

A 3D, fully Eulerian, VOF-based solver to study the interaction between two fluids and moving rigid bodies using the fictitious domain method

Ashish Pathak^a, Mehdi Raessi^{a,*}

^a*Department of Mechanical Engineering, University of Massachusetts-Dartmouth, North
Dartmouth, MA 02740*

Abstract

We present a three-dimensional (3D) and fully Eulerian approach to capturing the interaction between two fluids and moving rigid structures by using the fictitious domain and volume-of-fluid (VOF) methods. The solid bodies can have arbitrarily complex geometry and can pierce the fluid-fluid interface, forming contact lines. The three-phase interfaces are resolved and reconstructed by using a VOF-based methodology. Then, a consistent scheme is employed for transporting mass and momentum, allowing for simulations of three-phase flows of large density ratios. The Eulerian approach significantly simplifies numerical resolution of the kinematics of rigid bodies of complex geometry and with six degrees of freedom. The fluid-structure interaction (FSI) is computed using the fictitious domain method. The methodology was developed in a Message Passing Interface (MPI) parallel framework accelerated with Graphics Processing Units (GPUs). The computationally intensive solution of the pressure Poisson equation is ported to GPUs, while the remaining calculations are performed on CPUs. The performance and accuracy of the methodology is assessed using an array of test cases, focusing individually on the flow solver and the FSI in surface-piercing configurations. Finally, an application of the proposed methodology in simulations of the ocean wave energy converters is presented.

*Corresponding author

Email addresses: apathak@umassd.edu (Ashish Pathak), mraessi@umassd.edu (Mehdi Raessi)

Keywords: Multi-phase, Fluid-solid-interaction, Volume-of-fluid, Moving rigid bodies, Multi-material reconstruction

1. Introduction

Fluid-Structure Interaction (FSI) can be found in many diverse areas encompassing sediment transport in rivers, fluidized beds, blood flow in arteries, power plant condensers, and ocean wave energy converters (WECs). In this paper, we present a computational framework to model FSI between two fluids and a moving rigid structure. The framework is quite general and applicable to many diverse FSI applications. However, our focus is on WECs. WECs have been traditionally modeled using the potential flow theory, where both linear [1, 2] and nonlinear [3–6] models have been developed. A review on theory and applications of both types of models can be found in [7]. However, these methods cannot handle large topographical changes in the free-surface, e.g., breaking of waves around structures. Moreover, such methods employ an empirical relation, e.g., Morison’s equation [8], to account for viscous drag. Empirical relations for viscous drag exist only for simple structure geometries. Complex geometries will require additional wave tank tests.

Phenomena like viscous layer separation, turbulence, wave-breaking and overtopping are critical to correct numerical prediction of WEC response and can only be captured by solving the full Navier-Stokes (N-S) equations, including the viscous term. Approaches that employ the full N-S solutions to compute FSI include body conformal methods [9–13], immersed boundary (IB) methods [14, 15], and fictitious domain (FD) methods [16, 17]. A detailed review on different numerical modeling techniques of FSI can be found in [18]. In body conformal methods, the mesh is updated every time step to conform to the structure boundary or the free interface. The method becomes especially challenging if the boundary/interface is arbitrarily complex and undergoes large deformation. That issue is circumvented in IB and FD methods where the mesh for solving

the N-S equations does not have to conform to the boundary/interface. In the IB method first developed by Peskin [14], a forcing term is added to the N-S equations. In another variant of the IB method developed by Mohd-Yusof [19], the effect of immersed boundary is introduced by imposing velocity boundary condition at the grid points located in the vicinity of the interface. In the IB approach, the forces on the structure are computed by explicit integration of pressure and shear stresses at the immersed interface, e.g., [20]. In this regard, Hu and Joseph [21] and Fekken [22] observed that using the explicit integration to compute hydrodynamic forces on the structure is unstable if mass (m) of the moving structure is less than the virtual mass (m_v) of the surrounding fluid accelerated by the motion of the structure, i.e., if $m < m_v$. The explicit integration scheme will be unstable in spite of small time steps. To avoid this problem, Hu and Joseph [21] suggested alternately solving the equations for the structure and fluid in an iterative fashion until a convergence criterion is satisfied. The iterative procedure makes the scheme computationally expensive. Hesla's [23] combined weak formulation of the fluid-structure evolution obviates this iterative procedure. The formulation called the *fictitious domain method* was developed by Glowinski [24] using the distributed Lagrangian multiplier and later made computationally fast by Patankar [25, 26]. The fast method of Patankar was employed in a finite volume framework by Sharma [17]. In the fast fictitious domain method, the structure is considered as a fictitious fluid and the N-S equations are solved in the entire computational domain. Rigid body velocity is then imposed in the solid domain by conserving linear and angular momenta. The method was used by [27–29] to study FSI. In the present work, we employ the same method to compute FSI between two fluids and a moving rigid body.

Most of the previous work with FD method involved a structure completely submerged in fluid, e.g. [17, 24, 27, 29–31]. Of these, [29] and [30] used a fully Eulerian approach for computing both fluid and structure motions. FD method has been applied in 2D to surface piercing floating structures in [28, 32–34].

In 3D, [16, 35, 36] employed the FD method to surface piercing floating particles, where the free surface was represented by the level set function. The particle motion, however, was treated in a Lagrangian fashion. We present a fully Eulerian fictitious domain methodology to compute FSI in surface piercing configurations, i.e., rigid structures interacting with two immiscible incompressible fluids. Here, the kinematics of the solid boundary and the interface between the two fluids are resolved by using the Volume-of-Fluid (VOF) method through two separate color functions. Such representation allows us to handle arbitrarily shaped and complex solid bodies. The proposed methodology employs a single fully Eulerian finite volume grid for moving the structure and the two fluids. The Eulerian implementation of the FD method offers some advantages:

- It obviates the need for transfer functions, otherwise needed in Lagrangian treatment of the rigid body motion, to interpolate between Eulerian background grid and the Lagrangian nodes attached to the rigid body.
- Relying on the Eulerian transport schemes, e.g., the VOF method of Youngs [37, 38], simplifies appreciably the six degree-of-freedom (DOF) position update of the rigid body. Otherwise, in a Lagrangian framework, update of the rotational positions of a non-spherical body becomes a rather cumbersome task involving either Euler’s angles or quaternions. Recently, van Wachem et al. [39] implemented quaternion rotation to study the FSI involving non-spherical particles.
- Phase change processes can be easily handled by an Eulerian approach where the solid phase is represented by a color function such as VOF [29].

There are some challenges in implementing a fully Eulerian FD method to surface piercing configurations. In such configurations, there can exist computational cells that contain three phases (solid and the two fluids). Such cells require special treatment to reconstruct the interfaces and transport mass. We use the 3D error-minimization VOF method formulated by Pathak and Raessi [40] to reconstruct the phase interfaces and transport mass in three-phase cells. An alternative approach can be the moment-of-fluid (MoF) method first proposed by

Ahn and Shashkov [41], which requires additional information in the form of cell-wise centroid position of each phase. Recently, Li et al. [42] presented a MoF approach to track centroids along with volume fractions. The method [40] adopted in the present work requires only the available volume fraction distribution to perform interface reconstruction in three-phase cells. Furthermore, the density ratio among the three phases can be arbitrarily large. If not treated properly, non-physical deformations can be observed at the phase interfaces [43, 44]. We employ the consistent scheme proposed by Rudman [45] that can handle simulations involving large density ratios. In this scheme, mass and momentum are transported in a tightly coupled consistent fashion. The formulation is explained in Section 3.3.

In several previous studies, e.g., [22, 46], involving free surfaces, the N-S equations are solved only for the water phase. The effect of air is considered via pressure and shear stress boundary conditions. Such approach might not be appropriate for modeling phenomena like wave-breaking. Iafrati [47] showed significant dissipation of energy occurring in the air phase via formation of large scale dipoles during wave-breaking. The methodology proposed in the present paper solves the full N-S equations in both air and water phases making it suitable for modeling phenomena like breaking of waves around structures. Although, its immediate application is WECs, the computational tool was developed at a general level without any simplifications and by including all effects (e.g., surface tension) so that it is versatile and can be readily employed in other applications.

The organization of the paper is as follows. We present the governing equations and the boundary conditions in Section 2. The numerical implementation is provided in Section 3. Section 4 assesses the accuracy of our numerical framework via an array of test cases. Finally, the work is summarized in Section 5.

2. Governing equations

Consider a two-fluid flow interacting with a moving rigid body. The two fluids are assumed to be Newtonian, immiscible and incompressible. The conservation equations of mass and momentum are given by Eqs. 1 and 2, respectively.

$$\nabla \cdot \vec{U} = 0 \quad (1)$$

$$\frac{\partial}{\partial t}(\rho \vec{U}) + \nabla \cdot (\rho \vec{U} \vec{U}) = -\nabla p + \nabla \cdot \tau + \vec{F}_B + \vec{F}_{ST} + \vec{F}_S \quad (2)$$

Here, \vec{U} is the velocity vector, ρ is the density and p is the pressure. \vec{F}_B and \vec{F}_{ST} are the body force, e.g. gravity, and the surface tension force, respectively. \vec{F}_S denotes the fluid-structure interaction force. τ represents the stress tensor and is given by,

$$\tau = \mu(\nabla \vec{U} + \nabla \vec{U}^T) \quad (3)$$

where μ is the coefficient of dynamic viscosity.

The two fluids and the rigid solid body share the same Eulerian mesh. To track their volumes, we define two scalars f and ψ as,

$$f(\vec{x}) = \begin{cases} 1 & \vec{x} \in \text{fluid 1} \\ 0 & \vec{x} \notin \text{fluid 1} \end{cases} \quad (4)$$

$$\psi(\vec{x}) = \begin{cases} 1 & \vec{x} \in \text{solid} \\ 0 & \vec{x} \notin \text{solid} \end{cases} \quad (5)$$

In a discretized domain, we define volume fraction, F , as the fraction of cell volume V occupied by *fluid 1*, i.e.,

$$F = \frac{1}{V} \int_V f \, dv \quad (6)$$

Similarly, the discrete volume fraction, Ψ , for the solid phase is defined as,

$$\Psi = \frac{1}{V} \int_V \psi \, dv \quad (7)$$

The two fluids and the solid phase are tracked by solving the following transport equations:

$$\frac{\partial f}{\partial t} + (\vec{U} \cdot \nabla)f = 0 \quad (8)$$

$$\frac{\partial \psi}{\partial t} + (\vec{U} \cdot \nabla)\psi = 0 \quad (9)$$

Solid body motion has been treated by using a Lagrangian approach in several previous studies [17, 24, 27]. Adopting an Eulerian approach (given by Eqs. (5) and (9)), however, offers some advantages, for example, relatively straightforward handling of the motion of solid bodies with arbitrarily complex shapes.

The volume fractions F and Ψ are also used to compute the density and viscosity in the discretized form of the momentum equation (2):

$$\rho = F \rho_l + \Psi \rho_s + (1 - \Psi - F) \rho_g \quad (10)$$

where, ρ_l and ρ_g are the densities of the fluid 1 and 2, respectively, and ρ_s is the density of the solid phase. The coefficient of dynamic viscosity is defined by:

$$\frac{1}{\mu} = \frac{F}{\mu_l} + \frac{1 - F}{\mu_g} \quad (11)$$

Here, μ_l and μ_g are the coefficients of dynamic of viscosity of the two fluids. Such prescription of viscosity avoids severe accelerations in the lighter and less viscous fluid. The viscosity is applied only in the fluid domain. Inside the solid domain viscosity is of little significance because the strain rate ($\nabla \vec{U} + \nabla \vec{U}^T$) is zero due to the rigidity constraint.

At the wall boundaries of the numerical domain we impose no-slip and no-penetration boundary conditions:

$$\vec{U} \cdot \hat{n}_s = 0 \quad (12)$$

$$\vec{U} \cdot \hat{t}_s = 0 \quad (13)$$

Here, \hat{n}_s and \hat{t}_s are unit normal and tangential vectors to the wall. Additionally, a zero pressure gradient is imposed at the wall boundaries of the numerical

domain:

$$\frac{\partial p}{\partial n_s} = 0 \quad (14)$$

3. Numerical Implementation

We employ finite volume discretization of Eq. (2) on a staggered Cartesian grid according to the Marker and Cell (MAC) method [48]. In a staggered arrangement, the variables f , ψ , ρ and p are located at cell centres, while the velocities are defined at the cell faces. To solve Eq. (2), we implement the two-step projection method of Chorin [49].

$$\frac{\rho^{n+1}\vec{U}^* - \rho^n\vec{U}^n}{\Delta t} = -\nabla \cdot (\rho\vec{U}\vec{U})^n + \nabla \cdot (\tau)^n + \vec{F}_B^n + \vec{F}_S^n \quad (15)$$

$$\frac{\rho^{n+1}\vec{U}^{n+1} - \rho^{n+1}\vec{U}^*}{\Delta t} = -\nabla p^{n+1} + \vec{F}_{ST}^{n+1} \quad (16)$$

Here, the superscripts n and $n + 1$ denote the current and next time-steps, respectively. \vec{U}^* is the intermediate velocity. By applying continuity equation, Eq. (1), on Eq. (16) we obtain the pressure Poisson equation:

$$-\nabla \cdot \left(\frac{\nabla p^{n+1}}{\rho^{n+1}} \right) = -\frac{1}{\Delta t} \nabla \cdot \vec{U}^* - \nabla \cdot \left(\frac{\vec{F}_{ST}^{n+1}}{\rho^{n+1}} \right) \quad (17)$$

The pressure at the next time-step, p^{n+1} , is the solution of Eq. (17), which is solved by using the Incomplete Cholesky Conjugate Gradient (ICCG) solver. The pressure Poisson step is the bottleneck of the overall algorithm and has been accelerated by porting it to Graphics Processing Units (GPUs) by Codyer et. al [50]. The pressure is used to compute \vec{U}^{n+1} using Eq. (16). The solver employs Message Passing Interface (MPI) parallelism [51].

3.1. Fluid-Structure Interaction Force

The FSI force, \vec{F}_S , appearing in the Eq. (2), is computed using the fast-fictitious domain method of Sharma and Patankar [17]. Eqs. (1) and (2) are first solved in the entire computational domain after setting $\vec{F}_S = 0$. This yields a divergence free velocity \vec{U}^{n+1} . The velocity in the solid domain does

not satisfy the rigidity condition at this stage. Rigid body translational and rotational velocities, \vec{U}_s and $\vec{\omega}_s$, respectively, are first obtained by conserving linear and angular momentum in the solid domain:

$$M_s \vec{U}_s = \int_{\psi} \rho_s \vec{U}^{n+1} dv \quad (18)$$

$$I_s \vec{\omega}_s = \int_{\psi} \vec{r} \times \rho_s \vec{U} dv \quad (19)$$

where M_s and I_s are mass and moment of inertia of the solid body, respectively. v is volume and \vec{r} is the position vector relative to the center of mass of the solid body. The moment of inertia tensor, I_s , is given by:

$$I_s = \int_{\psi} \rho_s [(\vec{r} \cdot \vec{r}) I - \vec{r} \otimes \vec{r}] dv \quad (20)$$

where I is the identity matrix. Numerical integration of Eq. (20) will be discussed later. Next, rigid body velocity is imposed inside the solid domain as:

$$\vec{U}_{RBM} = \vec{U}_s + \vec{\omega}_s \times \vec{r} \quad (21)$$

Applying Eq. (21) is equivalent to applying FSI force \vec{F}_S^{n+1} inside the solid domain at the next time level t^{n+1} , where \vec{F}_S^{n+1} is given by,

$$\vec{F}_S^{n+1} = \frac{\rho_s}{\Delta t} (\vec{U}_{RBM}^{n+1} - \vec{U}^{n+1}) \quad (22)$$

where, \vec{U}^{n+1} is the solenoidal velocity proposed by the flow solver in Eq. (16).

3.2. Center of mass and moment of inertia tensor of the solid body

The solid body's centroid is needed to evaluate both I_s and \vec{U}_{RBM} in Eqs. (20) and (21), respectively. We discuss here how the centroid is numerically calculated for a solid body which is represented in an Eulerian mesh by a scalar, or color function, ψ . We assume that the solid body has a uniform density, in which case the center of mass and the centroid coincide. Centroid \vec{r}_c of a solid body is computed by:

$$M_s \vec{r}_c = \int_{\psi} \rho_s \vec{r}_o dv \quad (23)$$

where, \vec{r}_o is the position vector relative to the origin of a fixed reference frame. The integration in Eq. (23) is performed over cells that are: (a) fully filled with solid phase, i.e. $\psi = 1$, and (b) partially filled with solid phase, i.e. $0 < \psi < 1$. The integration term in Eq. (23) can be written as:

$$\int_{\psi} \rho_s \vec{r}_o dv = \sum_{full\ cells} \rho_s \vec{r}_f V_{cell} + \sum_{partial\ cells} \rho_s \vec{r}_p \psi V_{cell} \quad (24)$$

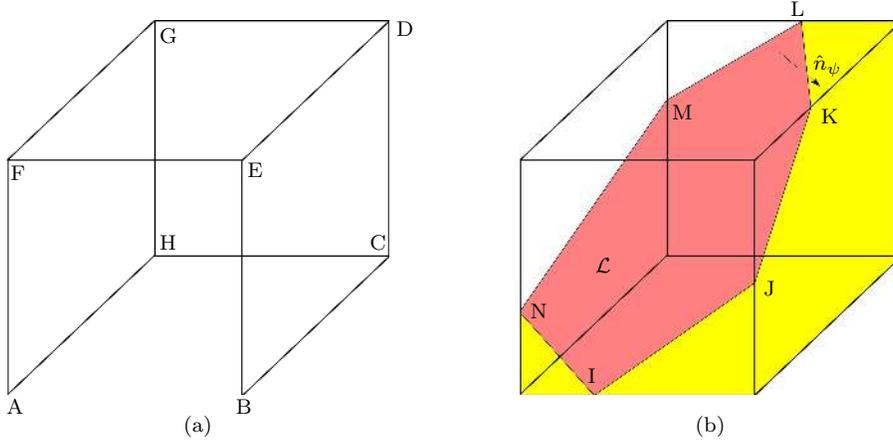


Figure 1: (a) A Cartesian computational cell. (b) Intersection between the computational cell and the plane IJKLMN, representing the solid surface. The volume of the cell occupied by the solid phase is shown in yellow.

Here, V_{cell} is the volume of the computational cell, \vec{r}_f is the cell center and \vec{r}_p denotes centroid of solid phase in a partial cell. Calculating \vec{r}_p will need a discussion here. Consider a computational cell shown in Fig. 1a. Suppose this cell contains the solid interface. In a Piecewise-Linear-Interface-Calculation (PLIC) scheme, this interface is approximated as a plane; see plane IJKLMN shown in red in Fig. 1b. The orientation of the plane is $\hat{n}_\psi = \frac{\nabla\psi}{|\nabla\psi|}$. The solid phase resides in the volume lying below this plane (in the direction of \hat{n}_ψ). This volume, shown in yellow in Fig. 1b, forms a polyhedron. Using the 3D geometric constructs presented in [40], one can construct this polyhedron. \vec{r}_p appearing in Eq. (24) denotes the centroid of this polyhedron. The centroid can be calculated

by decomposing the polyhedron into a number of tetrahedrons and computing the centroids of individual tetrahedrons. \vec{r}_p is then given by:

$$\vec{r}_p = \frac{\sum_j V_j \vec{r}_j}{\sum_j V_j} \quad (25)$$

where, V_j and \vec{r}_j are the volume and centroid position of the j^{th} tetrahedron and are given by:

$$V_j = \frac{1}{6} \begin{vmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \\ x_4 & y_4 & z_4 & 1 \end{vmatrix} \quad (26)$$

$$\vec{r}_j = \left(\frac{1}{4} \sum_{i=1}^4 x_i, \frac{1}{4} \sum_{i=1}^4 y_i, \frac{1}{4} \sum_{i=1}^4 z_i \right) \quad (27)$$

where (x_i, y_i, z_i) , $i = 1, 2, 3, 4$ represent the coordinates of the tetrahedron's vertices.

The inertia tensor calculation will follow a similar approach. The integration appearing in Eq. (20) is again performed over cells fully filled with solid and cells partially filled with solid separately.

$$\begin{aligned} I_s &= \int_{\psi} \rho_s [(\vec{r} \cdot \vec{r})I - \vec{r} \otimes \vec{r}] dv \\ &= \int_{full\ cells} \rho_s [(\vec{r} \cdot \vec{r})I - \vec{r} \otimes \vec{r}] dv + \int_{partial\ cells} \rho_s [(\vec{r} \cdot \vec{r})I - \vec{r} \otimes \vec{r}] dv \end{aligned} \quad (28)$$

The first term on the right hand side of Eq. (28) is due to the moment of inertia (M.I.) of cubes. Here, we are assuming the computational cells to be cubes. M.I. tensor $[I'_{ij}]$ of a cube, about the centroid \vec{r}_c of the solid body, is calculated by using the parallel axis theorem.

$$I'_{ij} = I_{ij} + M_{cube}(\delta_{ij}|\vec{a}|^2 - a_i a_j) \quad (29)$$

where M_{cube} is the cube's mass, δ_{ij} is the Kronecker delta function and $\vec{a} = \vec{r}_f - \vec{r}_c$, \vec{r}_f being the cube's centroid. I_{ij} is the cube's M.I. about its own

centroid, which is equal to $(1/6)M_{cube}\Delta x^2\delta_{ij}$. Δx is uniform cell size in each direction. The M.I. of the cube about the solid body's centroid is then given by:

$$I'_{ij} = \frac{1}{6}M_{cube}(\Delta x)^2\delta_{ij} + M_{cube}\delta_{ij}|\vec{a}|^2 - M_{cube}a_i a_j \quad (30)$$

Now, the second term appearing on the right hand side of Eq. (28) is due to several polyhedrons. To compute the M.I. of a polyhedron, we decompose it into a number of tetrahedron, calculate the M.I. of individual tetrahedrons and then add the individual M.I.s. Analytical formula provided by Tonon [52] allows us to compute the M.I.s of individual tetrahedrons analytically. The readers should be careful about a typo in the paper [52], where the expressions for the tensor elements b' and c' are swapped.

Next, we will present the numerical approach to transport mass/momentum and to compute the force terms appearing in the momentum equation (2).

3.3. Consistent mass and momentum transport

To handle large density ratios, for example $\sim 10^3$ or higher, special numerical schemes need to be employed to transport mass and momentum. At the fluid interface, the velocity is continuous, but due to sharp jump in density momentum is discontinuous. Therefore, a high-order or centered cell-edge estimates of the momentum near the interface can fail because the Taylor's series expansion of density is not valid in that region. Raessi [43] and Raessi and Pitsch [44] showed that a central difference estimate of density at cell-face will result in non-physical deformation of the interface only after a few time-steps into the simulation. The situation is exacerbated at large density ratios. To handle large density ratios we adopt the transport scheme proposed by Rudman [45]. Here we overlay a twice as fine grid on top of the regular grid as shown Fig. 2. The fine grid also has a staggered configuration; volume fractions (color functions) are defined at cell centres and the velocities are defined at cell faces. To avoid confusion, let us denote the color function at the regular grid by C and at the fine grid by c . For our purpose, the color function C can be either F or Ψ . Here,

we demonstrate a 2D example taken from [45], which has been extended to 3D in the present work. Let us denote by (U, V) the velocity at the regular grid and by (u, v) the velocity at the fine grid. The fine grid velocity components acquire their values from the regular grid:

$$\begin{aligned}
u_{2i+1/2,2j} &= U_{i+1/2,j} & v_{2i,2j+1/2} &= V_{i,j+1/2} \\
u_{2i+1/2,2j-1} &= U_{i+1/2,j} & v_{2i-1,2j+1/2} &= V_{i,j+1/2} \\
u_{2i-1/2,2j} &= 0.5(U_{i+1/2,j} + U_{i-1/2,j}) & v_{2i,2j-1/2} &= 0.5(V_{i,j+1/2} + V_{i,j-1/2}) \\
u_{2i-1/2,2j-1} &= 0.5(U_{i+1/2,j} + U_{i-1/2,j}) & v_{2i-1,2j-1/2} &= 0.5(V_{i,j+1/2} + V_{i,j-1/2})
\end{aligned} \tag{31}$$

The color function c is updated on the fine grid by solving Eqs. (8) and (9). In a three-phase configuration, e.g. solid/water/air, the transport equations (8) and (9) are solved using an error minimization based VOF approach presented by Pathak and Raessi in [40]. The color function C on the regular grid follows from c using the following expressions:

$$\begin{aligned}
C_{i,j} &= 0.25(c_{2i,2j} + c_{2i-1,2j} + c_{2i,2j-1} + c_{2i-1,2j-1}) \\
C_{i+1/2,j} &= 0.25(c_{2i+1,2j} + c_{2i,2j} + c_{2i+1,2j-1} + c_{2i,2j-1}) \\
C_{i,j+1/2} &= 0.25(c_{2i,2j+1} + c_{2i,2j} + c_{2i-1,2j+1} + c_{2i-1,2j})
\end{aligned} \tag{32}$$

The density on the regular grid are obtained from the color functions f and ψ .

$$\begin{aligned}
\rho_{i,j} &= f_{i,j}\rho_l + \psi_{i,j}\rho_s + (1 - f_{i,j} - \psi_{i,j})\rho_g \\
\rho_{i+1/2,j} &= f_{i+1/2,j}\rho_l + \psi_{i+1/2,j}\rho_s + (1 - f_{i+1/2,j} - \psi_{i+1/2,j})\rho_g \\
\rho_{i,j+1/2} &= f_{i,j+1/2}\rho_l + \psi_{i,j+1/2}\rho_s + (1 - f_{i,j+1/2} - \psi_{i,j+1/2})\rho_g
\end{aligned} \tag{33}$$

Next, we will focus on the advection of momentum. Figure 3 shows an x -momentum control volume (CV) located at $(i+1/2, j)$ and bounded by edges at (i, j) and $(i+1, j)$, respectively. We will consider the momentum flux advection at the $(i+1, j)$ edge of the CV during time step Δt . The momentum flux advection at the other three faces will follow the same suit. First the x -velocity component U at the right edge at $(i+1, j)$ is interpolated using van Leer's

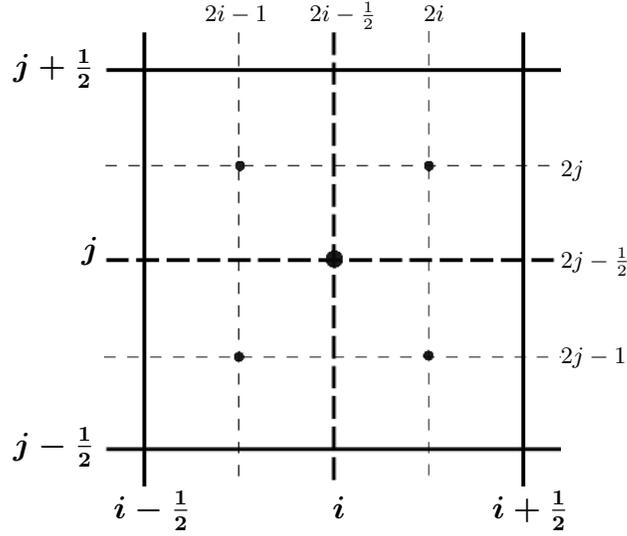


Figure 2: The regular grid cell (i, j) , shown in bold solid lines, is partitioned into four finer grid cells shown with bold dashed lines. Also shown are indices of the cells at the two grid sets. The figure is adopted from [45].

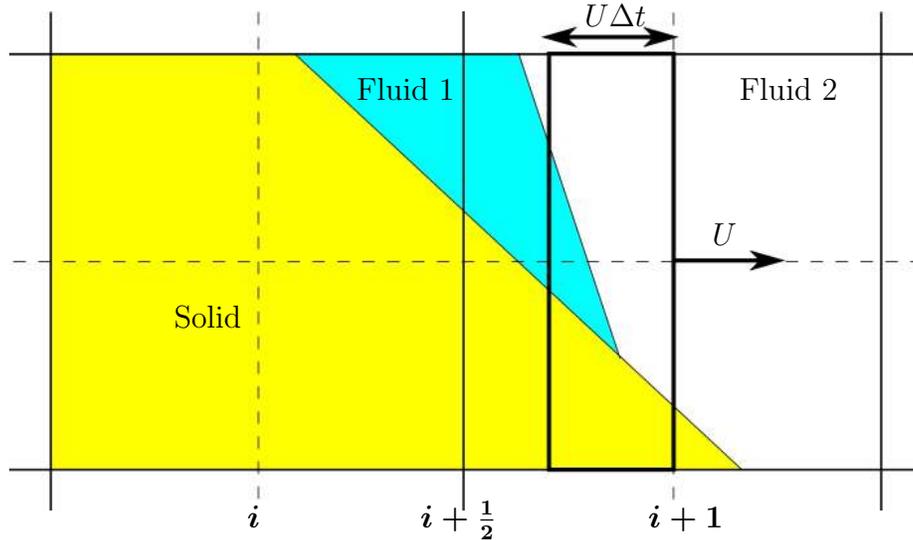


Figure 3: Momentum flux at the edge $(i+1, j)$ of the $(i+1/2, j)$ momentum CV is considered here. The volume flux of each phase is made available by the color function transport at the right edges of the fine grid cells $(2i+1, 2j)$ and $(2i+1, 2j-1)$. The yellow, blue and white areas in the bold rectangle correspond to the solid, fluid 1 and fluid 2 volume fluxes, respectively. The total mass flux, Δm , is obtained by multiplying the volume fluxes of individual phases by the corresponding densities and adding the results. Δm is used to compute the momentum flux at the edge $(i+1, j)$ of the x -momentum CV.

gradient [43, 53]. Near the interface, however, U is approximated by the upwind value. The volume flux of each phase at the right edge is obtained by adding the volume fluxes on the right faces of the fine grid cells at $(2i + 1, 2j)$ and $(2i + 1, 2j - 1)$. The total mass flux Δm at the right face of the momentum CV in time Δt is enclosed in the bold rectangle shown in Fig. 3. In the same figure, the volume fluxes (δV_s) , (δV_l) and (δV_g) of the solid phase and the two fluids, respectively, are represented by yellow, blue and white areas residing in the bold rectangle. The mass flux Δm is then given by:

$$\Delta m = (\delta V_s)\rho_s + (\delta V_l)\rho_l + (\delta V_g)\rho_g \quad (34)$$

The momentum flux donated by the momentum CV at $(i+1/2, j)$ to the momentum CV at $(i + 3/2, j)$ is then $\Delta m U$. Following the momentum flux advection, the x -velocity \tilde{U} located at $(i + 1/2, j)$ is obtained by dividing $(\rho U)_{i+1/2, j}$ by $\rho_{i+1/2, j}$. The density $\rho_{i+1/2, j}$ is obtained from Eq. (33), where the color functions are made available by F and Ψ at the fine grid. The same procedure is extended to the momentum CVs of the other velocity components. Because the same volume fluxes are used to update both mass and momentum, this procedure establishes a tight coupling between the two and their transports. That leads to a robust numerical scheme applicable to arbitrarily large density ratios. The efficacy of this scheme is demonstrated in Section 4.1.

In the above, we used the density $\rho_{i+1/2, j}$ to compute U after momentum advection. It is important to use the same density, computed from the fine grid color functions, in the pressure Poisson equation (17) and in Eq. (16) while applying the pressure. We have observed that using a density averaged as $(\rho_{i, j} + \rho_{i+1, j})/2$ in Eqs. (16) and (17) artificially produces large pressures just in front of an advancing interface, which tends to push it back resulting in non-physical interface topologies.

3.4. Viscosity

If $\rho u \Delta x / \mu \sim O(1)$ or less, the effect of viscosity becomes important. As presented in Eq. (11), we use a harmonic averaging approach [45, 54, 55] to compute the coefficient of dynamic viscosity in any cell. The volume fraction F in Eq. (11) is computed by adding the fine grid color functions as shown in Eq. (32). As an example let us consider Fig. 2. The volume fraction at regular grid location (i, j) is obtained from the fine grid locations $(2i, 2j)$, $(2i - 1, 2j)$, $(2i - 1, 2j - 1)$, $(2i, 2j - 1)$. Also, at regular grid corner $(i + 1/2, j + 1/2)$, the volume fraction is obtained from fine grid locations $(2i, 2j)$, $(2i + 1, 2j)$, $(2i + 1, 2j + 1)$ and $(2i, 2j + 1)$. The harmonic averaging approach avoids severe acceleration of the lighter, less viscous fluid in the vicinity of the interface, as noted by Rudman [45].

3.5. Surface Tension

Surface tension force is incorporated by implementing the balanced force algorithm of Francois et al. [56]. The algorithm imposes an exact balance between the pressure and surface tension, which is evident in Eqs. (16) and (17). The surface force term is given by:

$$\vec{F}_{ST} = \sigma \kappa \nabla f \quad (35)$$

where σ is the coefficient of surface tension between the fluids and κ is the fluid interface curvature:

$$\kappa = -\nabla \cdot \hat{n} \quad (36)$$

$$\hat{n} = \frac{\nabla f}{|\nabla f|} \quad (37)$$

In the discretized form, the above equations lead to the computation of interface curvature at the center of each cell. The numerical implementation of the balanced force algorithm for surface tension [56], however, requires the interface curvature at the face of each interfacial cell. To compute the face-centered curvature from the cell-centered values, a weighted averaging that is based on the volume fractions is performed, which is explained with the following 2D

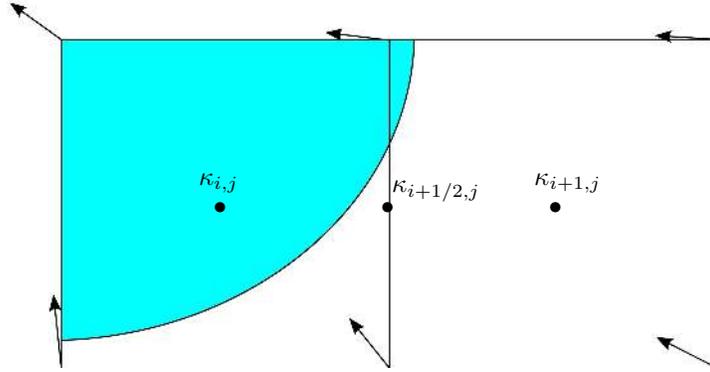


Figure 4: Shown are two adjacent cells at (i, j) and $(i + 1, j)$. Curvatures $\kappa_{i,j}$ and $\kappa_{i+1,j}$ are defined at the cell centers. The curvature $\kappa_{i+1/2,j}$ at the cell face is computed by taking a weighted average of $\kappa_{i,j}$ and $\kappa_{i+1,j}$.

example. Consider a 2D, two-fluid configuration illustrated in Fig. 4, which also shows the interface normal vectors located at the cell vertices along with the cell-centered curvatures, $\kappa_{i,j}$ and $\kappa_{i+1,j}$. To obtain the face-centered curvature, $\kappa_{i+1/2,j}$, the following weighted averaging is used:

$$\kappa_{i+1/2,j} = (\omega_1 \kappa_{i,j} + \omega_2 \kappa_{i+1,j}) / (\omega_1 + \omega_2) \quad (38)$$

where the weighting coefficients are a function of the volume fraction, F , as given below

$$\begin{aligned} \omega_1 &= 1 - |(2F_{i,j} - 1)|^3 \\ \omega_2 &= 1 - |(2F_{i+1,j} - 1)|^3 \end{aligned} \quad (39)$$

The weighting coefficient function is plotted in Fig. 5 against the volume fraction F . As can be seen in Fig. 5, by performing such weighted averaging, those cells with a volume fraction closer to 0.5 will contribute more to the calculation of face-centered curvatures. In other words, the cell-centered curvature of a cell that resides closer to the fluid interface is given more importance than of those located further. We have found in our tests, not shown here, that this approach yields better estimates of curvature, and consequently smaller magnitudes of “spurious currents” near the interface. The extension of the

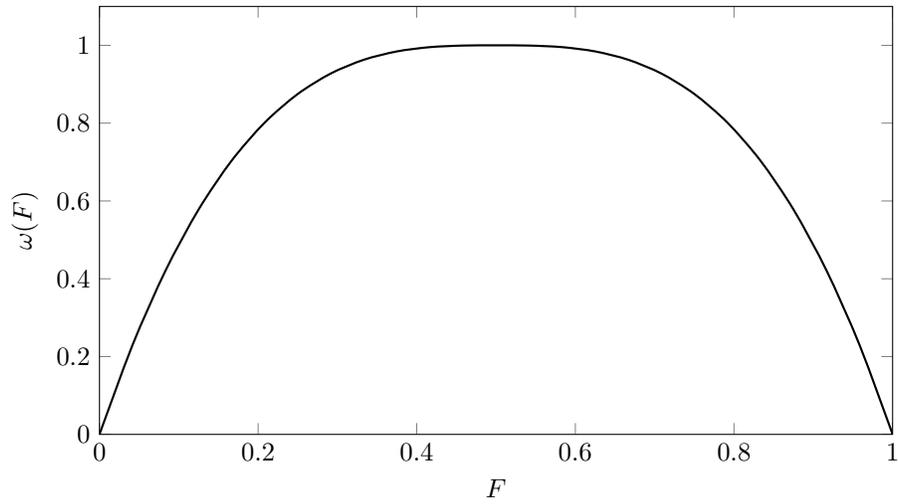


Figure 5: The weighting coefficient function $\omega(F) = 1 - |(2F - 1)^3|$ plotted against the volume fraction F . The weighting function is used in computing an averaged face-centered curvature via Eq. (38) and gives more weight to cell-centered curvatures located closer to the interface.

above weighted averaging scheme to 3D is straightforward and was used in the present work.

4. Results

We now present an array of test cases which will assess the performance of the proposed methodology. All the test cases except the first one employ air/water as the two fluids. The water (subscript l) and air (subscript g) properties that were used in the simulations are: $\rho_l = 1000 \text{ kg/m}^3$, $\rho_g = 1.226 \text{ kg/m}^3$, $\mu_l = 1.137 \times 10^{-3} \text{ kg/ms}$, $\mu_g = 1.78 \times 10^{-5} \text{ kg/ms}$, $\sigma = 0.0728 \text{ N/m}$. Gravitational acceleration g is 9.81 m/s^2 .

4.1. A dense drop moving in an ambient fluid

This test case was adopted from [44] to demonstrate the efficacy of the consistent mass and momentum advection in simulations involving large density ratios. We consider a spherical drop of fluid 1 with a radius of 0.1 initially positioned at (0.25, 0.5, 0.5) in a $1 \times 1 \times 1$ domain otherwise filled with fluid 2. The computational domain is discretized into $128 \times 128 \times 128$ cells. Surface tension, fluid viscosities and body forces in the momentum equation, Eq. 2, are set to zero, and the density ratio is $\rho_1/\rho_2 = 10^6$. The drop is given an initial velocity of (1, 0, 0) and its motion is simulated for a period of 0.5. The drop should remain undeformed due to the large density ratio.

Simulations were performed using both the present consistent method and a non-conservative formulation (see Raessi and Pitsch [44]). Figure 6a shows the initial drop. Figures 6b and 6c show the drop at the end of the simulation by a non-conservative formulation and the present method, respectively. The non-conservative formulation causes non-physical deformation of the drop only after few time steps due to error accumulation in the momentum transport. The issue is exacerbated at large density ratios. The present consistent scheme is well able to preserve the drop shape.

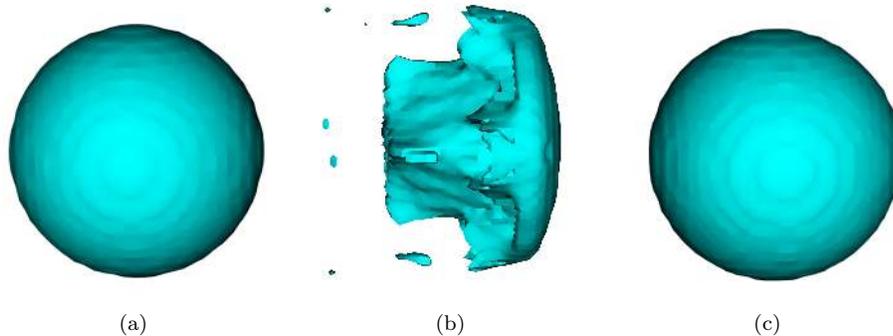


Figure 6: (a) Initial drop of a dense fluid which is given an initial velocity of $(1, 0, 0)$ and simulated for a period of 0.5. The same drop at the end of a simulation performed by using (b) a non-conservative formulation exhibiting non-physical deformations, and (c) the consistent mass and momentum transport scheme where the drop shape is preserved. Density ratio= $10^6 : 1$.

4.2. Dambreak: Collapsing water column

The present test case has been adopted from the experiments performed by Martin and Moyce [57]. Here, a water column collapses under the influence of gravity. This test case has been used by many to assess the performance of flow solvers in the context of various methods, including VOF [58–62], Smoothed Particle Hydrodynamics (SPH) [63, 64], Level Set [65, 66], Moving Particle Semi-Implicit [67, 68], and others [69–71]. The initial width and height of the dam are the same and equal to $a = 0.05715 \text{ m}$. The computational domain is $60 \text{ cm} \times 10 \text{ cm} \times 10 \text{ cm}$. The snapshots of the simulation are shown in Fig. 7.

Figure 8 illustrates the non-dimensional position z/a versus non-dimensional time $\tau = t\sqrt{g/a}$, where z is the distance of the surge front from the left wall. Simulations were performed at four resolutions and their results are plotted against the experimental data of Martin and Moyce [57] in Fig. 8. It was noted by Martin and Moyce [57, p. 314] that it was not possible to record the exact time of the onset of motion, which presents an uncertainty with $\tau = 0$. This uncertainty was also recognized by other researchers, e.g., in [67, p. 1122], who attributed the discrepancy between the numerical result and the experimental

data [57] to “the difficulty in realizing the ideal initial condition of infinitely sudden removal of front barrier”. To resolve this issue, the first experimental data point at $\tau = t\sqrt{g/a} = 0.43$ was identified as the “reference point”. Then, only for the highest mesh resolution, the simulation data curve was shifted in time just enough to establish a match between the reference point at $\tau = 0.43$ and the computational data point at this mesh resolution. Next, exactly the same amount of time shift, which corresponds to the data of the highest mesh resolution, was applied to the computational results at coarser resolutions. That means that the simulation data curves of the coarser mesh resolutions would not necessarily match the experimental data point at $\tau = 0.43$, i.e., the reference point.

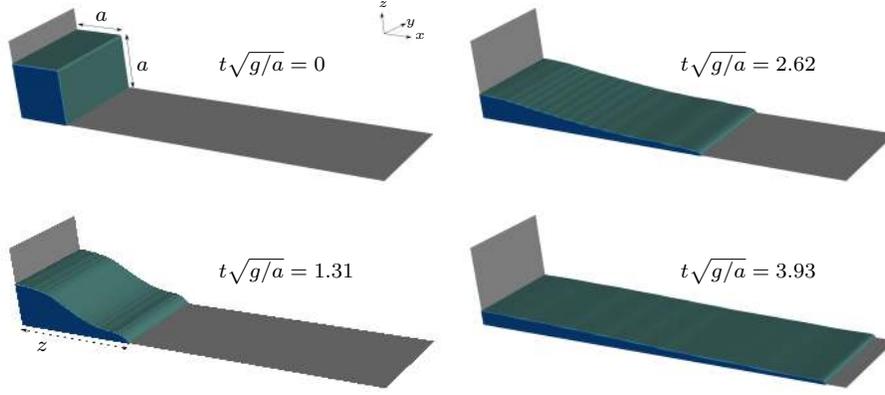


Figure 7: Snapshots of the simulated collapsing water column.

Taking the computational result at the highest resolution, $\Delta x = 1/2560$, as the benchmark, the error in the non-dimensional surge front position, $\tilde{z} = z/a$, which is denoted by $E_{\tilde{z}}$, was computed using Eq. (40),

$$E_{\tilde{z}} = \frac{1}{\tau_f - \tau_0} \int_{\tau_0}^{\tau_f} |\tilde{z}_{num.} - \tilde{z}_{bench.}| d\tau \quad (40)$$

where, $\tilde{z}_{num.}$ and $\tilde{z}_{bench.}$ are the numerical and benchmark dimensionless surge front positions, respectively. The calculation of error was carried out between non-dimensional time of $\tau_0 = 0.43$ and $\tau_f = 6.4$. The reason for using $\tau_0 = 0.43$ (as opposed to using $\tau_0 = 0$) was the time shift explained above. The error, $E_{\tilde{z}}$,

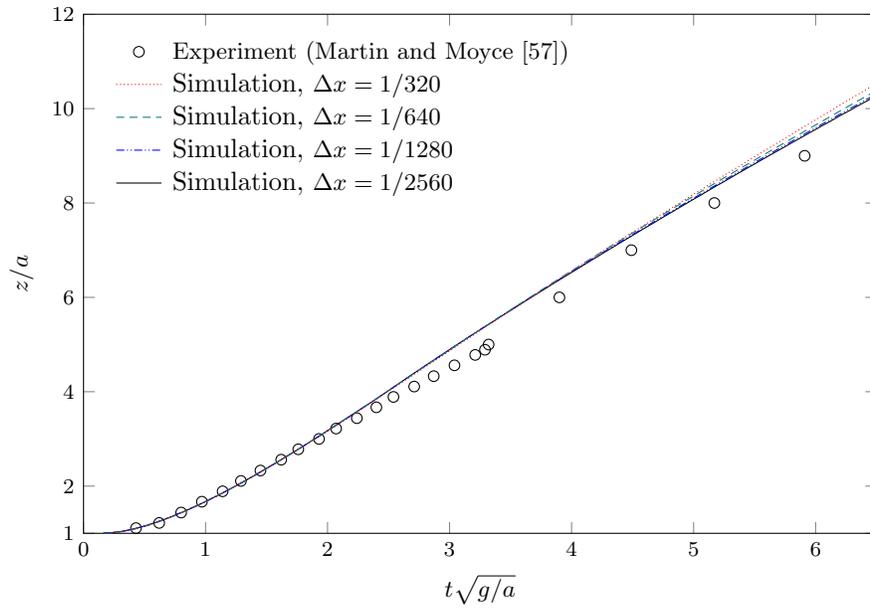


Figure 8: [The dimensionless surge front position versus dimensionless time in the collapsing water column. Simulated results are plotted against the experimental data of Martin and Moyce \[57\].](#)

is presented in Table 1 at each mesh resolution. A first-order error convergence is seen.

Δx	$E_{\bar{z}}$	order
1/320	5.80×10^{-2}	
		0.74
1/640	3.47×10^{-2}	
		1.10
1/1280	1.62×10^{-2}	

Table 1: Error $E_{\bar{z}}$, defined in Eq. (40), associated with the non-dimensional surge front z/a of the collapsing water column at different mesh resolutions.

4.3. Dambreak: Water column impact with an obstacle

Here, we present a dambreak problem where the collapsing water column impacts a stationary obstacle. The free water interface undergoes large deformation. The test case has been taken from Kleefsman et al. [46]. The experiments were performed at Maritime Research Institute Netherlands (MARIN). Figure 11a shows the initial setup. The dimension of the water tank is $3.22 \text{ m} \times 1 \text{ m} \times 1 \text{ m}$. The dimensions of the initial water column and the obstacle are shown in Fig. 9. The water column is released at $t = 0$, which travels towards the other end of the tank due to gravity. Figure 11 shows snapshots of the numerical simulation at various times. Also shown is a slice at $y = 0.475$. The simulation was performed for 6 s and a uniform finite volume grid consisting of $161 \times 50 \times 50$ cells was used.

In the experiments, sensors were located in the water tank to measure water height and pressure. The locations of the height sensors (H1-H4) are shown in Fig. 9, and of the pressure sensors (P1-P8) are shown in Fig. 10. The simulated height using the present method is compared with experiments at H2 and H4, and is shown in Fig. 12. Also shown in the plots are the simulation results of Kleefsman et al. [46]. Overall there is a good agreement between the present simulation and the experiments. Present simulation is well able to capture the

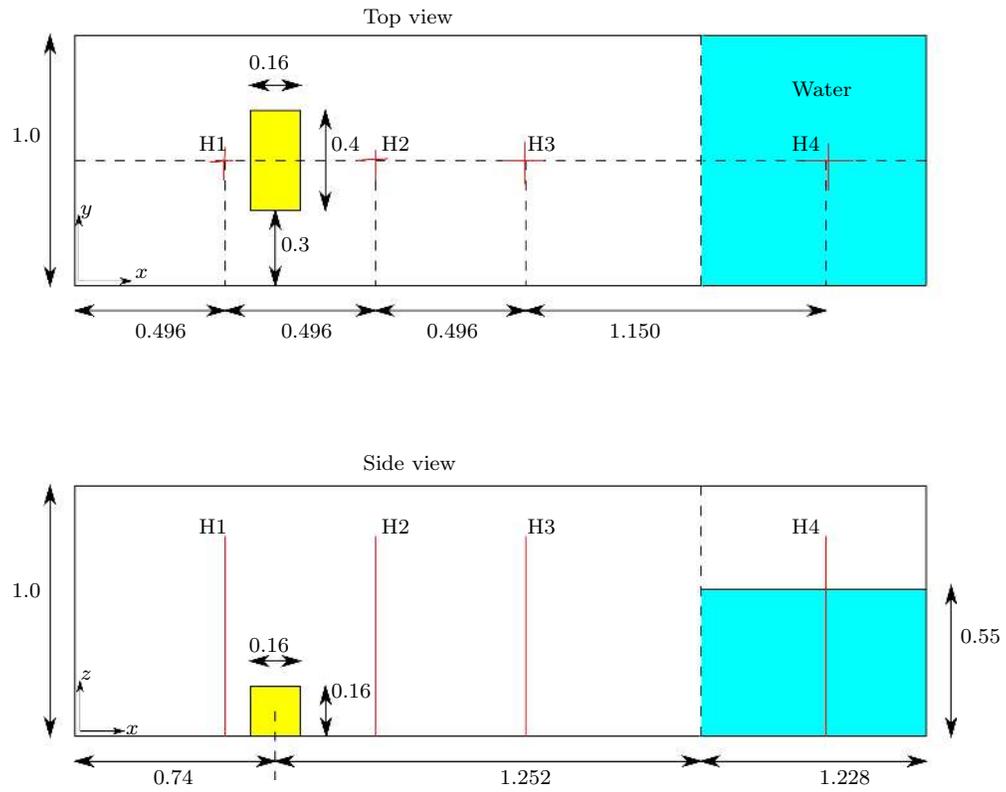


Figure 9: The dimensions of the initial water column (in blue) and the obstacle (in yellow) are shown. Also shown the location of the water height sensors. Extracted from [72].

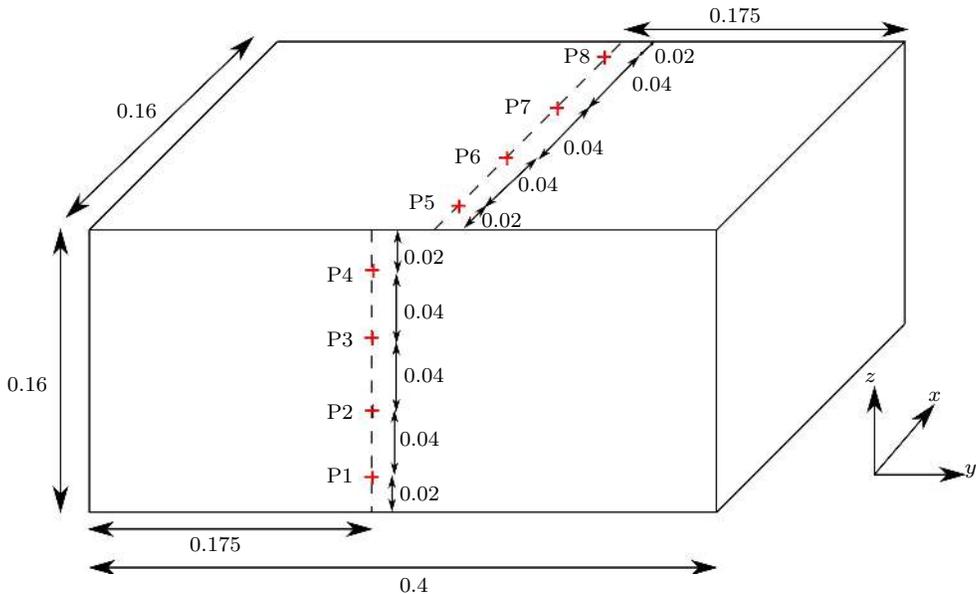
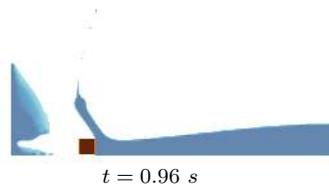
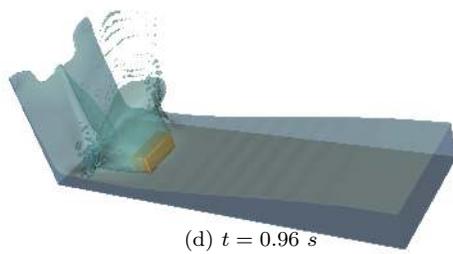
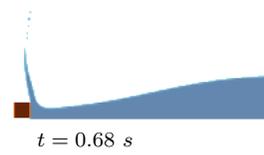
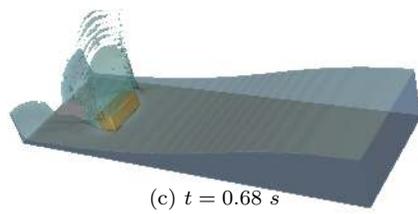
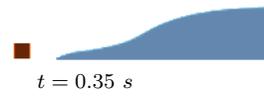
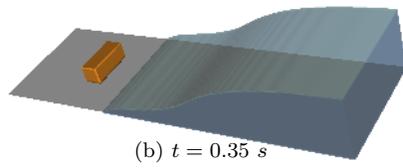
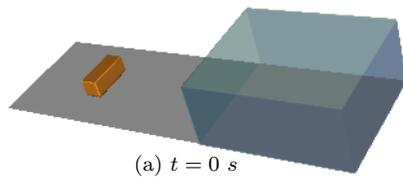


Figure 10: Location of pressure sensors on the obstacle. Extracted from [72].

oscillations in the water height.

Figure 13 compares the simulated pressure against experiments at P1, P3, P5 and P7. P1 and P3 sensors are located at the front face. The peak pressure is well captured by the simulations at P1, but underpredicted at P3. The wave returning at 4.7 s is evident by the pressure jump in both experiment and the simulations. Locations P5 and P7 lie on the top face of the obstacle. A few numerical spikes are visible in the pressure history of P5 and P7, both in the present simulation and that of Kleefsman et al. [46]. The present simulation shows a discrepancy for P7 pressure at $\sim 2\text{ s}$. We found that the momentary dip in the simulated pressure is due to an air void passing over the P7 location. Overall there is a fair agreement between the simulations and the experiments.



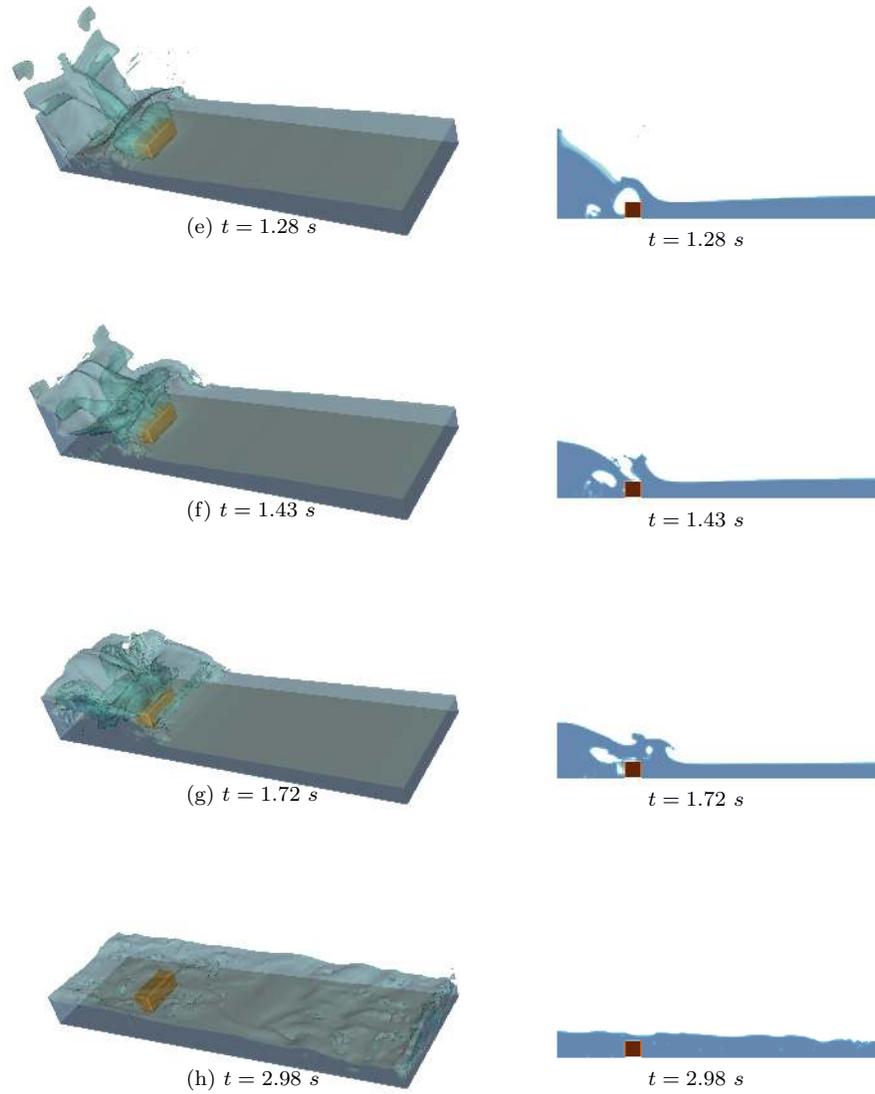


Figure 11: Computational simulation results of a water column released at $t = 0$ impacting a stationary obstacle. Also shown is a slice at $y = 0.475$.

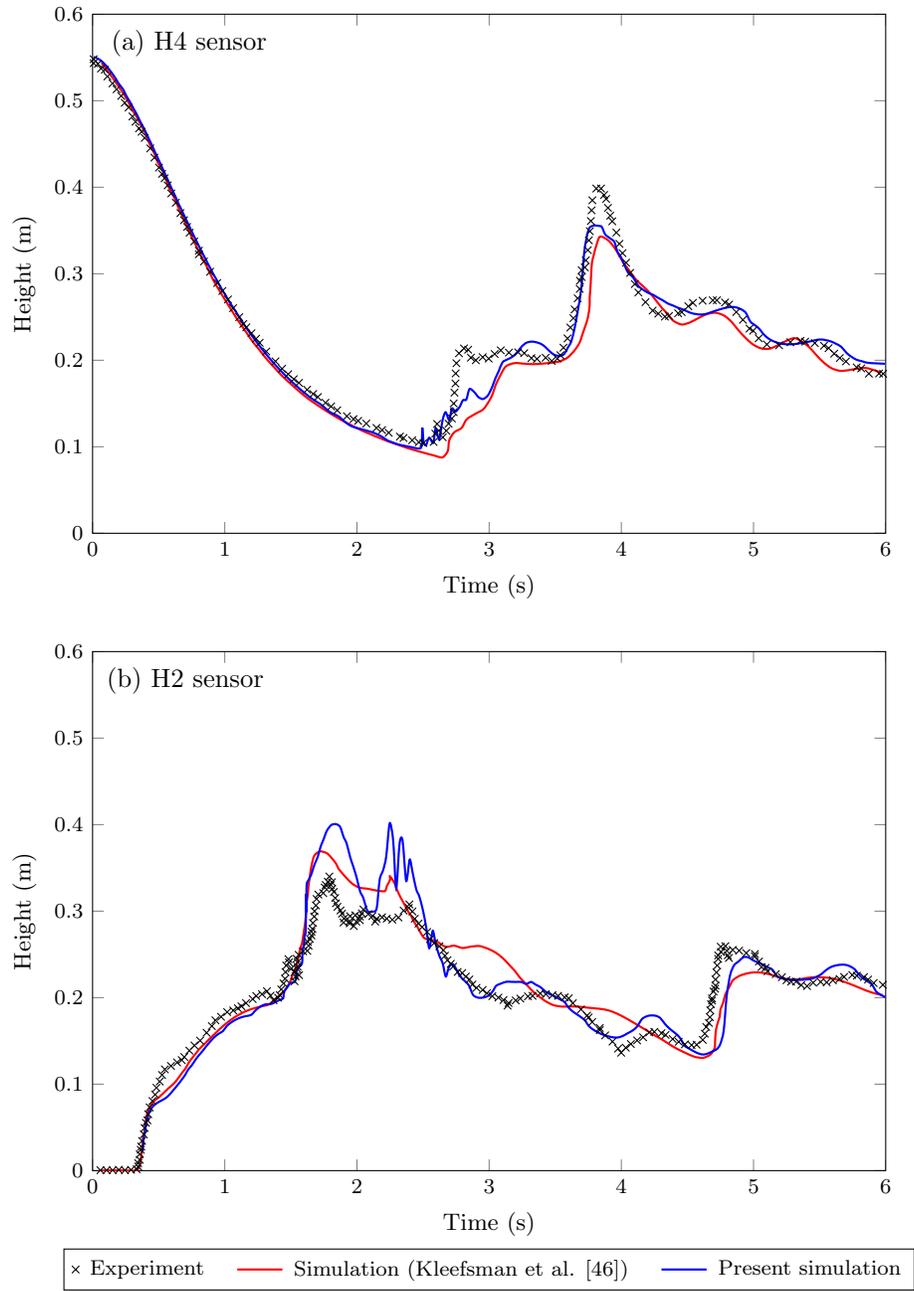
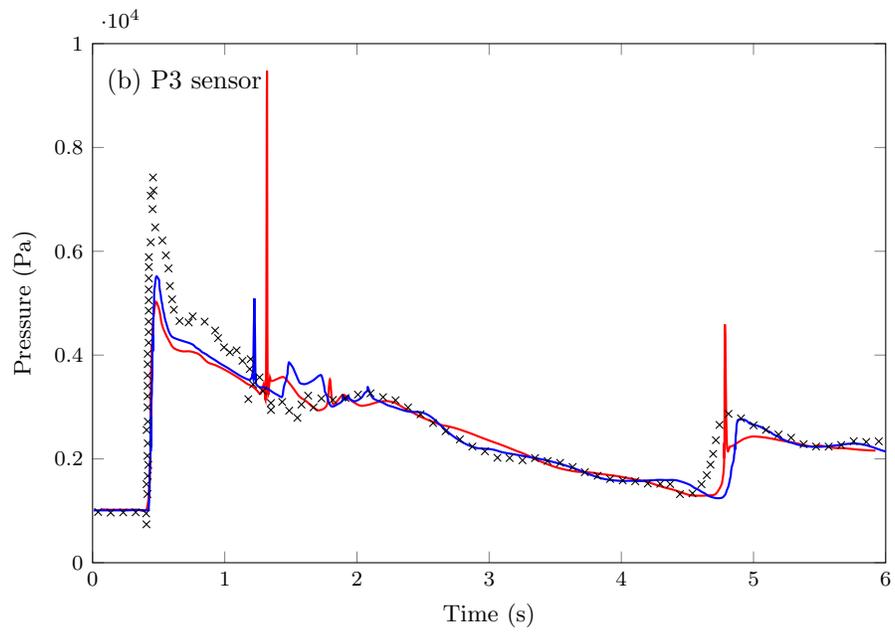
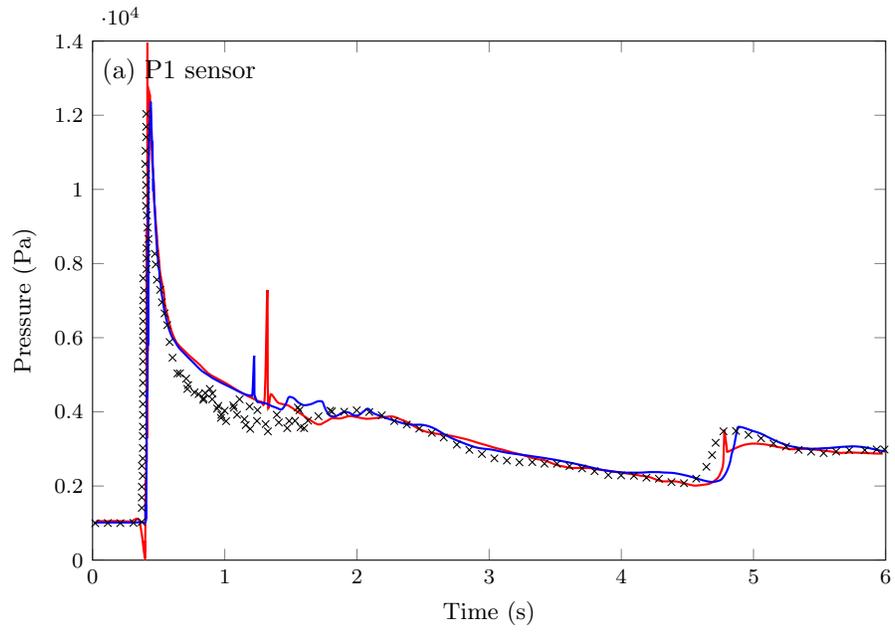


Figure 12: Water height history measured at (a) H4 and (b) H2 locations shown in Fig. 9.



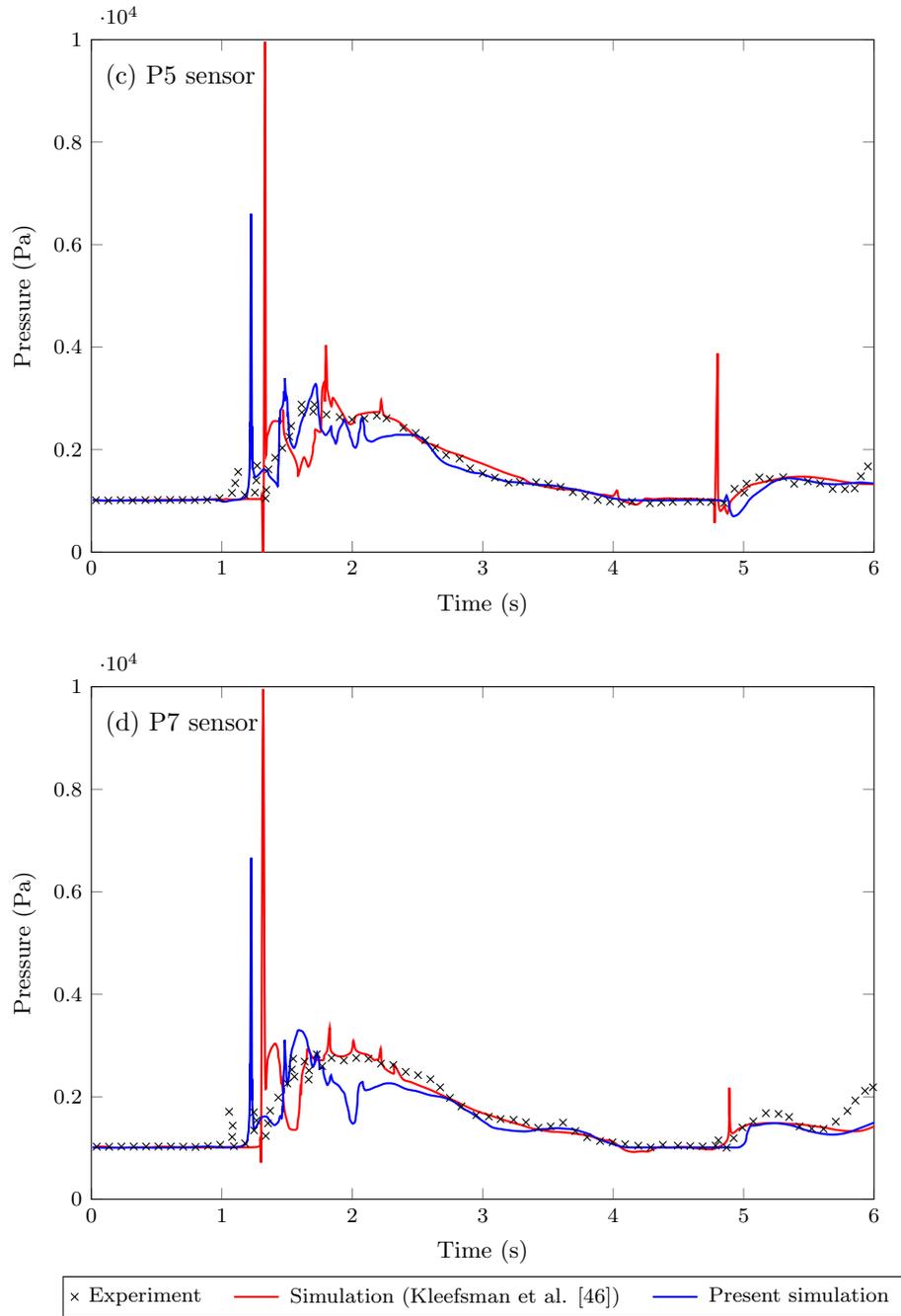


Figure 13: Pressure time history at locations: (a) P1, (b) P3, (c) P5 and (d) P7, depicted in Fig. 10

4.4. Free heave of a cylinder

Free heave of a cylinder was studied experimentally by Ito [73]. A cylinder of radius $a = 0.0762\text{ m}$ with a length of 2.5908 m was released into a rectangular water tank from $z_0 = 0.0254\text{ m}$ above the free surface. The density of the cylinder material was 500 kg/m^3 . The numerical water tank's length and width were set as 12.192 m and 2.5908 m , respectively. The water depth in the tank was 1.2192 m . Since the cylinder is half as dense as water, its equilibrium position aligns with the initial free surface of the water at $z = 0$. After being released into the tank, the cylinder starts performing heaving oscillations. The amplitude of the oscillations reduces over time as energy is radiated away from the cylinder via waves, which can be seen in Fig. 14.

Four simulations were performed using mesh resolutions of 10, 20, 40, and 80 cells per radius (CPR) of the cylinder. At each resolution and throughout the heave oscillations, the vertical position of the cylinder's centroid was tracked in time. The results are shown in Figure 15 in a dimensionless form, along with the experimental result by Ito [73]. The agreement between the computational results and the experimental data is very good, except at the coarsest resolution of 10 CPR. Using the computational result at the finest resolution, 80 CPR, as the benchmark, the error associated with the vertical position of the cylinder, which is denoted by $E_{\tilde{z}}$, was computed at each resolution by following Eq. (40), where $\tilde{z} = z/z_0$, $\tau = t\sqrt{g/a}$, $\tau_0 = 0$, and $\tau_f = 35$. The error $E_{\tilde{z}}$ is presented in Table 2 at different mesh resolutions and shows first-order convergence.

resolution	$E_{\tilde{z}}$	order
10 CPR	4.91×10^{-2}	
		1.19
20 CPR	2.15×10^{-2}	
		1.54
40 CPR	7.38×10^{-3}	

Table 2: Error $E_{\tilde{z}}$, defined in Eq. (40), associated with the vertical position of the heaving cylinder at different mesh resolutions.

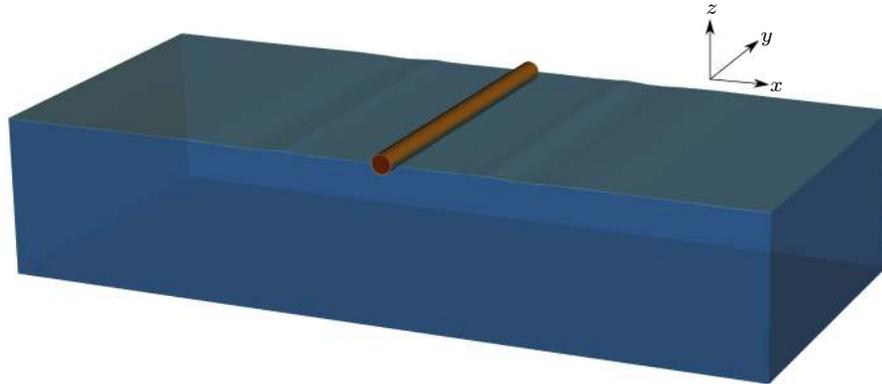


Figure 14: The heaving cylinder shown at the dimensionless time $t\sqrt{g/a} = 24.37$, which was released from an initial height. Waves radiating away from the cylinder are visible.

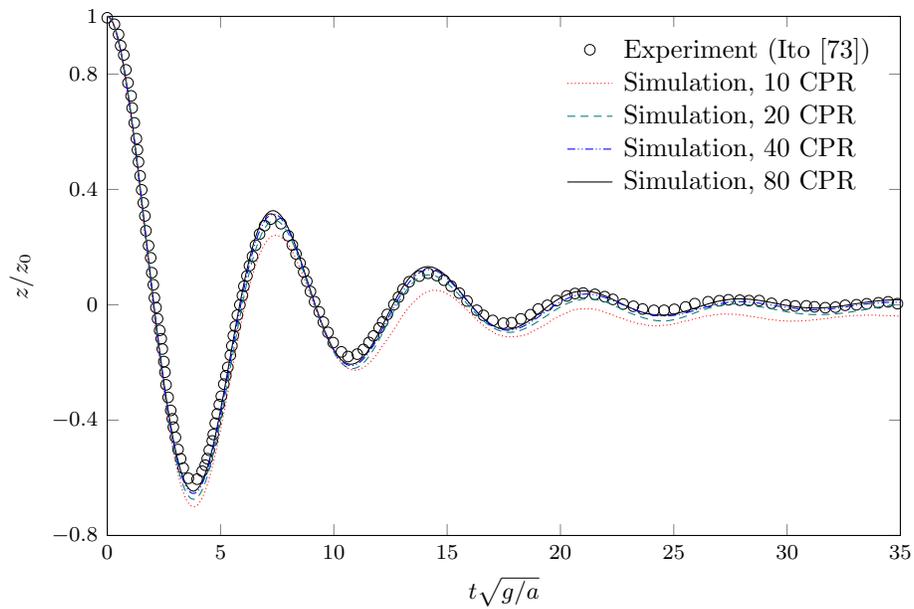


Figure 15: Vertical position of a freely heaving cylinder released from an initial height.

Using the potential flow theory, Maskell and Ursell [74] studied the free heave of a cylinder and developed an analytical model for the heave response. Figure 16 shows a comparison between the theoretical model [74], experiments by Ito [73], and our simulation results with the finest mesh resolution of 80 CPR. Note that the theoretical model [74] assumes small amplitudes of oscillations, i.e., a small initial height, while the initial height of the cylinder in the experiments [73] and our simulations was $a/3$. Despite the apparent difference in the range of the initial height, the theoretical response shows a good agreement with the simulated and experimental responses. That can suggest that the flow in the experiments and simulations is still close to the linear regime, where the nonlinear and viscous effects still remain small. As seen in Fig. 16, the theory [74] slightly over-predicts the heave response possibly because it neglects these small nonlinear and viscous effects.

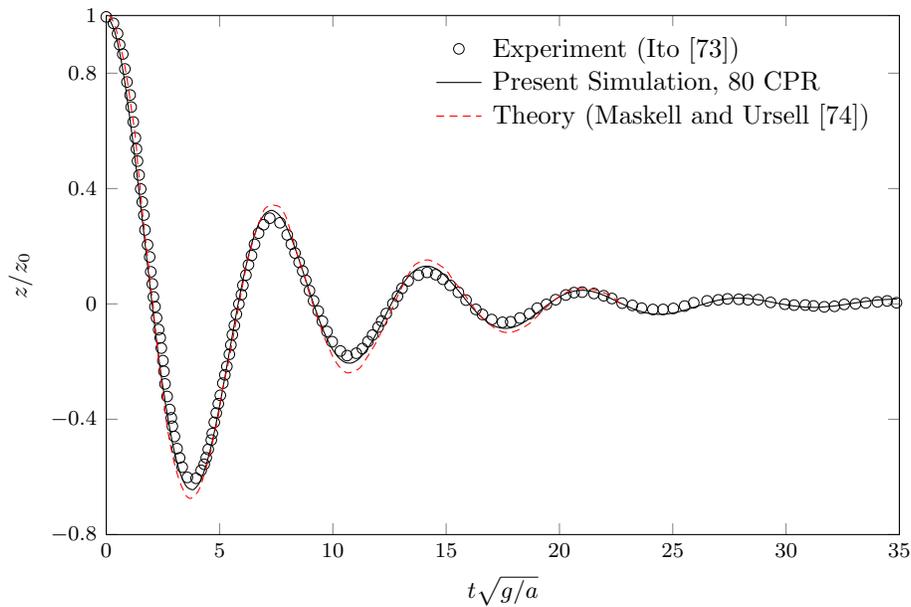


Figure 16: Vertical position of a freely heaving cylinder released from an initial height. Comparison is shown between the responses obtained from potential flow theory [74], present simulation and experiments [73].

4.5. Free roll of a rectangular cuboid barge

We now test the accuracy of the proposed method in capturing rotational motion of a rigid body. The test case presented here is based on experiments conducted by Jung et al. [75]. A rectangular cuboid barge 0.9 m long, 0.3 m wide and 0.1 m high is half submerged in a water tank. The water tank is 12 m long and 0.9 m wide; the water depth is 0.9 m. The barge is only allowed to rotate about an axis parallel to the y -axis and passing through the centroid of the barge. The density of the barge is 1049 kg/m^3 and is given an initial inclination of 15° as shown in Fig. 17a. The torque generated due to hydrodynamic forces causes roll motion in the barge. The roll amplitude decreases over time as the energy is damped and radiated away via waves.

Figure 17 shows the snapshots of the rolling barge at different times. The barge motion was simulated at four mesh resolutions that resolved the barge width (0.3 m) by 30, 60, 120 and 240 cells, respectively. Figure 18 shows the time history of the barge inclination angle obtained by our simulations at different mesh resolutions, by a simulation performed by Calderer et al. [76], and by the experiment of Jung et al. [75]. The simulations, both ours and that of Calderer et al. [76], overpredict the inclination angle compared to the experiment. However, the present simulation agrees very well with the simulation performed by Calderer et al. [76]. As argued by Calderer et al. [76], the suppressed response in the experimental results is attributed to the specific damping of the experimental apparatus, which was not known *a priori*, and was not taken into account in the computational simulations. Calderer et al. [76] performed additional simulations by adding an artificial damping and verified the above argument, which was not done in the present study.

Using the results of the finest resolution, 240 cells per width (CPW) of the barge, as the benchmark, the error associated with the inclination angle, which

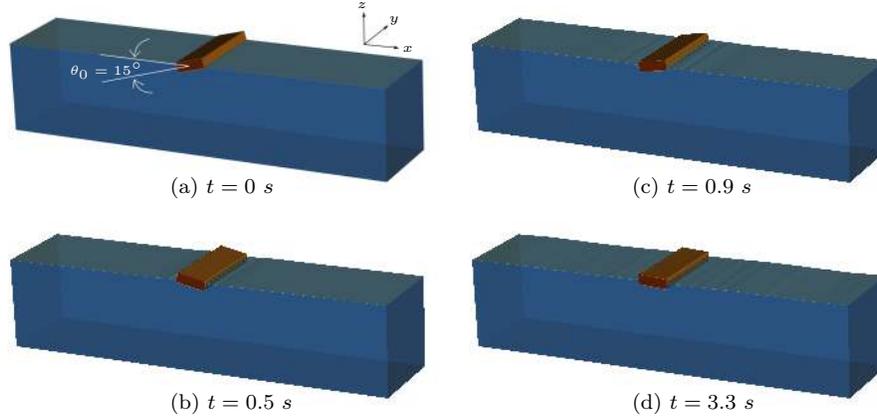


Figure 17: Snapshots of a rolling rectangular barge released from an initial inclination angle of 15° .

is denoted by E_θ , was calculated via Eq. (41) for each mesh resolution.

$$E_\theta = \frac{1}{t_f - t_0} \int_{t_0}^{t_f} |\theta_{bench.} - \theta_{num.}| dt \quad (41)$$

where, $t_0 = 0$ s, $t_f = 3.5$ s, $\theta_{bench.}$ and $\theta_{num.}$ are, respectively, the benchmark and simulated inclination angle (in degrees) of the barge. Table 3 reports the error E_θ at various mesh resolutions. As the mesh is refined, first-order convergence is observed. It would appear from the error table that the results obtained at 30 and 60 CPW are almost equally close to the benchmark. However, by closely observing Fig. 18, it is seen that two different types of errors are present in each case: the results at 30 CPW suffer from phase shift errors, which are most visible after 2.5 s, while the 60 CPW results exhibit errors mainly in the amplitude of oscillations.

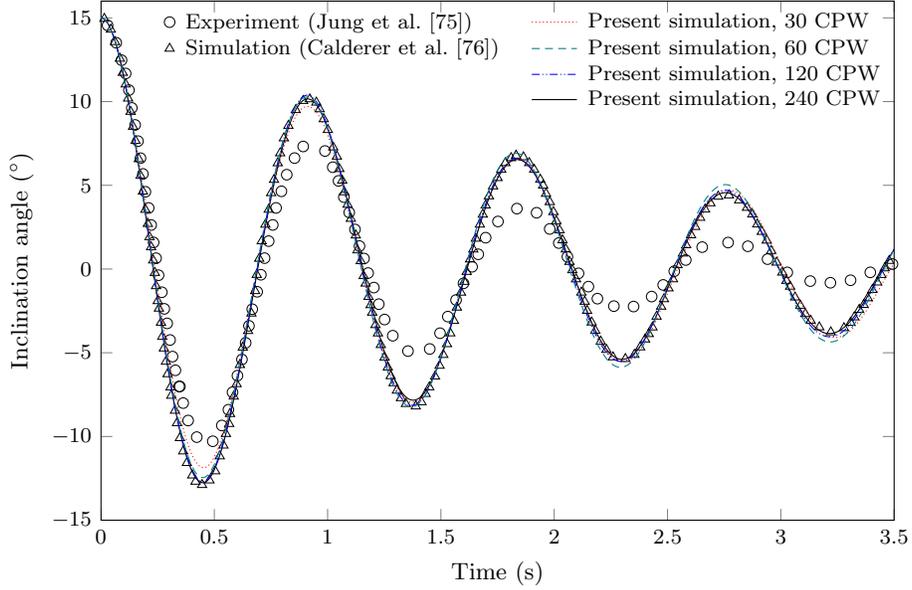


Figure 18: Time history of the inclination angle of the rolling rectangular cuboid barge, shown in Fig. 14, obtained by the present methodology at 30, 60, 120 and 240 cells per width (CPW) of the barge, and comparison with the experiments conducted by Jung et al. [75] and the computational simulation performed by Calderer et al. [76].

resolution	E_θ	order
30 CPW	3.12×10^{-1}	0.03
60 CPW	3.05×10^{-1}	1.19
120 CPW	1.33×10^{-1}	

Table 3: Error E_θ , defined in Eq. (41), associated with the inclination angle in the free roll motion of a rectangular cuboid barge at different mesh resolutions.

4.6. Free heave of a sphere

Next, we present the decaying heave motion of a spherical buoy. This problem has been studied experimentally, e.g., in [77, 78]. We adopted the experiments of Beck and Liapis [78], where a sphere with a radius of $a = 0.254 \text{ m}$ is released from an initial height of z_0 above the free water surface. The density of the sphere is 500 kg/m^3 . Therefore the equilibrium position coincides with the initial free surface location, $z = 0$. The experiments were performed in the Ship Hydrodynamics lab at the University of Michigan. The towing tank's specifications available at [79] were used for our numerical tank. Both length and width of the numerical tank were set at 13 m . The water depth was set at 3.2 m . After being released from the initial position, the sphere begins to perform oscillations, the amplitude of which dissipates over time. Simulations using the proposed methodology were performed for $z_0 = a/3$. Figure 19 shows a snapshot of the computational result at a dimensionless time of $t\sqrt{g/a} = 21.7$. The time history of the heaving sphere's centroid is presented in Fig. 20. The figure shows the results obtained from two simulations performed at the resolutions of 10 and 20 cells per radius (CPR) of the sphere. A comparison between the computational results and the experimental [78] and analytical [80] results are also presented in Fig. 20. The 10 CPR resolution does not capture the oscillation peaks well, but at the 20 CPR resolution, the agreement with the experimental and analytical results improves significantly.

It should be noted that a comparison between the experiments and simulations might not be exactly proper here because of the difference in the initial height of the sphere. The initial height in the experiments by Beck and Liapis [78] was $a/10$ in contrast to $a/3$ chosen in our simulations. The reason for such a choice was computational feasibility, because resolving a small initial height of $z_0 = a/10$ with an adequate number of grid points (e.g., six) leads to vast memory requirements. Nevertheless, as seen in Fig. 20, despite the larger initial height, the numerical solutions still follow the experimental and analytical responses. The experimental response seems to match well the analytical response model of Damaren [80], which assumes small amplitudes of oscillations and a

potential flow in the linear regime. From the results presented in Fig. 20, it can be argued that a larger initial height of $z_0 = a/3$ used in the simulations still leads to a linear flow regime.

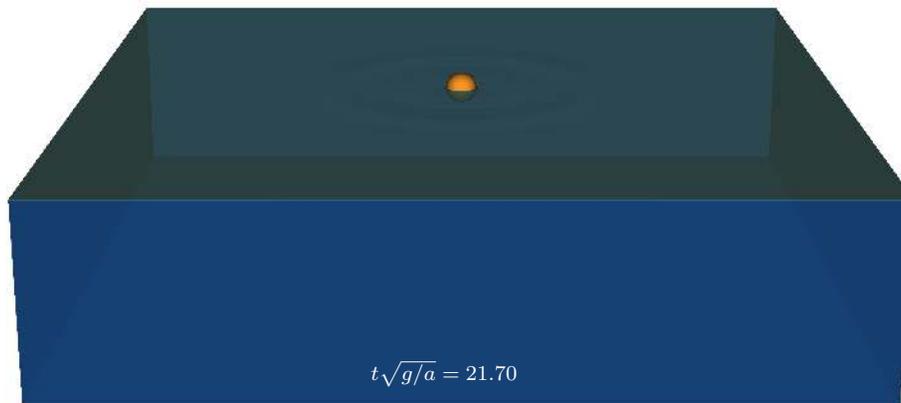


Figure 19: A snapshot of the computational result for the heave oscillations of a sphere released from an initial height.

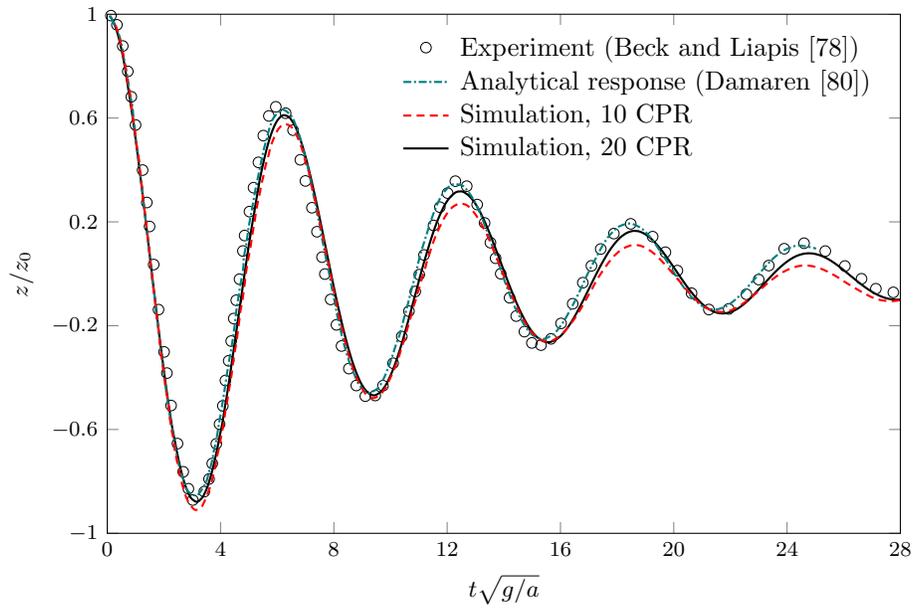


Figure 20: Dimensionless vertical position of a freely heaving sphere released from an initial height, z_0 , obtained from the present simulations at two resolutions of 10 and 20 cells per radius (CPR), and the comparison with the experimental and analytical results.

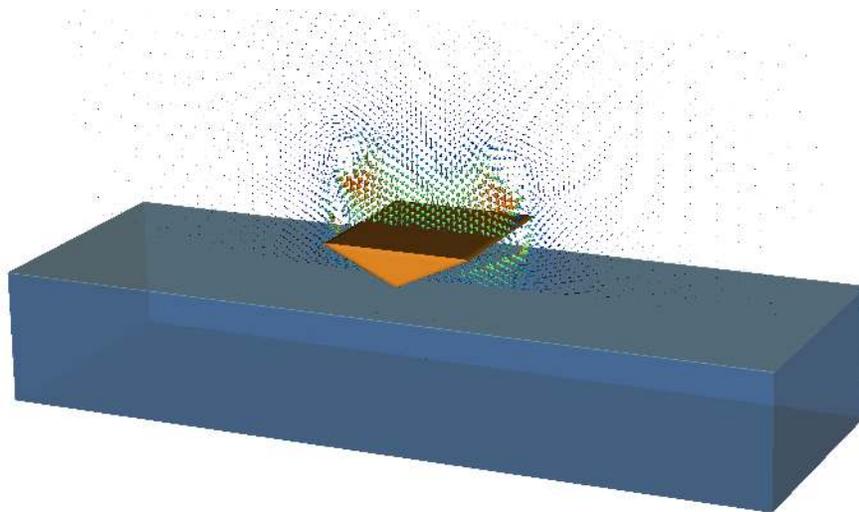
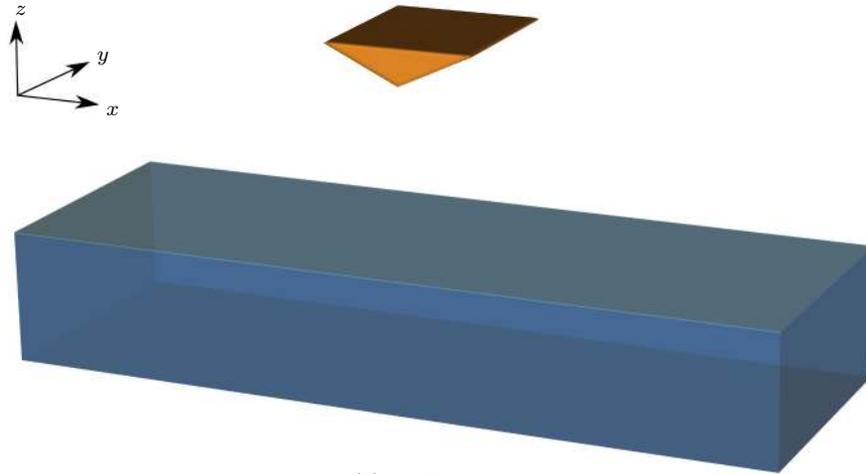
4.7. Water entry of a freely falling 3D wedge

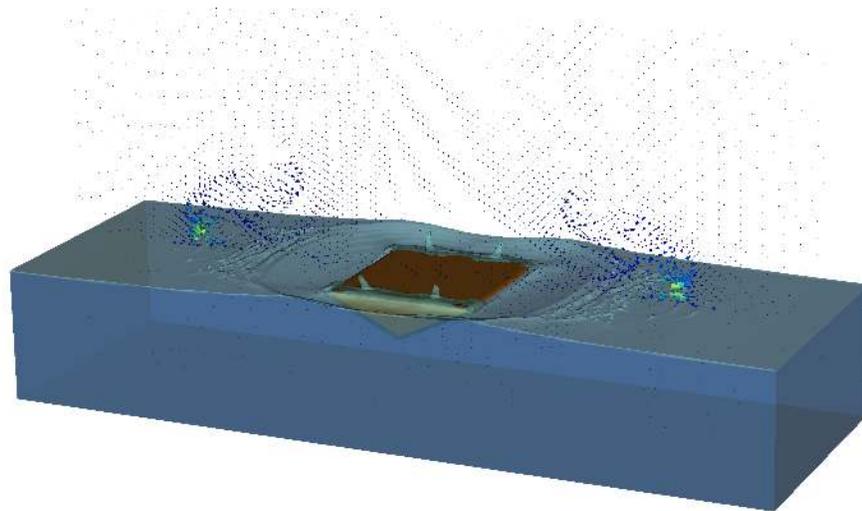
Here we simulate the water entry of a 3D wedge. Compared to the earlier test cases, the solid object has more complicated geometry and large velocity and pressure gradients are generated when the wedge keel impacts the free water surface. This test case was borrowed from the experiments of Yettou et al. [81]. The wedge has a square top $1.2\text{ m} \times 1.2\text{ m}$ and deadrise angle of 25° . The weight of the wedge is 94 kg corresponding to a density of 466.6 kg/m^3 . The wedge is initially positioned 1.3 m above the free water surface and there is a gap of 0.4 m between the wedge's vertical sides and water tank's lateral walls. The wedge is released into a water tank that is 12.0 m long and 2.0 m wide. The water depth is 1 m .

Figure 21 shows snapshots of the free fall and water entry of the wedge. The images also include the velocity vectors at the mid-plane normal to the y -axis. Vortices can be clearly seen in the wake of the wedge during its fall in the air; see Fig. 21b. The vortices are shed away from the wedge after its impact with the water surface. The wedge velocity and position have been plotted in Figs. 22 and 23, respectively, alongwith the experimental data of Yettou et al. [81]. The computational results include two uniform mesh resolutions resolving the wedge length (1.2 m) by 60 and 120 cells. In Fig. 22, we observe that the simulated wedge velocity is in good agreement with the experiment. The wedge position, in Fig. 23, is captured very well by the simulation when the wedge is falling in air. After the impact with water occurs, the simulation slightly underpredicts the water penetration length and the simulated wedge remains less submerged in water when compared with the experiment. Overall there is a fair agreement between the simulations and the experiment.

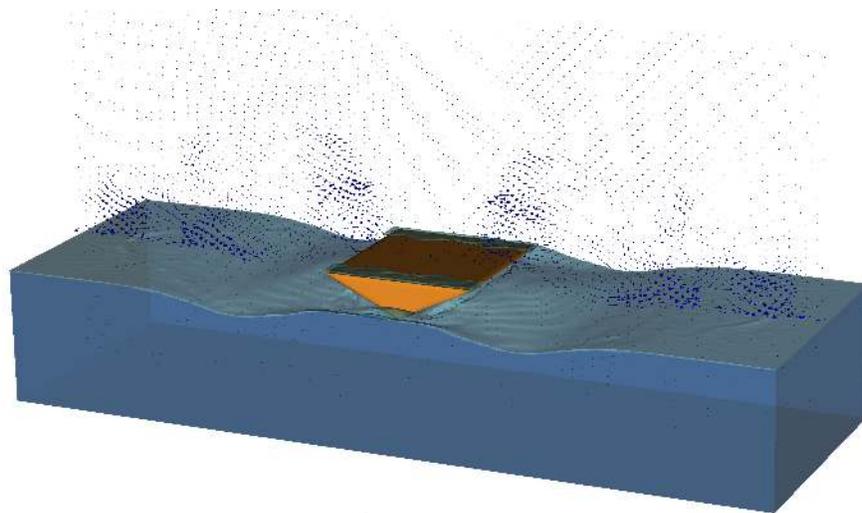
It is worth mentioning that the same test was simulated by Calderer et al. [76] using an immersed boundary method. There the hydrodynamic forces acting on the rigid body were calculated by explicit integration of surface forces, which led to diverged solutions. To ensure convergence they had to resort to strong

coupling that involved 4-5 subiterations in each time-step. Hu and Joseph [21] and Fekken [22] have stated earlier that an explicit coupling of fluid and rigid body motion, where the forces are computed explicitly, can lead to numerical instabilities. The methodology presented in the present paper is based on an implicit coupling of fluid/rigid body momentum equations. The forces on the rigid body are not calculated explicitly, but introduced implicitly in the form of rigid body velocities after solving simultaneously the momentum equations for fluid and the rigid body. The implicit formulation makes the new methodology stable and it posed no convergence issues while simulating the present test case.





(c) $t = 1.05 \text{ s}$



(d) $t = 1.6 \text{ s}$

Figure 21: Snapshots of a freely falling 3D wedge impacting the water surface, and subsequently floating.

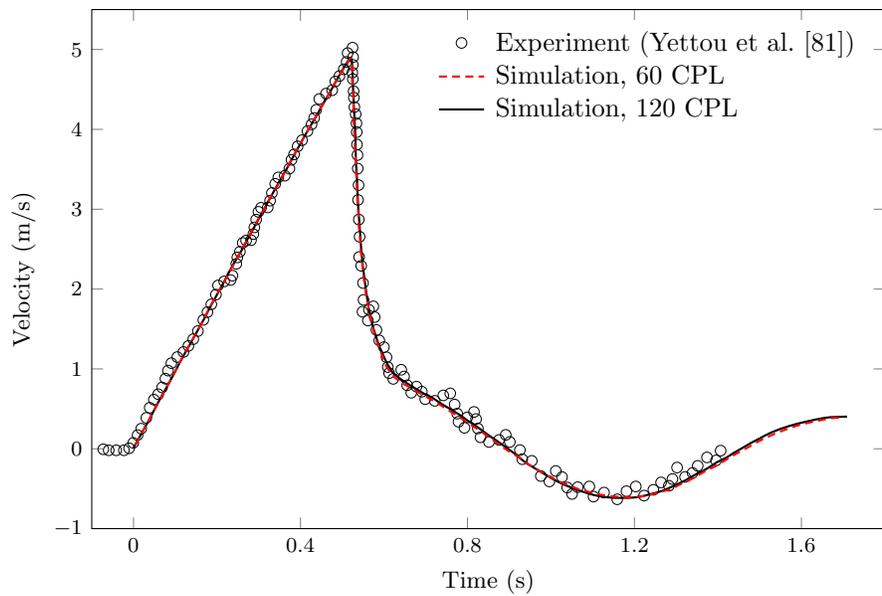


Figure 22: Velocity of a freely falling wedge entering the water, obtained from present simulations at resolutions of 60 and 120 cells per length (CPL) of the wedge, and the comparison with the experimental data of Yettou et al. [81].

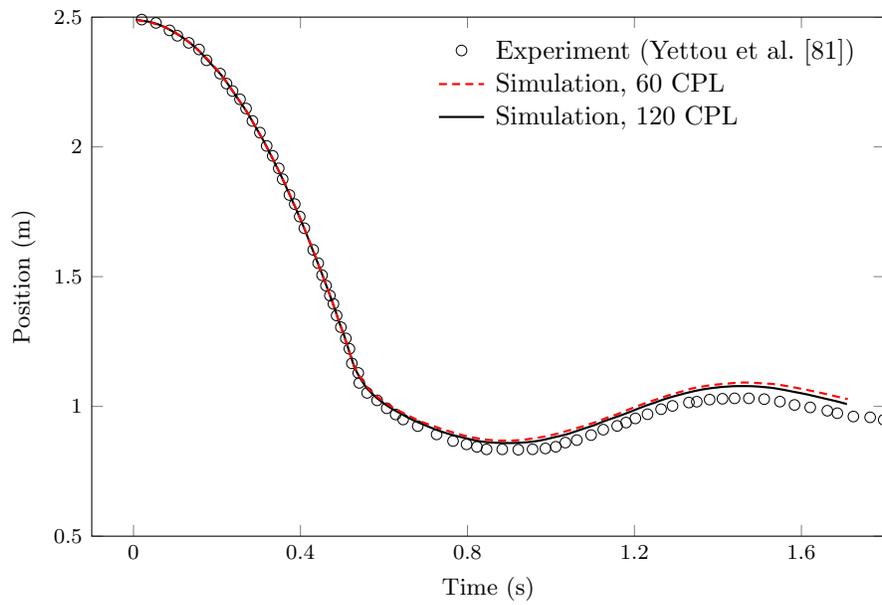


Figure 23: Position of a freely falling wedge entering the water, obtained from present simulations at the resolutions of 60 and 120 cells per length (CPL) of the wedge, and the comparison with the experimental data of Yettou et al. [81].

4.8. Interaction of multiple moving rigid bodies with waves

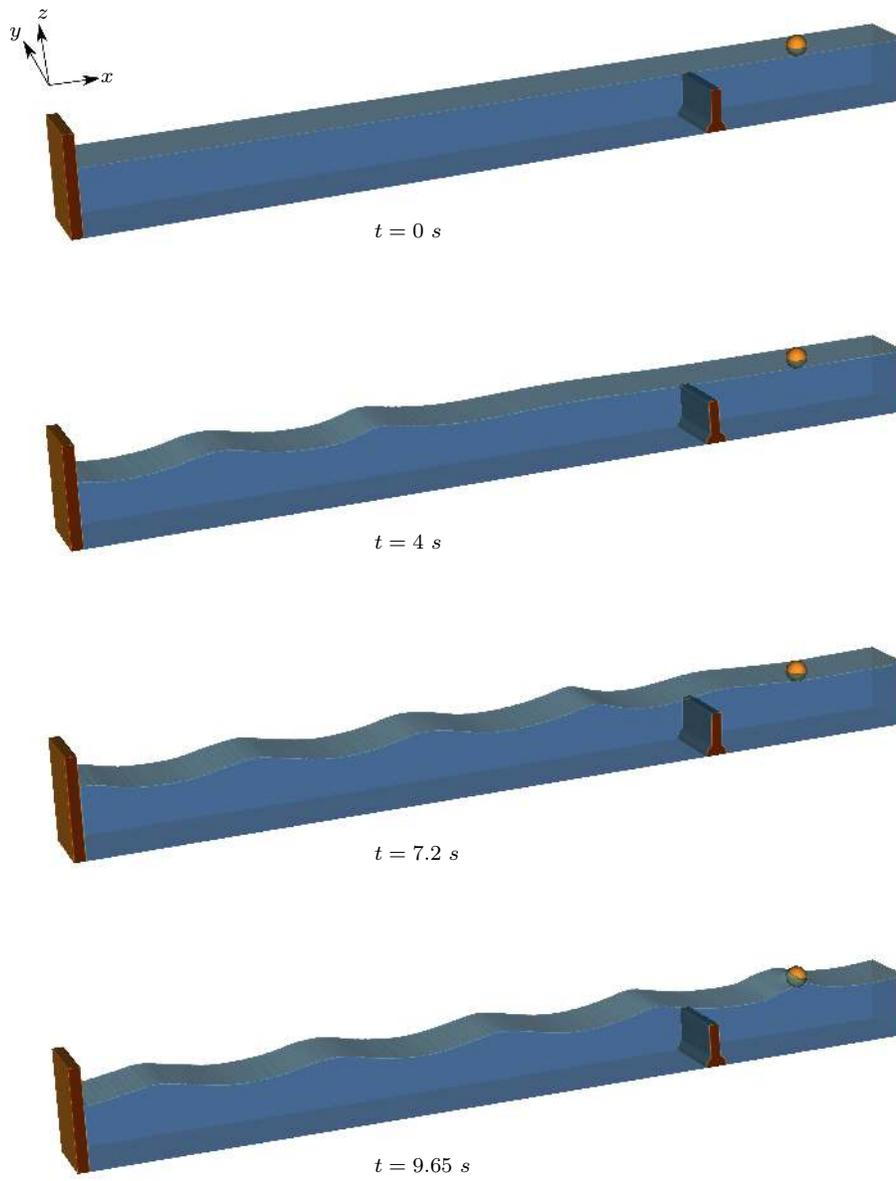
In the previous test cases, we have demonstrated the accuracy of the new methodology in capturing the interaction between two fluids and a single rigid body in both translational and rotational degrees of freedom. Now we present a numerical simulation of multiple rigid bodies interacting with waves. Two rigid bodies are representative examples of ocean wave energy converters (WECs); see Fig. 24. One is a flap type WEC, which can rotate about an axis passing through its base; the other is a point absorber spherical buoy that floats on the free surface. The third rigid body is the wave maker on the left. The numerical wave tank is 12 m long and 0.6 m wide. The water depth is 0.6 m . To generate waves we use a piston type wave maker that is 0.1 $m \times 0.6 m \times 0.9 m$ and is initially located at $x = 0.15 m$. The wave maker is prescribed a velocity in x -direction as:

$$V = \frac{S}{2} \frac{2\pi}{T} \sin\left(\frac{2\pi t}{T}\right) \quad (42)$$

where, stroke $S = 0.064 m$ and time period $T = 1 s$. The flap-type WEC has a cuboid part of dimensions 0.1 $m \times 0.6 m \times 0.3 m$. Below the cuboid there is a semicylinder base of radius 0.1 m and is located at $x = 6 m$. The spherical buoy has a radius of 0.1 m and is located at $x = 7.0 m$ half submerged in water. The sphere has a single degree of freedom in the vertical direction. The density of both WECs is 500 kg/m^3 . The computational domain, which is 12 $m \times 0.6 m \times 1.0 m$, is resolved using $1200 \times 60 \times 100$ cells.

Figure 24 shows snapshots of the simulation at various times. The waves cause rotational motion in the flap-type WEC and heave motion in the point-absorber WEC. The position of the heaving buoy and the roll angle of the flap-type WEC are plotted in Figs. 25 and 26, respectively. Such motions can be used to harness ocean wave energy using power take-off devices. Snapshots at $t = 12.15 s$ and $t = 16.65 s$ show that the waves become strongly nonlinear in the downstream of the flap-type WEC. As the waves around the sphere increase

in height, they break around it further adding to the wave's nonlinearity.



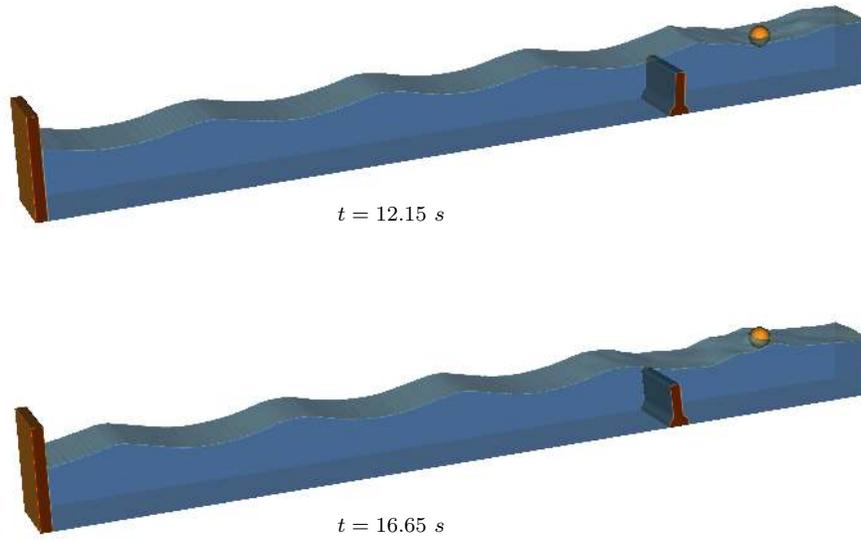


Figure 24: Snapshots of interaction between multiple moving rigid bodies with waves.

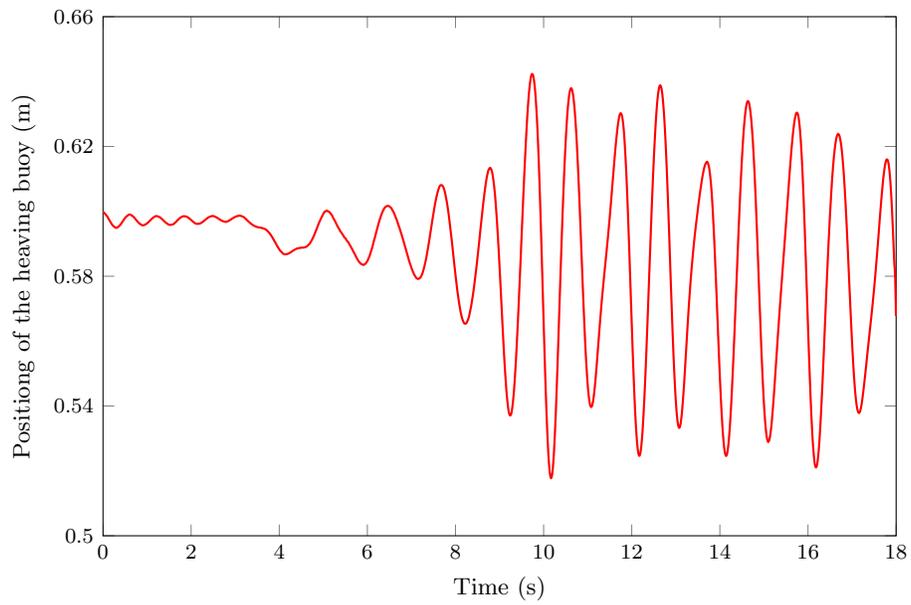


Figure 25: Position of the heaving spherical buoy, shown in Fig. 24, in response to waves.

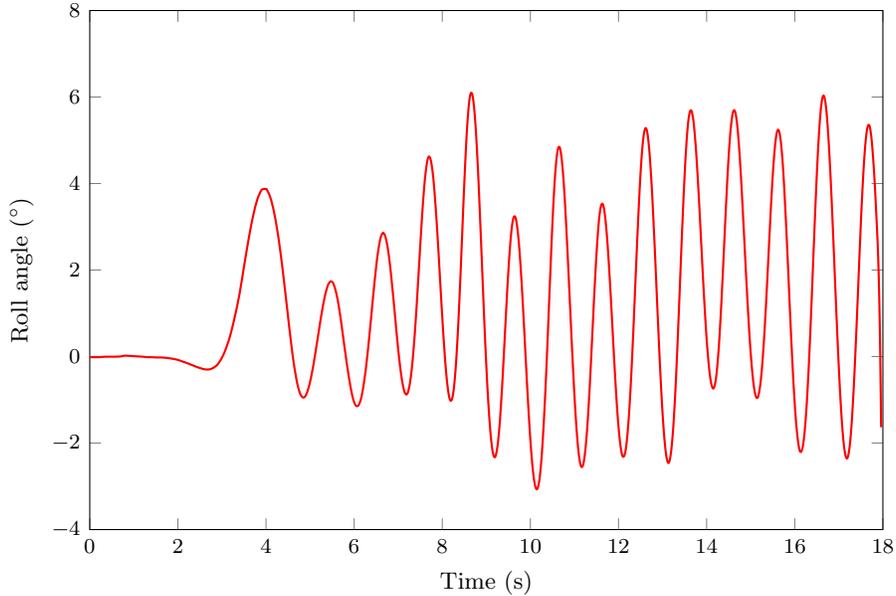


Figure 26: Roll angle of the flap-type WEC, shown in Fig. 24, in response to waves.

5. Summary and conclusion

We have presented a 3D fully Eulerian implementation of the fictitious domain method to study the interaction between two-fluid flows and moving rigid bodies. The proposed methodology is quite general and applicable to a variety of applications with fluid-solid interaction. In this work, however, our focus is on wave energy converters, and by considering two fluids we take into account the effect of air which can cause significant energy dissipation via formation of large dipoles in wave simulations. Unlike many numerical models for WEC simulations, we solve the full Navier-Stokes equations, thereby capturing the nonlinear phenomena and large topographical changes in the free water surface, which are critical to the WEC response.

Special numerical schemes were adopted to enable simulations of two fluids of arbitrarily large density ratio interacting with moving rigid body. Two color functions were used to track the rigid body and the fluid-fluid interface.

Interface reconstruction in three-phase cells was performed using an error minimization based VOF method [40]. A consistent mass and momentum transport scheme was adopted to handle large density ratios (e.g. air/water systems). The fluid-structure interaction was captured by the fast fictitious domain method of Sharma and Patankar [17], which solves a combined fluid and structure momentum equation. The method is implicit in nature and does not involve explicit integration of hydrodynamic forces on the boundary of the structure. The implicit formulation makes the proposed methodology numerically stable (as demonstrated in Section 4.7)

The proposed methodology was assessed using an array of test cases. First, the efficacy of the consistent mass and momentum transport and the ability of the solver to perform simulations involving large interfacial deformations was demonstrated. The remaining test cases assessed the accuracy with which the interaction between two-fluid flows and a moving rigid body is captured. Finally, a numerical simulation was presented where waves generated by a piston-type wave maker interacted with two different WECs. Oscillatory motion induced in the WECs as a result of this interaction was presented. The motion can be used to harness ocean wave energy using power take-off devices.

Acknowledgements

The research support from the National Science Foundation under the CBET Grant Nos. 1236462 and 1336232 is gratefully acknowledged. The simulations presented here were performed on the UMass Dartmouth HPC cluster, which was purchased through partial support from the NSF (MRI Grant No. CNS-0959382) and AFOSR (DURIP Grant No. FA9550-10-1-0354). AP gratefully acknowledges the Fellowship from the Office of the Associate Provost for Graduate Studies at UMass Dartmouth.

References

- [1] M. Eriksson, J. Isberg, M. Leijon, Hydrodynamic modelling of a direct drive wave energy converter, *International Journal of Engineering Science* 43 (17) (2005) 1377–1387.
- [2] X. Garnaud, C. C. Mei, Wave-power extraction by a compact array of buoys, *Journal of Fluid Mechanics* 635 (2009) 389–413.
- [3] Y. Kim, D. C. Kring, P. D. Sclavounos, Linear and nonlinear interactions of surface waves with bodies by a three-dimensional Rankine panel method, *Applied Ocean Research* 19 (5) (1997) 235–249.
- [4] W. Bai, R. E. Taylor, Higher-order boundary element simulation of fully nonlinear wave radiation by oscillating vertical cylinders, *Applied Ocean Research* 28 (4) (2006) 247–265.
- [5] P. Ferrant, D. L. Touzé, K. Pelletier, Non-linear time-domain models for irregular wave diffraction about offshore structures, *International journal for numerical methods in fluids* 43 (10-11) (2003) 1257–1277.
- [6] V. Vinayan, S. A. Kinnas, A numerical nonlinear analysis of two-dimensional ventilating entry of surface-piercing hydrofoils with effects of gravity, *Journal of Fluid Mechanics* 658 (2010) 383–408.
- [7] C. C. Mei, M. Stiassnie, D. K.-P. Yue, *Theory and applications of ocean surface waves: nonlinear aspects*, Vol. 23, World Scientific, 2005.
- [8] J. Morison, J. Johnson, S. Schaaf, The force exerted by surface waves on piles, *Journal of Petroleum Technology* 2 (05) (1950) 149–154.
- [9] T. Tezduyar, M. Behr, J. Liou, A new strategy for finite element computations involving moving boundaries and interfaces—The deforming—spatial—domain/space—time procedure: I. The concept and the preliminary numerical tests, *Computer Methods in Applied Mechanics and Engineering* 94 (3) (1992) 339 – 351.

- [10] R. Torii, M. Oshima, T. Kobayashi, K. Takagi, T. E. Tezduyar, Computer modeling of cardiovascular fluid–structure interactions with the deforming-spatial-domain/stabilized space–time formulation, *Computer Methods in Applied Mechanics and Engineering* 195 (13) (2006) 1885–1895.
- [11] T. E. Tezduyar, S. Sathe, Modelling of fluid–structure interactions with the space–time finite elements: Solution techniques, *International Journal for Numerical Methods in Fluids* 54 (6-8) (2007) 855–900.
- [12] H. Gan, J. Chang, J. Feng, H. Hu, Direct numerical simulation of the sedimentation of solid particles with thermal convection, *Journal of Fluid Mechanics* 481 (2003) 385–411.
- [13] D. Wan, S. Turek, An efficient multigrid-FEM method for the simulation of solid liquid two phase flows, *Computer Methods in Applied Mechanics and Engineering* 203 (2007) 561–580.
- [14] C. S. Peskin, Numerical analysis of blood flow in the heart, *Journal of computational physics* 25 (3) (1977) 220–252.
- [15] R. Mittal, G. Iaccarino, Immersed boundary methods, *Annu. Rev. Fluid Mech.* 37 (2005) 239–261.
- [16] M. Carlson, P. J. Mucha, G. Turk, Rigid fluid: animating the interplay between rigid bodies and fluid, *Transactions on Graphics* 23 (2004) 377–384.
- [17] N. Sharma, N. A. Patankar, A fast computation technique for the direct numerical simulation of rigid particulate flows, *Journal of Computational Physics* 205 (2005) 439–457.
- [18] S. Haeri, J. S. Shrimpton, On the application of immersed boundary, fictitious domain and body-conformal mesh methods to many particle multiphase flows, *International Journal of Multiphase Flow* 40 (2012) 38–55.

- [19] J. Mohd-Yusof, Combined immersed-boundary/b-spline methods for simulations of flow in complex geometries, Annual Research Briefs. NASA Ames Research Center, Stanford University Center of Turbulence Research, Stanford (1997) 317–327.
- [20] A. Gilmanov, S. Acharya, A hybrid immersed boundary and material point method for simulating 3D fluid–structure interaction problems, International journal for numerical methods in fluids 56 (12) (2008) 2151–2177.
- [21] H. H. Hu, D. D. Joseph, Direct simulation of fluid particle motions, Theoretical and Computational Fluid Dynamics 3 (1992) 285–306.
- [22] G. Fekken, Numerical simulation of free-surface flow with moving objects, Ph.D. thesis, University of Groningen, Groningen, The Netherlands, March. See also URL <http://www.ub.rug.nl/eldoc/dis/science/g.fekken> (2004).
- [23] T. Hesla, The dynamical simulation of two-dimensional fluid/particle systems, Unpublished notes.
- [24] R. Glowinski, T.-W. Pan, T. I. Hesla, D. D. Joseph, A distributed Lagrange multiplier/fictitious domain method for particulate flows, International Journal of Multiphase Flow 25 (5) (1999) 755–794.
- [25] N. A. Patankar, P. Singh, D. D. Joseph, R. Glowinski, T. W. Pan, A new formulation of the distributed Lagrange multiplier/fictitious domain method for particulate flows, International Journal of Multiphase Flow 26 (2000) 1509–1524.
- [26] N. Patankar, A formulation for fast computations of rigid particulate flows, Center for Turbulence Research Annual Research Briefs 2001 (2001) 185–196.
- [27] S. V. Apte, J. R. Finn, A variable-density fictitious domain method for particulate flows with broad range of particle-fluid density ratios, Journal of Computational Physics 243 (2013) 109–129.

- [28] A. Ghasemi, A. Pathak, M. Raessi, Computational simulation of the interactions between moving rigid bodies and incompressible two-fluid flows, *Computers & Fluids* 94 (0) (2014) 1 – 13.
- [29] P. Rauschenberger, B. Weigand, Direct numerical simulation of rigid bodies in multiphase flow within an Eulerian framework, *Journal of Computational Physics* 291 (2015) 238–253.
- [30] S. Gallier, E. Lemaire, L. Lobry, F. Peters, A fictitious domain approach for the simulation of dense suspensions, *Journal of Computational Physics* 256 (2014) 367–387.
- [31] S. Court, M. Fournié, A fictitious domain finite element method for simulations of fluid–structure interactions: The navier–stokes equations coupled with a moving solid, *Journal of Fluids and Structures* 55 (2015) 398–408.
- [32] M. Lage, H. Lopes, M. d. S. Carvalho, Flows with suspended and floating particles, *Journal of Computational Physics* 230 (20) (2011) 7736–7754.
- [33] H. Fadaifard, Computational two-phase flow and fluid-structure interaction with application to seabed scour, Ph.D. thesis, The University of Texas at Austin (2014).
- [34] M. Anbarsooz, M. Passandideh-Fard, M. Moghiman, Numerical simulation of a submerged cylindrical wave energy converter, *Renewable Energy* 64 (2014) 132–143.
- [35] P. Singh, D. Joseph, Fluid dynamics of floating particles, *Journal of Fluid Mechanics* 530 (2005) 31–80.
- [36] S. Madhavan, Investigation of vortical and interfacial particulate flows, Ph.D. thesis, University of Alberta (2011).
- [37] D. L. Youngs, Time-dependent multi-material flow with large fluid distortion, in: K. W. Morton, M. J. Baines (Eds.), *Numerical Methods for Fluid Dynamics*, Academic Press, 1982, pp. 273–285.

- [38] D. L. Youngs, An interface tracking method for a 3D Eulerian hydrodynamics code, Tech. Rep. 44/92/35, AWRE (1984).
- [39] B. van Wachem, M. Zastawny, F. Zhao, G. Mallouppas, Modelling of gas–solid turbulent channel flow with non-spherical particles with large Stokes numbers, *International Journal of Multiphase Flow* 68 (2015) 80–92.
- [40] A. Pathak, M. Raessi, A three-dimensional volume-of-fluid method for reconstructing and advecting three-material interfaces forming contact lines, *Journal of Computational Physics* 307 (2016) 550–573.
- [41] H. T. Ahn, M. Shashkov, Multi-material interface reconstruction on generalized polyhedral meshes, *Journal of Computational Physics* 226 (2007) 2096–2132.
- [42] G. Li, Y. Lian, Y. Guo, M. Jemison, M. Sussman, T. Helms, M. Arienti, Incompressible multiphase flow and encapsulation simulations using the moment-of-fluid method, *International Journal for Numerical Methods in Fluids* 79 (9) (2015) 456–490.
- [43] M. Raessi, On modeling surface tension-dominant, large density ratio, two-phase flows, Ph.D. thesis, University of Toronto, Canada (2008).
- [44] M. Raessi, H. Pitsch, Consistent mass and momentum transport for simulating incompressible interfacial flows with large density ratios using the level set method, *Computers and Fluids* 63 (2012) 70–81.
- [45] M. Rudman, A volume tracking method for incompressible multifluid flows with large density variations, *Int. J. Numer. Meth. Fluids* 28 (1998) 357–378.
- [46] K. Kleefsman, G. Fekken, A. Veldman, B. Iwanowski, B. Buchner, A volume-of-fluid based simulation method for wave impact problems, *Journal of Computational Physics* 206 (2005) 363–393.

- [47] A. Iafrati, A. Babanin, M. Onorato, Modulational instability, wave breaking, and formation of large-scale dipoles in the atmosphere, *Physical review letters* 110 (18) (2013) 184504.
- [48] F. H. Harlow, J. E. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface, *Physics of Fluids* 8 (1965) 2182–2190.
- [49] A. J. Chorin, Numerical solution of the Navier-Stokes equations, *Mathematics of Computation* 22 (1968) 745–762.
- [50] S. R. Codyer, M. Raessi, G. Khanna, Using graphics processing units to accelerate numerical simulations of interfacial incompressible flows, in: *ASME 2012 Fluids Engineering Division Summer Meeting collocated with the ASME 2012 Heat Transfer Summer Conference and the ASME 2012 10th International Conference on Nanochannels, Microchannels, and Minichannels*, American Society of Mechanical Engineers, 2012, pp. 625–634.
- [51] B. Avdicevic, A parallel, three-dimensional and two-fluid interfacial flow code with applications to pulsed water jets, Master’s thesis, University of Toronto (2007).
- [52] F. Tonon, Explicit exact formulas for the 3-D tetrahedron inertia tensor in terms of its vertex coordinates, *Journal of Mathematics and Statistics* 1 (1) (2005) 8–11.
- [53] D. B. Kothe, R. C. Mjolsness, M. D. Torrey, *RIPPLE: A computer program for incompressible flows with free surfaces*, available to DOE and DOE contractors from OSTI, 1991.
- [54] P. Stewart, N. Lay, M. Sussman, M. Ohta, An improved sharp interface method for viscoelastic and viscous two-phase flows, *Journal of Scientific Computing* 35 (1) (2008) 43–61.
- [55] D. B. Kothe, *Perspective on Eulerian finite volume methods for incompressible interfacial flows*, Springer, 1998.

- [56] M. M. Francois, S. J. Cummins, E. D. Dendy, D. B. Kothe, J. M. Sicilian, M. W. Williams, A balanced-force algorithm for continuous and sharp interfacial surface tension models within a volume tracking frameworks, *Journal of Computational Physics* 213 (2006) 141–173.
- [57] J. Martin, W. Moyce, An experimental study of the collapse of liquid columns on a rigid horizontal plane, *Philos. Trans. R. Soc. London* 244 (1952) 312–324.
- [58] M. Bussmann, D. B. Kothe, J. M. Sicilian, Modeling high density ratio incompressible interfacial flows, in: *ASME 2002 Joint US-European Fluids Engineering Division Conference*, American Society of Mechanical Engineers, 2002, pp. 707–713.
- [59] P. D. Hieu, T. Katsutoshi, V. T. Ca, Numerical simulation of breaking waves using a two-phase flow model, *Applied Mathematical Modelling* 28 (11) (2004) 983–1005.
- [60] V. Maronnier, M. Picasso, J. Rappaz, Numerical simulation of free surface flows, *Journal of Computational Physics* 155 (2) (1999) 439–455.
- [61] Y. Zhao, H. H. Tan, B. Zhang, A high-resolution characteristics-based implicit dual time-stepping VOF method for free surface flow simulation on unstructured grids, *Journal of Computational Physics* 183 (1) (2002) 233–273.
- [62] R. Löhner, C. Yang, E. Oñate, On the simulation of flows with violent free surface motion, *Computer Methods in Applied Mechanics and Engineering* 195 (41) (2006) 5597–5620.
- [63] A. Colagrossi, M. Landrini, Numerical simulation of interfacial flows by smoothed particle hydrodynamics, *Journal of Computational Physics* 191 (2) (2003) 448–475.

- [64] J. Bonet, T.-S. Lok, Variational and momentum preservation aspects of smooth particle hydrodynamic formulations, *Computer Methods in applied mechanics and engineering* 180 (1) (1999) 97–115.
- [65] C.-L. Lin, H. Lee, T. Lee, L. J. Weber, A level set characteristic Galerkin finite element method for free surface flows, *Int. J. Numer. Methods Fluids* 49 (5) (2005) 521–547.
- [66] E. Marchandise, J.-F. Remacle, A stabilized finite element method using a discontinuous level set approach for solving two phase incompressible flows, *Journal of computational physics* 219 (2) (2006) 780–800.
- [67] B.-H. Lee, J.-C. Park, M.-H. Kim, S.-C. Hwang, Step-by-step improvement of MPS method in simulating violent free-surface motions and impact-loads, *Computer methods in applied mechanics and engineering* 200 (9) (2011) 1113–1125.
- [68] B. Ataie-Ashtiani, L. Farhadi, A stable moving-particle semi-implicit method for free surface flows, *Fluid Dynamics Research* 38 (4) (2006) 241–256.
- [69] S. Y. Yoon, T. Yabe, The unified simulation for incompressible and compressible flow by the predictor-corrector scheme based on the CIP method, *Computer Physics Communications* 119 (2) (1999) 149–158.
- [70] F. Kelecy, R. Pletcher, The development of a free surface capturing approach for multidimensional free surface flows in closed containers, *Journal of computational physics* 138 (2) (1997) 939–980.
- [71] A. Murrone, H. Guillard, A five equation reduced model for compressible two phase flow problems, *Journal of Computational Physics* 202 (2) (2005) 664–698.
- [72] R. Issa, D. Violeau, Test-case 2, 3D dambreaking, Release 1.1, ERCOF-TAC, SPH European Research Interest Community SIG, Electricite De

- France, Laboratoire National d'hydraulique et Environnement (2006) 19–28.
- [73] S. Ito, Study of transient heave oscillation of a floating cylinder, Master's Thesis, Massachusetts Institute Technology (May 1997).
- [74] S. Maskell, F. Ursell, The transient motion of a floating body, *Journal of Fluid Mechanics* 44 (02) (1970) 303–313.
- [75] K. H. Jung, K. Chang, H. J. Jo, Viscous effect on the roll motion of a rectangular structure, *Journal of Engineering Mechanics* 132 (2006) 190–200.
- [76] A. Calderer, S. Kang, F. Sotiropoulos, Level set immersed boundary method for coupled simulation of air/water interaction with complex floating structures, *Journal of Computational Physics* 277 (2014) 201–227.
- [77] D. Bailey, D. Griffiths, S. Maskell, On the experimental observation of a heaving sphere, *Schiffstechnik; Forschungshefte für Schiffbau und Schiffsmaschinenbau* 23 (1976) 31–45.
- [78] R. Beck, S. Liapis, Transient motions of floating bodies at zero forward speed, *Journal of Ship Research* 31 (3).
- [79] <http://name.engin.umich.edu/facilities/mhl/facilities/basin>.
- [80] C. J. Damaren, Time-domain floating body dynamics by rational approximation of the radiation impedance and diffraction mapping, *Ocean Engineering* 27 (6) (2000) 687–705.
- [81] E.-M. Yettou, A. Desrochers, Y. Champoux, Experimental study on the water impact of a symmetrical wedge, *Fluid dynamics research* 38 (1) (2006) 47–66.