ARTICLE

A Numerical Model for Simulating Two-Phase Flow with Adaptive Mesh Refinement

Yunxing Zhang, Shan Ma, Kangping Liao and Wenyang Duan[*]

Harbin Engineering University, Harbin, 150001, China
[*]Corresponding Author: Wenyang Duan. Email: duanwenyang_heu@hotmail.com

ABSTRACT

In this study, a numerical model for simulating two-phase flow is developed. The Cartesian grid with Adaptive Mesh Refinement (AMR) is adopted to reduce the computational cost. An explicit projection method is used for the time integration and the Finite Difference Method (FDM) is applied on a staggered grid for the discretization of spatial derivatives. The Volume of Fluid (VOF) method with Piecewise-Linear Interface Calculation (PLIC) is extended to the AMR grid to capture the gas-water interface accurately. A coarse-fine interface treatment method is developed to preserve the flux conservation at the interfaces. Several two-dimensional (2D) and three-dimensional (3D) benchmark cases are carried out for the validation of the model. 2D and 3D shear flow tests are conducted to validate the extension of the VOF method to the AMR grid. A 2D linear sloshing case is considered in which the model is proved to have 2nd-order accuracy in space. The efficiency of applying the AMR grid is discussed with a nonlinear sloshing problem. Finally, 2D solitary wave past stage and 2D/3D dam break are simulated to demonstrate that the model is able to simulate violent interface problems.

KEYWORDS

Two-phase flow; adaptive mesh refinement; VOF; coarse-fine interface treatment

1 Introduction

Two-phase flow is very important as it is common in many engineering fields, ranging from naval hydrodynamics, geophysics to ocean engineering. The experimental study is a widely used approach for analyzing physical problems. However, due to the existence of small-scale structures and violent velocity change, experimental measurements on violent two-phase flow problems could be extremely difficult or even impossible. With the improvement of hardware and numerical algorithms, numerical simulation could be a powerful alternative for predicting the behaviors of these problems. However, there still exist many difficulties in numerical simulating two-phase flow.

Among the difficulties, the computational cost could be one of the major challenges in Computational Fluid Dynamics (CFD). Traditionally, a fixed structured grid is usually adopted for simulating two-phase flow problems. One of the remarkable features of this grid is that mesh refinement needs to be applied in priority on the range of all the caring regions, which is inefficient or even impossible when the caring regions are unpredictable. Adaptive Mesh Refinement (AMR)

grid was proposed by Berger et al. [1] which allows refining certain regions of the zone during the simulation to reduce the computational effort required. Since then, the AMR grid was widely used to simulate various kinds of flows.

For most of the research with AMR, single-phase flow is considered. As the method was proposed for hyperbolic partial differential equations, a natural implementation of AMR is the solution of the Euler equation [2,3]. Moreover, it was combined with the immersed boundary method for fluid-structure interaction problems of solid bodies [4,5] or flexible bodies [6,7] to combine the advantages of the two methods.

Research that applying AMR in two-phase flow is also numerous, relevant research can be found in surface-tension-driven interfacial flows such as bubbles [8–10], bubble/jet interact with bodies [11], and wave breaking or wave-body interactions [12–15]. However, it is still hard work to develop such a model. The difficulties include the large-density ratio and complex interface topology in two-phase flow, as well as the treatment of coarse-fine interface in the AMR grid where conservation is hard to preserve.

The objective of this paper is to develop a 3D numerical model for simulating two-phase flow with AMR. The model is the foundation of future work for wave-body interactions in the ship and ocean engineering field. To develop such a model, the difficulties discussed above should be considered. The first one is the complex interface topology, or in other words, to capture the gas-water interface accurately. For this problem, widely used methods include the front tracking method [16], the level set method [17], and the VOF method [18]. Among the methods, the accuracy and robustness of the VOF method were widely demonstrated [19]. For simplicity, the VOF method is adopted in this model. However, other methods can also be applied in this model as alternatives.

Large density ratios usually exist in two-phase flow, such as in bubble flow in which the density ratio could reach 10^6. This large ratio can result in numerical instability which may come from gas velocities being mixed with liquid densities [10,20]. It was demonstrated that a mass-momentum consistent transport can alleviate the instability [10,20–23]. The influence of high-density ratio on inconsistent transport of mass and momentum is complex. A detailed comparison between consistent and inconsistent transport of mass and momentum for high-density-ratio flows can be found in [21,22]. In fact, both inconsistent and consistent schemes can be found in two-phase flow simulations. With inconsistent transport schemes, acceptable results were observed for wave-body interactions [15] and bubble evolution [9]. Besides, consistent schemes were successfully applied to simulate wave energy converters [24–26] and water entry [27] problems. Consider the complexity of the numerical schemes and the objective of the solver, inconsistent transport of mass and momentum is adopted at present. Therefore, to validate the feasibility of inconsistent transport schemes is also one of the objectives of the paper.

A 3D numerical model for simulating two-phase flow with AMR is developed in this paper. The VOF method is adopted to capture the gas-water interface accurately. A coarse-fine interface treatment method for two-phase flow is developed to preserve the flux conservation at the coarse-fine interfaces. An inconsistent transport scheme is adopted for the advection of mass and momentum. The work is based on Zhang et al. [15] in which a 2D model is developed. The paper is organized as follows: The governing equations, single grid solver, and the AMR grid are introduced in Section 2, with special attention paid to discretization schemes and the coarse-fine interface treatment methods. The validations of the numerical model are given in Section 3 with selected cases. The accuracy of VOF on the AMR grid and the entire model is demonstrated.

Further validation of the model is presented in Section 4 with more serious problems to show the ability of the model on simulating wave breaking problems. Finally, the conclusions are drawn in Section 5.

2 Numerical Methods

2.1 Governing Equations

The single fluid formulation for multiphase flow is adopted in this model [28]. The viscous incompressible two-phase flow is considered as a fluid with spatially and temporally varying density and viscosity. The fluid is governed by the continuity equation and the Navier-Stokes equations, which in the non-conservative form are:

$$\nabla \cdot \mathbf{u} = 0$$
$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot \mathbf{u}\mathbf{u} = -\frac{1}{\rho}\nabla p + \frac{1}{\rho}\nabla \cdot \left[\mu\left(\nabla \mathbf{u} + \nabla \mathbf{u}^T\right)\right] + \mathbf{f} \tag{1}$$

in which $\mathbf{u} = (u, v, w)$ is the velocity of the fluid, p is the pressure, ρ and μ are the density and the dynamic viscosity coefficient, $\mathbf{f}$ stands for the body forces such as the gravitational force. The Eq. (1) is applied to a fixed Eulerian space with $\mathbf{x} = (x, y, z)$ to be the Eulerian coordinates.

The VOF method is utilized to capture the free surface. In the method, a scalar function φ is defined to represent the water phase. The value of φ for a cell is between 0 and 1 which denotes the fraction volume of the water phase within the cell. The evolution equation of φ is:

$$\frac{\partial \varphi}{\partial t} + \nabla \cdot (\mathbf{u}\varphi) = 0 \tag{2}$$

Appropriate initial and boundary conditions for Eqs. (1) and (2) are necessary. For all the simulations, water is initially at rest, and velocities are set to zero at the initial stage. The initial values of φ are defined with the topology of the water phase. All the boundaries are defined as walls, so the Dirichlet condition is used for the velocities and the Neumann condition is adopted for the pressure.

2.2 Single Grid Solver

In this section, the numerical methods on a single grid are introduced. For simplicity, the discretization schemes of the spatial derivatives are given in the 2D form. There is no extra difficulty for their extension to 3D.

2.2.1 Time Integration

The incompressible version of the projection method proposed by Xiao [29] is utilized for the time integration. The Navier–Stokes equation in Eq. (1) is split into three steps: one advection step and two non-advection steps.

Advection step

$$\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} + \nabla \cdot \mathbf{u}^n \mathbf{u}^n = 0 \tag{3}$$

Non-advection Step I

$$\frac{\mathbf{u}^{**} - \mathbf{u}^*}{\Delta t} = \frac{1}{\rho^{n+1}}\nabla \cdot \left[\mu^{n+1}\left(\nabla \mathbf{u}^* + \nabla \mathbf{u}^{*T}\right)\right] + \mathbf{f}^{**} \tag{4}$$

Non-advection Step II

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^{**}}{\Delta t} = -\frac{1}{\rho^{n+1}} \nabla \cdot p^{n+1} \tag{5}$$

Here the corner mark ^{n}, $^{n+1}$, *, and ** indicate the values of corresponding quantities at the current time step, the next time step, and the two predicted values between the two steps. This procedure was demonstrated to yield the first-order accuracy in time integration [29].

By applying the continuity equation in Eq. (1) to the Eq. (5), the following Pressure Poisson Equation (PPE) is obtained:

$$\nabla \cdot \left(\frac{1}{\rho} \nabla p^{n+1} \right) = \frac{1}{\Delta t} \nabla \cdot \mathbf{u}^{**} \tag{6}$$

The pressure is calculated by solving the equation after non-advection Step I and followed by non-advection Step II.

To ensure the temporal stability of the explicit projection method in the model, the time step Δt is defined concerning the convective effects. The time step Δt is determined with the following Courant–Friedrichs–Lewy (CFL) condition with C no more than 0.5 adopted for all the simulations.

$$\Delta t \leq \min \left(\frac{C\Delta x}{\max \|u\|_\infty}, \frac{C\Delta y}{\max \|v\|_\infty} \right) \tag{7}$$

2.2.2 Discretization of the Spatial Derivatives

The Finite Difference Method (FDM) on the Cartesian grid with a staggered arrangement of variables is adopted. A 2D case is illustrated in Fig. 1. The velocities are located on the cell side and scalars such as pressure and density are located on the cell center.
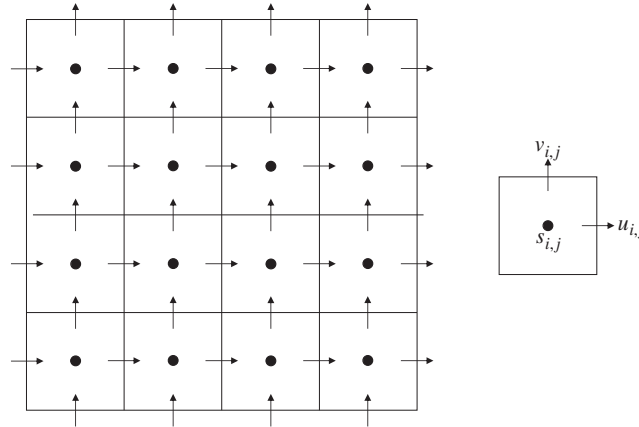

Figure 1: The staggered arrangement of the variables (s indicates for scalars)

For the discretization of the advection step (Eq. (3)), the upwind schemes are usually adopted to enhance the robustness of the code. A conservative discretization scheme is adopted as follow:

$$
\frac{\partial u_{i,j}}{\partial t} = -\frac{\left(u_{i,j}+u_{i+1,j}\right)\widetilde{u}_{i+1/2,j} - \left(u_{i,j}+u_{i-1,j}\right)\widetilde{u}_{i-1/2,j}}{2\Delta x}
$$
$$
-\frac{\left(v_{i,j}+v_{i+1,j}\right)\widetilde{u}_{i,j+1/2} - \left(v_{i,j-1}+v_{i+1,j-1}\right)\widetilde{u}_{i,j-1/2}}{2\Delta y}
\tag{8}
$$

$$
\frac{\partial v_{i,j}}{\partial t} = -\frac{\left(u_{i,j}+u_{i,j+1}\right)\widetilde{v}_{i+1/2,j} - \left(u_{i-1,j}+u_{i-1,j+1}\right)\widetilde{v}_{i-1/2,j}}{2\Delta x}
$$
$$
-\frac{\left(v_{i,j}+v_{i,j+1}\right)\widetilde{v}_{i,j+1/2} - \left(v_{i,j}+v_{i,j-1}\right)\widetilde{v}_{i,j-1/2}}{2\Delta y}
\tag{9}
$$

The values of $\widetilde{u}$ and $\widetilde{v}$ at the cell edge ($i\pm1/2,j$ or $i,j\pm1/2$) are calculated with the QUICK scheme [30] combined with Van Leer TVD limiter [31]. The interpolation stencil is determined based on the sign of the coefficient before it.

The central difference method is adopted for all the spatial derivatives in Eqs. (4)–(6). For non-advection step I (Eq. (4)), the discretization for the horizontal velocity u and the vertical velocity v is:

$$
\frac{\partial u_{i,j}}{\partial t} = fx_{i,j} + \frac{1}{\rho_{i+1/2,j}}\left\{\frac{2}{\Delta x}\left[\mu_{i+1,j}\frac{u_{i+1,j}-u_{i,j}}{\Delta x} - \mu_{i,j}\frac{u_{i,j}-u_{i-1,j}}{\Delta x}\right]\right.
$$
$$
+\frac{1}{\Delta y}\left[\mu_{i+1/2,j+1/2}\left(\frac{u_{i,j+1}-u_{i,j}}{\Delta y}+\frac{v_{i+1,j}-v_{i,j}}{\Delta x}\right)\right]
$$
$$
\left.-\frac{1}{\Delta y}\left[\mu_{i+1/2,j-1/2}\left(\frac{u_{i,j}-u_{i,j-1}}{\Delta y}+\frac{v_{i+1,j-1}-v_{i,j-1}}{\Delta x}\right)\right]\right\}
\tag{10}
$$

$$
\frac{\partial v_{i,j}}{\partial t} = fy_{i,j} + \frac{1}{\rho_{i,j+1/2}}\left\{\frac{1}{\Delta x}\left[\mu_{i+1/2,j+1/2}\left(\frac{v_{i+1,j}-v_{i,j}}{\Delta x}+\frac{u_{i,j+1}-u_{i,j}}{\Delta y}\right)\right]\right.
$$
$$
-\frac{1}{\Delta x}\left[\mu_{i-1/2,j+1/2}\left(\frac{v_{i,j}-v_{i-1,j}}{\Delta x}+\frac{u_{i-1,j+1}-u_{i-1,j}}{\Delta y}\right)\right]
$$
$$
\left.+\frac{2}{\Delta y}\left[\mu_{i,j+1}\frac{v_{i,j+1}-v_{i,j}}{\Delta y}-\mu_{i,j}\frac{v_{i,j}-v_{i,j-1}}{\Delta y}\right]\right\}
\tag{11}
$$

The discretization of Eq. (5) is easy to derive and is neglected here. The discretization of PPE (Eq. (6)) is as follows:

$$
\frac{1}{\Delta t}\left(\frac{u_{i,j}-u_{i-1,j}}{\Delta x}+\frac{v_{i,j}-v_{i,j-1}}{\Delta y}\right) = p_{i+1,j}\left(\frac{1}{\Delta x^2}\frac{1}{\rho_{i+1/2,j}}\right)+p_{i-1,j}\left(\frac{1}{\Delta x^2}\frac{1}{\rho_{i-1/2,j}}\right)
$$
$$
+p_{i,j+1}\left(\frac{1}{\Delta y^2}\frac{1}{\rho_{i,j+1/2}}\right)+p_{i,j-1}\left(\frac{1}{\Delta y^2}\frac{1}{\rho_{i,j-1/2}}\right)
$$
$$
-p_{i,j}\left[\frac{1}{\Delta x^2}\left(\frac{1}{\rho_{i+1/2,j}}+\frac{1}{\rho_{i-1/2,j}}\right)+\frac{1}{\Delta y^2}\left(\frac{1}{\rho_{i,j+1/2}}+\frac{1}{\rho_{i,j-1/2}}\right)\right]
\tag{12}
$$

2.2.3 VOF Equation

For the solution of Eq. (2), PLIC is utilized to reconstruct the interface [32]. The Lagrangian advection scheme is adopted to propagate the interfaces within the cells [33].

With the φ of a cell determined, the density and dynamic viscosity coefficient at the cell center are defined with:

$$\rho_{i,j} = \varphi_{i,j} \cdot \rho_{water} + \left(1 - \varphi_{i,j}\right) \cdot \rho_{gas} \tag{13a}$$

$$\mu_{i,j} = \varphi_{i,j} \cdot \mu_{water} + \left(1 - \varphi_{i,j}\right) \cdot \mu_{gas} \tag{13b}$$

In the discretizations of the two non-advection steps and the PPE equation (Eqs. (10)–(12)), the densities and dynamic viscosity coefficients at the cell edges and corners are necessary. If the viscous term in the Navier–Stokes equations is treated implicitly, the harmonic averaging method is suggested [10]. As the viscous term is treated explicitly in this model, the simple average method is adopted for simplicity:

$$s_{i+1/2,j} = \left(s_{i,j} + s_{i+1,j}\right)/2 \tag{14a}$$

$$s_{i+1/2,j+1/2} = \left(s_{i,j} + s_{i+1,j} + s_{i,j+1} + s_{i+1,j+1}\right)/4 \tag{14b}$$

where s stands for the density and dynamic viscosity coefficient.

2.3 AMR Grid
2.3.1 AMR Grid and Data Structure

As mentioned before, the AMR grid is employed in this model. The computation domain is defined as a quadtree (2D) or octree (3D) structure with all the nodes of the tree to indicate a grid block with the same grid number (see Figs. 2 and 3 for the grid and the corresponding data structure). During the simulation, the grid is allowed to refine/coarsen certain blocks based on the defined strategy. It is noted that the refinement level is not allowed to jump more than one at any location during the refining/coarsen process. In this paper, the open-source library Paramesh is utilized to manage the grid [34].
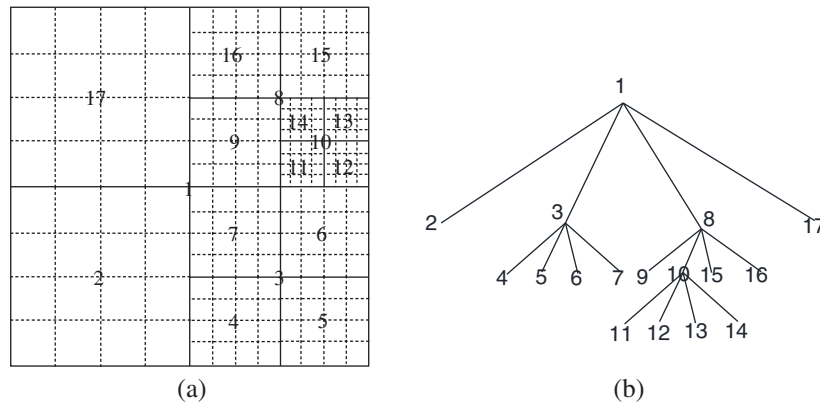

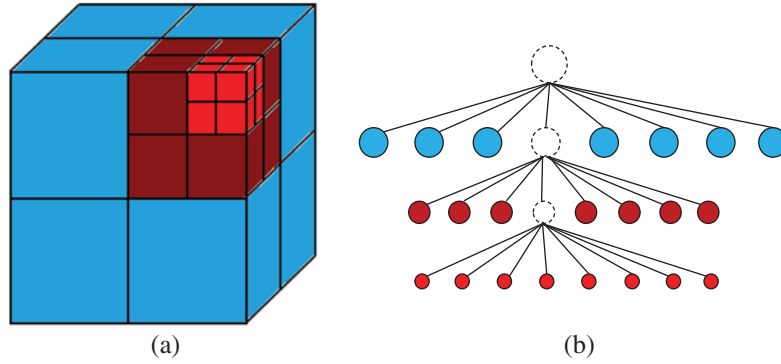**Figure 2:** (a) AMR grid and (b) the corresponding data structure in 2D

Figure 3: (a) AMR grid and (b) the corresponding data structure in 3D

A fixed time step strategy is adopted due to the incompressibility of the fluid. The single grid solver is applied on all the grid blocks without too many changes, except for the interpolation between blocks to provide boundary conditions.

2.3.2 Interpolation and Refining/Coarsen Operators

After each operation (advection, diffusion, and pressure correction) within each time step, interpolations are necessary as the grid blocks need ghost cells at their boundaries to complete the discretization stencils. Two layers of ghost cells are used for each block. Take the grid in Fig. 2 as an example, for the blocks at the same level (such as 9 and 16), the direct injection method is used. For the blocks whose ghost cell values are provided by a finer grid block (such as 17 provided by 9 and 16), the simple average method is used to preserve the accuracy as well as conservation. For the blocks whose ghost cell values are provided by a coarser grid block (such as 7 or 4 provided by 2), the algorithm is a little bit complex and we will interpret it in detail.

Due to the staggered arrangements of the variables, the interpolations for variables u and v are different. As shown in Fig. 4, $u_{i-1,j}$, $u_{i-2,j}$, $v_{i-1,j}$ and $v_{i,j}$ are needed for the fine grid. In this model, the bi-quadratic interpolation is adopted to achieve 2nd-order accuracy. The expressions are:

$$
\begin{aligned}
u_{i-2,j} &= \left(5U_{I,J+1} + 30U_{I,J} - 2U_{I,J-1}\right)/32 \\
u_{i-1,j} &= \left(u_{i-2,j} + u_{i,j}\right)/2 \\
v_{i-1,j} &= \left(V_{I,J} + 7V_{I+1,J} - v_{i+1,j}\right)/7 \\
v_{i,j} &= \left(8V_{I+1,J} + 10v_{i+1,j} - 3v_{i+2,j}\right)/15
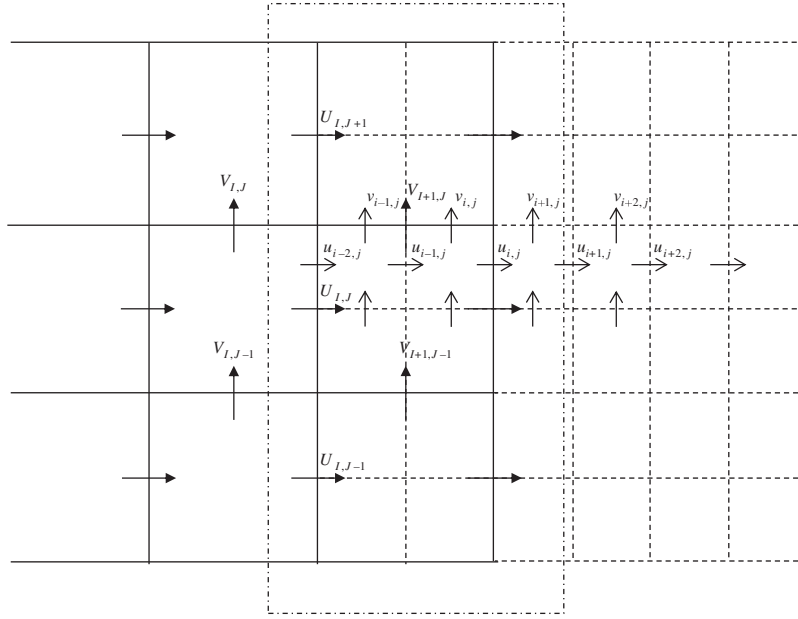\end{aligned}
\tag{15}
$$

Figure 4: Interpolations between coarse grid (solid lines with variables in the upper case) and fine grid (dashed lines with variables in the lower case), interpolation zone indicated with a box in the dash-dotted line

Other interpolations can be categorized into the expressions above. The interpolations in 3D are similar.

A refining/de-refining process is conducted based on the distance of a grid block from the free surface at the end of each time step. During the process, the divergence-free condition must be preserved. Here, the face averaging method is adopted for the de-refining process as it is a simple way to preserve the divergence of the velocity field on a staggered grid as well as global 2nd-order accuracy. For the refining process, the method proposed by Tóth et al. [35] in which a prolongation operator that can preserve the divergence and curl is employed.

2.3.3 Coarse-Fine Interface Treatment

A careful treatment of the coarse-fine interface is necessary during the solution process. Due to the staggered grid, velocities at the interface will share the same face for the coarse and fine grid as shown in Fig. 5. Flux conservation is necessary at the interface. The velocity on the finer grid is considered to be more accurate so a simple average is performed after each operation (advection step, diffusion step, and pressure correction step), which has the form of $U_{I,J}\Delta Y = (u_{i,j} + u_{i,j-1})\Delta y$. However, as the divergence-free condition is already fulfilled after the pressure correction step, the condition may no longer be preserved if an additional flux conservation operation is performed at the end of each time step.
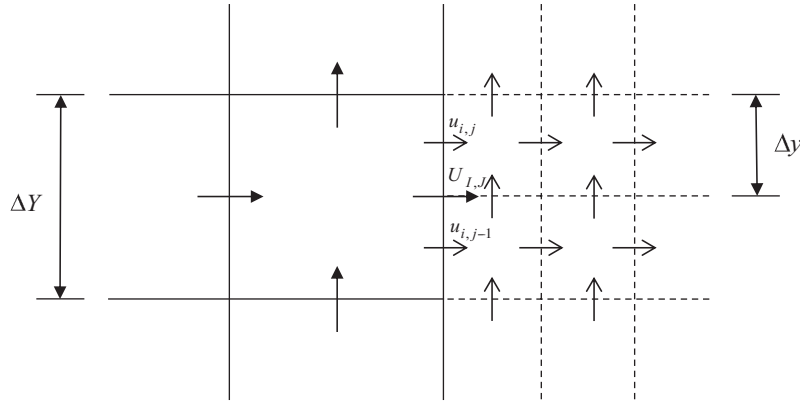
Figure 5: Flux conservation at the coarse-fine interface

A simple method to avoid the situation is to force the velocities to satisfy the flux conservation condition after the pressure correction step automatically. A pressure boundary condition method is proposed by Vanella et al. [4] for single-phase flow. The method could be extended to two-phase flow with the consideration of density variation. During the iterative solution of PPE, the gradient of the pressure normal to the coarse-fine interface is forced to satisfy the following equation:

$$\frac{1}{\rho} \frac{\partial p_{coarse}}{\partial x}\bigg|_{I,J} \Delta Y = \frac{1}{\rho} \frac{\partial p_{fine}}{\partial x}\bigg|_{i,j} \Delta y + \frac{1}{\rho} \frac{\partial p_{fine}}{\partial x}\bigg|_{i,j-1} \Delta y \tag{16}$$

Then the flux conservation condition will be automatically satisfied.

2.4 Algorithm

The complete algorithm of the model is as follows:

(1) Initialize the flow field with the initial topology of the water phase (φ^n), the velocity field ($\mathbf{u}^n$) and the pressure field (p^n).
(2) Calculate the position of the interface at the new time step to obtain φ^{n+1} by advecting the VOF equation (Eq. (2)).
(3) Obtaining the density and dynamic viscosity coefficient of the cell (ρ^{n+1}, μ^{n+1}) with φ^{n+1} (Eq. (13)).
(4) Compute the provisional velocity field $\mathbf{u}^*$ using Eq. (3).
(5) Compute the provisional velocity field $\mathbf{u}^{**}$ using Eq. (4).
(6) Calculate the pressure field p^{n+1} using Eq. (6).
(7) Compute the velocity field $\mathbf{u}^{n+1}$ using Eq. (5).
(8) Compute the distance from free surface for each grid block, refine/de-refine the blocks under the defined strategy.
(9) Proceed to Step (2) for the next time step.

3 Validations

In this section, three selected benchmark cases are carried out for the validation of the model. Shear flow tests are conducted to validate the extension of the VOF method to the AMR grid. A linear sloshing case is considered to check the accuracy of the model. A nonlinear sloshing problem is simulated to validate the accuracy of the model as well as discuss the efficiency with the AMR grid.

3.1 2D/3D Shear Flow Test

The shear flow test is a widely used benchmark to validate the VOF method [36]. In this problem, an initial scalar field evolves under a given velocity field. The forms of the velocity field are:

$$2D \quad \begin{aligned} u(x,y) &= \sin(\pi x)\cos(\pi y) \\ v(x,y) &= -\cos(\pi x)\sin(\pi y) \end{aligned} \tag{17a}$$

$$3D \quad \begin{aligned} u(x,y,z,t) &= 2\sin^2(\pi x)\sin(2\pi y)\sin(2\pi z)\cos(\pi t/T) \\ v(x,y,z,t) &= -\sin(2\pi x)\sin^2(\pi y)\sin(2\pi z)\cos(\pi t/T) \\ w(x,y,z,t) &= -\sin(2\pi x)\sin(2\pi y)\sin^2(\pi z)\cos(\pi t/T) \end{aligned} \tag{17b}$$

The zone scale is 1.0 for both 2D and 3D on all the dimensions. The initial shape of the scalar field is given as a circle in 2D and a sphere in 3D. So the corresponding volume fraction is set as:

$$2D \quad \varphi = \begin{cases} 1 & \sqrt{(x-0.5)^2 + (y-0.25)^2} < 0.15 \\ 0 & else \end{cases} \tag{18a}$$

$$3D \quad \varphi = \begin{cases} 1 & \sqrt{(x-0.35)^2 + (y-0.35)^2 + (z-0.35)^2} < 0.15 \\ 0 & else \end{cases} \tag{18b}$$

A 4-level AMR grid is used with the CFL number set to be 0.1 for all the simulations. The minimum scale of the grid is set as 1/128. The results at selected moments are shown in Figs. 6 and 7. For the 2D case, at time $t = 5$ s (Fig. 6a), discontinuity of the scalar field is observed. It is due to the incorrect normal direction and can be alleviated with a finer mesh. At time $t = 10$ s (Fig. 6b), the scalar field evolves back to a circle. For the 3D case, the sphere evolves to be disk type at $t = 0.5T$ (Fig. 7a) and then evolves back to a sphere at $t = T$ (Fig. 7b). The results are acceptable considering the results of previous works [13,36].
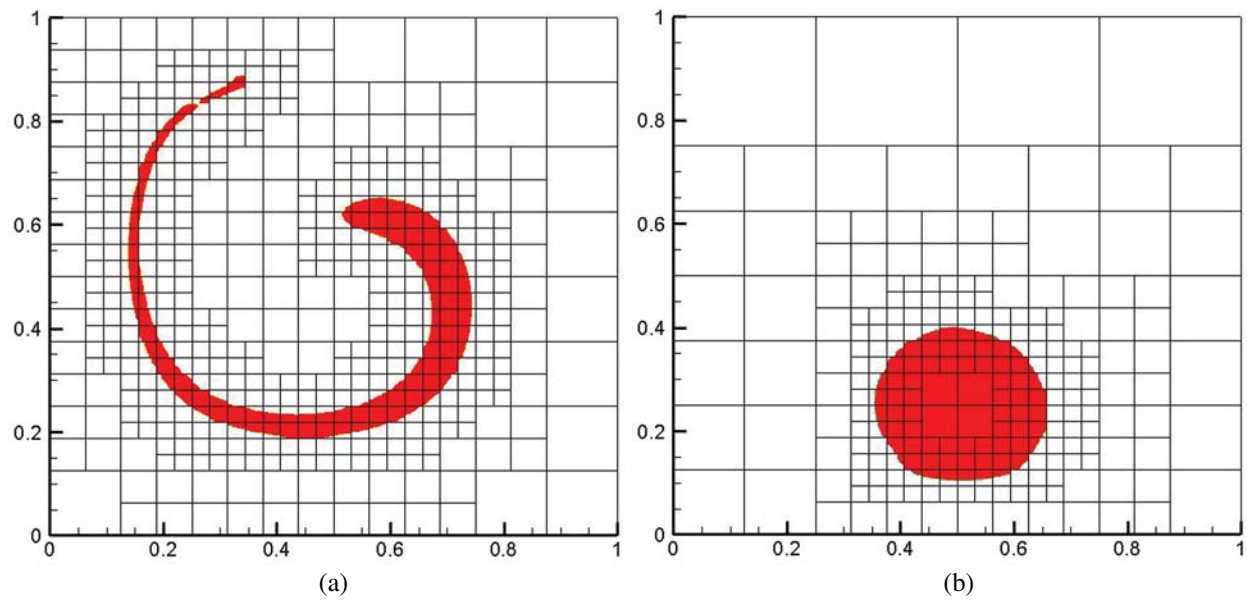
Figure 6: 2D scalar fields at selected moments (velocity field change sign symbol at $t = 5$ s): (a) $t = 5$ s; (b) $t = 10$ s
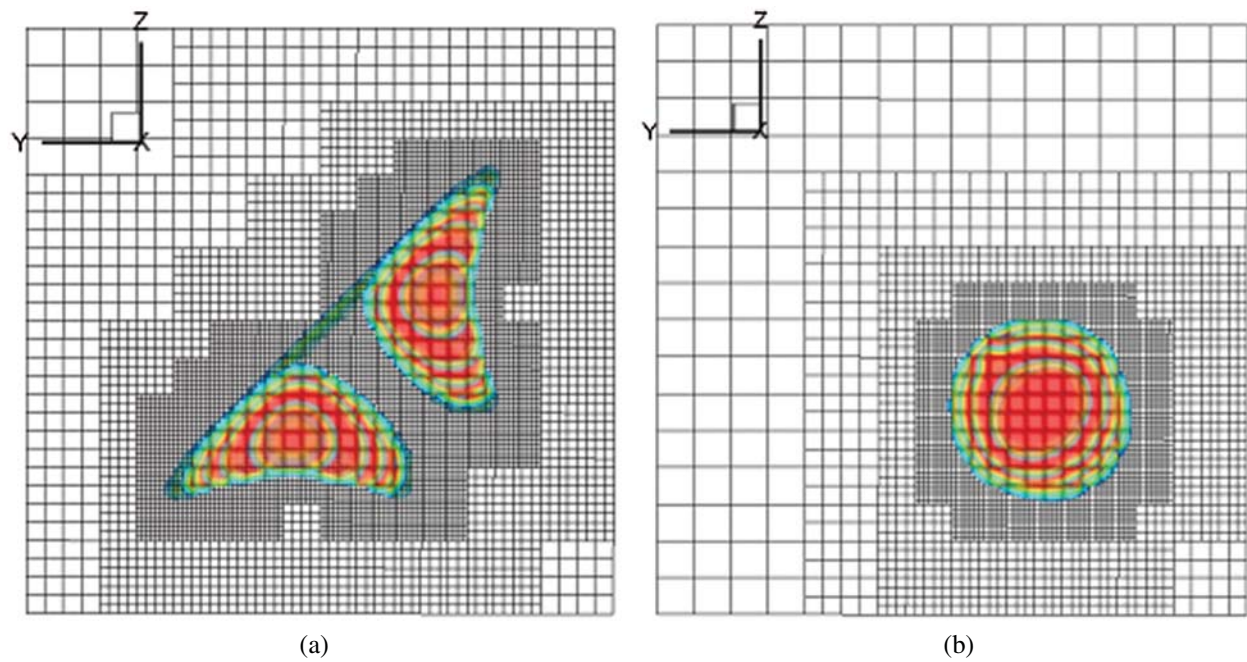

Figure 7: 3D scalar fields at selected moments ($T = 3$ s): (a) $t = 0.5\ T$; (b) $t = T$

3.2 2D Linear Free Sloshing with Viscous Effects

2D linear free sloshing is considered to validate the accuracy of the model. In this problem, a tank is filled with water and gas and the initial wave profile is given. The liquid is initially at rest and evolves submitted to the gravity force. An illustration of the problem is presented in Fig. 8.
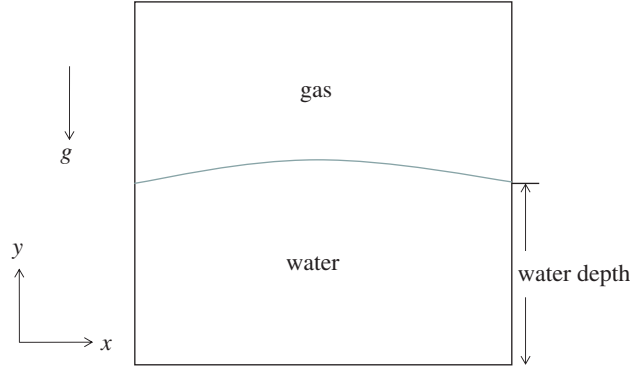

Figure 8: Illustration of the linear free sloshing problem

When the wave amplitude is small, a linear analytic solution for 2D viscous waves in a rectangular tank was derived by Wu et al. [37]. For the initial wave profile to be:

$$\eta_0 = a \cos k (x + W/2) \tag{19}$$

where W is the width of the tank, a is the initial wave amplitude and k is the wave number with $k = 2\pi/W$. When the condition that $\kappa = g/v^2 k^3 > 0.5814122$ is fulfilled where v is the kinematic viscosity of the liquid and g is the gravity acceleration, the analytical solution for free surface profile deformation can be expressed in the following form:

$$\frac{\eta(t)}{\eta_0} = 1 - 2\kappa \Re \left(\sum_{i=1}^{2} \frac{A_i \left\{ -\gamma_i e^{\left(-1+\gamma_i^2\right)vk^2 t} \left[1 + \mathrm{erf}\left(\gamma_i k \sqrt{vt}\right) \right] + \gamma_i + \mathrm{erf}\left(k\sqrt{vt}\right) \right\}}{1 - \gamma_i^2} \right) \tag{20}$$

in which $\Re$ means the real part of the complex number, γ_1 and γ_2 are any non-conjugate roots of the four possible roots of the following equation:

$$\left(x^2 + 1\right)^2 - 4x + \kappa = 0 \tag{21}$$

Here the width of the tank, the water depth, and the initial wave amplitude are set as 1.0 m, 0.5 m, and 0.01 m, respectively. A 4-level AMR grid is adopted for this case. The grids are uniform around the interface in both directions with the spacing of 1/128 m. A constant time step of 0.00001 s is used for all the simulations. Reynolds numbers of 20 and 200 are considered here to validate the model and both cases run up to 10 s. The results are shown in Fig. 9 with a comparison to the analytic solution and good agreement is obtained. It is observed that the decrease of wave amplitude for Re to be 20 is very fast, due to the large viscosity.
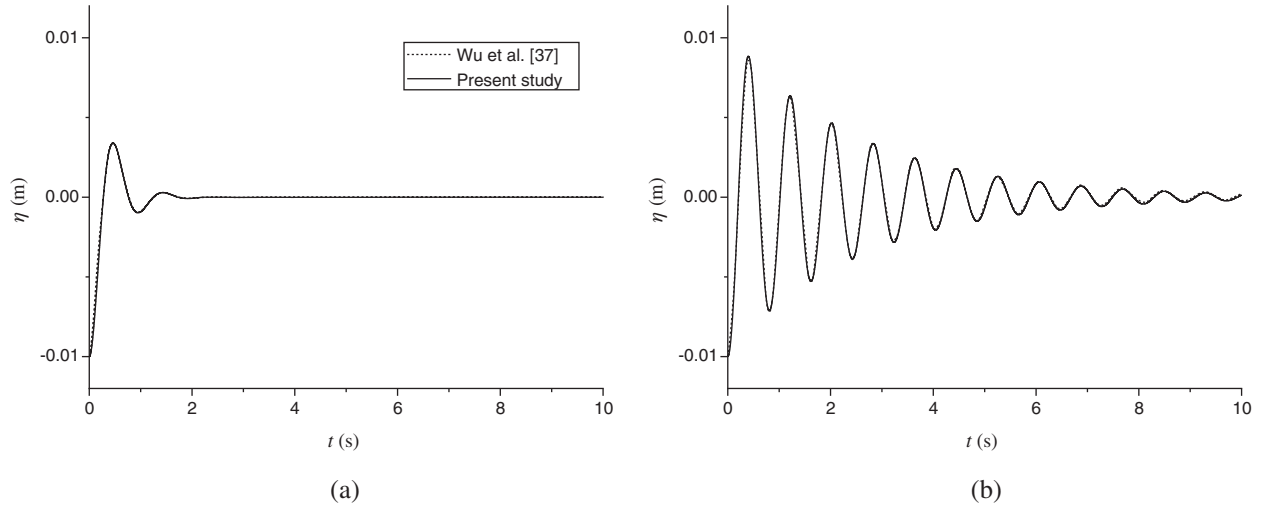
(a)   (b)

Figure 9: Time histories of the wave amplitudes for (a) Re = 20; (b) Re = 200. (Here $Re = h\sqrt{gh}/v$)

The spatial accuracy of the model is measured with the problem. The L_2 and L_∞ norm errors of the solution of free surface height are shown in Fig. 10. The solution on the 256×256 grid is defined as the reference solution. As shown in Figs. 10a and 10b, the L_2 norm error is nearly 2nd-order while the L_∞ norm error is between 1st-order and 2nd-order. The error should come from the smear near the free surface in the discretization (Eqs. (13) and (14)).
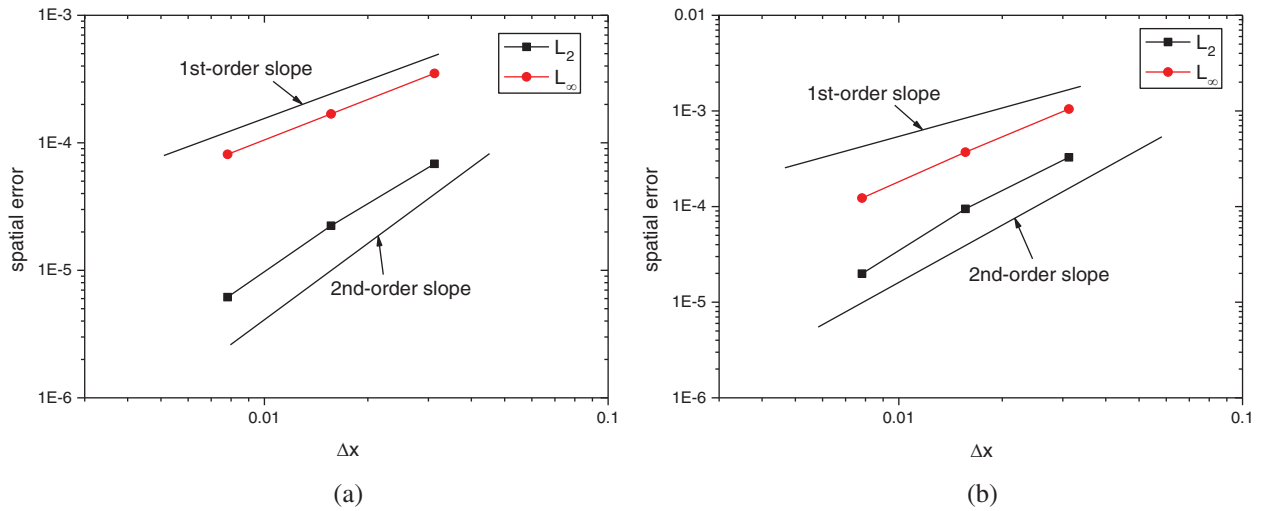

(a)   (b)

Figure 10: Variations of the norm errors of the free surface height with different grid spacing: (a) Re = 20; (b) Re = 200

3.3 2D Nonlinear Sloshing under Surge Motion

A 2D nonlinear sloshing resonance problem is considered with this model. In this problem, a partially filled tank submitted to the gravity force and an external horizontal force is simulated. The computational model with corresponding parameters consistent with Liu et al.'s [38]

experiment is shown in Fig. 11. When the frequency of the exciting force is close to the natural frequency of a tank, resonance will occur. For a 2D case, the natural frequency can be obtained with $\omega_n = \sqrt{g k_n \tanh (k_n d)}$ where d is the water depth and $k_n = n\pi/W$, here W is the width of the tank [39]. The width of the tank and the water depth are 0.57 m and 0.15 m, so the lowest natural frequency is $\omega_n = 6.0578$ s^{-1}. The water in the tank is at rest at $t = 0$ s. Under the external force, the movement of the tank follows $x = -a \sin \omega t$ in which the amplitude a is 0.005 m and $\omega = \omega_n$. So that resonance will occur after some time.
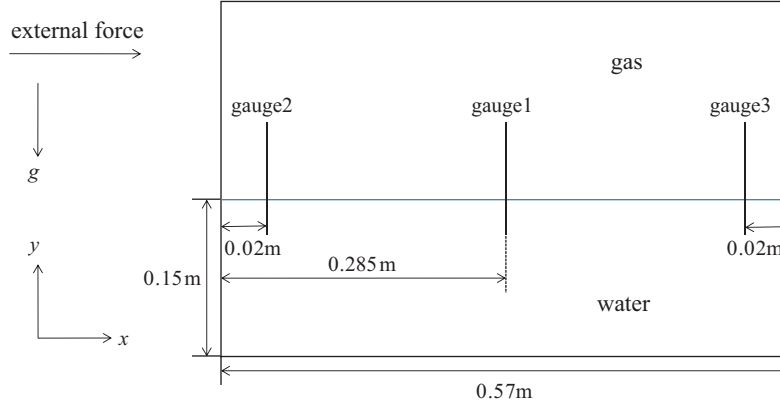

Figure 11: Computational model of 2D nonlinear sloshing

A 4-level AMR grid is adopted with the finest grid to be 0.0045 m in both directions. A constant time step of 0.001 s is used. Time histories of the free surface height at the three gauges for the time before 6.7 s are shown in Fig. 12. The linear analytic solution is also given. The results agree very well with the experimental data of [38]. While the linear analytic solution gives an acceptable prediction only for the time before about 3 s. Then it fails to predict the behavior due to the nonlinearity.
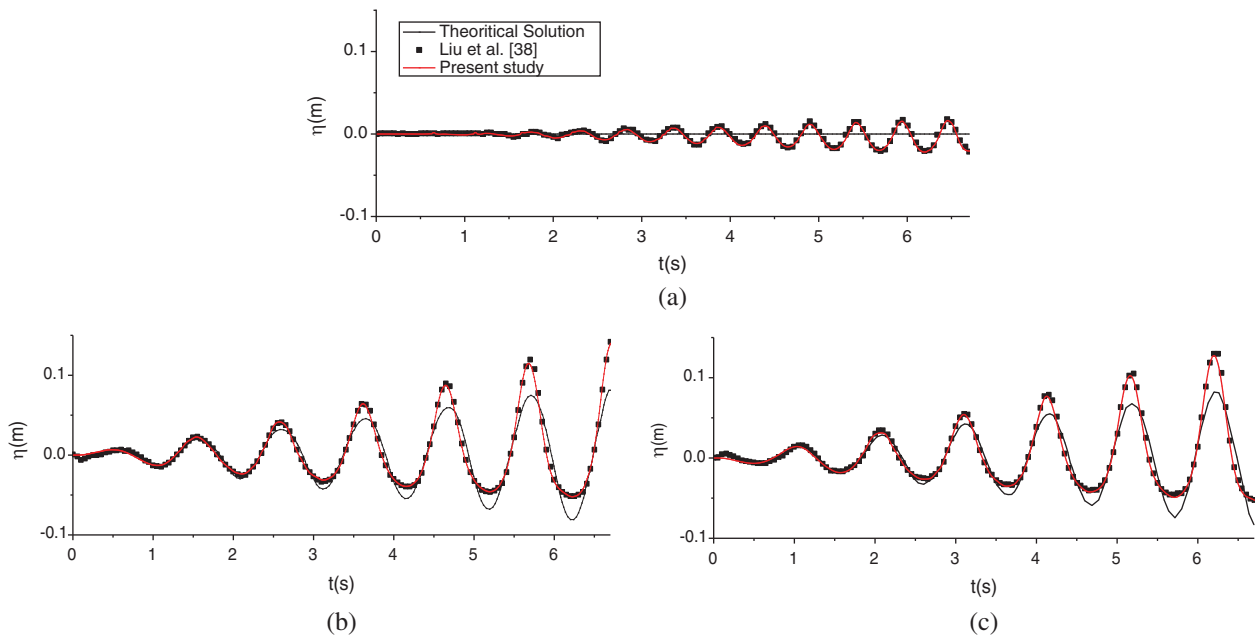

Figure 12: Time histories of free surface height at the gauges: (a) Gauge 1; (b) Gauge 2; (c) Gauge 3

With time going on, the wave amplitude will increase continuously until wave breaking occurs. Two selected moments are given in Fig. 13. Wave overturning and breaking occur on the left and right walls.
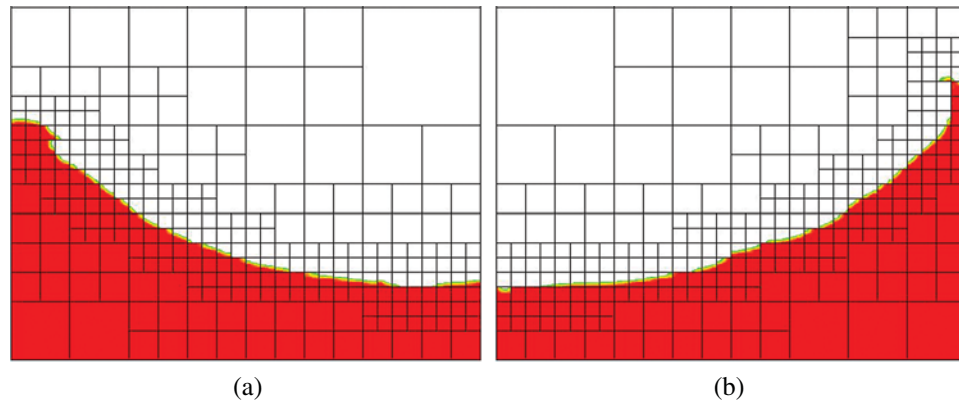

(a)   (b)

Figure 13: Wave overturning and breaking on the left and right wall: (a) $t = 7.8$ s; (b) $t = 8.2$ s

The efficiency of the AMR grid is discussed with this problem. An AMR grid with more levels usually leads to a smaller computational cost, if the minimum scale of the grid is set to be the same. To discuss it in quantitative, the stable phase of the nonlinear sloshing problem is selected ($t = 6\sim8$ s). Four grids with different levels (1 to 4) are considered and the finest grid scales are kept the same. The number of grid blocks varying with time is given in Fig. 14. Obviously, a grid with more levels leads to a smaller grid block number.
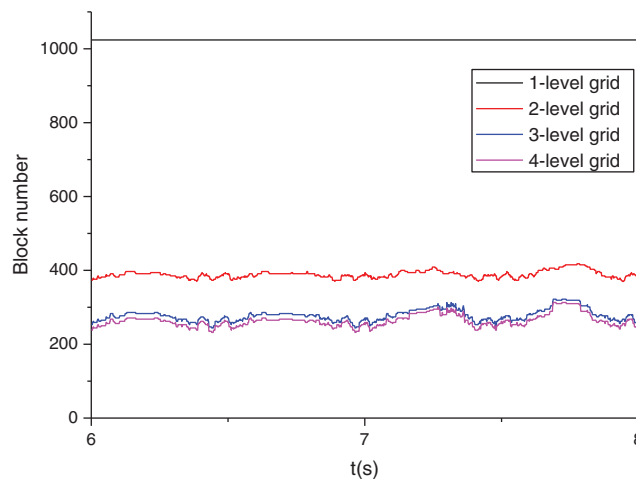

Figure 14: Time history of the grid block number for the four grids

The mean block number and corresponding CPU time for the four grids are shown in Tab. 1. With a 2-level AMR grid, more than 60 percent in grid number and more than 50 percent in CPU time are saved than the 1-level grid (uniform grid). More levels mean more savings and 70 percent in grid number and more than 60 percent in CPU time are saved for the 3-level and

4-level grid. The differences between the save in grid number and save in CPU time is also given in the Tab. 1. The difference increase with the increase of level, due to the increasing coarse-fine interface interpolation and refine/de-refine process for more level grids.

Table 1: Mean block number and efficiency for each level grid

	Mean block	Mean grid	Grid save (%)	CPU time (s)	CPU save (%)	Grid save—CPU save (%)
1	1024	16384	–	1964	–	–
2	388.86	6222	62.02	879	55.24	6.78
3	277.74	4444	72.88	731	62.78	10.10
4	264.04	4225	74.21	727	62.98	11.23

4 Results

In this section, two selected cases are utilized to show that the model is able to simulate violent interface problems.

4.1 2D Solitary Wave Past a Submerged Stationary Stage

To demonstrate the model's ability in dealing with wave breaking problems, a solitary wave past a submerged stationary stage is considered with the model. The computational model of the problem is shown in Fig. 15 with the parameters to be the same as the experiment conducted by Yasuda et al. [40]. Three gauges are set in the zone, referred to as P2, P3, and P4 by Yasuda et al. [40], respectively. The same problem was also considered by Lubin et al. [41] numerically.
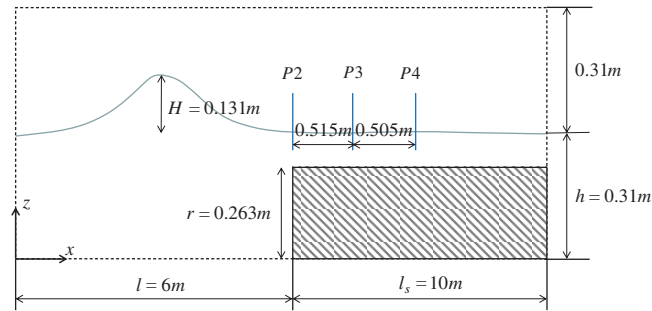

Figure 15: Computational model for solitary wave past a submerged stationary stage

A 4-level AMR grid is employed and a fixed time step of 0.001 s is adopted for the simulations. The minimum grid scales of the medium grid is 4.8×10^{-3} m on both horizontal and vertical directions after a careful convergence test. The time histories of free surface height at the gauges are compared with the experimental data from Yasuda et al. [40] and the numerical results from Lubin et al. [41], as shown in Fig. 16. All the results achieve good agreement including gauge P4, for which the free surface increases suddenly due to the existence of the stage.
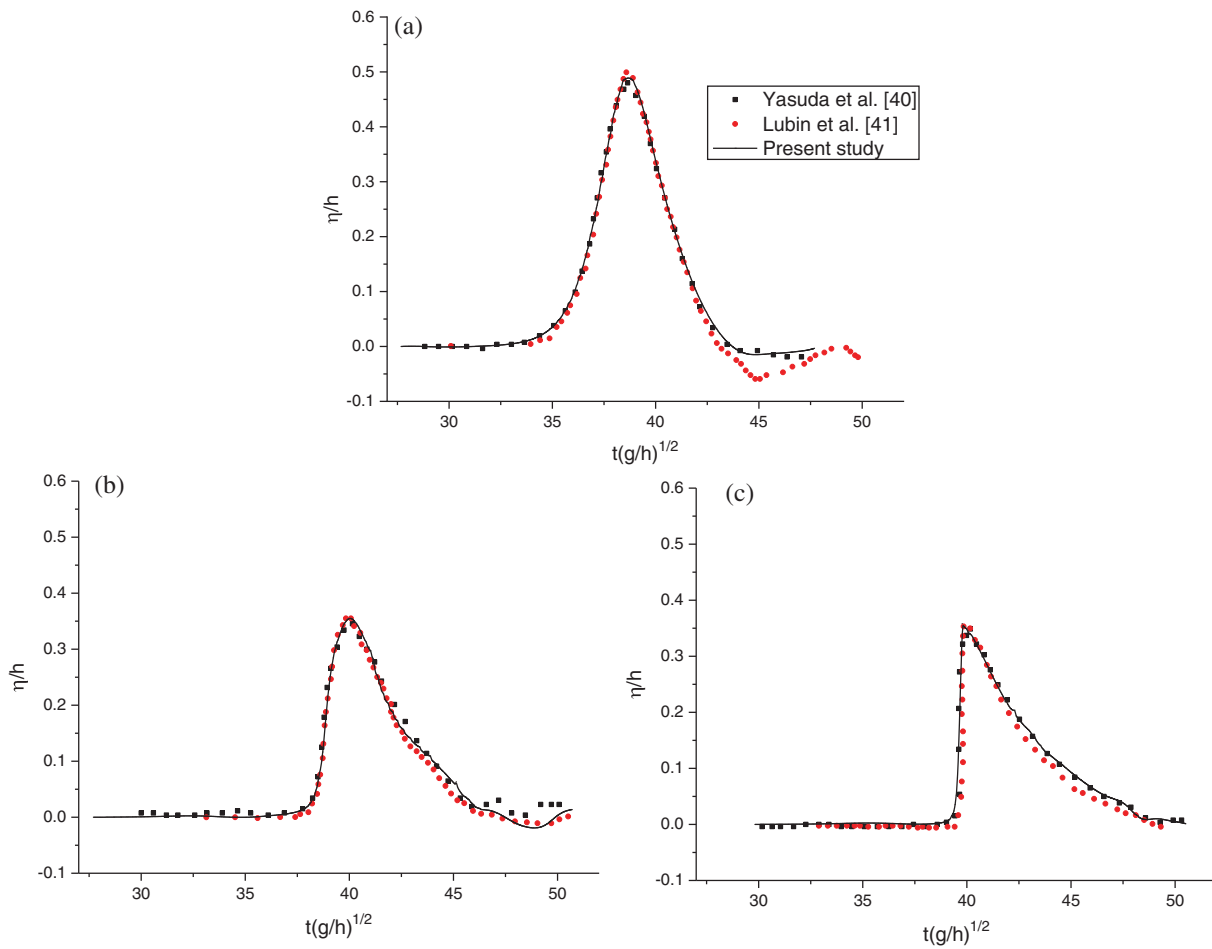
Figure 16: Time histories of free surface height at the gauges: (a) P2; (b) P3; (c) P4

Wave profiles at selected moments are shown in Fig. 17. The wave profile at the forehead becomes up straight first (Fig. 17a). Then the forehead rolls over as shown in Fig. 17b. The process means the current wave breaking mode is plungers [42].
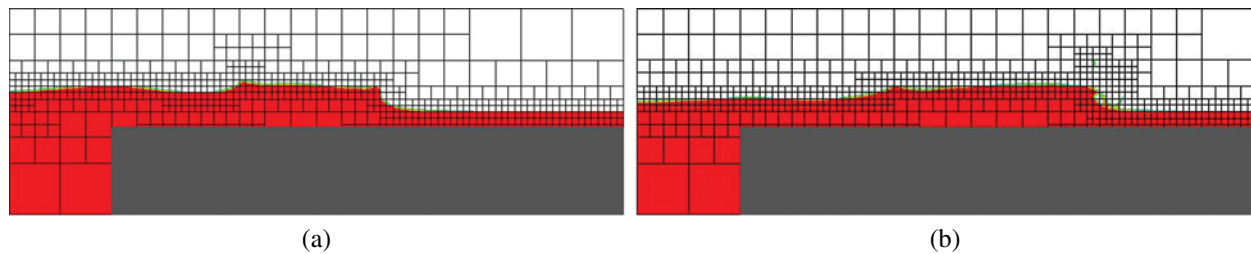

Figure 17: Wave profiles at selected moments: (a) forehead up straight; (b) forehead rolls over

4.2 2D/3D Dam Break

Dam break is a typical violent free-surface flow problem, as complicated phenomena including merging, overturning, and air entrainment occur during the process. The problem was studied by Martin et al. [43] experimentally, and then by researchers in various ways [44,45]. The computational layout of the 2D dam break problem is shown in Fig. 18, with $a = 0.2$ m corresponds to the parameters of Hu et al.'s [45] experiment. For the 3D case, the spanwise scale is a.
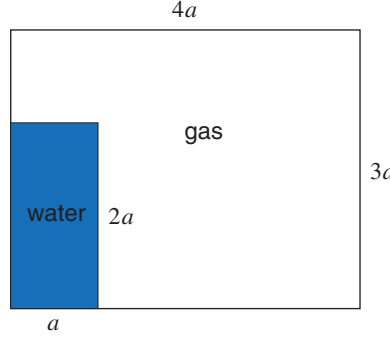

Figure 18: Computational layout of the dam break problem

Both 2D and 3D simulations are conducted, with the minimum grid scale to be 1/128 and 1/64 for 2D and 3D, respectively. A constant time step of 0.001 s is used. The water is initially at rest ($t = 0$ s) and then moves under the influence of gravity force. Time histories of wavefront location and water column height are presented in Fig. 19. Results on the three grids are compared with the experimental data of Martin et al. [43], Hu et al. [45], and the numerical results of Ubbink [46]. The results are non-dimensionalized with corresponding parameters to ease the comparison. It is observed that the results agree well with the experimental data and numerical results from other sources.
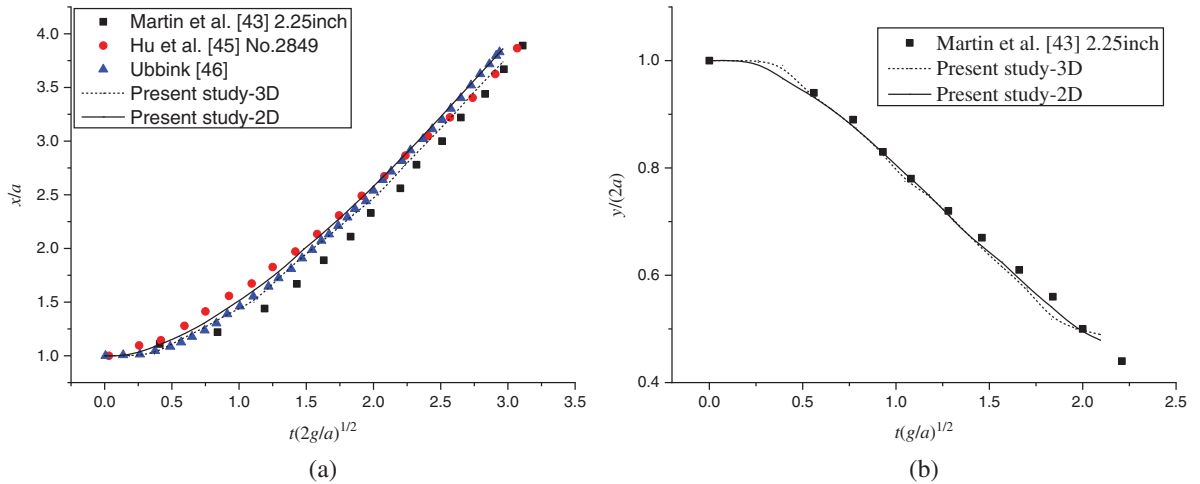

Figure 19: Comparison of non-dimensional time histories of: (a) the wavefront location; (b) the water column height
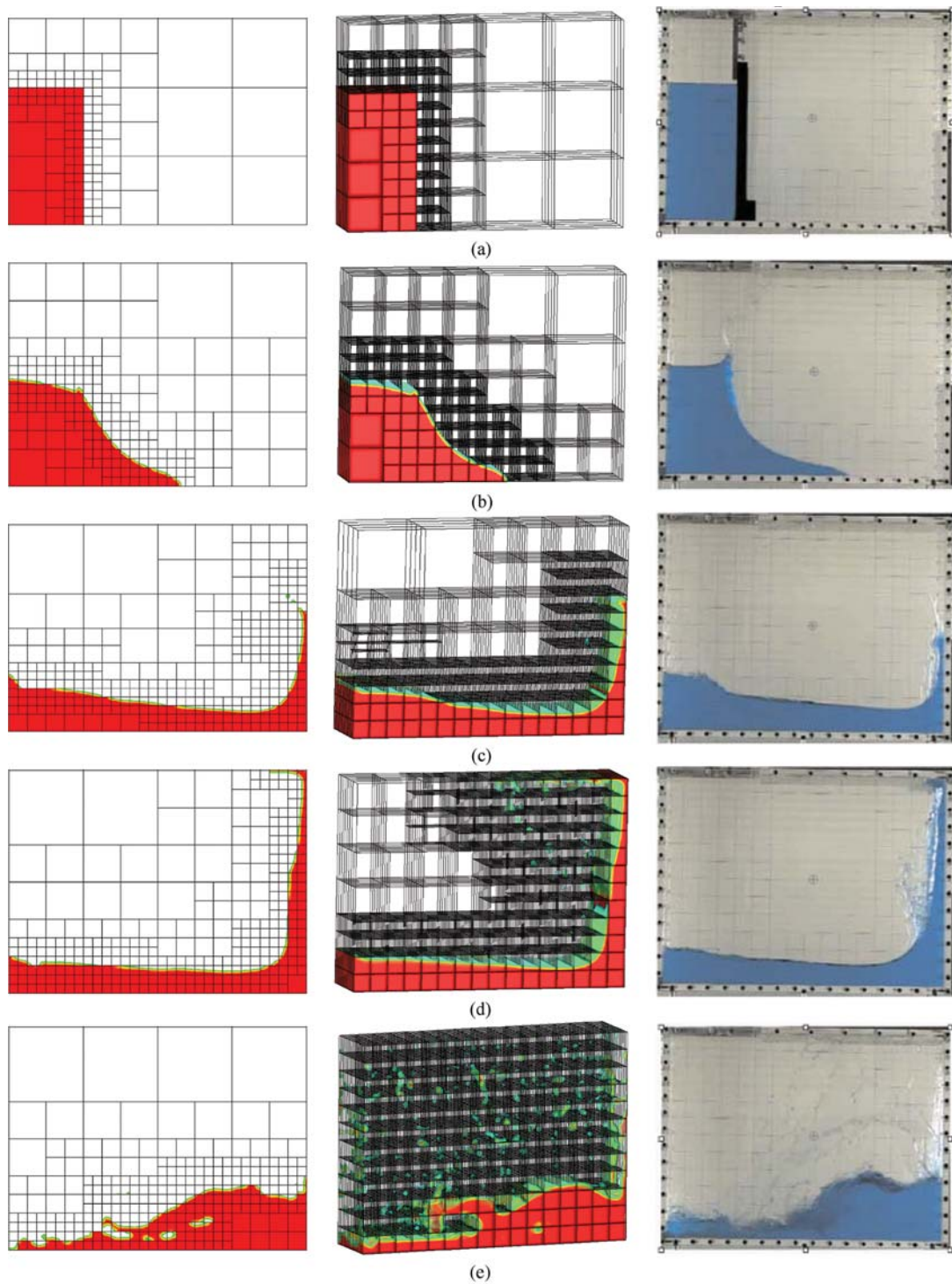
Figure 20: Comparison of the water run-up and overturning process (left: 2D results; medium: 3D results; right: experimental data from Hu et al. [45]). (a) t = 0.0 s, (b) t = 0.18 s, (c) t = 0.39 s, (d) t = 0.52 s, (e) t = 0.99 s

When the wavefront reaches the vertical wall on the right side, the water will run up and overturn. The process is shown in Fig. 20 with selected profiles, which are compared with the experimental data from Hu et al. [45]. The wavefront reaches the right wall and run-up along the wall (Fig. 20c). Then the water hits the roof with a jet formed (Fig. 20d). Finally, the water overturns to the bottom of the tank, with some bubbles exist in the bulk (Fig. 20e). It is observed that the present results agree well with the experimental data from Hu et al. [45].

5 Conclusions

A numerical model for simulating two-phase flow is developed with AMR. An explicit projection method is used to decouple the velocity and pressure. The free surface is captured with the VOF method combined with PLIC. A coarse-fine interface treatment method is developed to preserve the flux conservation at the interfaces. The adaptive grid is managed with the open-source PARAMESH library.

Several benchmark cases are considered to validate the model. The VOF method is validated with the shear flow test. The accuracy of the model is validated with a linear sloshing problem, for which analytic solutions can be derived. The model is demonstrated to have 2nd-order accuracy in space. The efficiency of AMR is demonstrated with a nonlinear sloshing problem.

The ability of the model to simulate more complex problems is demonstrated with two selected cases including solitary wave breaking and dam break. Good agreements are obtained between the current numerical results and experimental or numerical results from other sources. It is demonstrated that the model is able to simulate interface problems with wave breaking.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Berger, M. J., Oliger, J. (1984). Adaptive mesh refinement for hyperbolic partial differential equations. *Journal of Computational Physics, 53(3),* 484–512. DOI 10.1016/0021-9991(84)90073-1.

2. Popinet, S. (2003). Gerris: A tree-based adaptive solver for the incompressible Euler equations in complex geometries. *Journal of Computational Physics, 190(2),* 572–600. DOI 10.1016/S0021-9991(03)00298-5.

3. Liu, C., Hu, C. (2017). An immersed boundary solver for inviscid compressible flows. *International Journal for Numerical Methods in Fluids, 85(11),* 619–640. DOI 10.1002/fld.4399.

4. Vanella, M., Rabenold, P., Balaras, E. (2010). A direct-forcing embedded-boundary method with adaptive mesh refinement for fluid-structure interaction problems. *Journal of Computational Physics, 229(18),* 6427–6449. DOI 10.1016/j.jcp.2010.05.003.

5. Liu, C., Hu, C. (2018). An adaptive multi-moment FVM approach for incompressible flows. *Journal of Computational Physics, 359(1),* 239–262. DOI 10.1016/j.jcp.2018.01.006.

6. Roma, A. M., Peskin, C. S., Berger, M. J. (1999). An adaptive version of the immersed boundary method. *Journal of Computational Physics, 153(2),* 509–534. DOI 10.1006/jcph.1999.6293.

7. Griffith, B. E., Hornung, R. D., McQueen, D. M., Peskin, C. S. (2007). An adaptive, formally second order accurate version of the immersed boundary method. *Journal of Computational Physics, 223(1),* 10–49. DOI 10.1016/j.jcp.2006.08.019.

8. Popinet, S. (2009). An accurate adaptive solver for surface-tension-driven interfacial flows. *Journal of Computational Physics, 228(16),* 5838–5866. DOI 10.1016/j.jcp.2009.04.042.

9. Zuzio, D., Estivalezes, J. L. (2011). An efficient block parallel AMR method for two phase interfacial flow simulations. *Computers & Fluids, 44(1),* 339–357. DOI 10.1016/j.compfluid.2011.01.035.

10. Nangia, N., Griffith, B. E., Patankar, N. A., Bhalla, A. P. S. (2019). A robust incompressible Navier-Stokes solver for high density ratio multiphase flows. *Journal of Computational Physics, 390(1),* 548–594. DOI 10.1016/j.jcp.2019.03.042.

11. Liu, C., Hu, C. (2017). Adaptive THINC-GFM for compressible multi-medium flows. *Journal of Computational Physics, 342(2),* 43–65. DOI 10.1016/j.jcp.2017.04.032.

12. Sussman, M., Almgren, A. S., Bell, J. B., Colella, P., Howell, L. H. et al. (1999). An adaptive level set approach for incompressible two-phase flows. *Journal of Computational Physics, 148(1),* 81–124. DOI 10.1006/jcph.1998.6106.

13. Hu, C., Liu, C. (2018). Simulation of violent free surface flow by AMR method. *Journal of Hydrodynamics, 3(3),* 384–389. DOI 10.1007/s42241-018-0043-4.

14. Plas, P., Veldman, A., Seubers, H., Helder, J., Lam, K. (2018). Adaptive grid refinement for two-phase offshore applications. *37th International Conference on Ocean, Offshore and Arctic Engineering.* American Society of Mechanical Engineers, Madrid, Spain.

15. Zhang, Y., Duan, W., Liao, K., Ma, S., Xia, G. (2019). Numerical simulation of solitary wave breaking with adaptive mesh refinement. *38th International Conference on Ocean, Offshore and Arctic Engineering.* American Society of Mechanical Engineers, Glasgow, Scotland.

16. Tryggvason, G., Bunner, B., Esmaeeli, A., Juric, D., Al-Rawahi, N. et al. (2001). A front-tracking method for the computations of multiphase flow. *Journal of Computational Physics, 169(2),* 708–759. DOI 10.1006/jcph.2001.6726.

17. Osher, S., Fedkiw, R. P. (2001). Level set methods: An overview and some recent results. *Journal of Computational Physics, 169(2),* 463–502. DOI 10.1006/jcph.2000.6636.

18. Hirt, C. W., Nichols, B. D. (1981). Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics, 39(1),* 201–225. DOI 10.1016/0021-9991(81)90145-5.

19. Scardovelli, R., Zaleski, S. (1999). Direct numerical simulation of free-surface and interfacial flow. *Annual Review of Fluid Mechanics, 31(1),* 567–603. DOI 10.1146/annurev.fluid.31.1.567.

20. Arrufat, T., Crialesi-Esposito, M., Fuster, D., Ling, Y., Malan, L. et al. (2020). A mass-momentum consistent, volume-of-fluid method for incompressible flow on staggered grids. *Computers & Fluids, 215(2),* 104785. DOI 10.1016/j.compfluid.2020.104785.

21. Nangia, N., Patankar, N. A., Bhalla, A. P. S. (2019). A DLM immersed boundary method based wave-structure interaction solver for high density ratio multiphase flows. *Journal of Computational Physics, 398(17–18),* 108804. DOI 10.1016/j.jcp.2019.07.004.

22. Patel, J. K., Natarajan, G. (2017). A novel consistent and well-balanced algorithm for simulations of multiphase flows on unstructured grids. *Journal of Computational Physics, 350(1),* 207–236. DOI 10.1016/j.jcp.2017.08.047.

23. Pathak, A., Raessi, M. (2016). A 3D, fully Eulerian, VOF-based solver to study the interaction between two fluids and moving rigid bodies using the fictitious domain method. *Journal of Computational Physics, 311(17),* 87–113. DOI 10.1016/j.jcp.2016.01.025.

24. Pathak, A., Freniere, C., Raessi, M. (2017). Advanced computational simulations of water waves interacting with wave energy converters. *European Journal of Computational Mechanics, 26(1–2),* 172–204. DOI 10.1080/17797179.2017.1306829.

25. Ghasemi, A., Anbarsooz, M., Malvandi, A., Ghasemi, A., Hedayati, F. (2017). A nonlinear computational modeling of wave energy converters: A tethered point absorber and a bottom-hinged flap device. *Renewable Energy, 103(3),* 774–785. DOI 10.1016/j.renene.2016.11.011.

26. Dafnakis, P., Bhalla, A. P. S., Sirigu, S. A., Bonfanti, M., Bracco, G. et al. (2020). Comparison of wave-structure interaction dynamics of a submerged cylindrical point absorber with three degrees of freedom using potential flow and computational fluid dynamics models. *Physics of Fluids, 32(9),* 93307. DOI 10.1063/5.0022401.

27. Bhalla, A. P. S., Nangia, N., Dafnakis, P., Bracco, G., Mattiazzo, G. (2020). Simulating water-entry/exit problems using Eulerian-Lagrangian and fully-Eulerian fictitious domain methods within the open-source IBAMR library. *Applied Ocean Research, 94(14–15),* 101932. DOI 10.1016/j.apor.2019.101932.

28. Tryggvason, G., Scardovelli, R., Zaleski, S. (2011). *Direct numerical simulations of gas-liquid multiphase flows*. UK: Cambridge University Press.

29. Xiao, F. (1999). A computational model for suspended large rigid bodies in 3D unsteady viscous flows. *Journal of Computational Physics, 155(2),* 348–379. DOI 10.1006/jcph.1999.6340.

30. Leonard, B. P. (1979). A stable and accurate convective modelling procedure based on quadratic upstream interpolation. *Computer Methods in Applied Mechanics and Engineering, 19(1),* 59–98. DOI 10.1016/0045-7825(79)90034-3.

31. Harten, A. (1983). High-resolution schemes for hyperbolic conservation laws. *Journal of Computational Physics, 49(3),* 357–393. DOI 10.1016/0021-9991(83)90136-5.

32. Youngs, D. L. (1982). Time-dependent multi-material flow with large fluid distortion. *Numerical methods for fluid dynamics*. Cambridge: Academic Press.

33. 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. *Journal of Computational Physics, 152(2),* 423–456. DOI 10.1006/jcph.1998.6168.

34. MacNeice, P., Olson, K. M., Mobarry, C., de Fainchtein, R., Packer, C. (2000). PARAMESH: A parallel adaptive mesh refinement community toolkit. *Computer Physics Communications, 126(3),* 330–354. DOI 10.1016/S0010-4655(99)00501-9.

35. Tóth, G., Roe, P. L. (2002). Divergence-and curl-preserving prolongation and restriction formulas. *Journal of Computational Physics, 180(2),* 736–750. DOI 10.1006/jcph.2002.7120.

36. Rider, W. J., Kothe, D. B. (1998). Reconstructing volume tracking. *Journal of Computational Physics, 141(2),* 112–152. DOI 10.1006/jcph.1998.5906.

37. Wu, G. X., Taylor, R. E., Greaves, D. M. (2001). The effect of viscosity on the transient free-surface waves in a two-dimensional tank. *Journal of Engineering Mathematics, 40(1),* 77–90. DOI 10.1023/A:1017558826258.

38. Liu, D., Lin, P. (2008). A numerical study of three-dimensional liquid sloshing in tanks. *Journal of Computational Physics, 227(8),* 3921–3939. DOI 10.1016/j.jcp.2007.12.006.

39. Lin, P. (2008). *Numerical modeling of water waves*. UK: Taylor & Francis Routledge.

40. Yasuda, T., Mutsuda, H., Mizutani, N. (1997). Kinematics of overturning solitary waves and their relations to breaker types. *Coastal Engineering, 29(3),* 317–346. DOI 10.1016/S0378-3839(96)00032-4.

41. Lubin, P., Stéphane, G., Kimmoun, O., Branger, H. (2011). Numerical study of the hydrodynamics of regular waves breaking over a sloping beach. *European Journal of Mechanics, B/Fluids, 30(6),* 552–564. DOI 10.1016/j.euromechflu.2011.01.001.

42. Perlin, M., Choi, W., Tian, Z. (2013). Breaking waves in deep and intermediate waters. *Annual Review of Fluid Mechanics, 45(1),* 115–145. DOI 10.1146/annurev-fluid-011212-140721.

43. Martin, J. C., Moyce, W. J. (1952). Part IV. An experimental study of the collapse of liquid columns on a rigid horizontal plane. Philosophical Transactions of the Royal Society of London. *Series A, Mathematical and Physical Sciences, 244(882),* 312–324. DOI 10.1098/rsta.1952.0006.

44. Hu, C., Kashiwagi, M. (2004). A CIP-based method for numerical simulations of violent free-surface flows. *Journal of Marine Science and Technology, 9(4),* 143–157. DOI 10.1007/s00773-004-0180-z.

45. Hu, C., Sueyoshi, M. (2010). Numerical simulation and experiment on dam break problem. *Journal of Marine Science and Application, 9(2),* 109–114. DOI 10.1007/s11804-010-9075-z.

46. Ubbink, O. (1997). *Numerical prediction of two fluid systems with sharp interfaces (Ph.D. Thesis)*. University of London, UK.