation and evolution in water waves propagating over a submerged rectangular obstacle. Part I. Solitary waves. *Coastal Engineering* 2001; **44**:13–36.
49. Caiden R, Fedkiw R, Anderson C. A numerical method for two phase flow consisting of separate compressible and incompressible regions. *Journal of Computational Physics* 2001; **166**:1–27.

DOI: 10.1002/fld