Accurate wave generation using unstructured two-phase solver with a phase-field approach

By H. Hwang AND C. Gorlé

1. Motivation and objectives

The study of waves in coastal regions has gained significant attention due to the escalating vulnerability of coastal communities to extreme climate-related risks triggered by rising sea levels and human-induced warming (Chini *et al.* 2010; Dodet *et al.* 2019). The shallow water depths characteristic of coastal areas enhances the interplay between strong wave orbital motion and variations in the seabed (Nielsen 1992). This interaction leads to wave breaking within the surf zone, littoral drift (George & Kumar 2019) and coastal erosion (Zhang *et al.* 2022). Consequently, understanding these dynamics and their impact on coastal regions and infrastructure is crucial for fortifying the resilience of coastal communities, especially during severe events like storm surges and tropical cyclones (Janssen & Bidlot 2023).

Various numerical models, such as the WAVE WATCH III® (WW III®) (Tolman *et al.* 2009) and SWAN (Booij *et al.* 1999) models, have been successfully developed to predict waves by assuming a large-scale separation between the wave field and an individual wave. However, this assumption becomes questionable when applied to coastal regions, which are localized and limited in area. While global wave models can incorporate near-shore flow physics through methods like nesting with high-resolution wave models (Sandhya *et al.* 2014) or adding source terms to their governing equations (Tolman *et al.* 2009), their implementation remains rudimentary and is not yet adaptable to arbitrary shallow waters.

For a comprehensive understanding of complex flow dynamics in the coastal region, numerical simulations serve as powerful tools to analyze intricate local information and wave parameters, which can be integrated into existing global wave models. OpenFOAM® (v2212) is one of the most popular open-source computational fluid dynamics (CFD) toolboxes, providing numerous features for simulating wave interactions in coastal regions. The code uses a volume-of-fluid (VOF) method for interface capturing, employing algebraic-type (Higuera 2015) and geometric-type [e.g., isoAdvector by Roenby *et al.* (2016)] approaches. Various features related to wave dynamics, such as wave generation and absorption based on wave theories, have been rigorously validated across numerous cases (Lara *et al.* 2012; Higuera *et al.* 2013, 2018). However, validation of wave features relies solely on the VOF method, and its dependency on the dissipative nature of the numerical solver restricts its usage. Additionally, the extensive freedom in parameter selection in OpenFOAM® results in considerable sensitivity in simulations (Windt *et al.* 2019), indicating insufficient robustness in the current approach for wave simulations.

A diffuse-interface (DI) method, emerging as an alternative, is gaining traction among various numerical interface models due to its simple implementation and potential for efficient parallelization. Unlike the sharp interface representation in VOF, the DI method models the interface as a smooth gradient of phase indicators across multiple cells. Recently, an accurate conservative DI method has been proposed by Jain (2022) and ex-

tended to a Voronoi-based unstructured grid formulation by Hwang & Jain (2023). The authors rigorously derive the discrete kinetic energy equation showing the robustness of the DI method in terms of energy conservation even for collocated grids. The Voronoi-based mesh generation using `stitch`, developed by Cascade technologies, offers faster and more efficient mesh generation. These features make the proposed two-phase solver using DI method appealing for simulating arbitrary seabed geometries and coastal infrastructures.

This study aims to employ a two-phase solver using the DI method (Hwang & Jain 2023) for wave simulation by introducing a new hybrid method for wave generation compatible with the current interface capturing method. Furthermore, an accurate probing technique will be established. The performance of the solver will be compared to OpenFOAM® for linear waves in an intermediate water depth.

2. Methods

We consider incompressible and immiscible two-phase flows, described by the following governing equations as

$$\vec{\nabla} \cdot \vec{u} = 0, \tag{2.1}$$

$$\frac{\partial \rho \vec{u}}{\partial t} + \vec{\nabla} \cdot (\rho \vec{u} \otimes \vec{u}) = -\vec{\nabla} p^* - \vec{g} \cdot \vec{x} \vec{\nabla} \rho + \mu_{eff} \Delta \vec{u} + \sigma \kappa \vec{\nabla} \phi, \tag{2.2}$$

where ρ and $\vec{u}$ represent density and velocity, respectively. Here, p^* is the pseudo-dynamic pressure defined as $p^* = p - \rho \vec{g} \cdot \vec{x}$, with p and $\vec{g}$ as total pressure and gravity, respectively. The position vector $\vec{x}$ is relative to a specific reference point.

2.1. *Interface-capturing methods*

Both the VOF and DI methods are classified as one-fluid models employing a single additional variable to capture the interface. Each phase is characterized by the volume fraction of one of the fluids, denoted by the phase indicator variable $\phi = \phi_1$. Due to the definition of the volume fraction for the two fluids, ϕ_2 satisfies $\phi_2 = 1 - \phi_1$; thus only the transport equation for ϕ_1 is solved.

2.1.1. *Algebraic VOF method with limiter*

In OpenFOAM®, the algebraic-type VOF method is utilized along with the multi-dimensional universal limiter with explicit solution (MULES) (Damián & Nigro 2014) to ensure the boundedness of the phase indicator. The transport equation for the phase indicator ϕ is described by

$$\frac{\partial \phi}{\partial t} + \vec{\nabla} \cdot (\vec{u} \phi) = \vec{\nabla} \cdot (\vec{\mathcal{U}_r} \phi (1 - \phi)), \tag{2.3}$$

where $\vec{\mathcal{U}_r}$ represents the relative velocity, which controls the interface sharpness. Additional details can be found in Higuera (2015).

2.1.2. *DI method*

The phase indicator ϕ in the DI model is computed as (Jain 2022)

$$\frac{\partial \phi}{\partial t} + \vec{\nabla} \cdot (\phi \vec{u}) = \vec{\nabla} \cdot \left\{ \Gamma \left\{ \epsilon \vec{\nabla} \phi - \frac{1}{4} \left[1 - \tanh^2 \left(\frac{\psi}{2\epsilon} \right) \right] \frac{\vec{\nabla} \psi}{|\vec{\nabla} \psi|} \right\} \right\}, \tag{2.4}$$

FIGURE 1. Wave generation methods (a) SBM, (b) RZM and (c) hybrid method. Color gradients represent the relaxation zone.

where $\vec{u}$ represents velocity, Γ denotes the velocity-scale parameter and ϵ is the interface thickness scale parameter. Here, ψ is a signed-distance-like function from the interface, defined as

$$\psi = \epsilon \ln\left(\frac{\phi + \varepsilon}{1 - \phi + \varepsilon}\right), \tag{2.5}$$

with ε as a small number to avoid division by zero. For further information on the current DI method, readers are referred to Jain (2022) and Hwang & Jain (2023).

2.2. *Simulation setup*

Waves are generated in the simulation domain of length $L_x = 15$ m, height $L_y = 1.5$ m, and width $L_z = 0.3$ m. A uniform grid of size $\Delta = 0.01$ m in all directions is used. At $x = 0$ m, wave generation boundary conditions are used, which will be further discussed below. At $L_x = 15$ m, a wave absorption boundary condition based on the shallow water theory (Higuera *et al.* 2013) is prescribed to minimize the reflection of the waves, preventing contamination of the flow field within the simulation domain. Periodic and no-slip boundary conditions are employed at side walls and at the bottom, $L_y = 0$ m, respectively. At the top boundary, a Neumann-type boundary condition, which mimics the atmospheric open-boundary condition, is used for OpenFOAM® simulations, while a convective outlet boundary condition is used for DI method simulations.

2.3. *Wave generation*

A linear wave using the first-order Stokes wave theory is generated at the inlet. The wave condition is defined by its height $H = 0.05$ m, wave period $T = 1.5$ s, and water depth $h = 0.6$ m, which yields the wave length of $\lambda = 2.29$ m. Various kinds of wave generation methods can be used to introduce the selected wave.

2.3.1. *Stationary wave generation methods*

Existing methods for wave generation include static/dynamic, relaxation zone, mass- and impulse-source boundary methods as outlined in Windt *et al.* (2019). In scenarios involving a stationary mesh, static and relaxation zone boundary conditions are commonly favored due to their simplicity. However, implementing mass- and impulse-source wave generation necessitates not only incorporating a new source term into the governing equation but also an additional technique, such as convective boundary conditions, to absorb one of the waves generated both upstream and downstream.

The static boundary method (SBM) involves a straightforward time-varying Dirichlet boundary condition, where both velocity and the distribution of the phase indicator are computed and prescribed at the inlet boundary. The velocity and wave height are derived from the wave theories. The relaxation zone method (RZM), introduced by Jacobsen *et al.* (2012), integrates the desired solution θ_{target} with the computed solution within the simulation domain, θ_{computed}, utilizing a blending function $\alpha(x)$, which typically spans

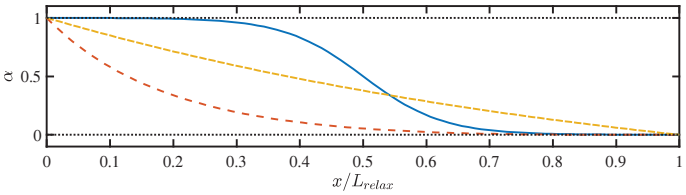

FIGURE 2. Blending function $\alpha(x)$ following Eq. (2.7) (blue solid line). Two functional forms recommended by Engsig-Karup *et al.* (2006): $(e^{1-(x/L_{\text{relax}})^{3.5}} - 1)/(e^1 - 1)$ (red dashed line), and $(e^{1-(x/L_{\text{relax}})} - 1)/(e^1 - 1)$ (yellow dash-dotted line).

multiple wavelengths. Subsequently, the integrated solution for any arbitrary variable θ is computed by

$$\theta = \alpha(x)\theta_{\text{target}} + (1.0 - \alpha(x))\theta_{\text{computed}}. \tag{2.6}$$

In practice, both the phase indicator ϕ and the velocity field $\mathbf{u}$ are blended with the theoretical solution and updated within the relaxation zone at every time step. Schematics of SBM and RZM are provided in Figure 1(a,b), respectively.

The study conducted by Windt *et al.* (2019) provides a comparative analysis of various existing wave generation methods. According to the authors, RZM demonstrated superior performance compared to others in terms of accuracy. However, implementing RZM incurs higher computational costs due to the additional grids in the relaxation zone. Furthermore, despite its effectiveness, RZM leads to an increase in the total volume of the liquid within the simulation domain, as highlighted by Jacobsen *et al.* (2012). In contrast, the static boundary condition maintains overall volume conservation and operates at lower computational costs.

2.3.2. *Hybrid method*

In pursuit of retaining the superior accuracy of RZM while mitigating excessive liquid volume increase within the simulation domain, we propose a novel hybrid method that combines aspects of both RZM and SBM. The hybrid method represents a variation of RZM wherein the solution for the velocity field is updated using a blending function similar to the conventional RZM, while the phase indicator ϕ is solely controlled at the inlet boundary face through the mass flux, which is introduced in solving the transport equation for ϕ [Eq. (2.4)]. The conceptual drawing is provided in Figure 1(c). Note that the velocity field is enforced in the relaxation zone after solving the momentum equation but before the correction step [see Hwang & Jain (2023)].

In this study, the blending function $\alpha(x)$ is chosen as

$$\alpha(x) = \frac{1}{2} - \frac{1}{2}\tanh\left\{a_1\left(\frac{x}{L_{\text{relax}}} - \frac{1}{2}\right)\right\}, \tag{2.7}$$

where we set $a_1 = 8.0$, and L_{relax} represents the length of the relaxation boundary, set at $L_{\text{relax}} = 5$ m. This specific choice of blending function (depicted as the blue solid line in Figure 2) aims to ensure a smooth transition at both ends of the relaxation zone, taking into account the mass flux at the inlet boundary face.

We note that the functional form of the blending function $\alpha(x)$ was proposed to statisfy $\alpha(0) = 1$ and $\alpha^{(n)}(0) = 0$ (Engsig-Karup *et al.* 2006) in the context of discontinuous Galerkin finite element methods. The superscript n denotes $n-$th derivatives. Thus, we

also show the blending function of the proposed form by Engsig-Karup *et al.* (2006) in Figure 2 for comparison. Note that the active wave absorption (Higuera 2015) is employed at the inlet boundary face, but it is not extensively discussed in this study.

2.3.3. *Quantities of interest*

Quantities of interest within this study are given below.

(1) Total volume of liquid phase:

$$\phi_{\text{total}} = \sum \phi V_{\text{cv}}, \tag{2.8}$$

where V_{cv} is the cell volume.

(2) Wave elevation: Further details on the probe are outlined in Section 2.4.

(3) Wave height error: Wave elevation data is decomposed into individual periods by identifying zero-crossing points at each probe location. Subsequently, wave height is computed for each period. The average of the wave heights $\overline{H}$ and its standard deviation σ_H are determined. The wave error is calculated as

$$\mathcal{E}_H = \frac{\overline{H} - H_{\text{theory}}}{H_{\text{theory}}}, \tag{2.9}$$

where $\sigma_{\mathcal{E}_H} = \sigma H / H_{\text{theory}}$ denotes its standard deviation.

(4) Wave velocity error at crests and troughs: Velocity field data are gathered within the region $x \in [5\text{ m}, 10\text{ m}]$ with a frequency of $f = 100$ Hz. Wave velocities at crest and trough locations are measured. Measurement heights are uniformly incremented by $\Delta_y = (h + H/2)/N$ for crests and $\Delta_y = (h - H/2)/N$ for troughs from the bottom wall. At each measurement height, the averaged velocity $\overline{u}_{max}$ ($\overline{u}_{min}$) and its standard deviation $\sigma_{u_{max}}$ ($\sigma_{u_{min}}$) for crests (troughs) are calculated. The velocity error is defined as

$$\mathcal{E}_u(y) = \frac{\overline{u}(y) - u_{\text{theory}}(y)}{u_{\text{theory}}(y)}, \tag{2.10}$$

where $\sigma_{\mathcal{E}_u} = \sigma_u / u_{\text{theory}}$ represents its standard deviation.

2.4. *Wave probe*

In order to accurately determine the interface location within the unstructured grid framework while employing the DI method, three distinct types of probes have been introduced: ϕ-based, ψ-based and isoplane-based probes. These probes utilize line configurations extending from point $\mathbf{x}_{p_1}$ to point $\mathbf{x}_{p_2}$, equipped with uniquely distributed measurement points along their lengths. Within this study, the probes are positioned along the centerline of the domain in the spanwise direction. The streamwise locations of these probes are defined as $x = 5$, 7.5, 10, 12.5 and 15 m. For consistency across all probes, fixed values of $\mathbf{x}_{p_1} = 0.5$ and $\mathbf{x}_{p_2} = 0.7$ are maintained for the initial and final points of the probes, respectively. Each probe comprises $N_p = 201$ measurement points, resulting in a uniform spacing between these points of $\Delta_{\text{probe}} = 0.01$.

2.4.1. *ϕ-based method*

The ϕ-based method is essentially an integration technique utilized for computing wave elevation η, expressed as

$$\eta = \mathbf{x}_{p_1} \cdot \hat{y} + \sum_{k=2}^{N_p} \phi_k \vec{\Delta}_{\text{probe}} \cdot \hat{y}. \tag{2.11}$$

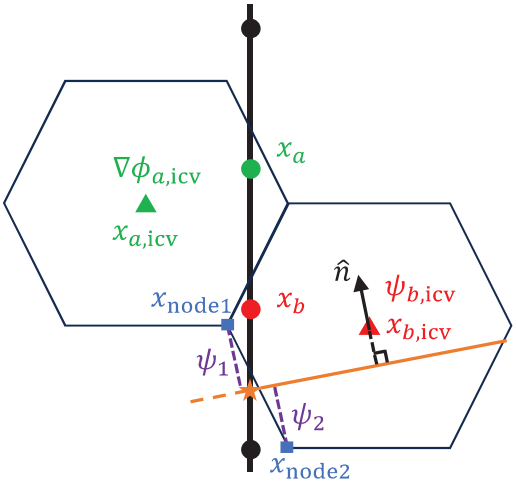

FIGURE 3. A schematic of probes.

Here, ϕ_k represents the phase indicator value interpolated onto the points distributed along the probes. For example, a probe is illustrated as a black solid line with discrete measurement points along the probe shown as circle symbols in Figure 3. ϕ_{x_a} (green circle) is interpolated using the values in the closest cell center $x_{a,\mathrm{icv}}$ (green triangle) as

$$\phi_{x_a} = \phi_{x_{a,\mathrm{icv}}} + \vec{\nabla}\phi_{a,\mathrm{icv}} \cdot (\vec{x}_a - \vec{x}_{a,\mathrm{icv}}), \tag{2.12}$$

where $\vec{\nabla}\phi$ is the gradient of the phase indicator ϕ at the cell center. OpenFOAM® implements this integration method as its default approach with linear interpolation. We note that this method relies on perfectly bounded phase indicators, adhering to the conditions $0 \leq \phi \leq 1$, and a sharp interface.

2.4.2. *ψ-based probe*

The auxiliary function ψ, resembling a signed-distance function, can be computed from the phase indicator ϕ according to (Jain 2022) as

$$\psi = 2\epsilon \tanh^{-1}(2\phi - 1). \tag{2.13}$$

The phase indicator ϕ is interpolated onto the probe points following the procedure above. Subsequently, utilizing Eq. (2.13), ψ is reconstructed along the probe. By determining the location $\mathbf{x}\eta$ where $\psi(\mathbf{x}\eta) = 0$, the wave elevation $\eta = \mathbf{x}_\eta \cdot \hat{y}$ is identified.

2.4.3. *Isoplane-based probe*

For each discrete point along the line probe (e.g., x_b in Figure 3), the closest cell center $x_{b,\mathrm{icv}}$ is identified. Subsequently, the auxiliary function $\psi_{x_{b,\mathrm{icv}}}$ is computed at this cell center using Eq. (2.13). With the known interface normal $\hat{n}$, the investigation of the closest cell intersection with the $\psi = 0$ isoplane involves computing

$$\psi_0 = \psi_{\mathbf{x}_{\mathrm{icv}}} - (\mathbf{x}_{\mathrm{node1}} - \mathbf{x}_{\mathrm{icv}}) \cdot \mathbf{n}, \tag{2.14}$$

$$\psi_1 = \psi_{\mathbf{x}_{\mathrm{icv}}} - (\mathbf{x}_{\mathrm{node2}} - \mathbf{x}_{\mathrm{icv}}) \cdot \mathbf{n}, \tag{2.15}$$

where ψ_0, ψ_1, x_{node1} and x_{node2} are illustrated in Figure 3. Iterating through every edge of the cell, the intersection point is determined by checking the condition $\psi_0\psi_1 <$

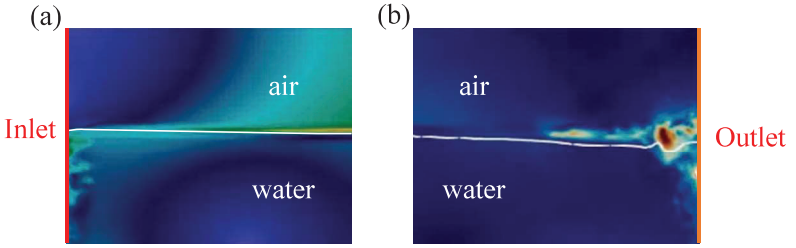

FIGURE 4. Erroneous velocity fluctuation generated at simulation (a) inlet and (b) outlet. Colormaps represent velocity magnitude field and the interface is denoted as a white solid line.

0. Once the intersection points between the isoplane and the edges are identified, the isoplane is reconstructed connecting those intersection points. Then, another intersection point between the probe and isoplane, which can be extended if necessary, is established. Finally, the y-component of the averaged value of all intersection points between the isoplanes and the probe is determined as the wave elevation η.

3. Results

Four simulation results are presented in this section,

- case1: OpenFOAM® with `nOuterCorrector`= 1
- case2: case1 with twice the vertical grid refinement $\Delta_y = 0.005$
- case3: OpenFOAM® with `nOuterCorrector`= 3
- case4: Unstructured two-phase solver utilizing DI method (Hwang & Jain 2023)

In OpenFOAM® simulations, the algebraic-type VOF method with MULES limiter is employed. `nOuterCorrector` signifies the number of PIMPLE (a combination of PISO and SIMPLE) outer corrections. Additional cases2 and 3 are conducted to investigate sensitivity factors beyond the original case1. Typically, a minimum of ten cells per wave height are recommended (Roenby *et al.* 2017). However, the current numerical setup only has five cells per wave height, categorizing it as a relatively underresolved case. Case2, having ten cells per wave height, addresses this resolution issue. Case3, featuring a higher `nOuterCorrector`= 3, aims to explore improved wave predictions compared to `nOuterCorrector`= 1 (Windt *et al.* 2019).

For the time advancement scheme and spatial discretization, the implicit Euler and second-order central difference are employed for In OpenFOAM® simulations, while fourth-order Runge-Kutta (RK4) and skew-symmetric-type second-order flux-split central difference are used (Hwang & Jain 2023).

3.1. *Erroneous velocity fluctuation*

In the simulation using the DI method (case4) with the SBM, colormaps representing the velocity field near the inlet and outlet are depicted in Figure 4. The orbital motion of the wave induces periodic in- and outflux through the boundary faces. Any deviation between the wave theory and the computed quantity, especially during the wave ramp-up period, results in an erroneous velocity fluctuation near the inlet and outlet. Additionally, active wave absorption introduces a constant mass flux (Higuera 2015), exacerbating this deviation and leading to further erroneous flow behavior. OpenFOAM® also exhibits similar phenomena, but the extent of error in the current situation is relatively smaller due to the dissipative nature of its algorithm.

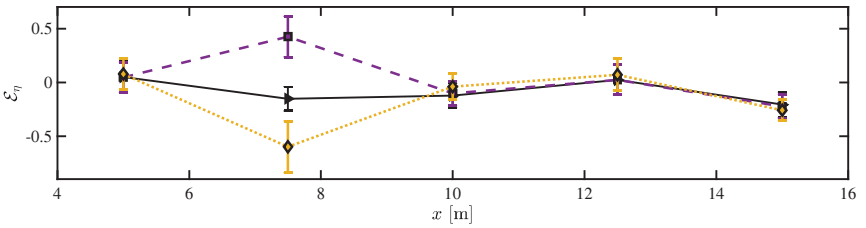

FIGURE 5. Wave height error calculated at each probe location. Lines represent ψ-based (black solid line), ϕ-based (purple dashed line) and isoplane-based (yellow dotted line). Symbols indicate the measurement locations.

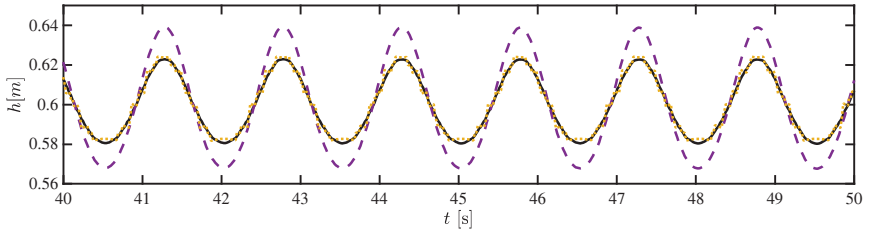

FIGURE 6. Wave elevation measured at probes location $x = 7.5$ m in simulations using DI method. Lines indicate G-based (black solid line), VOF-based (purple dashed line), and PLIC-based (yellow dotted line) probes.

In avoiding this erroneous fluctuation issue, the adoption of the hybrid method in DI method simulations proved beneficial. It effectively suppressed the erroneous velocity near the inlet (see Figure 12(d)) while ensuring satisfactory conservation of the total liquid volume (see Figure 7).

3.2. *Probes*

For the DI method simulation (case4), three different types of probes are compared. Figure 5 shows the height error at five locations. The ψ-based probe exhibited minimal height errors at the first four probe locations, with a maximum $\mathcal{E}_\eta < 0.15$. Given the presence of only five cells per wave height, this value corresponds to less than a single cell displacement. The error at the last probe, located at the wave outlet, though not significant, was not considered in this discussion.

Both the ϕ-based and isoplane-based probes showed substantial errors at the second probe location ($x = 7.5$ m). The ϕ-based probe yielded $\mathcal{E}\eta = 0.43$, while the isoplane-based probe recorded $\mathcal{E}\eta = -0.6$. Notably, the relaxation zone length ($L_{\text{relax}} = 5$ m) corresponds to the location of the large difference observed in wave elevation, implying a more diffused interface at this specific location. The time series of the wave elevation is depicted in Figure 6.

The ϕ-based probe calculates an increased wave height $H \approx 0.07$ m, which is larger than the prescribed wave. The isoplane-based probe shows nonsmooth wave elevation with spikes. The ψ-based probe results agree well with the results from OpenFOAM® simluation (cases 1 and 2; see Figure 8). Therefore, the ψ-based probe is selected as an appropriate probe for DI method simulations. Further analysis indicated that the ϕ-based probe calculated an increased wave height ($H \approx 0.07$ m), surpassing the prescribed wave. Additionally, the isoplane-based probe exhibited a nonsmooth wave elevations, characterized by spikes. Conversely, the ψ-based probe showed excellent agreement with results

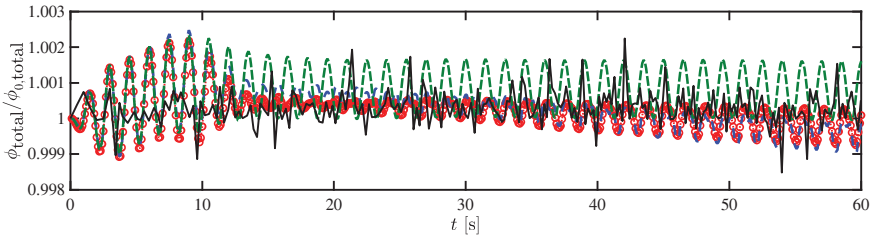

FIGURE 7. Total volume of liquid phase in simulations domain. The total volume is normalized by the initial total volume. Lines indicate case1 (red solid line), case2 (blue dashed line), case3 (green dash-dotted line) and case4 (black solid line).

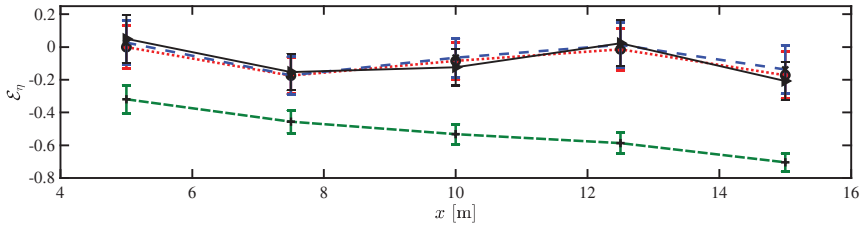

FIGURE 8. Height errors in each probe location. Lines indicate case1 (red solid line), case2 (blue dashed line), case3 (green dash-dotted line) and case4 (black solid line).

obtained from OpenFOAM® simulations (cases 1 and 2; see Figure 8). Consequently, the ψ-based probe was selected to measure the wave elevation for the subsequent simulation results.

3.3. *Total volume*

The total liquid volume tracked during the DI method simulation using the hybrid method for wave generation is presented in Figure 7 (depicted as a black line). We note that RZM introduces excessive mass into the simulation domain, while SBM does not. In comparison to the SBM results from OpenFOAM® simulations, the proposed hybrid method effectively conserves the total volume of the liquid phase at a satisfactory level.

3.4. *Height error*

Height error for four cases are reported in Figure 8. Unlike the findings reported in Windt *et al.* (2019), the case with a larger number of PIMPLE outer corrections (`nOuterCorrector`= 3) (depicted by the green dashed line) demonstrates higher height errors compared to other cases. The magnitude of the height error increases as the wave propagates downstream, indicating considerable wave damping. All cases, except case3, exhibit comparable height errors.

The time series of the wave elevation for all cases is illustrated in Figure 9. Notably, no discernible difference is observed between case1 (depicted by the red circle) and the refined case2 (illustrated by the blue dashed line), indicating identical wave elevation. Conversely, despite matching wave phases with case1 and 2, case3 (depicted by the green dashed line) shows a decrease in wave height. The DI method results reveal a shifted wave phase but satisfactory measures for wave height. The constant wave height observed throughout the simulation qualitatively suggests that the reflected wave effect is negligible in the current study.

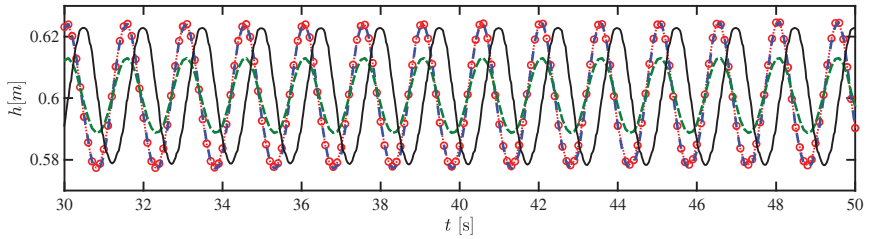

FIGURE 9. Wave elevation at probe location $x = 7.5$m. Lines indicate case1 (red solid line), case2 (blue dashed line), case3 (green dash-dotted line) and case4 (black solid line).

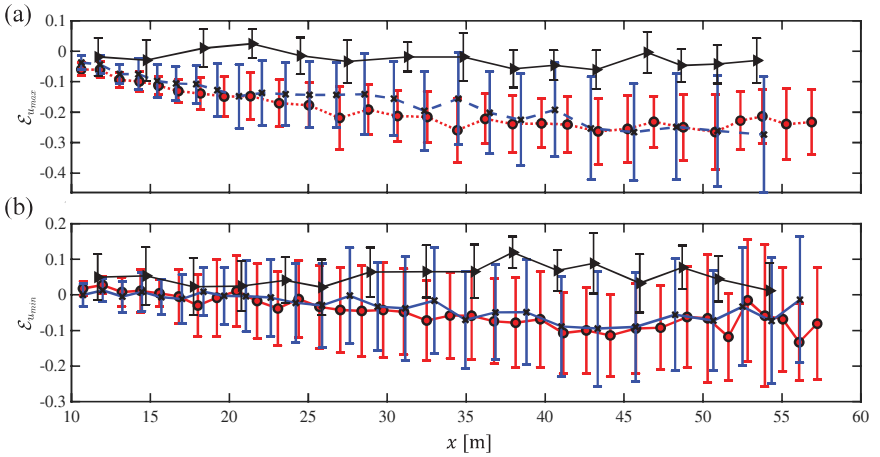

FIGURE 10. Velocity error at (a) crests $\mathcal{E}_{u_{\max}}$ and (b) troughs $\mathcal{E}_{u_{\min}}$ as a function of time. The error is measured at $y = 0.4861$ m for crests and at $y = 0.4472$ m for troughs. Colors of lines and symbols indicate case1 (red), case2 (blue) and case4 (black).

3.5. *Velocity error*

Velocity error is quantified for all cases following the approach detailed in Section 2.3.3. Initially, we measured the velocity error at crests $\mathcal{E}_{u_{\max}}$ and troughs $\mathcal{E}_{u_{\min}}$ as a function of time. Data were sampled after the transient period $t > 10$ s within the investigation window, with 20 samples grouped to yield a single data point where its x-axis location represents the averaged time over 20 samples.

Figure 10 displays the velocity error at the 8th point from the bottom, located at a height high enough to avoid the not–fully resolved bottom wall region but low enough to ensure that the measurement point consistently remains below the interface.

As simulation time progresses, the magnitude of the velocity errors as well as their standard deviation for case1 and case2 increases for both crests and troughs. Conversely, the DI method (case4) demonstrates relatively constant velocity errors with the least magnitude. Notably, case3, which exhibited the largest error $\mathcal{E}_u > 0.5$, deviates significantly from the targeted theoretical solution, and thus, is not included in the results.

Next, the velocity errors for the time window $t > 10$ s are displayed in Figure 11 as a function of water depth. The first point ($y = 0$ m) is omitted due to the no-slip condition at the wall. Additionally, the last point ($y = h + H/2$) is removed as the data points are at the interface, where a small height error could lead to sampling air velocity data.

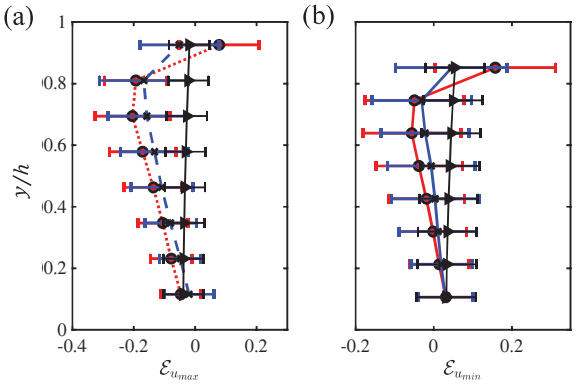

FIGURE 11. Velocity error at (a) crests $\mathcal{E}_{u_{\max}}$ and (b) troughs $\mathcal{E}_{u_{\min}}$ as a function of water depth. Colors of lines and symbols indicate case1 (red), case2 (blue) and case4 (black).

Velocity errors near the wall, where the velocity magnitude is small, exhibit similar error magnitudes ($\mathcal{E}_{u_{\max}} = -0.039$ at $y = 0.069$ m and $\mathcal{E}_{u_{\min}} = 0.032$ at $y = 0.063$ m) across all cases. Negative errors for crests and positive errors for troughs indicate a reduced velocity magnitude due to the presence of the wall. The maximum magnitude of the velocity error at crests for each case occurs at different heights, $\mathcal{E}_{u_{\max}} = -0.2$ at $y = 0.49$ m for case1, $\mathcal{E}_{u_{\max}} = -0.167$ at $y = 0.56$ m for case2, and $\mathcal{E}_{u_{\max}} = -0.02$ at the highest point ($y = 0.63$ m) for case4. The DI method (case4) demonstrates overall good accuracy across all heights. Abrupt sign changes of the velocity error near the interface may be related to errors induced by the interface advection. Similarly, for troughs, the maximum magnitude of the velocity error below the point nearest to the interface shows comparable values: $\mathcal{E}_{u_{\min}} = -0.05$ at $y = 0.45$ m for case1, $\mathcal{E}_{u_{\min}} = -0.03$ at $y = 0.51$ m for case2, and $\mathcal{E}_{u_{\min}} = 0.05$ at $y = 0.58$ m for case4. Abrupt sign changes are observed similar to the crest cases. For both crests and troughs, the DI method (case4) consistently demonstrates negative velocity errors for crests and positive errors for troughs, indicating a smaller orbital motion of the wave compared to the wave theory.

In addition to the averaged velocity error, the instantaneous velocity fields are provided in Figure 12 for all cases. Only case4 demonstrated the organized wave pattern throughout the simulation domain except near the outlet boundary face. This wave pattern should be maintained theoretically but all other cases showed disturbed velocity pattern.

4. Conclusions

This study investigates the performance of the two-phase solver utilizing an accurate and robust DI method on unstructured framework (Hwang & Jain 2023) against the well–established open-source CFD toolboxes, OpenFOAM®, in generating linear waves in intermediate water depth.

The conventional SBM technique induced an erroneous velocity fluctuation when utilized with the DI method. Consequently, a novel hybrid method for wave generation was proposed to address this issue. The hybrid method not only conserved almost the total liquid volume but also exhibited significantly lower fluctuations compared to the SBM method. Moreover, it successfully generated the targeted wave condition, allowing smooth propagation downstream. A thorough analysis of height and velocity errors high-

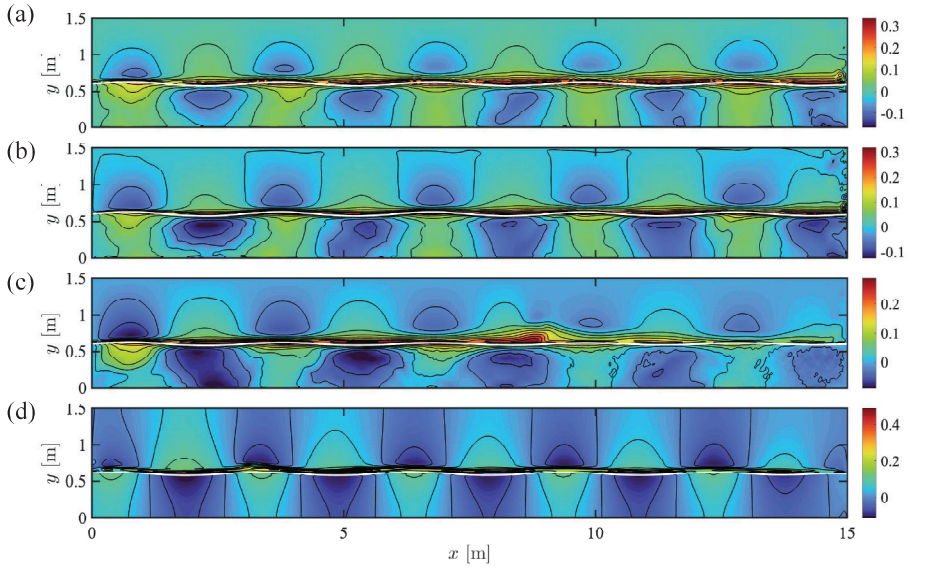

FIGURE 12. Instantaneous velocity fields at $t = 60$ s for (a) case1, (b) case2, (c) case3 and (d) case4. Colormaps and contours indicate velocity magnitudes. The interface is shown as white solid lines.

lighted the superior performance of the two-phase solver employing the DI method over other methodologies.

Furthermore, various probe techniques were assessed for the DI method within an unstructured framework. Among these techniques, the ψ-based probe, relying on the signed-distance-like auxiliary function ψ, demonstrated the most accurate interface-capturing capability. This probe method provided reliable and precise identification of the interface under consideration.

Acknowledgments

The authors would like to acknowledge the computing time on the Quartz machine at LLNL made available through the Predictive Science Academic Alliance Program (PSAAP) III at Stanford University.

REFERENCES

BOOIJ, N., RIS, R. C. & HOLTHUIJSEN, L. H. 1999 A third-generation wave model for coastal regions: 1. model description and validation. *J. Geophys. Res.-Oceans* **104**, 7649–7666.

CHINI, N., STANSBY, P., LEAKE, J., WOLF, J., ROBERTS-JONES, J. & LOWE, J. 2010 The impact of sea level rise and climate change on inshore wave climate: A case study for East Anglia (UK). *Coast. Eng.* **57**, 973–984.

DAMIÁN, S. M. & NIGRO, N. M. 2014 An extended mixture model for the simultaneous treatment of small-scale and large-scale interfaces. *Int. J. Numer. Method. Fluids* **75**, 547–574.

DODET, G., MELET, A., ARDHUIN, F., BERTIN, X., IDIER, D. & ALMAR, R. 2019 The

contribution of wind-generated waves to coastal sea-level changes. *Surv. Geophys.* **40**, 1563–1601.

Engsig-Karup, A. P., Hesthaven, J. S., Bingham, H. B. & Madsen, P. A. 2006 Nodal DG-FEM solution of high-order Boussinesq-type equations. *J. Eng. Math.* **56**, 351–370.

George, J. & Kumar, V. S. 2019 Long-term variations in nearshore wave climate and littoral drift at Ganpatipule coast, eastern Arabian sea and its link to Pacific climate variability. *J. Geophys. Res.-Oceans* **124**, 7307–7327.

Higuera, P. 2015 Application of computational fluid dynamics to wave action on structures. *PhD thesis, Universidade de Cantabria* .

Higuera, P., Lara, J. L. & Losada, I. J. 2013 Realistic wave generation and active wave absorption for Navier–Stokes models: Application to OpenFOAM®. *Coast. Eng.* **71**, 102–118.

Higuera, P., Liu, P. L.-F., Lin, C., Wong, W.-Y. & Kao, M.-J. 2018 Laboratory-scale swash flows generated by a non-breaking solitary wave on a steep slope. *J. Fluid Mech.* **847**, 186–227.

Hwang, H. & Jain, S. S. 2023 A robust phase-field method for two-phase flows on unstructured grids. arXiv: 2310.10795.

Jacobsen, N. G., Fuhrman, D. R. & Fredsøe, J. 2012 A wave generation toolbox for the open-source CFD library: OpenFOAM®. *Int. J. Numer. Method. Fluids* **70**, 1073–1088.

Jain, S. S. 2022 Accurate conservative phase-field method for simulation of two-phase flows. *J. Comput. Phys.* **469**, 111529.

Janssen, P. A. & Bidlot, J.-R. 2023 Wind–wave interaction for strong winds. *J. Phys. Oceanogr.* **53**, 779–804.

Lara, J. L., del Jesus, M. & Losada, I. J. 2012 Three-dimensional interaction of waves and porous coastal structures: Part II: Experimental validation. *Coast. Eng.* **64**, 26–46.

Nielsen, P. 1992 *Coastal Bottom Boundary Layers and Sediment Transport*, Advanced Series on Ocean Engineering, Vol. 4. World Scientific.

Roenby, J., Bredmose, H. & Jasak, H. 2016 A computational method for sharp interface advection. *Roy. Soc. Open Sci.* **3**, 160405.

Roenby, J., Larsen, B. E., Bredmose, H. & Jasak, H. 2017 A new volume-of-fluid method in OpenFOAM. In *MARINE VI: Proceedings of the VI International Conference on Computational Methods in Marine Engineering*, pp. 266–277. CIMNE.

Sandhya, K., Nair, T. B., Bhaskaran, P. K., Sabique, L., Arun, N. & Jeykumar, K. 2014 Wave forecasting system for operational use and its validation at coastal Puducherry, east coast of India. *Ocean Eng.* **80**, 64–72.

Tolman, H. L. *et al.* 2009 User manual and system documentation of WAVEWATCH III TM version 3.14. *Technical note, MMAB Contribution* **276** (220).

Windt, C., Davidson, J., Schmitt, P. & Ringwood, J. V. 2019 On the assessment of numerical wave makers in CFD simulations. *J. Mar. Sci. Eng.* **7**, 47.

Zhang, S., Zhang, Y., Xu, J., Guo, L., Li, G., Jia, Y., Qiao, L., Wu, J., Wen, M. & Zhu, C. 2022 In situ observations of hydro-sediment dynamics on the abandoned Diaokou lobe of the Yellow River Delta: Erosion mechanism and rate. *Estuar., Coast. Shelf Sci.* **277**, 108065.