Applied Ocean Research 158 (2025) 104523

Contents lists available at ScienceDirect

Applied Ocean Research

journal homepage: www.elsevier.com/locate/apor

Research paper

An improved direct forcing immersed boundary method for floating body simulations in waves

Ahmet Soydan *, Widar W. Wang, Hans Bihs

Norwegian University of Science and Technology, Høgskoleringen 7A, 7491 Trondheim, Norway

ARTICLE INFO

Keywords:
6DOF
Floating
Level set method
Immersed boundary

ABSTRACT

This paper introduces a novel direct forcing immersed boundary method tailored to simulate nonlinear interactions between ocean waves and arbitrarily complex free-floating structures. Within an open-source hydrodynamic framework, we couple the fluid–structure interaction (FSI) algorithm with the two-phase flow solver through forcing at the fluid–solid interface. We substantially enhance this coupling process by altering the density interpolation method, significantly reducing the interface smearing region, which improves both the stability and accuracy of the fluid flow in the vicinity of the floating objects. The tracking of the fluid–solid interface in the Eulerian domain is based on a level set function, thus avoiding the need for dynamically moving or overset meshes and greatly simplifying the mesh generation process. Rigid body dynamics are implemented using Euler parameters and Hamiltonian mechanics, allowing for arbitrarily large motions of the floating body. The presented approach is tested and validated with several 2D and 3D problems, including a full-scale simulation of a floating semi-submersible offshore wind turbine in waves. All numerical results demonstrate the accuracy and robustness of the new method, highlighting its potential as an outstanding alternative to existing numerical approaches for realistic floating-body simulations in waves.

1. Introduction

In the coastal and ocean engineering fields, fluid–structure interaction plays a key role in addressing real-world engineering problems. Given floating structures such as ships, floating wind turbines, or open-ocean aquaculture structures, interactions with the free surface lead to intricate hydrodynamic challenges. Various non-linear activities associated with diffraction, radiation, breaking waves, slamming, overtopping, and wave run-up may occur in fluid-floating object interactions in waves. Not all of these phenomena can be fully captured by potential-based solvers (Tran and Kim, 2016). Therefore, accurate, efficient and robust algorithms based on the Navier–Stokes equations are extremely vital for such problems.

Several numerical approaches for modeling the rigid floating body-fluid interaction have been introduced in the past and they can be broadly classified as explicit and implicit rigid-body representations in the fluid domain. The explicit method Arbitrary Lagrangian-Eulerian (ALE) approach, also known as body-fitted method, uses a single Eulerian grid around the structure which dynamically adapts to the motion of the structure by moving the grid lines accordingly. The implementation of this method is relatively straightforward and the boundary layer can be resolved by refining the cells around the body. However, it is relatively cumbersome due to the grid deformation and re-meshing procedure, as specified by Yang and Stern (2009). It is challenging to handle large deformations of the mesh while maintaining numerical stability. For example, Katsidoniotaki and Göteman (2022) demonstrate that large surge and pitch motions of the structure drastically reduce mesh quality, leading to numerical instability and causing the simulation to fail.

The overset method (Borazjani et al., 2013; Carrica et al., 2007; Chen et al., 2019) employs two distinct Eulerian grids. One of these grids remains fixed, while the other follows the movement of the floating object. This approach is capable of managing large motions and ensuring stability because the moving grid remains undisturbed. On the other hand, it does require an interpolation mechanism for communication between the overlapping grids (Hanaoka, 2013) which may affect the computational efficiency and accuracy. Vreman (2020) evaluated the computational efficiency of the overset method and showed that the moving boundary requires a large computational time because of the pressure interpolation on overset boundaries during each iteration of the Poisson equation. The overset method is widely used for several applications of CFD simulations. For example, high-fidelity fluid–structure interaction simulations of floating wind turbines (FOWT) based on the overset method was carried out using both

---

* Corresponding author.

E-mail addresses: ahmet.soydan@ntnu.no (A. Soydan), widar.w.wang@ntnu.no (W.W. Wang), hans.bihs@ntnu.no (H. Bihs).

https://doi.org/10.1016/j.apor.2025.104523
Received 16 October 2024; Received in revised form 7 March 2025; Accepted 10 March 2025

commercial software STAR-CCM+ (Tran and Kim, 2016, 2018; Zhang and Kim, 2018) and the open-source software OpenFOAM (Pinguet et al., 2020, 2021; Campaña-Alonso et al., 2023; Eskilsson et al., 2023; Barajas et al., 2024). Furthermore, the overset method is used in the numerical modeling of Wave Energy Converters (WECs) under extreme wave conditions (Katsidoniotaki and Göteman, 2022; Van Rij et al., 2021; Aliyar et al., 2022; Khan et al., 2024; Chen et al., 2024), as well as in the analysis of self-propulsion and maneuvering characteristics of ships (Delen et al., 2021; Sakamoto et al., 2019; Shen et al., 2015), as demonstrated by earlier studies. Numerical findings show that the overset method provides stability and can handle large motions of the structure, but comes with relatively large computational costs.

The immersed boundary method (IBM) (Peskin, 1972, 1977) is developed as an implicit rigid body representation in a fluid domain to overcome the constraints and limitations of explicit rigid body representation methods. The IBM, a non-boundary-conforming approach, was originally introduced by Peskin (1972) to simulate fluid–structure interaction (FSI) problems associated with human heart valves. It has gained attention due to its simplicity and flexibility, leading to the development of several variants aimed at improving its accuracy and expanding its applicability to real-world problems. Fadlun et al. (2000) introduced a direct forcing immersed boundary method (DFIBM), known as the sharp face direct forcing scheme. This method imposes the velocity boundary condition directly on a given immersed boundary and allows for the use of larger time steps compared to Peskin's method. While the initial applications of the direct forcing method focused on stationary bodies, it became apparent that when dealing with moving objects, spurious force oscillations could occur due to sudden changes in the relative positions between the fixed grid and the arbitrarily moving object. To address this issue, Uhlmann (2005) combined discrete delta function kernels with direct forcing formulations to resolve the problem of spurious force oscillations. In this approach, Eulerian velocities at fixed grid points are interpolated to Lagrangian points on the immersed boundary through a regularized Dirac delta function. Subsequently, the local forcing term is calculated corresponding to the desired velocity, which depends on the boundary conditions at the fluid–solid body interface. Finally, the local forcing is transferred to surrounding Eulerian locations using the regularized Dirac delta function. A similar interaction mechanism between Lagrangian and Eulerian locations for transferring quantities is also utilized by Kempe and Fröhlich (2012) and Kempe et al. (2015). A DFIBM was applied by Yang and Stern (2012) to address the fluid–solid interface by using a field extension method (Yang and Balaras, 2006). In scenarios involving moving boundaries, fluid cells may transition into solid cells and vice versa. This transition can result in unphysical values for the pressure and the velocities introducing spurious vorticity in the vicinity of the boundary. To mitigate this issue, the field extension method is used by means of extrapolation, ensuring the consistency of pressure gradients. Later, Yang and Stern (2015) proposed a non-iterative strongly coupled DFIBM for fluid–solid interaction problems based on the work of Yang and Stern (2012). Here, the field extension method was simplified, and the numerical stability was improved for low-density ratio problems.

As a further iteration in this regard, Yang and Stern (2009) presented a sharp interface DFIBM in a two-phase flow solver to represent the solid–fluid interface instead of overset grids, where a level set method was used for free-surface treatment. Calderer et al. (2014) presented a two-phase flow solver based on a level set approach for complex floating structures. In contrast to the Dirac function-based approach of Uhlmann (2005) and Yang (2018) proposed a formulation for two-phase fluid–structure interaction problems, where a Heaviside function is used to distribute the material properties and for interface tracking eliminating the necessity of using Lagrangian particles or overset meshes and simplifying the algorithm. The direct-forcing immersed boundary methods that have been presented in the literature solve either one-phase fluid problems for 6DOF motion (e.g Yang and Stern, 2012; Constant et al., 2017; Askarishahi, 2023) or address two-phase flow with prescribed motions, where the solid body is not freely moving (e.g Yang and Stern, 2009; Calderer et al., 2014; Yang, 2018; Zhang, 2020). In real-world applications, the solid body is freely moving due to the wave-structure interactions which create non-linearity with free-surface effects. Moreover, one of the biggest challenges for this method is to reduce the smoothing area separating the fluid and solid interface. The thicker smoothing area leads to non-sharp interfaces and a non-physical velocity field. For this reason, a thinner smoothing area is a key property to obtain accurate velocity and pressure fields.

In order to address those issues in the current study, a novel direct forcing immersed boundary method is implemented within the computational fluid dynamics (CFD) model of the open-source hydrodynamic framework REEF3D (Bihs et al., 2016). The software focuses on modeling ocean and coastal engineering problems such as breaking waves (Kamath et al., 2017a), as well as the analysis of breaking wave forces on cylindrical (Aggarwal et al., 2019a), porous (Sasikumar et al., 2020), and jacket structures (Aggarwal et al., 2019b). The method is built around a staggered rectilinear grid, resulting in tight pressure–velocity coupling, which is essential for stable two-phase flow modeling, due to the pressure and density jumps across the interface. Additionally, the rectilinear grid enables the use of high-order spatial and temporal discretization schemes, allowing the use of a relatively coarse mesh in the NWT. Moreover, the method is fully parallelized by using the domain decomposition method and MPI (Bihs et al., 2016), capable of scalability on a large number of processors.

A ghost-cell immersed boundary method, which was based on the approach of Berthelsen and Faltinsen (2008), was initially implemented in REEF3D to represent the motion of moving objects in a three-dimensional numerical wave tank (Kamath et al., 2017b; Bihs and Kamath, 2017). A triangulated STL file was used to represent the rigid body. A ray-casting algorithm was employed to determine the intersections of the cartesian grid and surface mesh of the solid body. The discrete surface area in each grid cell of the solid surface was calculated by means of a Dirac delta function with the level set representation of the solid surface. The position and orientation of the floating body were calculated with a second-order Adams–Bashforth time integration scheme. The method was governed by Euler angles, and the field extension method (Yang and Balaras, 2006) was implemented to prevent pressure oscillation due to the transition of the fluid cells into solid cells and vice versa. Subsequently, Martin et al. (2021b) improved the ghost-cell immersed boundary method by calculating the geometrical properties of the rigid body from a triangulated surface instead of a Dirac delta function with the level set representation of the solid surface. The motion of the floating body was captured by Euler parameters and Hamiltonian mechanics to overcome the Gimbal lock effect. Furthermore, a new quasi-static mooring model was implemented within this framework. This coupling model was validated across several applications (Martin et al., 2021b).

In this paper, the presented rigid-body FSI algorithm builds upon the original implementation of a continuous direct forcing approach in the numerical model (Martin et al., 2021a). The algorithm is significantly improved by introducing a new density interpolation method across the fluid–structure interface that removes unphysical spurious velocities in the vicinity of the interface, which may also affect the wave shape and amplitude at the free surface. The fluid density and viscosity are assumed to be the same inside and outside the immersed boundary. In that way, the material properties (density and viscosity) in two phases (air and water) can be smeared out through the solid–fluid interface. Consequently, it removes artificial density gradients in the vicinity of the floating body. This approach strongly reduces the smoothing area between the fluid and solid interface and ensures stable and accurate wave load calculations on floating objects. An arbitrarily complex structure is represented with an STL geometry. Sufficient information to create the level set function is provided through the use of an STL geometry by means of a ray casting algorithm (Bihs et al., 2017) which simplifies the mesh generation process significantly.

The presented method is based on Euler parameters and Hamiltonian mechanics preventing the gimbal lock effect, which results in the loss of a degree of freedom for Euler angles, allowing for the modeling of arbitrary large motions. An explicit third-order low-storage Runge–Kutta scheme (Spalart et al., 1991) is implemented for the calculation of equations of motion to enhance computational efficiency and stability. These features of the new rigid-body FSI algorithm enable modeling the free motion of arbitrarily complex structures in wave conditions.

The presented paper describes the new rigid-body FSI algorithm and demonstrates its capability for fluid–structure interaction problems. The new method is tested and validated with several benchmark cases including a full-scale floating semisubmersible offshore wind turbine. The numerical findings are then compared with experimental data to evaluate the accuracy and reliability of the method.

2. Numerical model

2.1. Governing equations

The conservation of mass and momentum written in convective form should be satisfied for an incompressible fluid.

$$\nabla\cdot\mathbf{u}=0 \tag{1}$$

$$\frac{\partial\mathbf{u}}{\partial t}+\mathbf{u}\cdot\nabla\mathbf{u}=-\frac{1}{\rho}\nabla p+\nu\nabla\cdot\left(\left[\nabla\mathbf{u}+\nabla\mathbf{u}^{T}\right]\right)+\mathbf{g} \tag{2}$$

Here, $\mathbf{u}$ [m/s] is the velocity vector, ρ [kg/m^3] is the density of the fluid, p [m^2/s^2] is the pressure, ν [m^2/s] is the kinematic viscosity, and $\mathbf{g}$ [m/s^2] the acceleration vector due to gravity. Turbulence effects are taken into account with a modified k-ω turbulence model (Bihs et al., 2016).

For the transition between the air and water phases, the level set function (Osher and Sethian, 1988) is used, which is defined as a signed distance function:

$$\phi(\vec{x},t)=\begin{cases}>0 & \text{if } \vec{x}\in \text{phase 1}\\ =0 & \text{if } \vec{x}\in \Gamma\\ <0 & \text{if } \vec{x}\in \text{phase 2}\end{cases} \tag{3}$$

A convection equation for the level set function is solved using the fluid velocity field $\mathbf{u}$.

$$\frac{\partial\phi}{\partial t}+\mathbf{u}\cdot\nabla\phi=0 \tag{4}$$

The level set function is reinitialized after each time step to maintain the signed distance properties. In the method, a PDE based reinitialization equation is solved (Sussman et al., 1994):

$$\frac{\partial\phi}{\partial t}+\operatorname{sign}(\phi)\left(|\nabla\phi|-1\right)=0 \tag{5}$$

where sign(ϕ) is the smoothed sign function (Peng et al., 1999). Eq. (5) helps to satisfy the Eikonal equation $|\nabla\phi|=1$ in order to ensure mass conservation. The density and viscosity are calculated using:

$$\rho=\rho_w H(\phi)+\rho_a(1-H(\phi)) \tag{6}$$

$$\nu=\nu_w H(\phi)+\nu_a(1-H(\phi)) \tag{7}$$

Here, the subscripts w and a indicate water and air properties, respectively. The following Heaviside function $H(\phi)$ is used for the smoothing of the sharp change of the fluid properties at the interface

$$H(\phi)=\begin{cases}0 & \text{if } \phi<-\epsilon\\ \frac{1}{2}\left(1+\frac{\phi}{\epsilon}+\frac{1}{\pi}\sin\left(\frac{\pi\phi}{\epsilon}\right)\right) & \text{if } |\phi|\le\epsilon\\ 1 & \text{if } \phi>\epsilon\end{cases} \tag{8}$$

with an interface thickness of $\epsilon=2.1\Delta x$. In the staggered grid approach, the density and viscosity are only required at the cell faces and are thus evaluated at this location directly in order to avoid unphysical oscillations arising from an interpolation procedure (see Bihs et al., 2016).

The system of equations is solved using finite differences on a rectilinear staggered grid. A fifth-order accurate weighted essentially non-oscillatory (WENO) scheme (Jiang and Shu, 1996) is applied for the convection terms. The fifth-order accurate Hamilton–Jacobi WENO method of Jiang and Peng (2000) is used for discretization of the convection term in Eq. (4). For the diffusion terms, second-order accurate central finite differences are applied. An incremental pressure-correction algorithm (Timmermans et al., 1996) is used for the solution of the pressure gradient term in the Reynolds-averaged Navier–Stokes equations (RANS) as described in Martin et al. (2021a). In the predictor step, the conservation equation for momentum Eq. (2) is solved considering the pressure gradient from the preceding Runge–Kutta step. An intermediate velocity field $\mathbf{u}^{(*)}$ is calculated by an explicit third-order low-storage Runge–Kutta scheme (Spalart et al., 1991) using the following equation:

$$\begin{aligned}\frac{\mathbf{u}^{(*)}-\mathbf{u}^{(k-1)}}{\Delta t} &= 2\alpha_k\nu\nabla\cdot\left(\left[\nabla\mathbf{u}+\nabla\mathbf{u}^{T}\right]\right)^{(k-1)}-2\alpha_k\nabla\left(\frac{p^{k-1}}{\rho}\right)\\ &\quad-\gamma_k\mathbf{u}^{k-1}\cdot\nabla\mathbf{u}^{k-1}-\zeta_k\mathbf{u}^{k-2}\cdot\nabla\mathbf{u}^{k-2}+\mathbf{g}\end{aligned} \tag{9}$$

Here, $\alpha_k = 4/15,\ 1/15,\ 1/6$, $\gamma_k = 8/15,\ 5/12,\ 3/4$, $\zeta_k = 0,\ -17/60,\ -5/12$ and $k = 1,\ 2,\ 3$. The third-order Total Variation Diminishing (TVD) Runge–Kutta scheme (Shu and Osher, 1988) is applied for the solution of the time derivatives for Eqs. (4) and (5). The time step size is controlled under the restriction of the Courant–Friedrichs–Lewy (CFL) condition to ensure numerical stability efficiently. An implicit Euler method is applied for the time advancement of the turbulent kinetic energy k [m^2/s^2] and the specific turbulent dissipation ω [1/s] equations to prevent a significantly smaller time step size due to the CFL criterion (Bihs et al., 2016). The diffusion term of the RANS equation is treated implicitly to remove it from the CFL restriction. The Poisson equation for the pressure correction is formed by the insertion of the intermediate velocities into the continuity equation.

$$\nabla\cdot\left(\frac{1}{\rho}\nabla p_{corr}\right)=\frac{1}{2\alpha_k\Delta t}\nabla\cdot\mathbf{u}^{(*)} \tag{10}$$

The Poisson equation (Eq. (10)) is solved for the pressure correction terms p_{corr} with the fully parallelized BiCGStab algorithm of the hypre library (van der Vorst, 1992) with the geometric multigrid preconditioner PFMG (Ashby and Flagout, 1996) to increase the computational performance. Finally, the pressure and the divergence-free velocity field are calculated using the updated pressure correction term:

$$p^{(k+1)}=p^{(k)}+p_{corr} \tag{11}$$

$$\mathbf{u}^{(k+1)}=\mathbf{u}^{*}-\frac{2\alpha_k\Delta t}{\rho}\nabla p^{(k+1)} \tag{12}$$

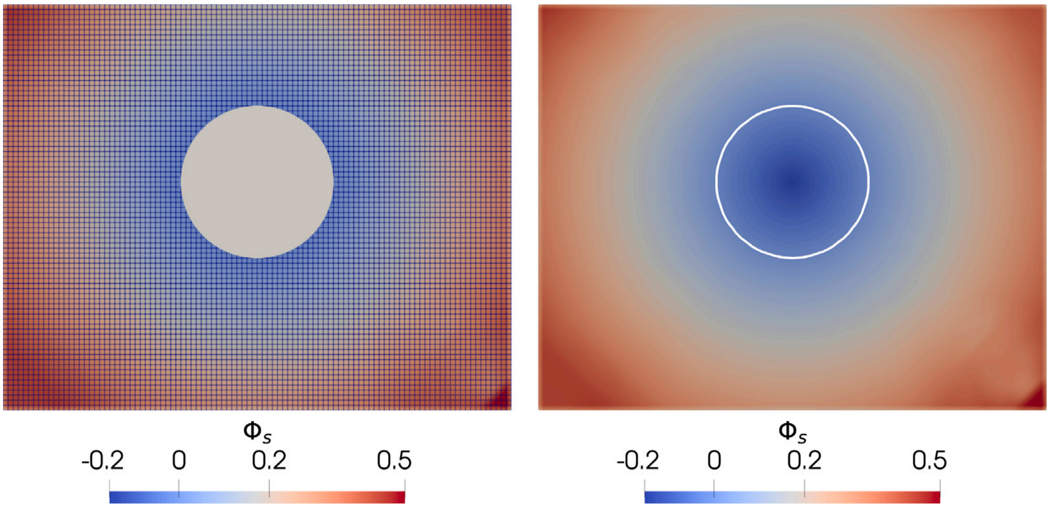
2.2. Rigid-body FSI algorithm

In this section, the fluid–structure interaction algorithm for the floating body is described. The rigid-body dynamics model is detailed in Section 2.2.1, where the new implementation of the equations of motion using an explicit third-order low-storage Runge–Kutta scheme (Spalart et al., 1991) is explained. The new density interpolation method is introduced in Section 2.2.2, and the corresponding algorithm is outlined in Section 2.2.3.

The governing equations are extended by adding the forcing term $\mathbf{f}$ in the momentum equations accounting for the rigid body in the fluid domain following the derivation in (Yang, 2018).

$$\nabla\cdot\mathbf{u}=0 \tag{13}$$

$$\frac{\partial\mathbf{u}}{\partial t}+\mathbf{u}\cdot\nabla\mathbf{u}=-\frac{1}{\rho}\nabla p+\mathbf{g}+\mathbf{f} \tag{14}$$



(a) STL geometry with mesh and level set function. White circle shows the STl geometry.

(b) Signed distance field with the zero level set for the fluid-solid interface. The $\Phi_s = 0$ are shown in white circle.

Fig. 1. Representation of the rigid-body with a level set function.

and the forcing term on the fluid and solid phases is as follows

$$\mathbf{f} = \begin{cases} \frac{\partial \mathbf{P}(\mathbf{u})}{\partial t} + \mathbf{P}(\mathbf{u}) \cdot \nabla \mathbf{P}(\mathbf{u}) + \frac{1}{\rho} \nabla p - \mathbf{g} & \text{if } \Phi_s < 0 \\ \nu \nabla \cdot \left(\left[\nabla \mathbf{u} + \nabla \mathbf{u}^T \right] \right) & \text{if } \Phi_s > 0 \end{cases} \tag{15}$$

where $\mathbf{P(u)}$, the projection operator, is used to project the velocity field into a divergence-free rigid body velocity field. The smoothed Heaviside function $H(\Phi_s)$ is introduced to represent the transition from fluid to solid. For this, the level set function Φ_s (Osher and Sethian, 1988), a signed distance function, is used to identify the fluid–solid interface. The information to create the signed distance field Φ_s is taken from an STL geometry representing the solid object consisting of multiple non-connected triangles.

A ray-casting algorithm (Bihs et al., 2017) is used to determine the closest distance information from neighboring fluid cells to the solid boundary exactly. This ensures that the signed distance values closest to the solid surface remain unaffected by the grid size, leading to a sharp representation of the solid–fluid interface. The closest distance is transformed into a signed distance function Φ_s representing the floating body in the Eulerian domain, with negative distance values indicating the inside of the solid and positive values representing the fluid region as can be seen in Fig. 1. The signed distance function Φ_s is reinitialized with the PDE-based Eq. (5) by Sussman et al. (1994) only for the cells away from the interface. This way the solid boundary obtained from the ray casting remains sharp and unaffected by any smoothing effects of the reinitialization. The level set function Φ_s is required as input for the Heaviside function in Eq. (19) for the activation of the forcing terms. The forcing term $\mathbf{f}$ at the next Runge–Kutta substep k+1 can be written in a discrete manner as:

$$\mathbf{f}^{(k+1)} = H\left(\Phi_s^{(k+1)}\right) \cdot \left(\frac{\mathbf{P}\left(\mathbf{u}^{(k+1)}\right) - \mathbf{P}\left(\mathbf{u}^{(k)}\right)}{\Delta t} + \mathbf{P}\left(\mathbf{u}^{(k)}\right) \cdot \nabla \mathbf{P}\left(\mathbf{u}^{(k)}\right) + \frac{1}{\rho} \nabla p^{(k+1)} - \mathbf{g} \right) \tag{16}$$

The pressure is taken from the previous time step and $\mathbf{P}\left(\mathbf{u}^{(n)}\right) = \mathbf{u}^{(n)}$ is used as an approximation to solve Eq. (16).

$$\mathbf{f}^{(k+1)} = H\left(\Phi_s^{(k+1)}\right) \cdot \left(\frac{\mathbf{P}\left(\mathbf{u}\right)^{(k+1)} - \mathbf{u}^{(k)}}{\Delta t} + \mathbf{u}^{(k)} \cdot \nabla \mathbf{u}^{(k)} + \frac{1}{\rho} \nabla p^{(k)} - \mathbf{g} \right) \tag{17}$$

Eq. (18) can be written by comparing with Eqs. (9) and (17).

$$\mathbf{f}^{(k+1)} = H\left(\Phi_s^{(k+1)}\right) \cdot \left(\frac{\mathbf{P}(\mathbf{u}^{(k+1)}) - \mathbf{u}^{*}}{\Delta t} \right) \tag{18}$$

$\mathbf{u}^{(*)}$ itself can be chosen as the updated velocity field. Then, the calculation of the predictor step (Eq. (9)) is first performed without the forcing term. Therefore, $\mathbf{f}^{(*)}$ is calculated as below.

$$\mathbf{f}^{(k+1)} \approx \mathbf{f}^{(*)} = H\left(\Phi_s^{(*)}\right) \cdot \left(\frac{\mathbf{P}(\mathbf{u}^{(*)}) - \mathbf{u}^{*}}{2\alpha_k \Delta t} \right) \tag{19}$$

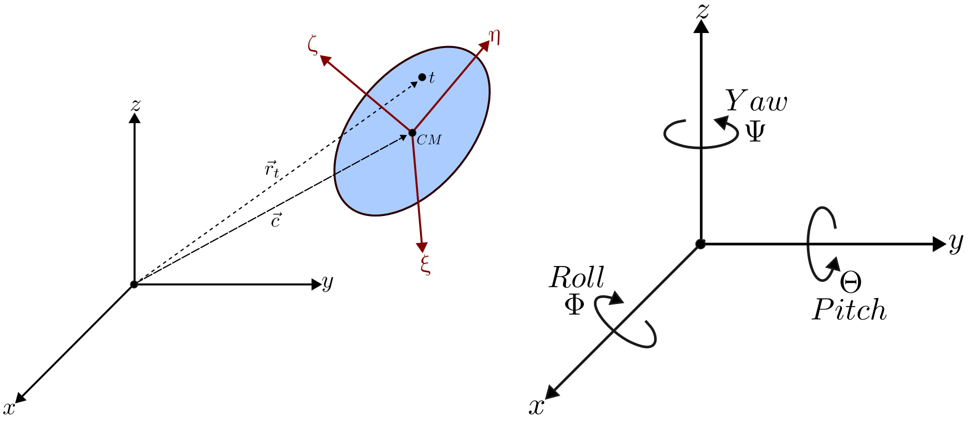
The forcing term then is added to the intermediate velocity field $\mathbf{u}^{(*)}$. Then, the Poisson equation (Eq. (10)) is solved, and with the pressure correction, the final velocity is obtained.

This approach is known as a direct-forcing immersed boundary method (Fadlun et al., 2000). The presented method employs the *one-fluid* formulation (Yang, 2018) since the three phases (air, water, and rigid body) are collectively represented by converting Eqs. (1) and (2) into Eqs. (13) and (14) for the entire domain. Consequently, there is no need to utilize Lagrangian particles or overset meshes in the formulation, providing significant simplicity and flexibility.

2.2.1. Rigid-body dynamics

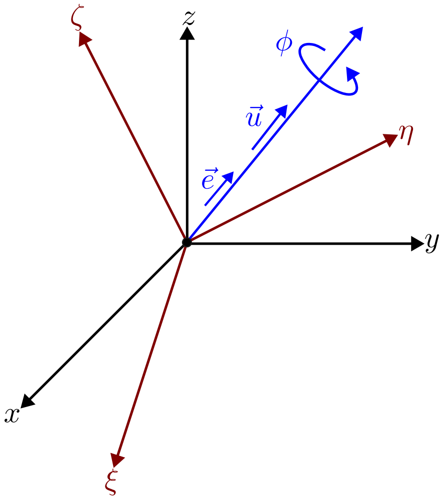
In this section, the approach for 3D rigid-body rotations and the coupling of rigid-body dynamics with the new FSI solver are explained. The coordinate systems and conventions which are used in this study are explained below:

- The center of mass of the rigid body $\mathbf{c} = (c_0, c_1, c_2)^T$ [m] is defined in the fixed inertia system (x, y, z) and the rotational motion of the rigid body is defined in a body-fixed coordinate system (ξ, η, ζ) as depicted in Fig. 2(a). The origin of the body-fixed coordinate system CM is located at the center of mass of the rigid body.
- The rotation of the rigid body around each axis is described by the three Euler angles (Φ, Θ, Ψ) [rad] as can be seen in Fig. 2(b). The angles follow the right-hand criteria on a right-handed Cartesian coordinate system so that rotations are positive clockwise in the direction of a positive axis.
- It is possible to use different conventions of Euler angles. However, the most common convention is the Tait-Bryan type of Euler angles. In this convention, the rotation around a different axis is defined by each of the three Euler angles. Here, the sequence of rotation matters because different sequences produce different results. The sequence in this study is defined in which the first rotation is the yaw angle Ψ about the z-axis, the second is the pitch angle Θ about the y-axis, and the last one is the roll angle Φ about the x-axis. This convention is known as *xyz convention* (Goldstein et al., 2001).



(a) Inertia (x, y, z) and body-fixed (ξ, η, ζ) coordinate systems. CM: The origin of the body-fixed coordinate system, $\vec{c}$: The coordinates of the rigid body center of mass, $\vec{t}$: A point of the triangles of the discrete geometry in the body-fixed coordinate system, $\vec{r}_t$: The coordinate of $\boldsymbol{t}$ in the inertia system.

(b) Illustration of the Euler angles. Φ: Roll angle, Θ: Pitch angle, Ψ: Yaw angle.



(c) Illustration of the Euler parameters (Nikravesh and Chung, 1982). $\vec{e}$: The quaternion, $\vec{u}$: A unit vector along the rotation axis, ϕ: Magnitude of rotation.

Fig. 2. Illustration of the coordinate systems, Euler angles and Euler parameters.

- Each rotation is executed on the body-fixed coordinate system that is rotated after the previous rotations. For example, the first rotation is executed around the z-axis. The second rotation then is executed around the y-axis in its new position updated by the first rotation. And, the last rotation is executed around the x-axis updated by the first two rotations. This system is known as *Intrinsic rotation* (Aye, 2011) system.

The orientation of a rigid body can be defined by a so-called rotation matrix. This matrix is most commonly constructed with Euler angles or Euler parameters (Flores, 2015). Here, Euler parameters (quaternions) $\mathbf{e} = (e_0, e_1, e_2, e_3)^T$ are selected for their significant advantages. The Euler parameters are free of the singularities, unlike the Euler angles. When using Euler angles in the rotation matrix, some ambiguities can occur in case the pitch angle Θ is $\pi/2$ or $-\pi/2$. In this case, $\cos\Theta = 0$ causes a singularity in the rotation matrix. As a result, the yaw and roll rotation axes align with each other, and therefore there is no unique solution for roll and yaw angles. This ambiguity is called the *gimbal lock effect*. Hence, the Euler parameters are free from the gimbal lock effect and can model arbitrarily large motions. Another advantage of using Euler parameters comes up when performing a concatenation of rotations. The quaternion multiplication is computationally faster and numerically more stable than the rotation matrix associated with Euler angles. This is because the rotation matrix constructed with the Euler angles is highly nonlinear (Flores, 2015) and a 3 × 3 rotation matrix multiplication is computationally expensive. In summary, the implementation of the new FSI algorithm is based on the Euler parameters, the coordinate system follows the right-hand criteria with xyz convention, considering the intrinsic rotation system.

The quaternion has four parameters with one real term and three imaginary terms. The real term of the quaternion e_o is defined as in Eq. (20) and the three imaginary terms are defined as in Eq. (21) (Nikravesh and Chung, 1982). Here, $\vec{e}$ is defined along the rotation axis with a ϕ the magnitude of the rotation and $\vec{u}$ a unit vector along the rotation axis. The property of this representation is that $\boldsymbol{e}^T\boldsymbol{e} = 1$.

$$e_0 = cos\frac{\phi}{2} \tag{20}$$

$$(e_1, e_2, e_3)^T = \boldsymbol{u}sin\frac{\phi}{2} \tag{21}$$

The Tait-Bryan type of Euler angles can be converted to quaternions as in Eq. (22) (here, c is cosine function and s is sine function) (Goldstein et al., 2001). These four parameters are then used to create the orthogonal rotation matrix. A vector in the body-fixed coordinate system can be transformed to a corresponding vector in the inertial system by means of the orthogonal rotation matrix, also called the quaternion matrix, as defined in Eq. (23) (Shivarama and Fahrenthold, 2004).

$$\begin{aligned}
e_0 &= c\left(\frac{\Phi}{2}\right)\cdot c\left(\frac{\Theta}{2}\right)\cdot c\left(\frac{\Psi}{2}\right) + s\left(\frac{\Phi}{2}\right)\cdot s\left(\frac{\Theta}{2}\right)\cdot s\left(\frac{\Psi}{2}\right)\\
e_1 &= s\left(\frac{\Phi}{2}\right)\cdot c\left(\frac{\Theta}{2}\right)\cdot c\left(\frac{\Psi}{2}\right) - c\left(\frac{\Phi}{2}\right)\cdot s\left(\frac{\Theta}{2}\right)\cdot s\left(\frac{\Psi}{2}\right)\\
e_2 &= c\left(\frac{\Phi}{2}\right)\cdot s\left(\frac{\Theta}{2}\right)\cdot c\left(\frac{\Psi}{2}\right) + s\left(\frac{\Phi}{2}\right)\cdot s\left(\frac{\Theta}{2}\right)\cdot s\left(\frac{\Psi}{2}\right)\\
e_3 &= c\left(\frac{\Phi}{2}\right)\cdot c\left(\frac{\Theta}{2}\right)\cdot s\left(\frac{\Psi}{2}\right) - s\left(\frac{\Phi}{2}\right)\cdot s\left(\frac{\Theta}{2}\right)\cdot c\left(\frac{\Psi}{2}\right)
\end{aligned} \tag{22}$$

$$\mathbf{R} = \mathbf{E}\mathbf{G}^T \tag{23}$$

with

$$\mathbf{E} = \begin{pmatrix} -e_1 & e_0 & -e_3 & e_2 \\ -e_2 & e_3 & e_0 & -e_1 \\ -e_3 & -e_2 & e_1 & e_0 \end{pmatrix} \tag{24}$$

$$\mathbf{G} = \begin{pmatrix} -e_1 & e_0 & e_3 & -e_2 \\ -e_2 & -e_3 & e_0 & -e_1 \\ -e_3 & e_2 & -e_1 & e_0 \end{pmatrix} \tag{25}$$

Euler angles can be calculated with knowledge of the Euler parameters as in Eq. (26).

$$\begin{aligned}
\Psi &= \text{arctan2}(2\cdot(e_1\cdot e_2 + e_3\cdot e_0), 1 - 2\cdot(e_2\cdot e_2 + e_3\cdot e_3))\\
\theta &= \text{arcsin}(2\cdot(e_0\cdot e_2 - e_1\cdot e_3))\\
\phi &= \text{arctan2}(2\cdot(e_2\cdot e_3 + e_1\cdot e_0), 1 - 2\cdot(e_1\cdot e_1 + e_2\cdot e_2))
\end{aligned} \tag{26}$$

The rigid-body dynamics modeling is built upon the combination of Euler parameter kinematics and Hamiltonian mechanics. Hamiltonian mechanics is an energy-based formulation and it offers a model suitable for numerical implementation. This combination offers a convenient numerical integration model when addressing strongly nonlinear problems (Shivarama and Fahrenthold, 2004). The momentum vector is defined as in Eq. (27), where $\boldsymbol{\omega}_b$ [rad/s] is the angular velocity vector in the body-fixed coordinate system and $\mathbf{I}$ [kg m^2] is the moment of inertia tensor.

$$\mathbf{h} = \boldsymbol{I}\boldsymbol{\omega}_b \tag{27}$$

The kinematic equations for the rotational motion of the rigid body concerning the time derivatives of the Euler parameters are defined in the body-fixed coordinate system as in Eq. (28).

$$\dot{\mathbf{e}} = \frac{1}{2}\mathbf{G}^T\boldsymbol{I}^{-1}\mathbf{h} \tag{28}$$

A first-order ODE is derived for the time derivative of the momentum vector $\dot{\mathbf{h}}$ using a Hamiltonian system (Shivarama and Fahrenthold, 2004). The equation can be written by forcing the potential energy function to zero and imposed moments $\boldsymbol{M_b}$ [N m] in the body-fixed system as:

$$\dot{\mathbf{h}} = -2\mathbf{G}\dot{\mathbf{G}}^T\mathbf{h} + \boldsymbol{M_b} \tag{29}$$

The moments in the body-fixed system $\boldsymbol{M_b}$ can be obtained from the moments in the inertia system $\boldsymbol{M_i}$ by means of the transformation matrix (Eq. (23)), $\boldsymbol{M_b} = \boldsymbol{R}^{-1}\boldsymbol{M_i}$. In the Hamiltonian model, the time derivative of the mass center $\dot{\boldsymbol{c}}$ [m/kg s] is defined as in Eq. (30). Here, $\mathbf{p}$ [m/s] is the velocity of the mass center in the inertia system and m_s [kg] is the mass of the floating body.

$$\dot{\mathbf{c}} = \frac{\mathbf{p}}{m_s} \tag{30}$$

Thus, the time derivative of the velocity of the mass center $\dot{\mathbf{p}}$ [kg m/s^2] can be defined in the inertia system as a first-order differential equation by applying Newtons' second law as below.

$$\dot{\mathbf{p}} = \mathbf{F}_i \tag{31}$$

The position and velocity of a floating body's center of gravity are calculated in the inertia system by integrating Eqs. (30) and (31). Overall, Eqs. (30), (31), (28) and (29) comprise a system of thirteen first-order ODEs for equations of motion. The system of equations of motion is integrated with the same explicit third-order low-storage Runge–Kutta scheme (Spalart et al., 1991) as the fluid solver to calculate the velocity of the mass center, the position of the mass center, the quaternion, and the momentum vector as given Eqs (32)–(35).

$$\boldsymbol{p}^k = \boldsymbol{p}^{k-1} + \gamma_k\Delta t\dot{\boldsymbol{p}}^k + \zeta_k\Delta t\dot{\boldsymbol{p}}^{k-1} \tag{32}$$

$$\boldsymbol{c}^k = \boldsymbol{c}^{k-1} + \gamma_k\Delta t\dot{\boldsymbol{c}}^k + \zeta_k\Delta t\dot{\boldsymbol{c}}^{k-1} \tag{33}$$

$$\boldsymbol{h}^k = \boldsymbol{h}^{k-1} + \gamma_k\Delta t\dot{\boldsymbol{h}}^k + \zeta_k\Delta t\dot{\boldsymbol{h}}^{k-1} \tag{34}$$

$$\boldsymbol{e}^k = \boldsymbol{e}^{k-1} + \gamma_k\Delta t\dot{\boldsymbol{e}}^k + \zeta_k\Delta t\dot{\boldsymbol{e}}^{k-1} \tag{35}$$

During each Runge–Kutta substep, the rigid-body position is adjusted by updating the level set position of the rigid body using the ray-casting algorithm after calculating the translational and rotational motion of the rigid body. Let us suppose that $\boldsymbol{t} = (\xi, \eta, \zeta)$ is a point of the triangles of the discrete geometry in the body-fixed coordinate system, $\vec{r}_t(x, y, z)$ is the coordinate of $\boldsymbol{t}$ in the inertia system and $\vec{c}(x, y, z)$ is the coordinates of the rigid body center of mass as depicted in Fig. 2(a). The updated coordinates of the point in the inertia system are calculated with the rotation matrix in each Runge–Kutta substep by applying Eq. (36).

$$\boldsymbol{r}_t(x, y, z) = \boldsymbol{c}(x, y, z) + \boldsymbol{R}\boldsymbol{t}(\xi, \eta, \zeta) \tag{36}$$

The forces $\boldsymbol{F_i}$ $[N]$ and momentum $\boldsymbol{M_i}$ [N m] acting on the floating body in the inertia system are calculated by integrating the fluid properties (pressure p, the viscous stress tensor τ) over the discrete solid surface Ω via the discrete areas of the triangular surface mesh of the rigid body.

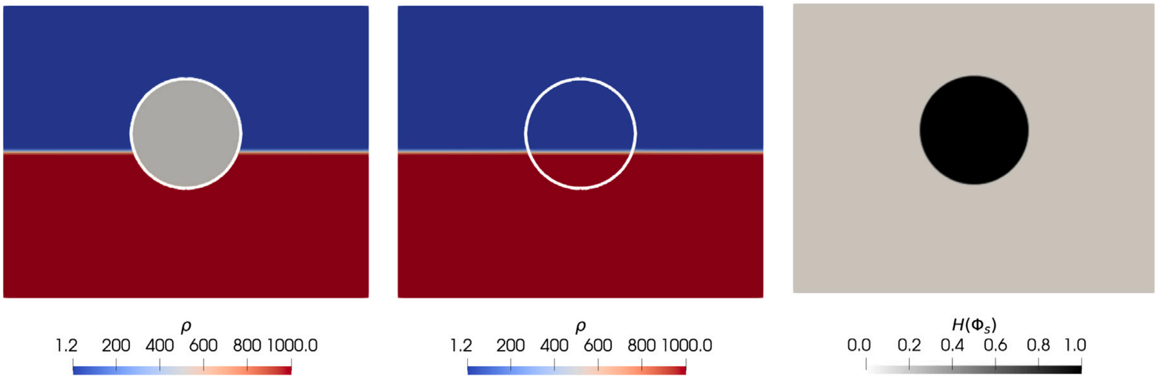
$$\boldsymbol{F_i} = \int_\Omega (-\mathbf{n}p + \rho\nu\mathbf{n}\tau)d\Omega(\mathbf{x}) = \sum_{i=1}^{N}(-\mathbf{n}p + \rho\nu\mathbf{n}\tau)_i\cdot\Delta\Omega_i \tag{37}$$

$$\boldsymbol{M_i} = \int_\Omega \mathbf{r}\times(-\mathbf{n}p + \rho\nu\mathbf{n}\tau)d\Omega(\mathbf{x}) = \sum_{i=1}^{N}\mathbf{r}_i\times(-\mathbf{n}p + \rho\nu\mathbf{n}\tau)_i\cdot\Delta\Omega_i \tag{38}$$

Here, $\mathbf{n}$ is the surface normal vector on the solid body surface pointing outwards, and $\mathbf{r}$ is the distance vector from the center point of an individual triangle of the discrete solid surface to the center of gravity of the solid body. Additionally, the mooring forces are independently calculated as external forces, using a finite-element-based quasi-static mooring model (Martin et al., 2021b), and subsequently added to the total force during each Runge–Kutta substep.

The rigid-body velocity field in the inertial reference frame is calculated as Eq. (39), where $\boldsymbol{\omega}_i$ is the angular rigid-body velocity vector in the inertial reference frame. Here, the angular rigid-body velocity vector $\boldsymbol{\omega}_b$ is transferred to the inertial reference frame by using the transformation matrix as $\boldsymbol{\omega}_i = \mathbf{R}\boldsymbol{\omega}_b$.

$$\mathbf{P}\left(\mathbf{u}^{(*)}\right) = \frac{\boldsymbol{p}}{m_s} + \boldsymbol{\omega}_i\times\boldsymbol{r} \tag{39}$$



(a) Discontinuous solid density. White circle represents the riding body.

(b) Continuous fluid density. White circle represents the riding body.

(c) Solid Heaviside Function, representing the transition layer.

Fig. 3. Floating body density treatment.

2.2.2. *Density interpolation method*

In the presented method, the material properties of the three phases (air, water and rigid body) are assigned for the entire Eulerian domain with the level set function. The pressure and the density information is needed to solve the correction of the velocity field in Eq. (12). The pressure information is obtained by solving the Poisson equation in Eq. (10). In Fig. 3, two different density interpolation methods are represented along with the representation of the solid Heaviside function given in Eq. (8). As can be seen in Fig. 3(c), the Heaviside function takes the values of 1 or 0 inside and outside of the rigid body, respectively. In the transition layer, the function is smeared out. In the previous approach (Martin et al., 2021a) depicted in Fig. 3(a), the material properties of the solid body are assigned using the solid Heaviside function $H(\Phi_s)$ as:

$$\rho(\phi,\Phi_s) = \rho_s H(\Phi_s) + (1 - H(\Phi_s)) \cdot \left(\rho_w H(\phi) + \rho_a (1 - H(\phi))\right) \tag{40}$$

$$\nu(\phi,\Phi_s) = (1 - H(\Phi_s)) \cdot (\nu_w H(\phi) + \nu_a (1 - H(\phi))) \tag{41}$$

This approach results in a strong density gradient in the vicinity of the body and leads to spurious velocity when solving the correction of velocity in Eq. (12). In addition, the spurious velocity causes instabilities in solving the Poisson equation in Eq. (10). Maintaining stability requires a relatively large transition layer (ϵ) between the fluid and the solid body, typically around $2.1dx$. In the proposed new approach, the continuous fluid density treatment is implemented based on Eqs. (6) and (7), as illustrated in Fig. 3(b). Here, the fluid density and viscosity are assumed to be the same inside and outside the immersed boundary. In that way, the material properties (density and viscosity) in the two phases (air and water) can be smeared out through the solid–fluid interface. This approach excludes the floating body's density from the formulation and it removes artificial density gradients in the vicinity of the floating body. Consequently, it becomes feasible to employ relatively coarser grid cells in the vicinity of the floating body, along with a thinner transition layer ($\epsilon = 0.6dx$) separating the fluid and the solid body. This methodology is subjected to testing and validation within the scope of the current research study.

2.2.3. *Algorithm of the method*

The flowchart of the algorithm is shown in Fig. 4. The new rigid-body FSI algorithm can be summarized as follows:

1. Initializing all variables or taking the values from the previous time step for fluid (the fluid velocity and pressure) and rigid body (the position, velocity, and angular momentum) to be used in the next step.
2. For k = 1, the momentum predictor step (Eq. (14)) starts without the forcing term.
3. Calculate the forces on the discrete rigid body (Eqs. (37) and (38)).
4. Solve the equations of motion (Eqs. (28)–(31)).
5. Update the Euler parameters and the level set position of the rigid body by transforming the coordinates of the triangles of the discrete geometry (Eq. (23)).
6. Calculate the rigid body velocity field (Eq. (39)).
7. Calculate the forcing term (Eq. (19)).
8. Add the forcing term to the predicted momentum (Eq. (14)).
9. Solve the Poisson equation to obtain new pressure and velocity values (Eqs. (10), (11), and (12)).
10. The momentum corrector step starts (Eq. (9)).
11. Repeat until the last step of the 3rd-order Runge–Kutta scheme.
12. Start the next time step.

2.3. *Comparison of the numerical models*

Several numerical models are used for FOWT simulations, such as potential theory (PT) (see e.g., Holcombe et al., 2023), Smoothed Particle Hydrodynamics (SPH) (see e.g., Tagliafierro et al., 2023b,a), finite volume method (FVM) with overset mesh (see e.g., Tran and Kim, 2016; Barajas et al., 2024; Eskilsson et al., 2023), and mesh deforming (see e.g., Ransley et al., 2023). The presented numerical model has notable advantages and some inherent limitations compared to existing numerical models for simulating FOWTs, as summarized in Table 1. FVM with mesh deforming often struggles to handle large motions, as demonstrated in Katsidoniotaki and Göteman (2022). Overset mesh-based FVM, on the other hand, requires computationally expensive pressure interpolation between overlapping grids during each iteration of the Poisson equation, as explained in Vreman (2020). SPH-based methods can handle large motion responses in extreme waves; however, they require a high number of particles for accuracy and are computationally expensive. The presented model uses a fixed background Cartesian grid and complex solid geometries are modeled with immersed boundary treatment. This model avoids mesh deformation and interpolation, achieving stable numerical calculations with Euler parameters and Hamiltonian mechanics for arbitrarily large motions. Additionally, generating a proper grid around the solid structure is cumbersome in FVM-based models. The presented method represents the solid structure with an STL geometry, simplifying the mesh generation process.

PT-based methods are computationally much faster and are widely used for early-stage FOWT design. However, PT-based methods are not suitable for nonlinear problems. Unlike FVM-based methods, where the

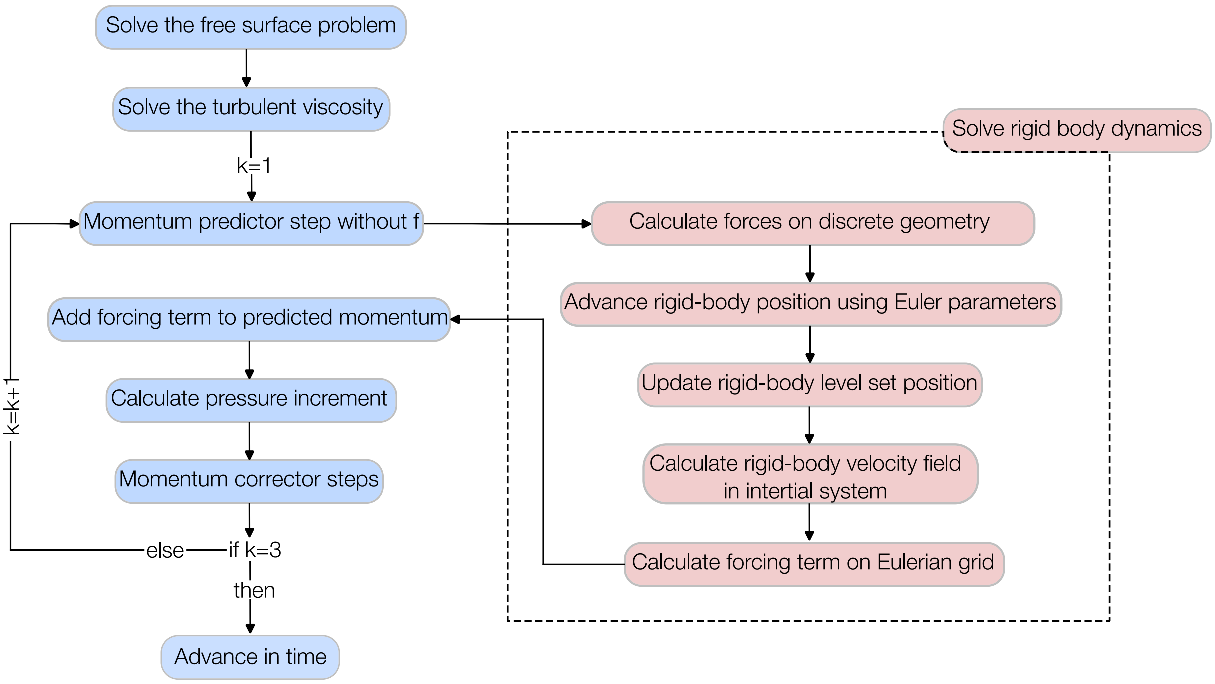


Fig. 4. Illustration of the 6DOF algorithm, k represents the Runge–Kutta substep iteration number.

Table 1
Comparison of the numerical models for FOWT simulations.

Model	Advantages	Limitations
FVM (Overset mesh)	Stable for large motions Suitable for complex geometries Resolve boundary layer	Expensive grid interpolation
FVM (Mesh deforming)	Resolve boundary layer	Instable for large motions
SPH	Mesh-free Handles extreme waves	High computational cost Numerical dissipation
PT (linear)	Fast Suitable for early-stage design	Inaccurate for strong nonlinear effects
DF-IBM (Present work)	No mesh distortion Stable for large motions High-order discretization	No body-fitting near-wall refinement

near-wall region can be refined, the present method relies on a uniform Cartesian grid, and the viscous sub-layer cannot be resolved explicitly. This can limit viscous force accuracy unless very fine grids are used, reducing computational efficiency. As future work, block mesh refinement will be implemented in the model to improve its performance for viscous force-dominant problems (e.g., ship hydrodynamic simulations).

3. Results

3.1. 2D heaving of a circular cylinder

For the validation of the new floating algorithm, a 2D heaving cylinder is simulated. The motion is restricted to the vertical direction so that it can oscillate up and down freely. The numerical result is compared to the experimental result taken from Itō (1977). The numerical setup is carried out similarly to Bihs and Kamath (2017) and Yang and Stern (2009). The dimensions of the numerical domain are set to 10 m × 1.8 m. The initial water depth is $d = 1.22$ m and the water density $\rho = 1000\ \mathrm{kg/m^3}$. The schematic representation of the simulation is illustrated in Fig. 5. Wall boundary conditions are implemented at the start, end, and bottom of the NWT, while symmetry conditions are applied to the boundaries along the width and top of the NWT. k-ω turbulence model is used for the simulation. The circular cylinder is placed in the center of the tank following Ito's configurations (Itō, 1977). The circular cylinder has a radius of $r = 0.0762$ m and density $\rho = 500\ \mathrm{kg/m^3}$. The centroid position of the cylinder is initially set to $h_1 = 0.02454$ m above the free surface. Then, the cylinder is dropped from the initial position in free fall. Four different uniform grids are applied for the grid convergence study, and three different CFL numbers, are considered for the temporal convergence by using the finest grid size dx = 0.002.

$$Normalization = \frac{Z - d}{h_1} \tag{42}$$

Fig. 6 shows the time series of the numerical vertical velocity of the cylinder. Fig. 7 illustrates the motion of the cylinder with the vorticity iso-surface at different times. The cylinder is released from its initial position and enters the water. The cylinder reaches its highest velocity of 0.121 m/s in the first period at $t = 0.5$ s. It generates waves on each side, which start to dissipate after $t = 2.5$ s. The waves are completely dispersed by the end of $t = 3.0$ s, and the motion of the cylinder is no longer observable. The results illustrate the capability of the new FSI method to predict the interaction of a rigid body with the free surface.

Fig. 8(a) and (b) show the heave motion results along with the experimental data. The y-axis represents the normalized position calculated by Eq. (42). The coarsest grid shows a good match with the

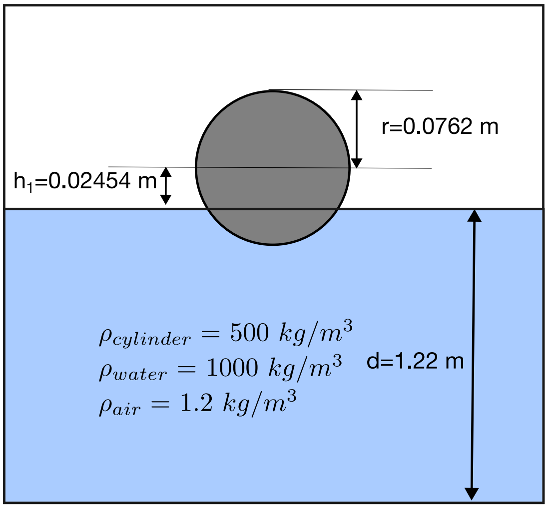


Fig. 5. Schematic description of the heaving cylinder.

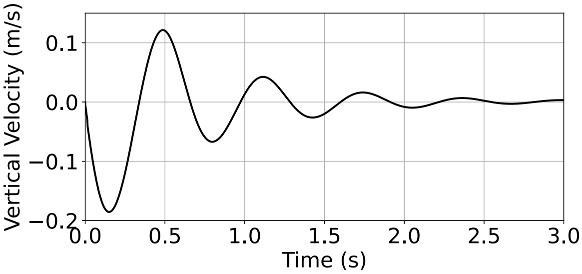


Fig. 6. Numerical vertical velocity of the cylinder.

experimental results until 0.5 s, after which it starts deviating after the first period of the motion. As the grid sizes are refined, the results converge monotonically with the experimental data. The results obtained with grid sizes $dx = 0.005$ m and $dx = 0.002$ m represent the heave motion more accurately. Furthermore, Fig. 8(b) indicates that the simulation with CFL 0.5 is slightly different from simulations with CFL 0.25 and 0.1. The presented numerical findings represent the robustness and accuracy of the new method.

For comparison of the density interpolation methods introduced in Section 2.2.2, vertical velocity contours at t = 1.0 s (Fig. 9) and the heave motion results (Fig. 10) are provided. Here, $dx = 0.005$ m and a CFL number of 0.25 are chosen for numerical simulations. The spurious velocity in the vicinity of the cylinder is evident with the old density interpolation method and it is removed with the new method. A discrepancy is observed in the phase and period of the heave motion results with the old density interpolation method, which is improved with the new approach. The maximum deviation between the experimental results and those obtained with the old and new methods is 12.2% and 5.9%, respectively. The numerical calculations are carried out on a M1 Max Macbook Pro. The computational times for 3-s simulations using 8 processors with the old and new approaches are 2097.05 s and 1093.65 s, respectively.

3.2. 2D disc entry

In this section, the new 6DOF algorithm is tested for the disc entry problem. This benchmark case has been studied by several researchers (Calderer et al., 2014; Yang and Stern, 2009; Bihs and Kamath, 2017) to validate their models. All the parameters used in this test case are non-dimensional. The disc enters the water phase with a constant vertical velocity V = −1. The center of the disc is placed at h_1 = 1.25 above the free-surface level. The radius of the disc is r = 1.

A uniform mesh size $dx = 0.025$ m is used, and the two-dimensional computational domain size is set as 30R × 22R. The gravity acceleration g is −1, water density $\rho_{water} = 1$, water viscosity $\mu_{water} = 0.001$, air density $\rho_{air} = 0.001$, and air viscosity $\mu_{air} = 0.018$. No turbulence model is considered for this case, and the CFL number is set as 0.1. The schematic representation of the simulation is depicted in Fig. 11. The induced vorticity field with the vertical velocity contour is shown in Fig. 12. The disc moves downwards with a constant velocity V = −1 and enters the liquid phase, and symmetric breaking waves are generated by the interaction of the disc and the free-surface. After the waves break, they start to move away towards the side boundaries. The water then begins to return to the point where it was displaced, as can be seen in Fig. 12(f). At $t = 2.5$, the disc is fully submerged in the water phase. At the end, a vertical jet is formed as the water returns. The numerical calculation is carried out on an M1 Max MacBook Pro. The computational time for 6-s simulations using 8 processors is 2.1 h. The results highlight that the level set function, coupled with the PDE-based reinitialization, enables a precise representation of the free surface.

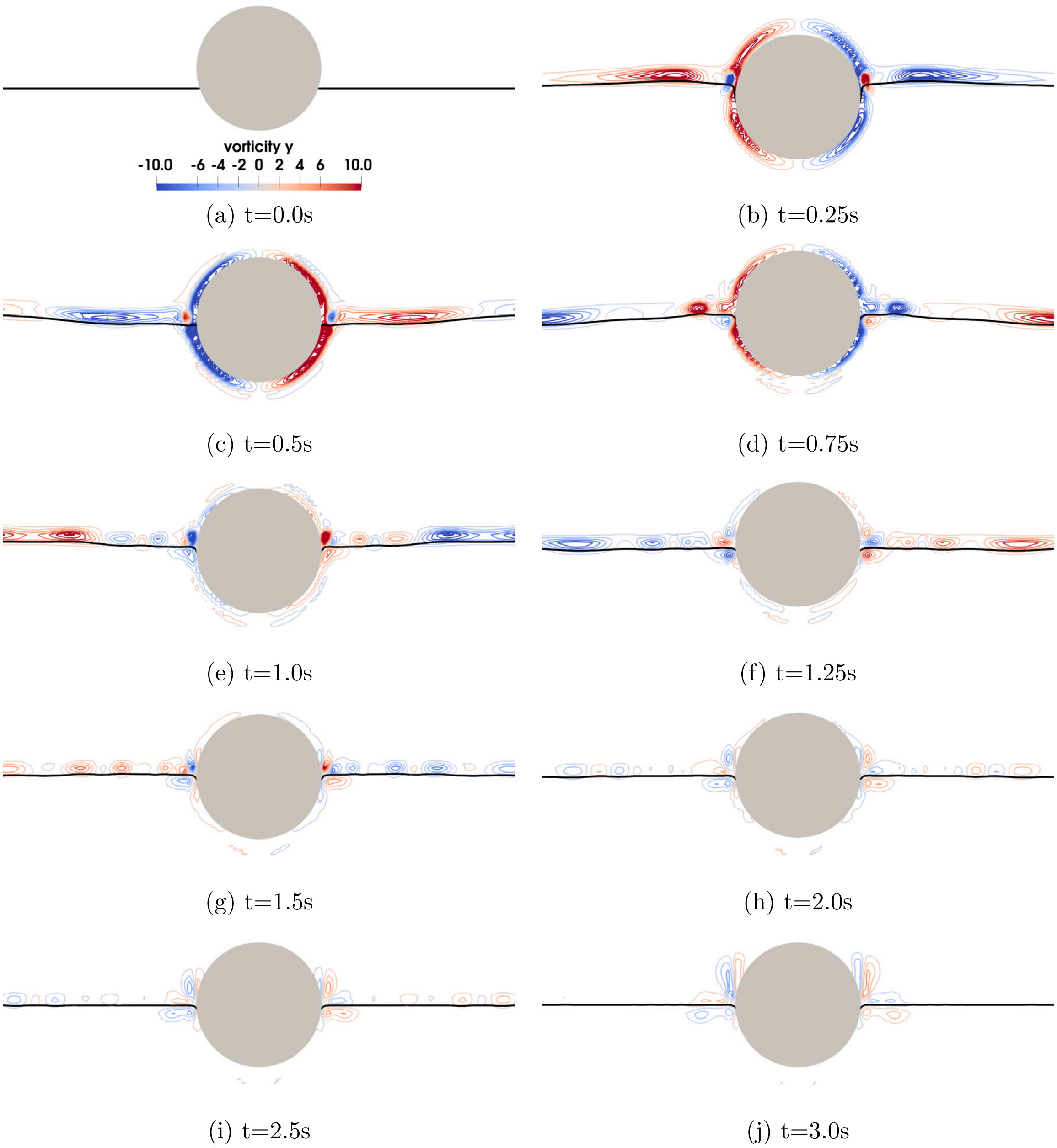
3.3. 2D free floating box

In this section, a 2D benchmark case of a free-floating box without mooring is addressed to validate the new FSI algorithm. The experimental data is taken from Ren et al. (2015) and the numerical setup is set based on the work of Martin et al. (2021a). Fig. 13 illustrates the numerical wave tank (NWT) with a length of 20 m × 0.8 m and the water depth $d = 0.4$ m. The two-dimensional barge (0.3 m × 0.2 m) is placed in the NWT at $x = 7$ m and $z = 0.4$ m. The density of the barge is $\rho_s = 500$ kg/m^3. Fig. 14 shows the solid Heaviside function in the vicinity of the box.

The barge is simulated under regular wave conditions (wave period $T = 1.2$ s, wave height $H = 0.04$ m, and wavelength $\lambda = 1.936$ m) with 2nd-order Stokes theory. The length of the NWT is determined based on the wavelength. The numerical framework, as described in Miquel et al. (2018), incorporates several wave generation and absorption methods, including relaxation, the Dirichlet-type, and the active wave absorption methods. In this study, the relaxation method is used for both the wave generation and absorption to avoid reflection. The dimensions of the wave generation zone and absorption beach are equivalent to one wavelength and two wavelengths, respectively.

Fig. 15 shows the time series for the wave elevation η at $x = 5.5$ m, the heave motion ζ, the pitch motion θ and the surge motion ξ of the free-floating barge for three different cell sizes ($dx = 0.003$ m, 0.0024 m and 0.0020 m) along with the experimental results. For the grid convergence study, the CFL number is kept at 0.1, and a uniform grid is used for the simulations. In Fig. 15(a) and (b), the wave elevation at $x = 5.5$ m in front of the floating barge is presented. The results show good agreement with the experiments. The heave motion in Fig. 15(c) and (d) has minor over-underestimations but overall shows a good trend and monotonic convergence. The pitch motion is illustrated in Fig. 15(e) and (f). Fig. 15(g) and (h) show the surge motion results over time. It is more sensitive to the grid size and gets closer to the experiment as the grid size is refined. The fine grid size dx = 0.002 m is much closer to the experiment. In Fig. 16, the rectangular box is presented at different stages of a wave cycle with the vertical velocity contour.

The assessment of spatial and temporal convergence for the new floating model follows the methodology outlined in Celik et al. (2008). To estimate convergence, the third peak of the motions is considered as the extrapolated value, with relative error computation according to Eq. (43). Temporal convergence is determined with a constant CFL number of 0.1, while spatial convergence is assessed using a fine grid (dx = 0.002 m). Tables 2 and 3 present the results of the spatial convergence study. In the mean period results, all variables exhibit oscillatory



(a) t=0.0s

(b) t=0.25s

(c) t=0.5s

(d) t=0.75s

(e) t=1.0s

(f) t=1.25s

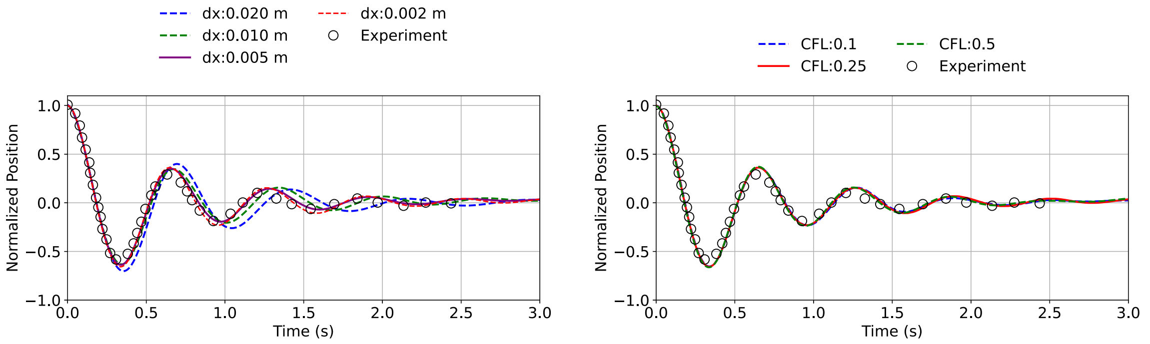
(g) t=1.5s

(h) t=2.0s

(i) t=2.5s

(j) t=3.0s

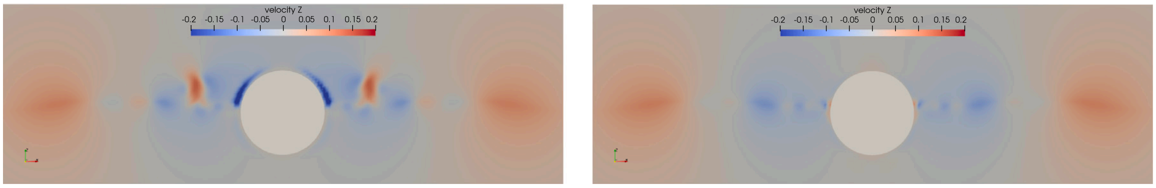
Fig. 7. Iso-surface of vorticity at different times for 2D heaving cylinder. The black line represents the free-surface.



(a) Grid convergence study of the heaving cylinder

(b) Temporal convergence study of the heaving cylinder

Fig. 8. Comparison of heave motion results with the experimental result.



(a) Old density interpolation method (b) New density interpolation method

Fig. 9. Comparison of old and new density interpolation method. Contour shows the vertical velocity at t = 1.0 s.

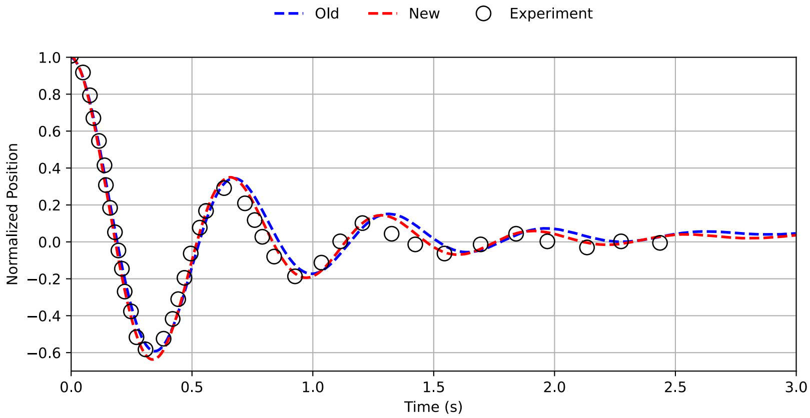


Fig. 10. Comparison of heave motion results with old and new density interpolation method.

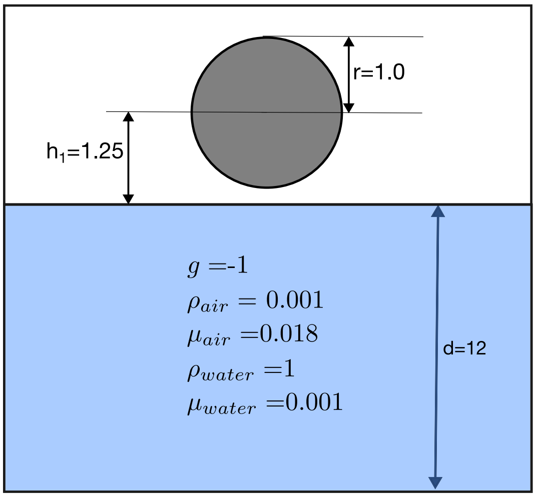


Fig. 11. Schematic description of the disc entry.

convergence, with maximum deviations below 4%. Conversely, the mean amplitude variables demonstrate monotonic convergence behavior. Notably, the wave elevation (η) and pitch motion (θ) tend to over-predict experimental results, whereas the mean amplitude of heave motion (ζ) and surge motion (ξ) tend to under-predict. Temporal convergence findings are presented in Tables 4 and 5. While the mean period of the pitch motion demonstrates oscillatory convergence, both wave elevation and heave motion exhibit monotonic convergence. All variables of mean amplitude show oscillatory convergence, with calculations closely aligning with experimental data.

Numerical calculations are carried out on a Linux cluster with an AMD EPYC 7702 64-Core Processor (16,384 MB L3 Cache, 2.0 GHz, and 64 cores). The 15-s simulation takes approximately 7.5 h using 512 processors. This numerical study shows evidence that the new FSI

Table 2
Spatial convergence of the numerical mean period in comparison to the experimental results for 2D free floating box.

Motion	Coarse	Medium	Fine	Extrapol.	Exp	Error (%)
η	0.9974	0.9956	0.9981	1.0035	1.0414	3.6
ζ	1.0138	1.0040	1.0110	1.0310	1.0330	0.2
θ	1.0495	1.0463	1.1308	1.1321	1.1472	1.3

Table 3
Spatial convergence of the numerical mean amplitude in comparison to the experimental results for 2D free floating box.

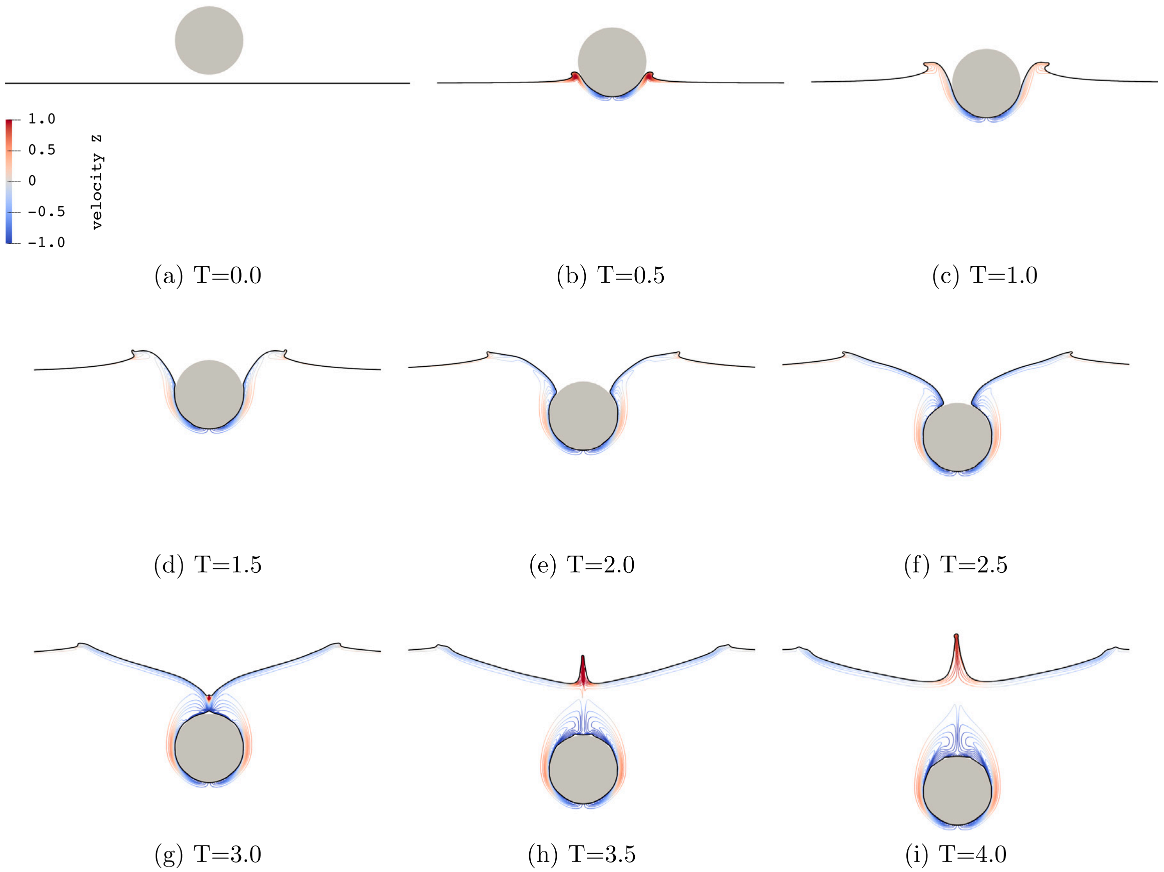
Motion	Coarse	Medium	Fine	Extrapol.	Exp	Error (%)
η	0.0601	0.0598	0.0595	0.0586	0.0545	−7.6
ζ	0.0606	0.0597	0.0595	0.0593	0.0630	6.0
θ	0.0473	0.0464	0.0440	0.0433	0.0394	−9.9
ξ	0.0740	0.0707	0.0668	0.0590	0.0738	20.1

Table 4
Temporal convergence of the numerical mean period in comparison to the experimental results for 2D free floating box.

Motion	CFL 0.5	CFL 0.3	CFL 0.1	Extrapol.	Exp	Error (%)
η	0.9803	0.9898	0.9981	1.0011	1.0414	3.9
ζ	1.0167	1.0112	1.0110	1.0110	1.0330	2.1
θ	1.1173	1.1487	1.1308	1.1179	1.1472	2.6

Table 5
Temporal convergence of the numerical mean amplitude in comparison to the experimental results for 2D free floating box.

Motion	CFL 0.5	CFL 0.3	CFL 0.1	Extrapol.	Exp	Error (%)
η	0.0575	0.0605	0.0595	0.0593	0.0545	−8.9
ζ	0.0591	0.0605	0.0595	0.0578	0.0630	8.3
θ	0.0394	0.0496	0.0440	0.0403	0.0394	−2.5
ξ	0.0628	0.0676	0.0668	0.0668	0.0738	9.5



(a) T=0.0 (b) T=0.5 (c) T=1.0

(d) T=1.5 (e) T=2.0 (f) T=2.5

(g) T=3.0 (h) T=3.5 (i) T=4.0

Fig. 12. Disc entry problem, the contour illustrates the velocity magnitude on the iso-surface of vorticity. Black line shows the free-surface.

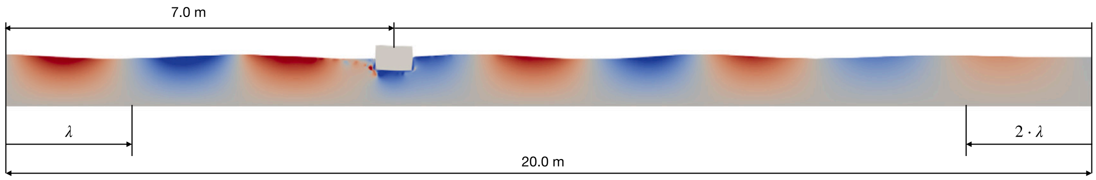


Fig. 13. Numerical domain for the simulation of a two-dimensional barge in a wave tank.

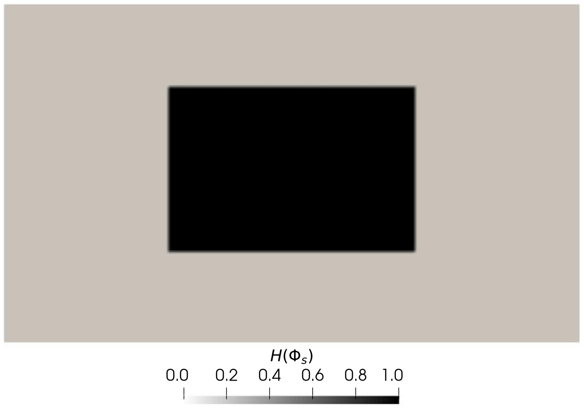


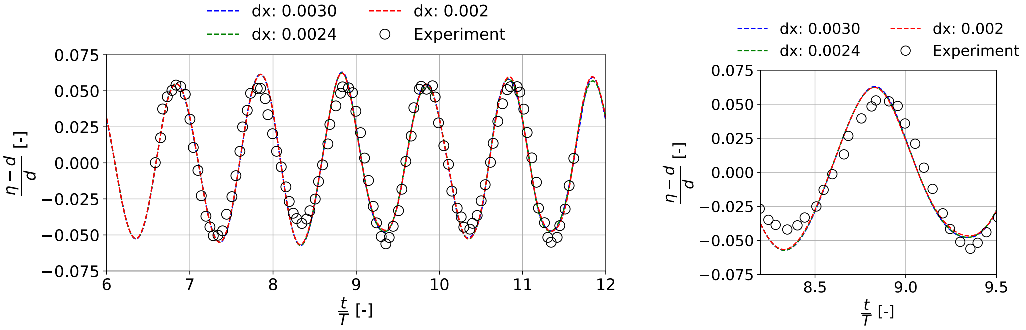
Fig. 14. Zoomed image in the vicinity of the box, showing the solid Heaviside function field.

algorithm is capable of modeling arbitrary motions of a rigid body in waves accurately. The numerical calculations exhibit a satisfactory level of agreement with experimental results, falling within a range of 10%. Furthermore, these results indicate the stability and robustness of the new floating algorithm, even when subjected to higher CFL numbers.

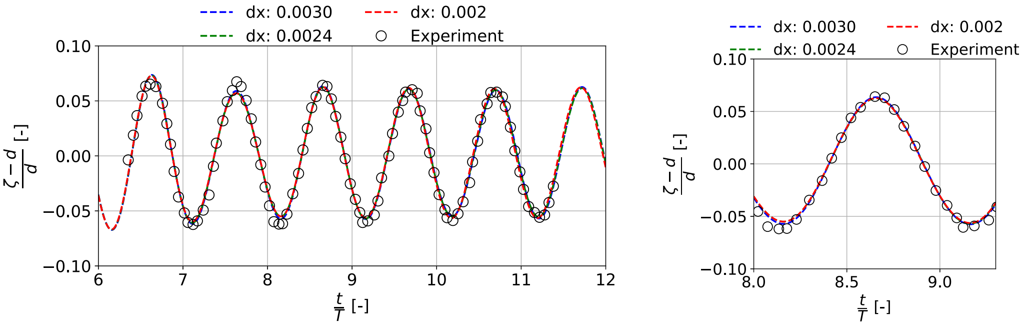
$$Error\ (\%) = \frac{Exp. - Extrapol.}{Exp.} \times 100 \tag{43}$$

3.4. 2D free floating box in steep waves

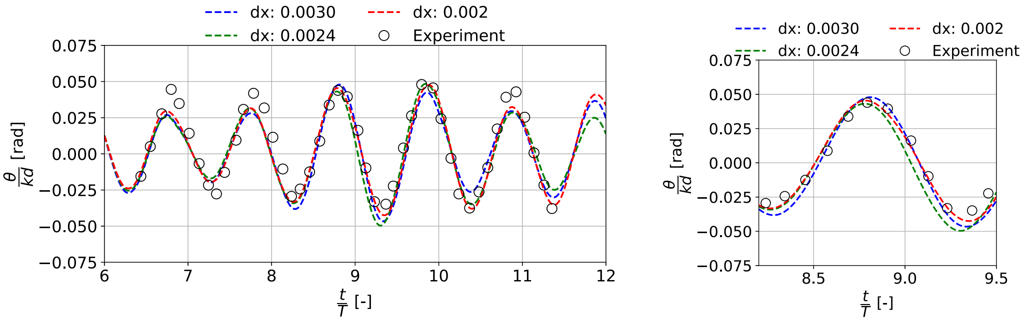
In the previous section, verification and validation studies were presented in a regular wave condition using a 2D benchmark case of a free-floating box without mooring. Here, the new FSI algorithm is tested in steep waves to demonstrate its capability in extreme conditions. 9 different wave cases addressed in this study are given in Table 6. The barge is simulated under regular wave conditions with the 5th-order Stokes theory. The same numerical setup is considered as in the previous section. Response Amplitude Operators (RAO) are calculated from power spectra using a discrete Fast Fourier Transform (FFT) based on Eq. (44). Here, S_{motion} and S_{wave} are the power spectra of the motion responses and the amplitudes of waves. Welch method (Welch, 1967) in Scipy library (Virtanen et al., 2020) is used to calculate the power



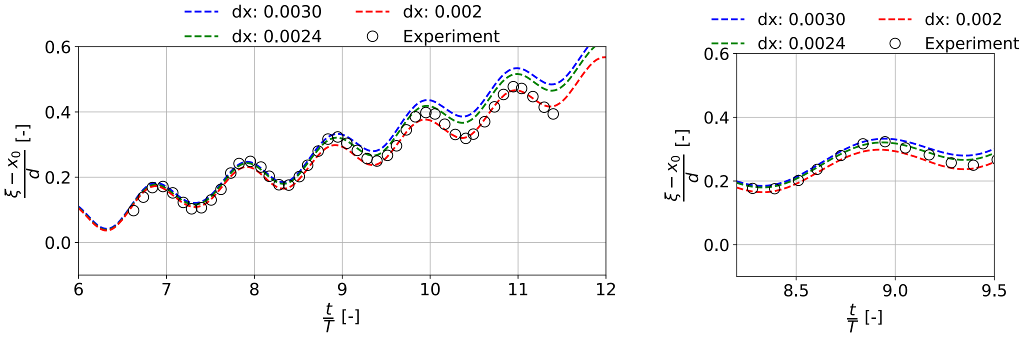
(a) Wave elevation over time at $x = 5.5m$ (b) Wave elevation-3rd peak



(c) Heave motion over time (d) Heave motion-3rd peak



(e) Pitch motion over time (f) Pitch motion-3rd peak



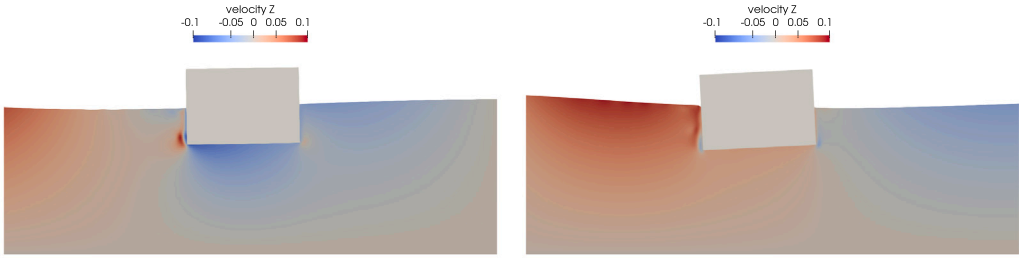
(g) Surge motion over time (h) Surge motion-3rd peak

Fig. 15. 3DOF motion of the two-dimensional barge over time. Comparison of numerical and experimental results for CFL = 0.1.

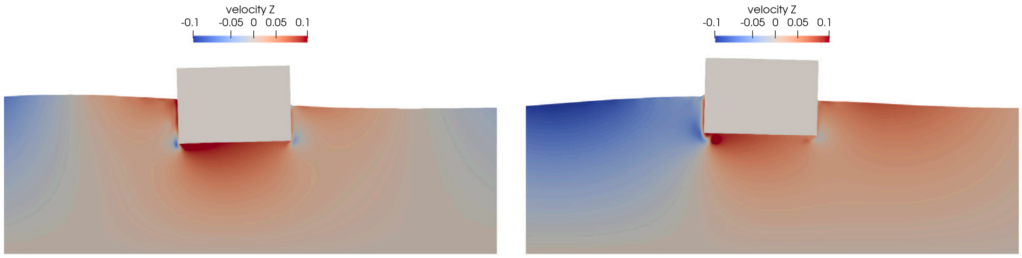
spectra.

$$RAO_{motion} = \sqrt{\frac{S_{motion}}{S_{wave}}} \tag{44}$$

The impact of the wave loads on the structure can be very large under extreme wave events. The force in the wave propagation direction acting on the floating body leads to large motions and strongly non-linear wave-structure interaction occurs. Hence, maintaining numerical stability becomes quite challenging when simulating such problems. To simulate these conditions, the wave height is gradually increased to increase the wave steepness and non-linearity for three different wave periods. The results of the simulations are presented in Fig. 17 as RAO of the motion responses. The amplitudes of the surge and pitch motions

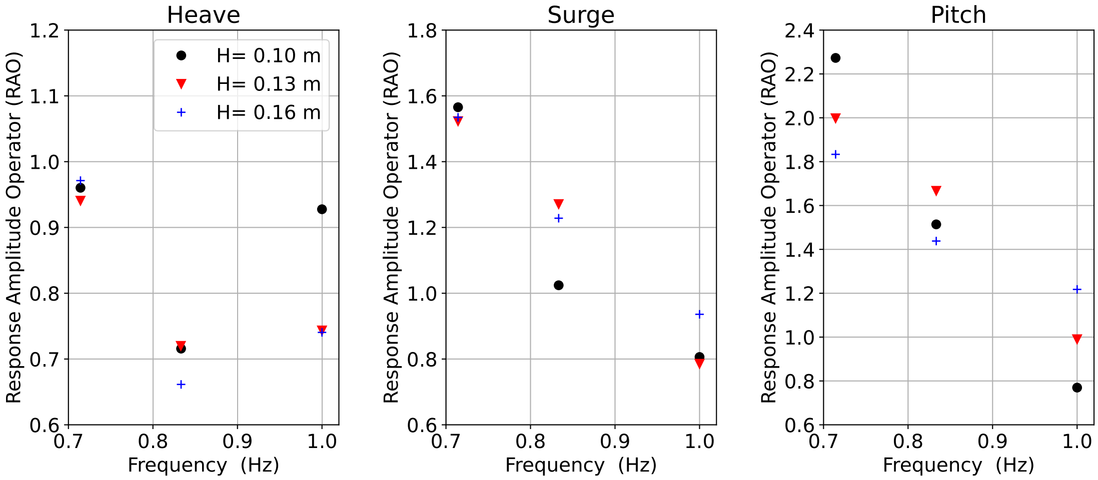


(a) t/T=0.0 (b) t/T=0.25



(c) t/T=0.5 (d) t/T=0.75

Fig. 16. The motion of the floating box in waves, showing the vertical velocity in t/T = 0.25 intervals.



(a) Heave Motion (b) Surge Motion (c) Pitch Motion

Fig. 17. RAO of the motions of the floating box for varying wave heights (H) and wave frequencies.

decrease with increasing wave frequency, while the heave motions are less effected by changing the frequency. The pitch motions are more dominant than the heave and surge motions, especially for the wave conditions with frequencies below 1.0 Hz.

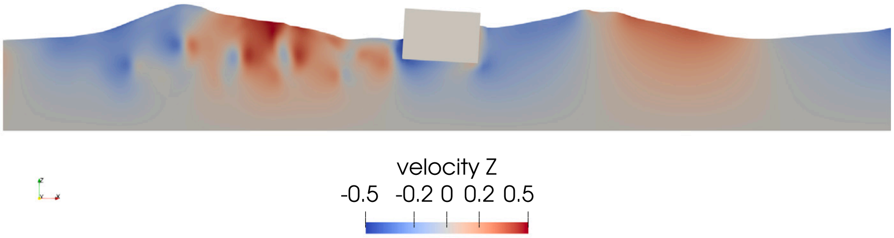
Table 6
The wave conditions for investigating the motion responses of the floating box.

Wave conditions	H [m]	T [s]	Frequency [Hz]	Steepness
Wave1	0.10			0.066
Wave2	0.13	1.0	1.000	0.083
Wave3	0.16			0.099
Wave4	0.10			0.050
Wave5	0.13	1.2	0.833	0.064
Wave6	0.16			0.077
Wave7	0.10			0.041
Wave8	0.13	1.4	0.714	0.052
Wave9	0.16			0.063

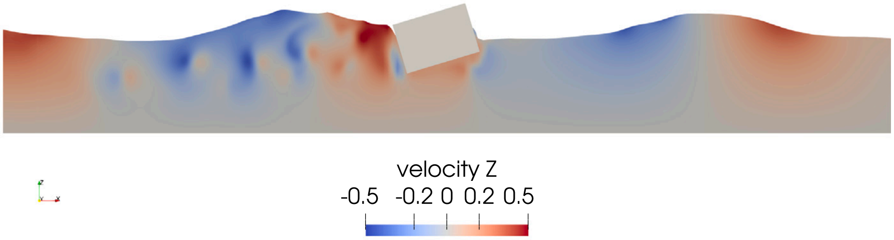
Fig. 18 shows the different stages of a wave cycle in the steepest wave condition, Wave3, with the vertical velocity contour. The interaction between the free surface and the floating body under the steep wave condition results in a complex hydrodynamic event, presenting the robustness of the new algorithm. Numerical calculations are carried out on a Linux cluster with an AMD EPYC 7702 64-Core Processor (16,384 MB L3 Cache, 2.0 GHz, and 64 cores). The 16-s simulation takes approximately 7.7 h using 512 processors.

3.5. 3D buoy with mooring

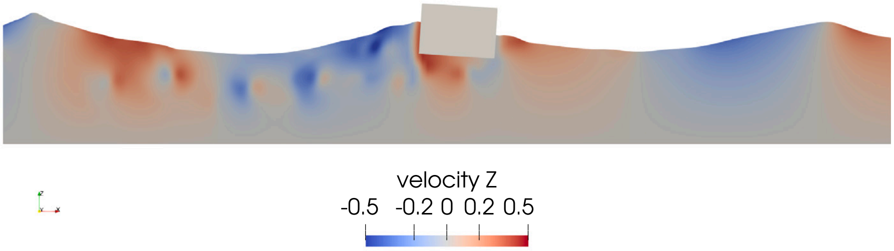
The motions of a moored buoy in a 3D NWT are represented with the new floating algorithm. The experiments were carried out at the Danish Hydraulics Institute in the offshore basin (Fonseca et al., 2011). The basin dimensions are 20 m × 30 m × 3 m. Fig. 19 depicts the numerical wave tank (NWT) with identical length dimensions as the basin with 3 m water depth. The investigated buoy in these experiments features a radius of 0.325 m and a rounded bottom with a radius of 0.1 m, transitioning to a flat bottom. The vertical sides have a length of 0.4 m, and the draught is 0.2 m. The underwater volume of the buoy measures



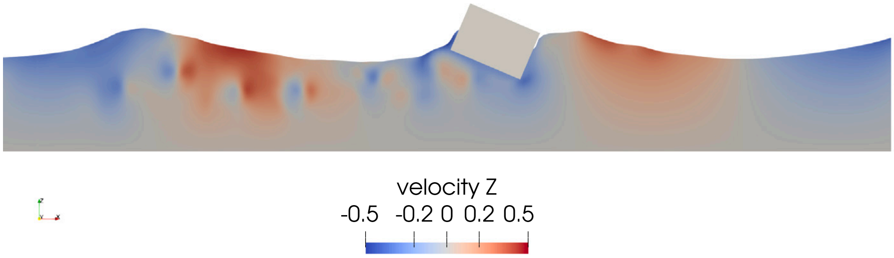
(a) t/T=0.0



(b) t/T=0.3



(c) t/T=0.6



(d) t/T=0.9

Fig. 18. The motion of the floating box in the steepest wave condition, Wave3, showing the vertical velocity in t/T = 0.3 intervals.

0.063 m^3, and its radius of gyrations (R_x, R_y, R_z) is 0.2 m, 0.2 m, and 0.23 m, respectively. Fig. 20 shows the solid Heaviside function in section view in the vicinity of the box. The buoy is represented with an STL file and simulated under regular wave conditions (wave period T = 1.3 s, wave height H = 0.02 m) with 2nd-order Stokes theory. The relaxation method is used for both wave generation and absorption to avoid reflection. A finite element based quasi-static mooring model is used (Martin et al., 2021b). Four symmetrical mooring lines are fixed to the buoy and are out of the water, as depicted in Fig. 21. Each of these chains possesses a weight of 0.910 kg/m and extends to a length of 1.5 m. The coordinates of the mooring lines are given in Table 7.

Table 7
Coordinates of mooring lines.

Location	Coordinates (m) (Start)	Coordinates (m) (End)
Line 1	−7.5, −9.2, −1.322	−0.2, −0.24, −0.057
Line 2	−7.5, 9.2, −1.322	−0.2, 0.24, −0.057
Line 3	7.5, −9.2, −1.322	0.2, −0.24, −0.057
Line 4	−7.5, 9.2, −1.322	0.2, 0.24, −0.057

Fig. 20 shows the solid Heaviside function in the vicinity of the buoy. The grid stretching applied around the buoy and the free surface. The mesh below the free-surface is depicted in Fig. 22. Fig. 23 shows the heave motion, the pitch motion and the surge motion of the 3-D buoy for three different cell sizes (dx = 0.040 m, 0,030 m and 0.025 m) along with the experimental results. In consideration of the prior application involving the free-floating box, it is possible to consider a higher CFL number 0.3 for the grid convergence study. Nevertheless, in order to minimize truncation errors, CFL = 0.1 is used for the grid convergence study. An additional damping is applied in Eq. (30) in the z-direction to calibrate the heave motion. Fig. 23 reveals the time

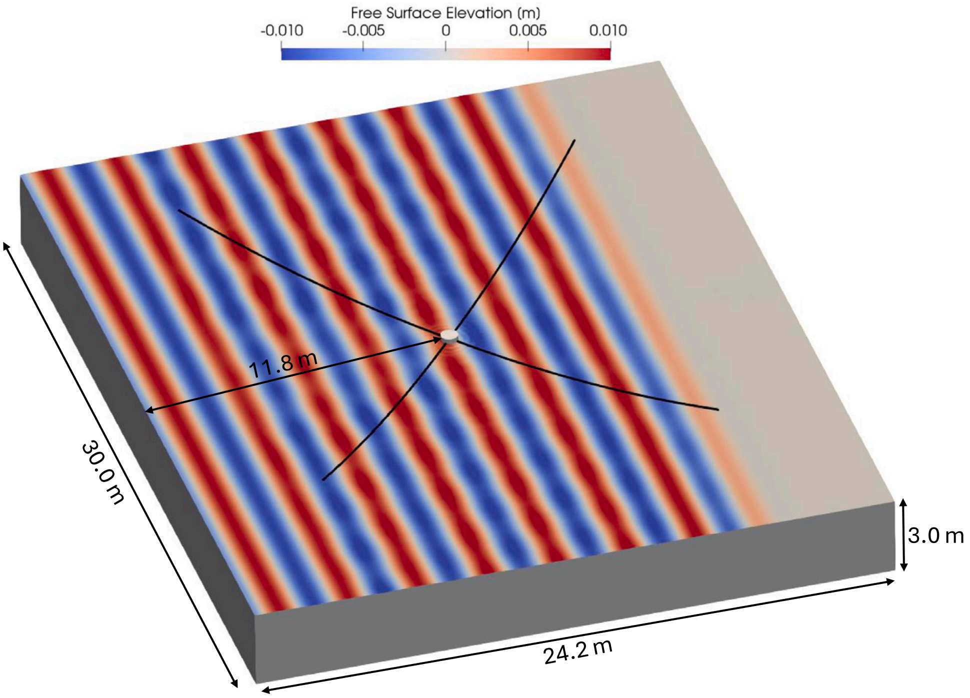


Fig. 19. Numerical domain for the simulation of a three-dimensional buoy with mooring. The contour shows the free surface elevation.

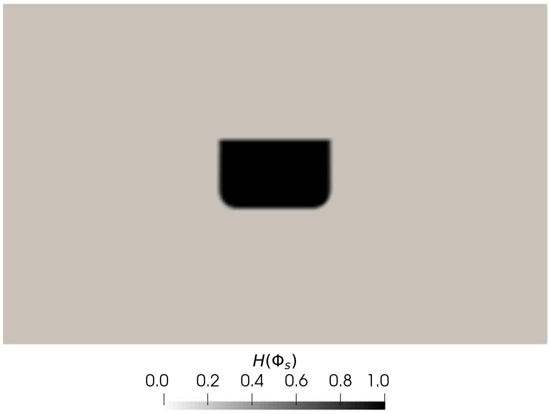


Fig. 20. Zoomed image in the vicinity of the buoy, showing the solid Heaviside function field in section view.

series of the wave excitation motions of the buoy with the experimental results. The coarse grid size $dx = 0.040$ m shows a phase shifting for all motions. Nevertheless, the amplitude of the surge and pitch motions match the experimental data. The mesh sizes $dx = 0.030$ m and $dx = 0.025$ m can capture the phase for all motions very well, while the mesh size $dx = 0.030$ m shows an offset in the surge motion. The fine grid size $dx = 0.025$ m aligns closely with the experimental data in the heave, surge and pitch motions.

The same procedure is followed as in Section 3.3 to assess spatial convergence. To estimate convergence, the fourth peak of the motions is considered as the extrapolated value, with relative error computation according to Eq. (43). Tables 8 and 9 present the outcomes of the spatial convergence study. In the mean period results, all variables exhibit monotonic convergence, with maximum deviations below 5%. The mean amplitude variables also show monotonic convergence behavior except for the heave motion that exhibits oscillatory convergence. All motion variables tend to over-predict the experimental results within the range of 6% error. The new method can capture small motion amplitudes with medium and fine grids.

Table 8
Spatial convergence of the numerical mean period in comparison to the experimental results for 3D buoy.

Motion	Coarse	Medium	Fine	Extrapol.	Exp	Error (%)
Heave	1.3100	1.3080	1.2920	1.2919	1.3000	0.6
Pitch	1.3020	1.3020	1.3100	1.3100	1.2500	−4.8
Surge	1.3200	1.2840	1.2700	1.2390	1.2500	0.9

Table 9
Spatial convergence of the numerical mean amplitude in comparison to the experimental results for 3D buoy.

Motion	Coarse	Medium	Fine	Extrapol.	Exp	Error (%)
Heave	0.0146	0.0133	0.0138	0.0141	0.0138	−2.6
Pitch	0.9360	0.9414	0.9509	0.9548	0.9060	−5.4
Surge	0.0066	0.0067	0.0067	0.0067	0.0066	−1.5

Numerical calculations are carried out on a Linux cluster with an AMD EPYC 7702 64-Core Processor (16,384 MB L3 Cache, 2.0 GHz, and 64 cores). The 200-s simulation takes approximately 19.3 h using 512 processors. The numerical calculations demonstrate a satisfactory level of agreement with experimental results with medium and fine grids around the structure. This is attributed to the new density interpolation method, which reduces the smoothing area in the vicinity of the fluid and solid interface. Overall, the numerical results indicate the flexibility and robustness of the new method.

3.6. DeepCwind floating semisubmersible offshore wind turbine

The new floating algorithm presented in this study is tested using a semisubmersible structure sourced from the OC5-DeepCwind floating semisubmersible offshore wind test campaign, which was carried out at MARIN's offshore wave basin (Robertson et al., 2017). The structure geometry and mooring line configurations are taken from Robertson et al. (2017) and are presented in Fig. 24. The structural properties can be found in Table 10. In this study, wave conditions, structure geometry, and mooring line configurations are carried out at full scale to demonstrate that the proposed method can efficiently simulate arbitrarily large motions of FOWTs while ensuring numerical stability

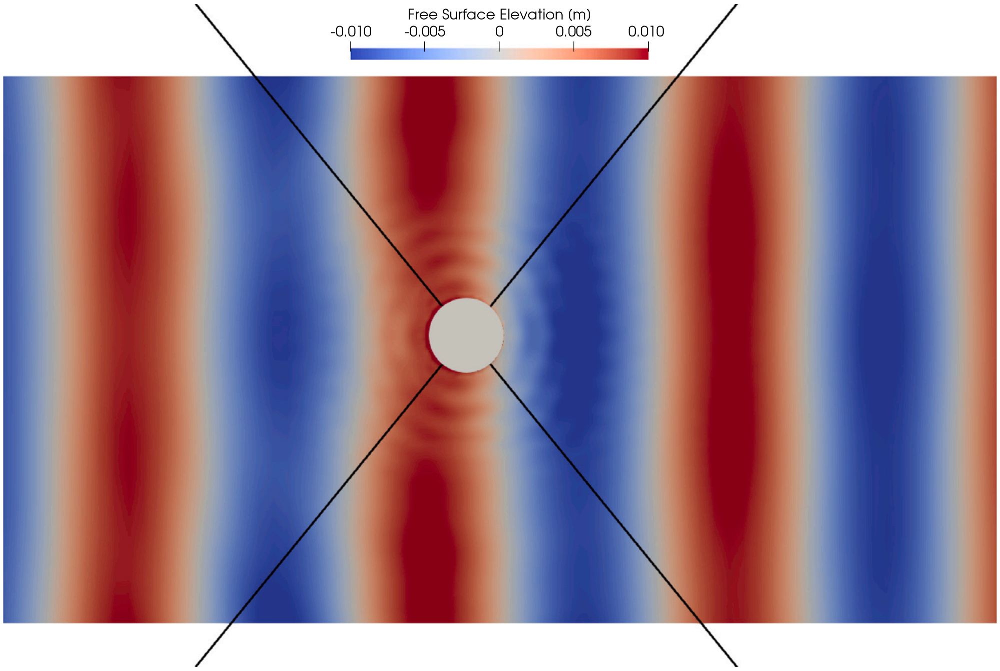


Fig. 21. Free surface elevation around the buoy. The black lines represent the mooring lines.

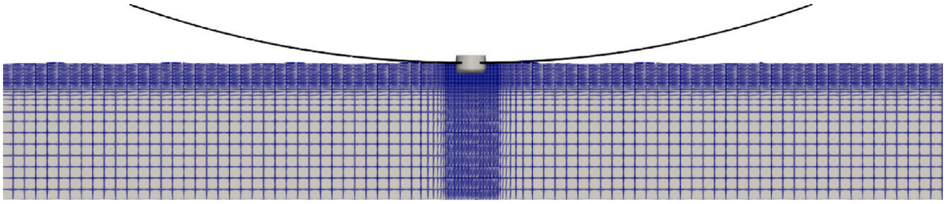


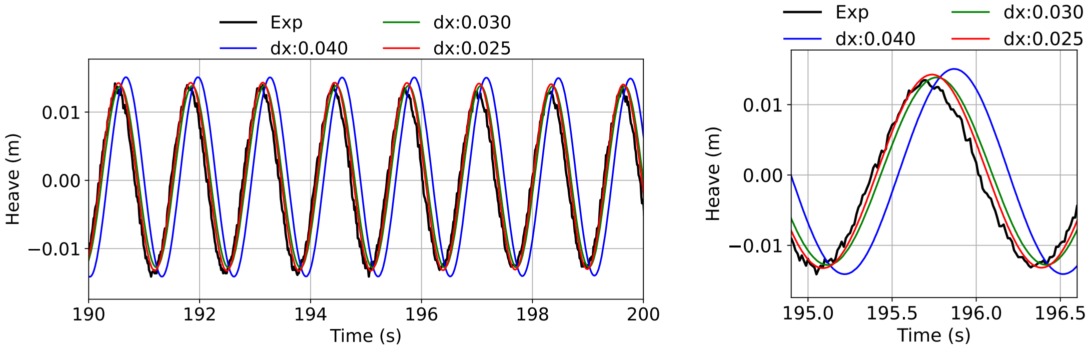
Fig. 22. Grid stretching around the three-dimensional buoy and the free-surface. The black lines represent the mooring lines.

Table 10
OC5 System structural properties.

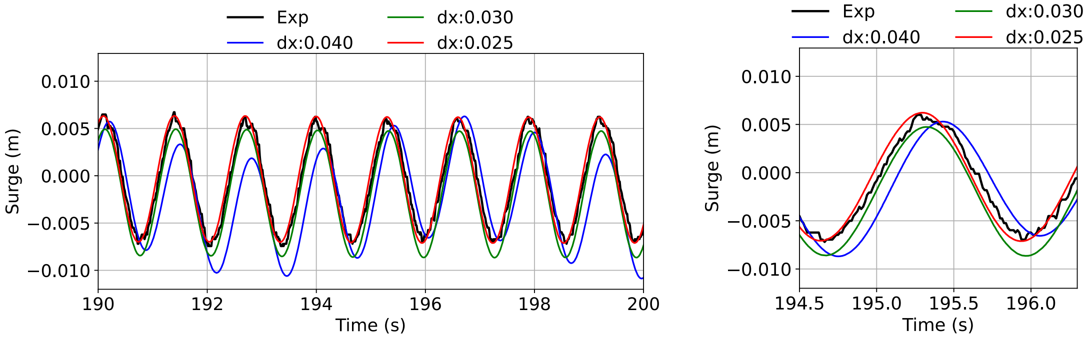
Mass	1.3958E+7 kg
Draft	20 m
Displacement	1.3917E+4 m^3
CM location below SWL	8.07 m
Roll inertia about system CM	1.3947E+10 kg m^2
Pitch inertia about system CM	1.5552E+10 kg m^2
Yaw inertia about system CM	1.3692E+10 kg m^2

and accuracy. The wind turbine is not simulated; therefore, the wind effect is not considered in this study. The platform is positioned at the center of an NWT with dimensions of 1500 m in length, 210 m in width, and 250 m in height and a water depth of 200 m. The semi-submersible structure is simulated under two distinct regular wave conditions Wave-1 (with a period of $T = 12.07$ s and wave height $H = 7.37$ m) and Wave-2 (with a period of $T = 14.3$ s and wave height $H = 9.41$ m), considering 2nd-order Stokes theory. The relaxation method is employed for both wave generation and absorption, while the mooring dynamics are addressed with a quasi-static mooring model (Martin et al., 2021b). The grid stretching implemented around the semisubmersible and the free surface is depicted in Fig. 24(b). Fig. 25 shows the solid Heaviside function in section view in the vicinity of the OC5 structure. Fig. 26 presents the heave, pitch, and surge motions of the semisubmersible for three different cell sizes ($dx = 1.56$ m, 1.2 m, and 1.0 m), alongside the corresponding experimental results. For the grid convergence study, a CFL number of 0.1 is considered. In Fig. 26(a) and (b), the heave motion during Wave-1 exhibits minor overestimations but follows a generally good trend. Surge motion is depicted in 26(c) and (d), showing overestimations in the peaks, yet demonstrating good agreement in the troughs when comparing the numerical results to experimental data. Fig. 26(e) and (f) display pitch motion results over time, showing a good degree of agreement with the experimental results. To estimate the spatial convergence, the fourth peak of the motions is considered as the extrapolated value, with relative error computation according to Eq. (43). Tables 11 and 12 present the outcome of the spatial convergence study. In the mean period results, all variables exhibit oscillatory convergence, with maximum deviations below 7%. All motion variables tend to over-predict the experimental results except the surge motion. The mean amplitude variables exhibit monotonic converging behavior except for the pitch motion which shows an oscillatory converging behavior. The surge motion shows a somewhat large discrepancy, yet the numerical calculations exhibit a satisfactory level of agreement with the experimental results. Building upon the findings from the previous Wave-1 simulations, a grid size of 1.0 m and a CFL number of 0.1 are adopted for the simulation of Wave-2 conditions. The outcomes of this simulation, encompassing the heave motion, pitch motion, and surge motion of the semisubmersible, are presented in Fig. 29. In Fig. 29(a), the heave motion peaks during Wave-2 exhibit minor underestimations but generally follow a good trend. The surge motion, as depicted in 29(b), shows slight underestimation of the peaks while demonstrating good agreement in the troughs when compared to experimental data. Fig. 29(c) displays pitch motion results over time. The peaks of the motion align well with the experimental results; however, the troughs exhibit some discrepancies.

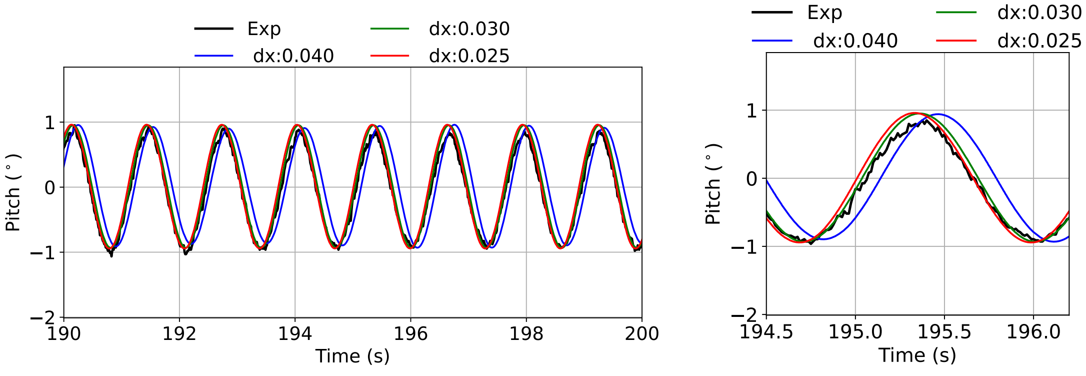
This concluding numerical study highlights all the features and capabilities of the new FSI algorithm. The method accurately represents



(a) Heave Motion (b) Heave Motion-4th peak



(c) Surge Motion (d) Surge Motion-4th peak



(e) Pitch Motion (f) Pitch Motion-4th peak

Fig. 23. Motion responses of the 3D buoy over time. Comparison of numerical and experimental results.

Table 11
Spatial convergence of the numerical mean period in comparison to the experimental results for OC5 structure.

Motion	Coarse	Medim	Fine	Extrapol.	Exp	Error (%)
Heave	12.0620	12.1080	11.9840	11.9530	11.2000	−6.7
Pitch	11.9120	12.2100	12.0540	11.8158	11.6000	−1.9
Surge	12.0540	12.1500	12.0740	11.7059	12.2000	4.0

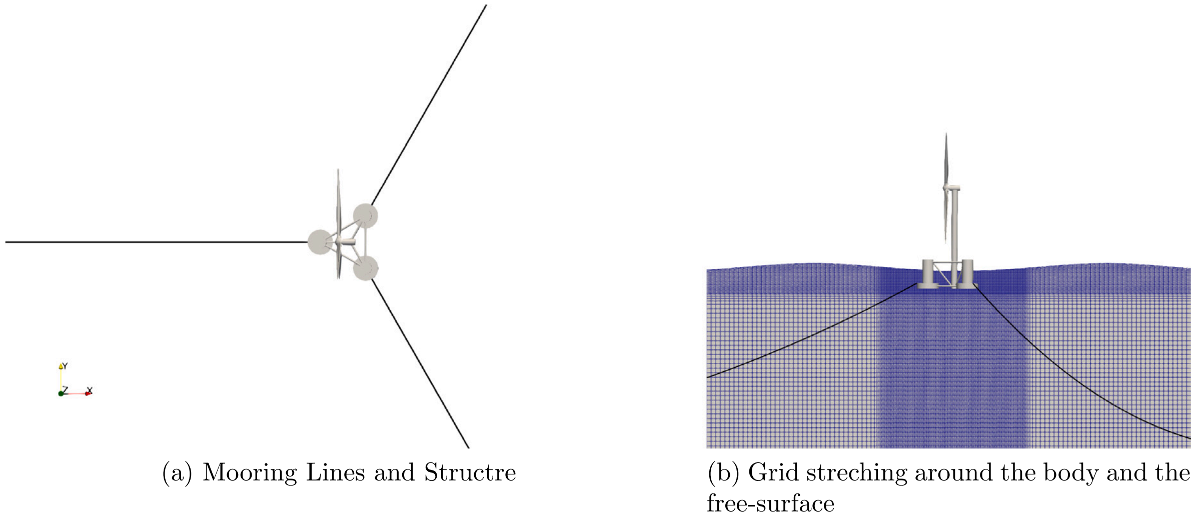
Table 12
Spatial convergence of the numerical mean amplitude in comparison to the experimental results for OC5 structure.

Motion	Coarse	Medim	Fine	Extrapol.	Exp	Error (%)
Heave	1.2565	1.2480	1.1875	1.1857	1.1136	−6.5
Pitch	1.0237	1.0124	1.0166	1.0203	1.0461	2.5
Surge	2.2570	2.2665	2.3545	2.3561	1.9920	−18.3

a full-scale, DeepCwind Floating Semisubmersible Offshore Wind Turbine with an STL file using a level set function accompanied by a ray casting algorithm. The motion of the FOWT is restrained by mooring lines, as illustrated in Fig. 27(a) and (b), demonstrating the rigid-body free-surface interaction in waves with vertical velocity. Fig. 28(a) and (b) show the free surface elevation. The new density interpolation method eliminates unphysical spurious velocity in the vicinity of the interface, facilitating stable and accurate wave load calculations on floating objects. The complex free surface behavior resulting from the interaction is evident in these figures. Numerical computations for the semisubmersible are carried out on a Linux cluster with an AMD EPYC 7702 64-Core Processor (16.384 MB L3 Cache, 2.0 GHz, and 64 cores). Employing a 1.0 m grid size with grid stretching results in 3.4 million cells. The 1000-s simulation takes approximately 11 h using 128 processors.

4. Conclusion

A new direct forcing immersed boundary method for simulating floating bodies in waves is presented. The method is designed on a



(a) Mooring Lines and Structre

(b) Grid streching around the body and the free-surface

Fig. 24. Mooring lines arrangement and numerical grid for OC5 structure.

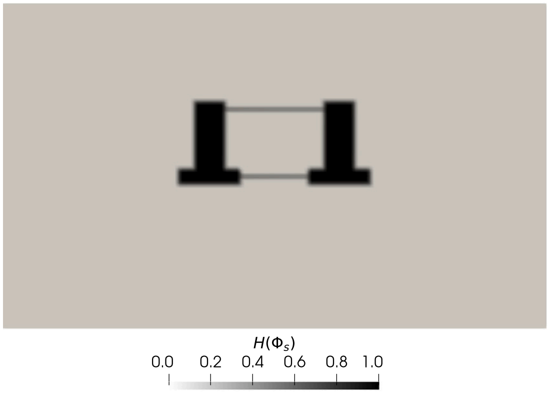


Fig. 25. Zoomed image in the vicinity of the OC5 structure, showing the solid Heaviside function field in section view.

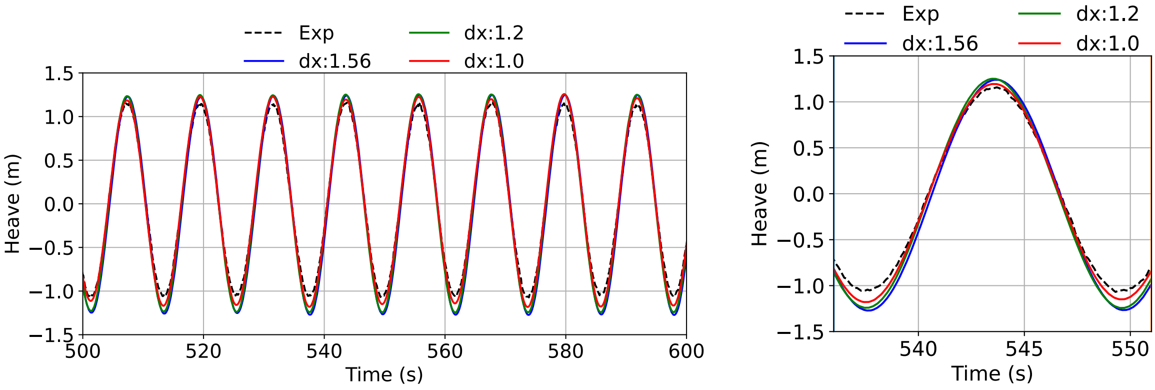
staggered rectilinear grid, ensuring tight pressure–velocity coupling. It enables the use of high-order spatial and temporal discretization methods, such as a 5th-order WENO scheme in space and an third-order Runge–Kutta scheme in time, resulting in good stability, robustness, and accuracy. The method is fully parallelized based on the domain decomposition method and MPI library, capable of scalability on a large number of processors. The complex floating body geometry underlying the fixed Eulerian domain is represented with a level set function. The level set function is created with information obtained by an STL geometry. A ray-casting algorithm determines the inside/outside of the solid body and the closest distance information from neighboring fluid cells to the solid boundary. This feature makes the mesh generation process both flexible and efficient and avoids the drawbacks associated with the moving and overset mesh approaches.

Special attention was given to the new density interpolation method for the fluid–structure interface. The algorithm is improved by a continuous density interpolation treatment. The fluid density and viscosity are assumed to be the same inside and outside the rigid body. This removes the artificial density gradient in the vicinity of the floating body. This way, the smoothing area is strongly reduced, it allows the employment of relatively coarser grid cells in the vicinity of the floating body, along with a thinner transition layer separating the fluid and the solid body. This treatment ensures stable and accurate wave load and response calculations on the floating body. In the method, the rigid-body dynamics is governed by Euler parameters and Hamiltonian mechanics. Euler parameters are free from the gimbal lock effect and computationally more efficient and stable than Euler angles. Hamiltonian mechanics provides a suitable model for numerical implementation when dealing with strongly nonlinear problems. The combination of Euler parameters kinematics and Hamiltonian mechanics allows for the modeling of arbitrary motions of a structure in waves. Furthermore, an explicit third-order low-storage Runge–Kutta scheme (Spalart et al., 1991) was implemented for the calculation equations of motion, enhancing computational efficiency and stability.

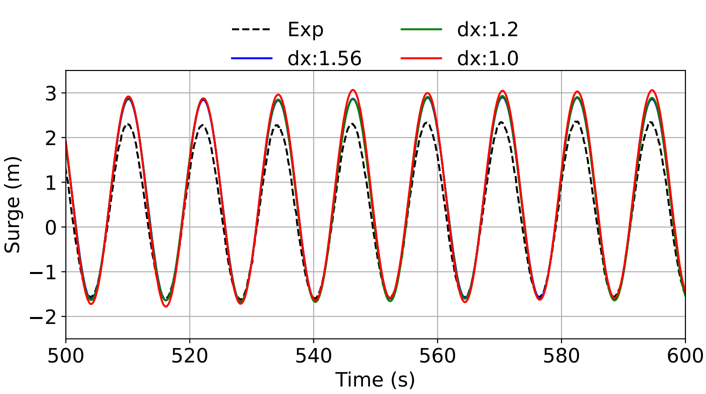
The new method was validated with several benchmark problems. The complexity of the problems was increased gradually. First, the 2D heaving of a circular cylinder case was simulated to investigate the capability of the new FSI method to predict the rigid-body and free-surface interaction. The numerical results showed very good accuracy even with larger CFL numbers. In the second case, a 2D disc entry problem was addressed with a prescribed motion. Numerical findings demonstrate that the level set function accompanied by the PDE-based reinitialization can capture the waves generated by a structure and the free surface interaction precisely.

As explained in the introduction part, the FSI algorithms based on the direct-forcing immersed boundary method in the literature solve either one-phase fluid problems for 6DOF motion or two-phase flow with prescribed motions. As an advancement in this domain, the proposed new FSI algorithm is capable of solving a freely moving rigid body in waves. To demonstrate this capability, a 2D free floating box under regular wave conditions was simulated. Comprehensive verification and validation studies were presented in Section 3.3. The numerical results showed the accuracy of the new FSI algorithm, falling within a range of 10%. Additionally, the method was tested in steep waves by generating 5th-order Stokes waves to reveal its stability and robustness. Under steep wave conditions, large motions and strongly non-linear wave-structure interaction can occur. The presented results show that the new method is able to keep the numerical stability intact under the complex hydrodynamic events.

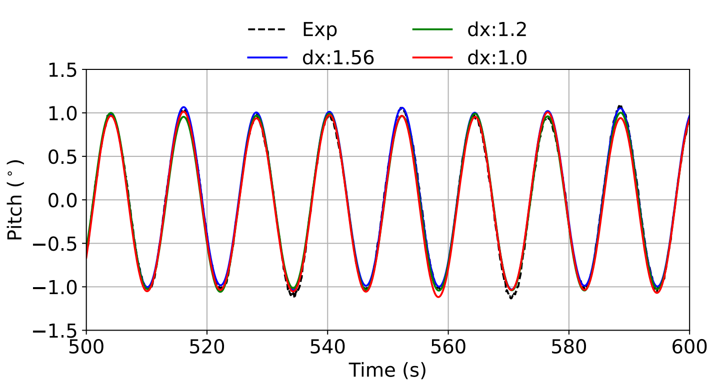
In realistic FSI problems, the mooring dynamics can play a role and the presented method is incorporated with a finite element base quasi-static mooring model (Martin et al., 2021b) to take into account mooring dynamics. In the fourth case, a 3D moored buoy was simulated in an NWT under regular wave conditions, and the results of the grid convergence study were given with the experimental data (Fonseca et al., 2011). The method showed quite a good accuracy for the motion



(a) Heave Motion (b) Heave motion-4th peak

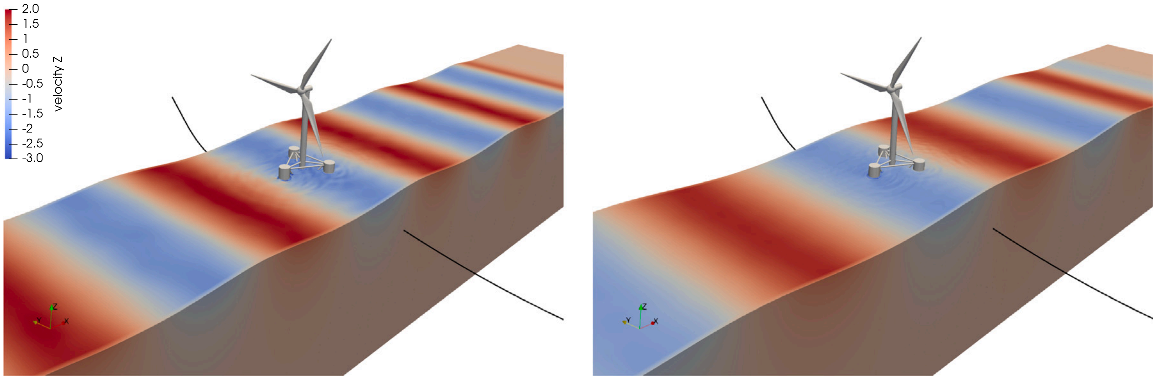


(c) Surge Motion (d) Surge motion-4th peak



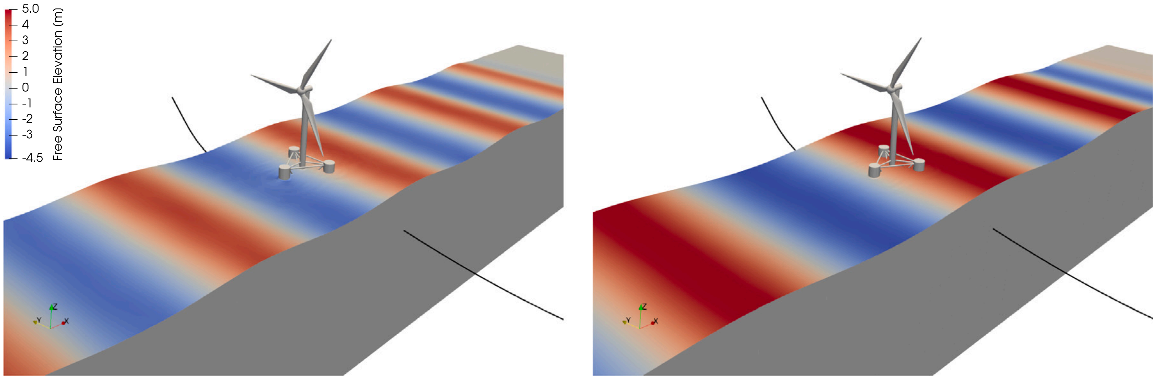
(e) Pitch Motion (f) Pitch motion-4th peak

Fig. 26. Motion responses of the OC5 structure over time in Wave-1. Comparison of numerical and experimental results.



(a) Wave-1 (b) Wave-2

Fig. 27. The OC5 structure in waves, showing the vertical velocity.



(a) Wave-1 (b) Wave-2

Fig. 28. The OC5 structure in waves, showing the free-surface elevation.

responses even with a relatively coarse cell. This can be attributed to the new density interpolation method, which reduces the smoothing area in the vicinity of the fluid and solid interface, along with a thinner transition layer separating the fluid and the solid body.

The last numerical study was the most challenging case of a full-scale floating semisubmersible offshore wind turbine subjected to real engineering problems. This study revealed the new FSI algorithm's capability and features. The geometry of the structure is represented with an STL file, which underlies the fixed Eulerian grid. The ray-casting algorithm determines the closest distance information from neighboring fluid cells to the solid boundary, and a level set function is created with that information. The motions of the semi-submersible structure were restricted by three mooring lines, modeled using the integrated mooring dynamic solver. The motion responses of the structure were simulated under two different regular wave conditions. Good agreement was shown between the numerical results and the experimental data in heave and pitch motions in Wave-1 and heave and surge motions in Wave-2. A discrepancy was observed in surge motion during Wave-1 and pitch motion during Wave-2. Nonetheless, the computational efficiency was promising. The complex wave events, as the results of wave-structure interaction, were captured by the new FSI method with a relatively coarse mesh, as a result of the new density interpolation method.

Coastal engineering problems are often driven by inertia forces, making the method presented in this study well-suited for such scenarios. However, viscous forces play a critical role in other applications, such as ship hydrodynamic simulations. In future work, the method will be enhanced to address these challenges. All these results presented in this study indicate that realistic fluid–structure interaction problems can be efficiently modeled using the new algorithm. In summary, the new rigid-body FSI algorithm makes it possible to efficiently model the free motion of arbitrarily complex structures in waves.

CRediT authorship contribution statement

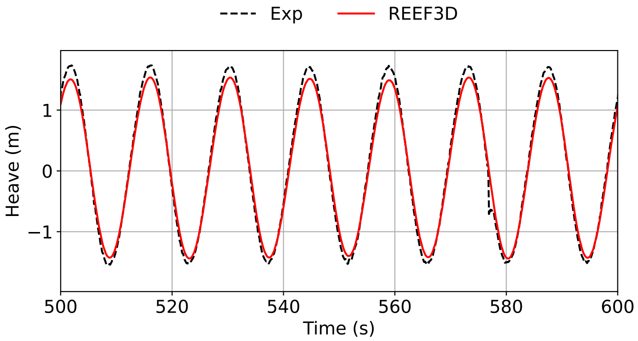
Ahmet Soydan: Writing – original draft, Visualization, Validation, Software, Methodology, Conceptualization. **Widar W. Wang:** Writing – review & editing, Supervision. **Hans Bihs:** Writing – review & editing, Supervision, Software, Methodology, Conceptualization.

Declaration of competing interest

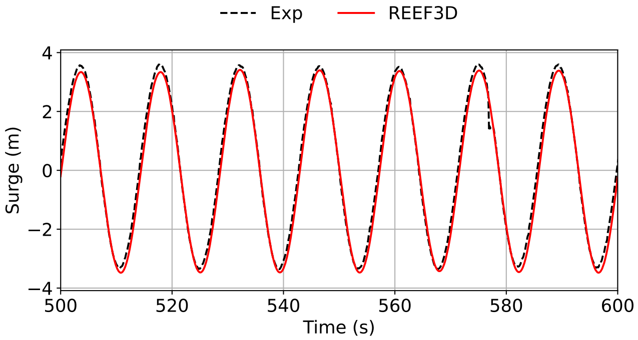
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

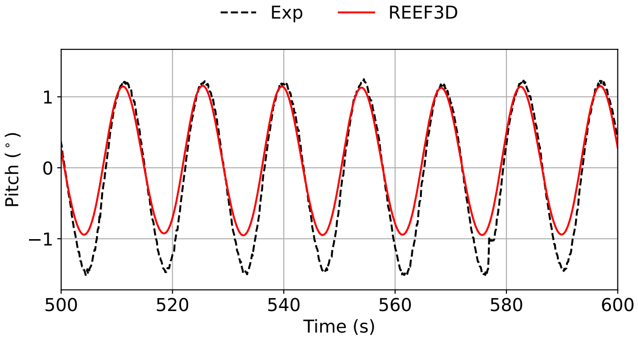
This work was made possible through the knowledge-building project Improving Performance in Real Sea, IPIRiS, financially supported by the Norwegian Research Council, NFR project number 308843, and the consortium partners.



(a) Heave Motion



(b) Surge Motion



(c) Pitch Motion

Fig. 29. Motion responses of the OC5 structure over time in Wave-2. Comparison of numerical and experimental results.

References

Aggarwal, A., Bihs, H., Myrhaug, D., Chella, M.A., 2019a. Characteristics of breaking irregular wave forces on a monopile. Appl. Ocean Res. 90, 101846.

Aggarwal, A., Bihs, H., Shirinov, S., Myrhaug, D., 2019b. Estimation of breaking wave properties and their interaction with a jacket structure. J. Fluids Struct. 91, 102722.

Aliyar, S., Ducrozet, G., Bouscasse, B., Bonnefoy, F., Sriram, V., Ferrant, P., 2022. Numerical coupling strategy using HOS-OpenFOAM-MoorDyn for OC3 hywind SPAR type platform. Ocean Eng. 263, 112206.

Ashby, S., Flagout, R., 1996. A parallel mulitgrid preconditioned conjugate gradient algorithm for groundwater flow simulations. Nucl. Sci. Eng. 124 (1), 145–159.

Askarishahi, M., 2023. Immersed-boundary/soft-sphere method for particle–particle-fluid interaction in a viscous flow: An OpenFOAM solver. Adv. Powder Technol. 34 (11), 104204.

Aye, M.M.M., 2011. Analysis of Euler angles in a simple two-axis gimbals set. World Acad. Sci. Eng. Technol. 81, 389–394.

Barajas, G., Lara, J.L., Losada, I.J., 2024. Hydrodynamic response of a floating offshore wind turbine using OpenFOAM. In: ISOPE International Ocean and Polar Engineering Conference. ISOPE, pp. ISOPE–I.

Berthelsen, P.A., Faltinsen, O.M., 2008. A local directional ghost cell approach for incompressible viscous flow problems with irregular boundaries. J. Comput. Phys. 227 (9), 4354–4397.

Bihs, H., Kamath, A., 2017. A combined level set/ghost cell immersed boundary representation for floating body simulations. Internat. J. Numer. Methods Fluids 83 (12), 905–916.

Bihs, H., Kamath, A., Alagan Chella, M., Pakozdi, C., 2017. Complex geometry handling for a cartesian grid based solver. In: MekIT'17-Ninth National Conference on Computational Mechanics. International Center for Numerical Methods in Engineering (CIMNE).

Bihs, H., Kamath, A., Chella, M.A., Aggarwal, A., Arntsen, Ø.A., 2016. A new level set numerical wave tank with improved density interpolation for complex wave hydrodynamics. Comput. & Fluids 140, 191–208.

Borazjani, I., Ge, L., Le, T., Sotiropoulos, F., 2013. A parallel overset-curvilinear-immersed boundary framework for simulating complex 3D incompressible flows. Comput. & Fluids 77, 76–96.

Calderer, A., Kang, S., Sotiropoulos, F., 2014. Level set immersed boundary method for coupled simulation of air/water interaction with complex floating structures. J. Comput. Phys. 277, 201–227.

Campaña-Alonso, G., Martín-San-Román, R., Méndez-López, B., Benito-Cia, P., Azcona-Armendáriz, J., 2023. OF 2: coupling OpenFAST and OpenFOAM for high-fidelity aero-hydro-servo-elastic FOWT simulations. Wind. Energy Sci. 8 (10), 1597–1611.

Carrica, P.M., Wilson, R.V., Noack, R.W., Stern, F., 2007. Ship motions using single-phase level set with dynamic overset grids. Comput. & Fluids 36 (9), 1415–1433.

Celik, I.B., Ghia, U., Roache, P.J., Freitas, C.J., Coleman, H., Raad, P.E., 2008. Procedure for estimation and reporting of uncertainty due to discretization in CFD applications. J. Fluids Eng. (ISSN: 0098-2202) 130 (7), 078001. http://dx.doi.org/10.1115/1.2960953.

Chen, H., Medina, T.A., Cercos-Pita, J.L., 2024. CFD simulation of multiple moored floating structures using openfoam: An open-access mooring restraints library. Ocean Eng. 303, 117697.

Chen, H., Qian, L., Ma, Z., Bai, W., Li, Y., Causon, D., Mingham, C., 2019. Application of an overset mesh based numerical wave tank for modelling realistic free-surface hydrodynamic problems. Ocean Eng. 176, 97–117.

Constant, E., Favier, J., Meldi, M., Meliga, P., Serre, E., 2017. An immersed boundary method in openfoam: verification and validation. Comput. & Fluids 157, 55–72.

Delen, C., Can, U., Bal, S., 2021. Prediction of resistance and self-propulsion characteristics of a full-scale naval ship by CFD-based GEOSIM method. J. Ship Res. 65 (04), 346–361.

Eskilsson, C., Fernandez, G.V., Andersen, J., Palm, J., 2023. Hydrodynamic simulations of a FOWT platform (1st FOWT comparative study) using openfoam coupled to moodycore. In: ISOPE International Ocean and Polar Engineering Conference. ISOPE, pp. ISOPE–I.

Fadlun, E., Verzicco, R., Orlandi, P., Mohd-Yusof, J., 2000. Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations. J. Comput. Phys. 161 (1), 35–60.

Flores, P., 2015. Concepts and Formulations for Spatial Multibody Dynamics. Springer.

Fonseca, N., Pessoa, J., Mavrakos, S., Le Boulluec, M., 2011. Experimental and numerical investigation of the slowly varying wave exciting drift forces on a restrained body in bi-chromatic waves. Ocean Eng. 38 (17–18), 2000–2014.

Goldstein, H., Poole, C., Safko, J., 2001. CPP, Safko JL: Classical Mechanics, vol. 6, Addison-Wesley, p. 5.

Hanaoka, A., 2013. An Overset Grid Method Coupling an Orthogonal Curvilinear Grid Solver and a Cartesian Grid Solver. The University of Iowa.

Holcombe, A., Edwards, E., Tosdevin, T., Greaves, D., Hann, M., 2023. A comparative study of potential-flow-based numerical models to experimental tests of a semi-submersible floating wind turbine platform. In: ISOPE International Ocean and Polar Engineering Conference. vol. All Days, ISOPE–I–23–067.

Itō, S., 1977. Study of the Transient Heave Oscillation of a Floating Cylinder (Ph.D. thesis). Massachusetts Institute of Technology.

Jiang, G., Peng, D., 2000. Weighted ENO schemes for hamilton Jacobi equations. SIAM J. Sci. Comput. 21, 2126–2143.

Jiang, G.-S., Shu, C.-W., 1996. Efficient implementation of weighted ENO schemes. J. Comput. Phys. 126 (1), 202–228.

Kamath, A., Alagan Chella, M., Bihs, H., Arntsen, Ø.A., 2017a. Energy transfer due to shoaling and decomposition of breaking and non-breaking waves over a submerged bar. Eng. Appl. Comput. Fluid Mech. 11 (1), 450–466.

Kamath, A., Bihs, H., Arntsen, Ø.A., 2017b. Study of water impact and entry of a free falling wedge using computational fluid dynamics simulations. J. Offshore Mech. Arct. Eng. 139 (3).

Katsidoniotaki, E., Göteman, M., 2022. Numerical modeling of extreme wave interaction with point-absorber using OpenFOAM. Ocean Eng. 245, 110268.

Kempe, T., Fröhlich, J., 2012. An improved immersed boundary method with direct forcing for the simulation of particle laden flows. J. Comput. Phys. 231 (9), 3663–3684.

Kempe, T., Lennartz, M., Schwarz, S., Fröhlich, J., 2015. Imposing the free-slip condition with a continuous forcing immersed boundary method. J. Comput. Phys. 282, 183–209.

Khan, M.A., Barajas, G., Gaeta, M.G., Lara, J.L., Archetti, R., 2024. Hydrodynamic analysis and optimization of a floating wave energy converter with moonpool using OpenFOAM®. Appl. Ocean Res. 142, 103847.

Martin, T., Kamath, A., Bihs, H., 2021b. Accurate modeling of the interaction of constrained floating structures and complex free surfaces using a new quasistatic mooring model. Internat. J. Numer. Methods Fluids 93 (2), 504–526.

Martin, T., Tsarau, A., Bihs, H., 2021a. Numerical framework for modelling the dynamics of open ocean aquaculture structures in viscous fluids. Appl. Ocean Res. 106, 102410.

Miquel, A.M., Kamath, A., Alagan Chella, M., Archetti, R., Bihs, H., 2018. Analysis of different methods for wave generation and absorption in a CFD-based numerical wave tank. J. Mar. Sci. Eng. 6 (2), 73.

Nikravesh, P.E., Chung, I.S., 1982. Application of Euler parameters to the dynamic analysis of three-dimensional constrained mechanical systems. J. Mech. Des. 104 (4), 785–791. http://dx.doi.org/10.1115/1.3256437.

Osher, S., Sethian, J.A., 1988. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton–Jacobi formulations. J. Comput. Phys. 79, 12–49.

Peng, D., Merriman, B., Osher, S., Zhao, H., Kang, M., 1999. A PDE-based fast local level set method. J. Comput. Phys. 155, 410–438.

Peskin, C.S., 1972. Flow patterns around heart valves: a numerical method. J. Comput. Phys. 10 (2), 252–271.

Peskin, C.S., 1977. Numerical analysis of blood flow in the heart. J. Comput. Phys. 25 (3), 220–252.

Pinguet, R., Kanner, S., Benoit, M., Molin, B., 2020. Validation of open-source overset mesh method using free-decay tests of floating offshore wind turbine. In: ISOPE International Ocean and Polar Engineering Conference. ISOPE, ISOPE–I.

Pinguet, R., Kanner, S., Benoit, M., Molin, B., 2021. Modeling the dynamics of freely-floating offshore wind turbine subjected to waves with an open-source overset mesh method. In: International Conference on Offshore Mechanics and Arctic Engineering. vol. 84768, American Society of Mechanical Engineers, V001T01A008.

Ransley, E.J., Brown, S.A., Greaves, D.M., Hann, M.R., 2023. Modelling the hydrodynamic response of a floating offshore wind turbine using openfoam and a quasi-static mooring model. In: ISOPE International Ocean and Polar Engineering Conference. vol. All Days, ISOPE–I–23–071.

Ren, B., He, M., Dong, P., Wen, H., 2015. Nonlinear simulations of wave-induced motions of a freely floating body using WCSPH method. Appl. Ocean Res. 50, 1–12.

Robertson, A.N., Wendt, F., Jonkman, J.M., Popko, W., Dagher, H., Gueydon, S., Qvist, J., Vittori, F., Azcona, J., Uzunoglu, E., et al., 2017. OC5 project phase II: validation of global loads of the DeepCwind floating semisubmersible wind turbine. Energy Procedia 137, 38–57.

Sakamoto, N., Ohashi, K., Araki, M., Kume, K.-i., Kobayashi, H., 2019. Identification of KVLCC2 manoeuvring parameters for a modular-type mathematical model by RaNS method with an overset approach. Ocean Eng. 188, 106257.

Sasikumar, A., Kamath, A., Bihs, H., 2020. Modeling porous coastal structures using a level set method based VRANS-solver on staggered grids. Coast. Eng. J. 62 (2), 198–216.

Shen, Z., Wan, D., Carrica, P.M., 2015. Dynamic overset grids in openfoam with application to KCS self-propulsion and maneuvering. Ocean Eng. 108, 287–306.

Shivarama, R., Fahrenthold, E.P., 2004. Hamilton's equations with Euler parameters for rigid body dynamics modeling. J. Dyn. Syst. Meas. Control. 126 (1), 124–130.

Shu, C., Osher, S., 1988. Efficient implementation of essentially non-oscillatory shock-capturing schemes. J. Comput. Phys. 77 (2), 439–471.

Spalart, P.R., Moser, R.D., Rogers, M.M., 1991. Spectral methods for the Navier–Stokes equations with one infinite and two periodic directions. J. Comput. Phys. 96 (2), 297–324.

Sussman, M., Smereka, P., Osher, S., 1994. A level set approach for computing solutions to incompressible two-phase flow. J. Comput. Phys. 114, 146–159.

Tagliafierro, B., Capasso, S., Martínez-Estévez, I., Göteman, M., Bernhoff, H., Karimirad, M., Domínguez, J.M., Altomare, C., Viccione, G., Crespo, A.J.C., Gómez-Gesteira, M., 2023a. Hydrodynamic validation of a semi-submersible floating platform supporting a 15 MW wind turbine tower under extreme loading scenarios with DualSPHysics and MoorDyn+. In: ISOPE International Ocean and Polar Engineering Conference. vol. All Days, ISOPE–I–23–037.

Tagliafierro, B., Karimirad, M., Altomare, C., Göteman, M., Martínez-Estévez, I., Capasso, S., Domínguez, J.M., Viccione, G., Gómez-Gesteira, M., Crespo, A.J., 2023b. Numerical validations and investigation of a semi-submersible floating offshore wind turbine platform interacting with ocean waves using an SPH framework. Appl. Ocean Res. (ISSN: 0141-1187) 141, 103757. http://dx.doi.org/10.1016/j.apor.2023.103757.

Timmermans, L.J., Minev, P.D., Van De Vosse, F.N., 1996. An approximate projection scheme for incompressible flow using spectral elements. Internat. J. Numer. Methods Fluids 22 (7), 673–688.

Tran, T.T., Kim, D.-H., 2016. Fully coupled aero-hydrodynamic analysis of a semi-submersible FOWT using a dynamic fluid body interaction approach. Renew. Energy 92, 244–261.

Tran, T.T., Kim, D.-H., 2018. A CFD study of coupled aerodynamic-hydrodynamic loads on a semisubmersible floating offshore wind turbine. Wind. Energy 21 (1), 70–85.

Uhlmann, M., 2005. An immersed boundary method with direct forcing for the simulation of particulate flows. J. Comput. Phys. 209 (2), 448–476.

van der Vorst, H., 1992. BiCGStab: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. SIAM J. Sci. Comput. 13, 631–644.

Van Rij, J., Yu, Y.-H., Tran, T.T., 2021. Validation of simulated wave energy converter responses to focused waves. Proc. Inst. Civ. Eng.-Eng. Comput. Mech. 174 (1), 32–45.

Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S.J., Brett, M., Wilson, J., Millman, K.J., Mayorov, N., Nelson, A.R.J., Jones, E., Kern, R., Larson, E., Carey, C.J., Polat, İ., Feng, Y., Moore, E.W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E.A., Harris, C.R., Archibald, A.M., Ribeiro, A.H., Pedregosa, F., van Mulbregt, P., SciPy 1.0 Contributors, 2020. SciPy 1.0: Fundamental algorithms for scientific computing in python. Nature Methods 17, 261–272. http://dx.doi.org/10.1038/s41592-019-0686-2.

Vreman, A., 2020. Immersed boundary and overset grid methods assessed for stokes flow due to an oscillating sphere. J. Comput. Phys. 423, 109783.

Welch, P., 1967. The use of fast Fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms. IEEE Trans. Audio Electroacoust. 15 (2), 70–73.

Yang, L., 2018. One-fluid formulation for fluid–structure interaction with free surface. Comput. Methods Appl. Mech. Engrg. (ISSN: 0045-7825) 332, 102–135. http://dx.doi.org/10.1016/j.cma.2017.12.016.

Yang, J., Balaras, E., 2006. An embedded-boundary formulation for large-eddy simulation of turbulent flows interacting with moving boundaries. J. Comput. Phys. 215 (1), 12–40.

Yang, J., Stern, F., 2009. Sharp interface immersed-boundary/level-set method for wave–body interactions. J. Comput. Phys. 228 (17), 6590–6616.

Yang, J., Stern, F., 2012. A simple and efficient direct forcing immersed boundary framework for fluid–structure interactions. J. Comput. Phys. 231 (15), 5029–5061.

Yang, J., Stern, F., 2015. A non-iterative direct forcing immersed boundary method for strongly-coupled fluid–solid interactions. J. Comput. Phys. 295, 779–804.

Zhang, C., 2020. Sdfibm: a signed distance field based discrete forcing immersed boundary method in openfoam. Comput. Phys. Comm. 255, 107370.

Zhang, Y., Kim, B., 2018. A fully coupled computational fluid dynamics method for analysis of semi-submersible floating offshore wind turbines under wind-wave excitation conditions based on OC5 data. Appl. Sci. 8 (11), 2314.