A 3D parallel Particle-In-Cell solver for extreme wave interaction with floating bodies

Qiang Chen[a,b], Jun Zang[a,b,*], Dezhi Ning[b], Chris Blenkinsopp[a], Junliang Gao[c]

[a]*Research Unit for Water, Environment and Infrastructure Resilience (WEIR), Department of Architecture and Civil Engineering, University of Bath, BA2 7AY, U.K.*
[b]*State Key Laboratory of Coastal and Offshore Engineering, Dalian University of Technology, Dalian, 116024, China*
[c]*School of Naval Architecture and Ocean Engineering, Jiangsu University of Science and Technology, Zhenjiang, 212003, China*

Abstract

Floating structures are widely used for vessels, offshore platforms, and recently considered for deep water floating offshore wind system and wave energy devices. However, modelling complex wave interactions with floating structures, particularly under extreme conditions, remains an important challenge. Following the three-dimensional (3D) parallel particle-in-cell (PIC) model developed for simulating wave interaction with fixed bodies, this paper further extends the methodology and develops a new 3D parallel PIC model for applications to floating bodies. The PIC model uses both Lagrangian particles and Eulerian grid to solve the incompressible Navier-Stokes equations, attempting to combine both the Lagrangian flexibility for handling large free-surface deformations and Eulerian efficiency in terms of CPU cost. The wave-structure interaction is resolved via inclusion of a Cartesian cut cell method based two-way strong fluid-solid coupling algorithm that is both stable and efficient. The numerical model is validated against 3D experiments of focused wave interaction with a floating moored buoy. Good agreement between the numerical and experimental results has been achieved for the motion of the buoy and the mooring force. Additionally, the PIC model achieves a CPU efficiency of the same magnitude as that of the state-of-the-art OpenFOAM® model for an extreme wave-structure interaction scenario.

Keywords: Wave-structure interaction, Extreme wave, Floating bodies, Particle-In-Cell method, OpenFOAM® model

*Corresponding author.
Email address: `j.zang@bath.ac.uk` (Jun Zang)

1. Introduction

In the past few decades, computational fluid dynamics (CFD) methods have become more and more popular within the ocean engineering field. Typical examples are the grid-based Eulerian model such as OpenFOAM® and the particle-based Lagrangian model such as the smoothed particle hydrodynamics (SPH) method based SPHysics. While the former models are relatively efficient due to the use of a fixed grid, the latter solvers are more suitable for handling large free-surface deformations via using particles. In an attempt to combine the advantages of Eulerian and Lagrangian methods, the Particle-In-Cell (PIC) method was invented through a combined use of particles and grid (Harlow, 1955, 1964). Typically, the particles are used to solve the transport terms and track the free-surface position, while the gird is employed for solving the non-advection terms. Thus, it is possible to achieve both Lagrangian flexibilities and Eulerian efficiency in the PIC framework. However, sophisticated schemes must be developed for the interaction between the fixed grid and the scattered particles in order to drive the computation and maintain numerical accuracy and stability. The early versions of the PIC method are successful, see e.g. Harlow (1964), but have a few drawbacks such as high numerical dissipation, low accuracy and demanding memory storage requirement. Later, many attempts have been made to improve this method (see Brackbill and Ruppel (1986); Brackbill et al. (1988); Nishiguchi and Yabe (1982, 1983)). More recently, high-order PIC variations have become possible (see Edwards and Bridson (2012), Maljaars et al. (2018) and Wang and Kelly (2018)). However, so far this hybrid method has not been very well exploited for use in the ocean engineering field, where modelling complex wave-structure interaction with computational efficiency still remains an important challenge.

Early attempts of developing a PIC method based numerical model for modelling wave-structure interaction processes in the coastal and offshore environment can be found in Kelly (2012); Kelly et al. (2015); Chen et al. (2016a,b, 2017, 2018). These studies nevertheless have shown great potential of the PIC method in becoming a high quality CFD tool. In particular, Chen et al. (2016b) developed a Cartesian cut cell method based two-way strong fluid-solid coupling algorithm for wave interaction with floating bodies in their two-dimensional (2D) PIC framework. The key point of this coupling methodology is that the velocity of the rigid floating body has been implicitly represented by the pressure in cells immediately surrounding the solid. Thus, any implicit calculations of the velocity fluxes along the solid surface required by the cut cell method can be integrated into the procedure for solving a suitably amended pressure Poisson equation (PPE). This makes the proposed scheme both stable and efficient, as no iterations are needed when dealing with wave interaction with freely moving structures. Very recently, Chen et al. (2018) extended the PIC model of Chen et al. (2016b) to three spatial dimensions, and parallelised the model using the domain decomposition based Message Passing Interface (MPI) approach. Nevertheless, they only managed to apply the three-dimensional (3D) parallel PIC solver to wave interaction with fixed or motion prescribed structures.

Cut cell method has been widely employed in CFD modelling as an alternative to the traditional structured or unstructured body-fitted grid. Instead of having to regenerate the body-fitted grid as the boundary moves, the cut cell method uses the boundary segment to intersect with a stationary background grid, leading to simply different cut cells that are composed of the boundary segment and grid cell segments to represent the boundary surface. Yang et al. (1997) developed a Cartesian cut cell method applicable to compressible flows around static and moving bodies. Causon et al. (2000, 2001) proposed a Cartesian cut cell method for shallow water flows involving fixed and moving boundaries. Qian et al. (2006) later employed the cut cell method developed in the aforementioned papers to their two-fluid solver involving fluid interaction with moving solids. While in the aforementioned papers the cut cell method is developed in a collocated Cartesian grid environment, Ng et al. (2009) proposed a cut cell method within a staggered grid arrangement for fluid interaction with fixed and motion prescribed structures. Later, their cut cell approach was developed by Chen et al. (2016b) to simulate 2D freely moving structures as mentioned above. In this paper, the cut cell approach of Chen et al. (2016b) is further extended to model 3D floating bodies.

In the open literature, investigations on wave interaction with floating bodies have been carried out extensively using various numerical models and physical experiments. Physical experimental data is required to validate the numerical models, which in turn can help select experimental conditions and reduce the cost of physical modelling studies. Hann et al. (2015) experimentally studied focused wave interaction with a simplified wave energy converter (WEC), consisting of a free-floating buoy and a mooring system that encourages the occurrence of extreme snatch load. Ransley et al. (2017b) simulated regular wave interaction with a freely-pitching, 1:10 scale model of the Wavestar using OpenFOAM® and successfully reproduced the fully coupled motion of the device. Using the same OpenFOAM® model, Ransley et al. (2017a) studied focused wave interaction with the simplified WEC presented in Hann et al. (2015), with an alternative mooring system that does not encounter snatch loads. Their OpenFOAM® model well reproduced the motion of the buoy and mooring load measured in physical experiments. Omidvar et al. (2013) applied the SPH method with variable mass distribution to a single heaving-float WEC, known as the 'Manchester Bobber', in extreme waves and compared the results with experiments in a wave tank. Lind et al. (2016) simulated the experiment of Hann et al. (2015) using SPH with the Froude-Krylov approximation. Their SPH model well reproduces the snatch and non-snatch mooring load in non-breaking waves, but predicts the snatch mooring load less accurately in breaking waves. Gunn et al. (2018) investigated regular wave interaction with a floating moored spherical buoy also using the SPH method and provided experimental data for validation. Their numerical results based on the SPH method are very promising and compare well with the experimental measurements of the motion of the buoy. These studies provide useful data for validating other computational methods.

In this paper we further extend the 3D parallel PIC model proposed in Chen et al. (2018) to simulate wave interaction with floating bodies using the fluid-solid coupling algorithm proposed in Chen et al. (2016b). In particular, as there is still a lack of confidence in the capability of numerical models on handling extreme wave events and their interaction with floating structures (Ransley et al., 2017a), physical experiments of focused wave interaction with a moored floating buoy, encountering both snatch and non-snatch mooring load, are used to validate the present numerical model. We show that the newly developed 3D parallel PIC solver is capable of modelling extreme wave interaction with floating bodies both accurately and efficiently.

The paper is organised as follows: Section 2 gives an overview of the current PIC model including the governing equations and major numerical implementations. Next, in Section 3 the numerical model is validated against an existing experiment of focused wave interaction with a moored floating buoy. Finally, in Section 4 conclusions are drawn.

2. Numerical Model

2.1. Governing equations

The present PIC model solves the incompressible Newtonian Navier-Stokes equations for single-phase flow:

$$\nabla\cdot \boldsymbol{u} \;=\; 0, \tag{1}$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u}\cdot\nabla)\boldsymbol{u} = \boldsymbol{f} - \frac{1}{\rho}\nabla p + \nu\nabla^2\boldsymbol{u}, \tag{2}$$

where, in 3D, $\boldsymbol{u} = [u, v, w]^T$ is the velocity field; $\boldsymbol{f}$ = [0.0,0.0,-9.81 m/s^2]^T represents the body force due to gravity; p is pressure; t is time, and ν and ρ are the kinematic viscosity and density of the fluid respectively. Both a set of particles and a underlying grid are employed to discretise the computational domain. Following Harlow and Welch (1965), a staggered grid is used where pressures are computed at cell centres, whose positions along the x-, y- and z-directions are labelled by the indices i, j and k respectively, and velocities are calculated at the centres of relevant cell faces, whose positions are numbered with half-integer values of the indices. Fig. 1 shows a schematic of the computational setup, where both the staggered grid and the fluid particles are sketched. The particles carry the fluid properties such as the mass and momentum, and are used to solve the nonlinear advection term (the second term on the left hand side (LHS) of Eq. 2) in a Lagrangian manner and hence track the configuration of the fluid including the free-surface position, while the underlying grid is employed solely for computational convenience for solving the non-advection terms in an Eulerian sense. Initially, eight particles are seeded in each cubic cell accommodating the fluid area, and as the simulation progresses cells occupied by the particles are marked as fluid cells.

Two main steps are used to solve the governing equations, and they are an Eulerian step and a Lagrangian step. First, in the Eulerian step the governing equations are solved on the grid with the nonlinear advection term being ignored. Then, in the Lagrangian step the solution on the grid including a divergence-free velocity field and an acceleration field are used to update the velocity field carried by the particles, and the remaining advection term is solved by moving the particles in a Lagrangian manner. The fluid-structure interaction is resolved during the Eulerian step and the velocity and position of the structure are advanced during the Lagrangian step. It is noted that no turbulence models are incorporated in the present numerical model, thus the test case used for validation study in Section 3 is carefully selected. For full details of the solution procedure, the interested reader is referred to Chen et al. (2018). In what follows, the major components and equations used in the 3D PIC model are briefly introduced, with the implementation of the fluid-structure interaction algorithm for freely moving structures being highlighted.

2.2. Eulerian step

In the Eulerian step, the governing equations ignoring the nonlinear advection term in the momentum equation are solved on the grid. Note that prior to the solutions, the velocity field carried by the particles $\boldsymbol{v}_p^n$ at the time step n is mapped onto the grid to form a velocity field $\boldsymbol{u}^n$. This is done by using a kernel interpolation that conserves mass and momentum (see more details in Chen et al. (2018)). The solution uses the pressure projection method proposed in Chorin (1968). The governing equations are solved and the time is advanced in

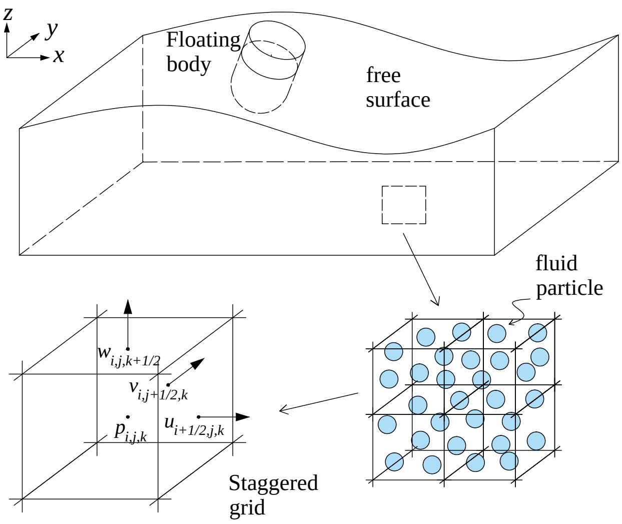

Fig. 1: Sketch of the computational domain, the staggered grid and fluid particles.

the following steps:

$$\frac{\tilde{\boldsymbol{u}} - \boldsymbol{u}^n}{\Delta t} = \nu \nabla^2 \boldsymbol{u}^n + \boldsymbol{f}, \tag{3}$$

$$\frac{(\boldsymbol{u}^{n+1} - \tilde{\boldsymbol{u}})}{\Delta t} = -\rho^{-1} \nabla p^{n+1}, \tag{4}$$

$$\Delta t \rho^{-1} \nabla^2 p^{n+1} = \nabla \cdot \tilde{\boldsymbol{u}}\,, \tag{5}$$

$$\boldsymbol{u}^{n+1} = \tilde{\boldsymbol{u}} - \Delta t \rho^{-1} \nabla p^{n+1}, \tag{6}$$

where $\tilde{\boldsymbol{u}}$ is a tentative velocity between $\boldsymbol{u}^n$ and $\boldsymbol{u}^{n+1}$ and Δt is the time step. Eq. 5 is a pressure Poisson equation (PPE), which is discretised and solved in a finite volume sense in the current solver. In addition, during the solution of the PPE, the boundary conditions on both the free surface and the structure surface are resolved.

Following Ng et al. (2009), the boundary conditions imposed on the structure surface are:

$$\boldsymbol{n} \cdot \boldsymbol{u} = \boldsymbol{n} \cdot \boldsymbol{U}_b \quad \text{and} \quad \boldsymbol{n} \cdot (\Delta t \rho^{-1} \nabla p) = \boldsymbol{n} \cdot (\tilde{\boldsymbol{U}}_b - \boldsymbol{U}_b^{n+1}) \quad \text{on } \partial\Omega_S(\boldsymbol{x}, t), \tag{7}$$

where $\tilde{\boldsymbol{U}}_b$ represents a tentative velocity on the structure surface; $\boldsymbol{U}_b^{n+1}$ is the velocity on the structure surface at time step $n+1$; $\boldsymbol{n}$ is the unit outward normal vector of the structure surface and $\partial\Omega_S$ represents the structure surface. Integrating both sides of the PPE (Eq. 5) over a fluid cell, G_{ijk}, that is partially occupied by a solid structure and evoking the divergence theorem and Eq. 7, a discretised PPE can be expressed by:

$$
\begin{aligned}
E_{i-\frac{1}{2},j,k} \cdot \frac{\Delta t(p^{n+1}_{i-1,j,k} - p^{n+1}_{i,j,k})}{\rho \Delta x} + E_{i+\frac{1}{2},j,k} \cdot \frac{\Delta t(p^{n+1}_{i+1,j,k} - p^{n+1}_{i,j,k})}{\rho \Delta x} + \\
E_{i,j-\frac{1}{2},k} \cdot \frac{\Delta t(p^{n+1}_{i,j-1,k} - p^{n+1}_{i,j,k})}{\rho \Delta y} + E_{i,j+\frac{1}{2},k} \cdot \frac{\Delta t(p^{n+1}_{i,j+1,k} - p^{n+1}_{i,j,k})}{\rho \Delta y} + \\
E_{i,j,k-\frac{1}{2}} \cdot \frac{\Delta t(p^{n+1}_{i,j,k-1} - p^{n+1}_{i,j,k})}{\rho \Delta z} + E_{i,j,k+\frac{1}{2}} \cdot \frac{\Delta t(p^{n+1}_{i,j,k+1} - p^{n+1}_{i,j,k})}{\rho \Delta z} \\
= E_{i+\frac{1}{2},j,k} \cdot \tilde{u}_{i+\frac{1}{2},j,k} - E_{i-\frac{1}{2},j,k} \cdot \tilde{u}_{i-\frac{1}{2},j,k} + E_{i,j+\frac{1}{2},k} \cdot \tilde{v}_{i,j+\frac{1}{2},k} - E_{i,j-\frac{1}{2},k} \cdot \tilde{v}_{i,j-\frac{1}{2},k} + \\
E_{i,j,k+\frac{1}{2}} \cdot \tilde{w}_{i,j,k+\frac{1}{2}} - E_{i,j,k-\frac{1}{2}} \cdot \tilde{w}_{i,j,k-\frac{1}{2}} - \int_{G_{ijk} \bigcap \partial\Omega_S} \boldsymbol{n} \cdot \boldsymbol{U}_b^{n+1} \, dA \, , \quad (8)
\end{aligned}
$$

where the subscripts are the space indices described in Section 2.1; E represents the area of a cell face that is not occupied by structures; dA represents the area differential; Δx, Δy and Δz are the grid sizes in the x-, y- and z-directions respectively and note that a uniform grid is currently employed in the solver, i.e. $\Delta x = \Delta y = \Delta z$. The interested reader is referred to Ng et al. (2009) for the derivation of Eq. 8. It is noticed that the last term on the right hand side (RHS) of Eq. 8 is a velocity integral on the structure surface within the computational cell $G_{i,j,k}$, and the integral involves the velocity $\boldsymbol{U}_b^{n+1}$ imposed on the structure surface at the time step $n+1$. For fixed and motion prescribed structures (e.g. a wavemaker), $\boldsymbol{U}_b^{n+1}$ is known. However, for freely moving structures, $\boldsymbol{U}_b^{n+1}$ is unknown at the time step n when Eq. 8 needs to be solved. The Cartesian cut cell based two-way strong fluid-solid coupling algorithm presented in Chen et al. (2016b) is employed to resolve this issue. Here, the solution is to transfer the structure velocity to the fluid pressures in cells immediately surrounding the structure:

$$
\boldsymbol{U}^{n+1} = \boldsymbol{U}^n + \Delta t \mathbf{M}_s^{-1} \mathbf{J} p^{n+1} + \Delta t \mathbf{M}_s^{-1} (\boldsymbol{F}_g + \boldsymbol{F}_{ext}), \quad (9)
$$

where $\boldsymbol{U}^{n+1}$ and $\boldsymbol{U}^n$ are the structure velocities at time steps $n+1$ and n, respectively; $\mathbf{M}_s$ is the mass matrix of the structure; $\mathbf{J}$ is an operator that maps the pressures to net forces and torques on the structure; $\boldsymbol{F}_g$ denotes the force and torque on the structure due to gravity; $\boldsymbol{F}_{ext}$ represents the external forces and torques due to, for example, moorings. Once the structure velocity $\boldsymbol{U}^{n+1}$ is constructed using Eq. 9, the velocity integral in Eq. 8 can be expressed purely in terms of the pressures to be solved for, leading to a revised PPE. The construction of operator $\mathbf{J}$ and the handling of the velocity integral on the structure surface are discussed in Section 2.4.1. The resulting linear system of equations are solved using the bi-conjugate gradient (BCG) method (Press et al., 1992).

On the free surface, the boundary condition enforced is:

$$p = 0 \quad \text{on } \zeta(\boldsymbol{x}, t), \tag{10}$$

where $\zeta(\boldsymbol{x}, t)$ represents the free-surface position reconstructed on the grid based on the particle position. The implementation of the free-surface boundary condition within the current PIC model is detailed in Chen et al. (2018) and is not repeated here.

2.3. Lagrangian step

In this step, the velocity field carried by the particles is updated and the particles are moved to solve the remaining nonlinear advection term in a Lagrangian manner. To update the particle velocity in the PIC framework, two approaches are commonly used. One is to directly interpolate the velocity field from the grid, and the other is to increment the particle velocity field through an acceleration field, $\boldsymbol{a}^{n+1} = \boldsymbol{u}^{n+1} - \boldsymbol{u}^n$, on the grid. While the former approach is commonly referred to as "classical" PIC (Harlow, 1964), the latter one is characterised as "full particle" PIC (Brackbill and Ruppel, 1986). While "classical" PIC is more dissipative and stable due to the velocity interpolation back and forth, "full particle" PIC leads to much less numerical dissipation because the velocity increment is relatively small at each time step. Nevertheless, by incrementing the particle velocity at each time step "full particle" PIC also allows the associated numerical errors to accumulate which can cause numerical instability (Jiang et al., 2015). As a trade-off between numerical stability and accuracy, Zhu and Bridson (2005) proposed using an empirical blending coefficient between "classical" PIC and "full particle" PIC, which calculates the final particle velocity by:

$$\boldsymbol{v}_p^{n+1} = c(\boldsymbol{v}_p^n + \sum_i \boldsymbol{a}^{n+1} S_i) + (1-c) \sum_i \boldsymbol{u}^{n+1} S_i \,, \tag{11}$$

where $\boldsymbol{v}_p$ is the particle velocity; S_i represents an interpolation function, and c is the blending coefficient. Eq. 11 is used in the current PIC framework, and c is set at 0.96 following Chen et al. (2016b) so as to stabilise the code while keeping the associated numerical dissipation as low as possible. After the velocity field carried by the particles are updated, the particles are then moved through the divergence-free velocity field on the grid using the third-order accurate Runge-Kutta scheme of Ralston (1962). Details of these implementations are introduced in Chen et al. (2018). Finally, after the particles are advected one computational cycle of solving the Navier-Stokes equations is completed.

As mentioned above, the velocity and position of the structure are also advanced in this step. Following Chen et al. (2016b), the velocity of the structure is updated using Eq. 9, with $\mathbf{J}p^{n+1}$ being replaced by an integral of the fluid pressure over the wetted area of the structure surface. Once the structure velocity is updated, the translational displacement and rotational angle of the structure, $\boldsymbol{D}^n$, is calculated by:

$$\boldsymbol{D}^n = \frac{(\boldsymbol{U}^n + \boldsymbol{U}^{n-1})}{2}\Delta t\,. \tag{12}$$

Assuming that all rotations are small at each time step, the sequence of rotation becomes unimportant and the Euler angles are used in the current implementation. Take (x, y, z) to be a point on the structure surface with reference to a coordinate system localised at the moving structure. After the rotations involving three angles $(\theta_x, \theta_y, \theta_z)$ with reference to the axes of the local coordinate system, the new coordinate of that point (X, Y, Z) within the local coordinate system is calculated by:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} \cos\theta_y\cos\theta_z & -\cos\theta_x\sin\theta_z + \sin\theta_x\sin\theta_y\cos\theta_z & \sin\theta_x\sin\theta_z + \cos\theta_x\sin\theta_y\cos\theta_z \\ \cos\theta_y\sin\theta_z & \cos\theta_x\cos\theta_z + \sin\theta_x\sin\theta_y\sin\theta_z & -\sin\theta_x\cos\theta_z + \cos\theta_x\sin\theta_y\sin\theta_z \\ -\sin\theta_y & \cos\theta_y\sin\theta_x & \cos\theta_y\cos\theta_x \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{13}$$

2.4. Additional numerical implementations

2.4.1. Construction of operator **J**

As discussed in Section 2.2, the two-way strong fluid-solid coupling algorithm employed for floating bodies requires an operator $\mathbf{J}$ that maps the fluid pressure to net forces and torques on the structure. The operator $\mathbf{J}$ is formed following Batty et al. (2007). For example, the x-component of the translational force on the structure can be written as:

$$F_x = -\iint_{\partial\Omega_S} p\boldsymbol{n}\,dA = -\iiint_{\Omega_S} \nabla p\,dV \simeq -\sum_{i,j,k} V_{i+1/2,j,k}\frac{p_{i+1,j,k} - p_{i,j,k}}{\Delta x}\,, \tag{14}$$

where dA and dV are the area and volume differential respectively; $V_{i+1/2,j,k}$ is the volume of velocity cell that is occupied by the structure; the velocity cell in this case is the cubic cell whose centre is located at $u_{i+1/2,j,k}$. Here, in the 3D code the volume of velocity cell is computed in the same manner as that proposed in Chen et al. (2016b). Rewriting Eq. 14, the x-component of the translational force part of the operator $\mathbf{J}$ is obtained:

$$J_{1,(i,j,k)} = \frac{V_{i+1/2,j,k} - V_{i-1/2,j,k}}{\Delta x} \,. \tag{15}$$

The y- and z-components of the translational force part $J_{2,(i,j,k)}$ and $J_{3,(i,j,k)}$ are formed in the same manner.

Similarly, the torque on a structure can be expressed by:

$$\boldsymbol{T} = -\iint_{\partial\Omega_S} (\boldsymbol{r} - \boldsymbol{r}_c) \times p\boldsymbol{n}\, dA = \iiint_{\Omega_S} \nabla p \times (\boldsymbol{r} - \boldsymbol{r}_c)\; dV \,, \tag{16}$$

where $\boldsymbol{r}_c$ is the structure rotation centre and $\boldsymbol{r}$ is the point of action of a fluid force fraction. Discretising and rewriting Eq. 16 in the same manner as above, the torque part of the operator $\mathbf{J}$ with reference to the x-axis, for example, is finally obtained:

$$J_{4,(i,j,k)} = -\frac{V_{i,j+1/2,k} - V_{i,j-1/2,k}}{\Delta y} \left(z_{i,j,k} - Z_c\right) + \frac{V_{i,j,k+1/2} - V_{i,j,k-1/2}}{\Delta z} \left(y_{i,j,k} - Y_c\right) \,, \tag{17}$$

where Y_c, Z_c and $y_{i,j,k}$, $z_{i,j,k}$ are the coordinates of the structure rotation centre and the point of action, respectively. Note that the torque parts with reference to the y- and z-axes $J_{5,(i,j,k)}$ and $J_{6,(i,j,k)}$ are computed in the same manner.

With the operator $\mathbf{J}$ being constructed, the structure velocity $\boldsymbol{U}^{n+1}$ can be explicitly expressed in terms of the pressure via Eq. 9, Eq. 15 and Eq. 17. Therefore, the velocity integral at the RHS of Eq. 8 can also be expressed as a function of the pressure in cells immediately surrounding the structure. This is because the velocity at any point on the structure surface can be calculated by:

$$\boldsymbol{U}_b^{n+1} = \boldsymbol{U}_t^{n+1} + \boldsymbol{U}_w^{n+1} \times \boldsymbol{R} \,, \tag{18}$$

where $\boldsymbol{U}_t^{n+1}$ and $\boldsymbol{U}_w^{n+1}$ are the translational and the angular velocities of the structure at time step $n+1$ respectively, and $\boldsymbol{R} = \boldsymbol{r} - \boldsymbol{r}_c$ denotes a vector pointing from the structure rotation centre to a point on the structure surface.

The structure boundary is discretised into a set of triangular elements in the pre-processing. Fig. 2 shows a schematic of a computational cell cut by a structure surface (the grey area), for which the triangular elements are also depicted. So, in the cell $G_{i,j,k}$, for

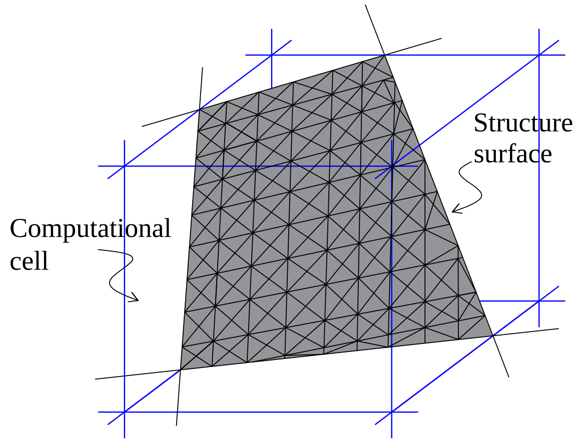

Fig. 2: A sketch showing a computational cell being occupied by a structure whose surface (the grey area) is discretised into a set of triangular elements.

example, the velocity integral of Eq. 8 can be approximated by:

$$\int_{G_{ijk}\bigcap\partial\Omega_S} \boldsymbol{n}\cdot\boldsymbol{U}_b^{n+1}dA \simeq \sum_{n_{ijk}} \boldsymbol{n}_k\cdot\left(\boldsymbol{U}_k^{temp}+\boldsymbol{Q}_k(\mathbf{M}_s,\mathbf{J},\Delta t,p^{n+1})\right)\Delta A_k\,, \tag{19}$$

where the subscript k represents the kth triangular element; $\boldsymbol{n}_k$ is the outward pointing unit normal vector; $\boldsymbol{U}_k^{temp}$ represents the updated velocity on the structure surface due to Eq. 18 and $\boldsymbol{U}^n+\Delta t\mathbf{M}_s^{-1}(\boldsymbol{F}_g+\boldsymbol{F}_{ext})$ in Eq. 9, of which the variables are all knowns at the time step n; $\boldsymbol{Q}_k$ denotes the boundary velocity transferred from the pressure immediately surrounding the structure; ΔA_k is the area of the triangular element; n_{ijk} is the total number of triangular elements located inside cell $G_{i,j,k}$. n_{ijk} is computed at each time step by detecting whether the centroid of a triangular element is located inside the cell or not. This could lead to some errors when the discretisation elements of the structure surface are relatively coarse. Therefore, in the present work the triangular elements are generated with a characteristic area of approximately $(\Delta x)^2/55$. Note that $\boldsymbol{n}_k$, $\boldsymbol{U}_k^{temp}$ and $\boldsymbol{Q}_k$ are all defined at the centroid of each triangular element, whose coordinates are denoted by (XCE, YCE, ZCE).

In Eq. 19, $\boldsymbol{U}_k^{temp}$ and $\boldsymbol{Q}_k$ are calculated/constructed according to Eq. 18, with $\boldsymbol{U}_t^{n+1}$ and $\boldsymbol{U}_w^{n+1}$ (see Eq. 18) being the velocity components due to $\boldsymbol{U}^n+\Delta t\mathbf{M}_s^{-1}(\boldsymbol{F}_g+\boldsymbol{F}_{ext})$ (for calculation of $\boldsymbol{U}_k^{temp}$) and $\Delta t\mathbf{M}_s^{-1}\mathbf{J}p^{n+1}$ (for construction of $\boldsymbol{Q}_k$), respectively. For example, the x-direction component of $\boldsymbol{Q}_k = (UX, VY, WZ)$ is expressed by (assuming that the rotational centre coincides with the centre of mass):

$$UX = \Delta t M^{-1} \sum_{q=1}^{m} J_{1,q} p_q^{n+1} + \Delta t \left(I_{21} \sum_{q=1}^{m} J_{4,q} p_q^{n+1} + I_{22} \sum_{q=1}^{m} J_{5,q} p_q^{n+1} + I_{23} \sum_{q=1}^{m} J_{6,q} p_q^{n+1} \right) (ZCE - Z_c) - \Delta t \left(I_{31} \sum_{q=1}^{m} J_{4,q} p_q^{n+1} + I_{32} \sum_{q=1}^{m} J_{5,q} p_q^{n+1} + I_{33} \sum_{q=1}^{m} J_{6,q} p_q^{n+1} \right) (YCE - Y_c) = \sum_{q=1}^{m} B_{x,q} p_q^{n+1}\,, \quad (20)$$

where M is the mass of structure; the subscript q denotes the index of the cells immediately surrounding the structure (i.e. partially occupied by the structure) and m is the total number of such cells at each time step; I_{ab} ($a = 1, 2, 3$ and $b = 1, 2, 3$) is the element of the 3×3 inverse matrix of the moment of inertia matrix of the structure; $B_{x,q}$ is the x-direction component of a coefficient vector $\boldsymbol{B}_q = (B_{x,q}, B_{y,q}, B_{z,q})$ that is related to the calculations of UX, VY and WZ. Note that VY and WZ are constructed in the same manner, as are the coefficients $B_{y,q}$ and $B_{z,q}$, respectively. The $\boldsymbol{Q}_k$ related term in Eq. 19 (i.e. $\sum_{n_{ijk}} \boldsymbol{n}_k \cdot \boldsymbol{Q}_k(\mathbf{M}_s, \mathbf{J}, \Delta t, p^{n+1}) \Delta A_k$) connects all pressures immediately surrounding the structure, and is added to the LHS of Eq. 8, modifying the coefficient matrix of the linear system of equations. The coefficient matrix is now not necessarily symmetric or positive definite due to the above manipulation, as the $\boldsymbol{Q}_k$ related term changes between cells due to the different cell volumes occupied by the structure in each velocity cell. However, the linear system of equations under these conditions are still solvable using the BCG solver (Press et al., 1992).

2.4.2. Numerical wave tank

In the present work, a numerical wave tank (NWT) is established following Chen et al. (2018). Uni-directional waves are generated in the x-direction by a piston-type wave paddle employed at one end of the NWT, and the waves are absorbed at the other end of the NWT by a relaxation zone. For full details of the NWT in the current PIC model, the reader is referred to Chen et al. (2018).

2.4.3. Numerical algorithm

The numerical algorithm used in the present model basically follows that presented in Chen et al. (2018). For presentation simplicity, only the major components with respect to the modelling of freely moving structures are given below.

(1) Calculate $\boldsymbol{U}^n + \Delta t \mathbf{M}_s^{-1}(\boldsymbol{F}_g + \boldsymbol{F}_{ext})$ in Eq. 9;

(2) Move the piston-type wave paddle according to the wave generation method;

(3) Map the mass and momentum carried by the particles to the grid and reconstruct the free-surface position on the grid based on the particle location;

(4) Construct Eq. 19 and Eq. 8 and solve the resulting linear system of equations;

(5) Project the tentative velocity field $\tilde{\boldsymbol{u}}$ onto a divergence-free velocity field through Eq. 6;

(6) Calculate the velocity acceleration field $\boldsymbol{a}^{n+1} = \boldsymbol{u}^{n+1} - \boldsymbol{u}^n$ on the grid;

(7) Update the structure velocity and then update the structure position through Eq. 12 and Eq. 13;

(8) Update the velocity field carried by the particles through Eq. 11 and then advect the particles;

(9) Conduct wave absorption in the relaxation zone;

(10) Update the time step (see details in Chen (2017)) and repeat steps (1)-(10).

3. Results and Discussions

In this section, the present numerical model is validated against the laboratory measurements of focused wave interaction with a floating, hemispherical-bottomed, cylindrical buoy with different mooring configurations: (1) a linearly-elastic mooring that encounters non-snatch loads (Ransley et al., 2017a); (2) more complex mooring system that encourages snatch loads (Hann et al., 2015). In both test cases the numerical model is validated first for the focused wave generation in the absence of the buoy, and then for the motion of the buoy and mooring force under focused wave action.

3.1. Test case 1: mooring configuration with non-snatch loads

3.1.1. Experimental setup

The experiment of Ransley et al. (2017a) was performed in the Ocean Basin at Plymouth University's Coastal, Ocean And Sediment Transport (COAST) laboratory. The basin is 35 m long and 15.5 m wide, with 24 flap-type wave paddles installed at one end and a parabolic beach at the other. The water depth at the wavemaker was 4 m and decreased to 2.8 m in

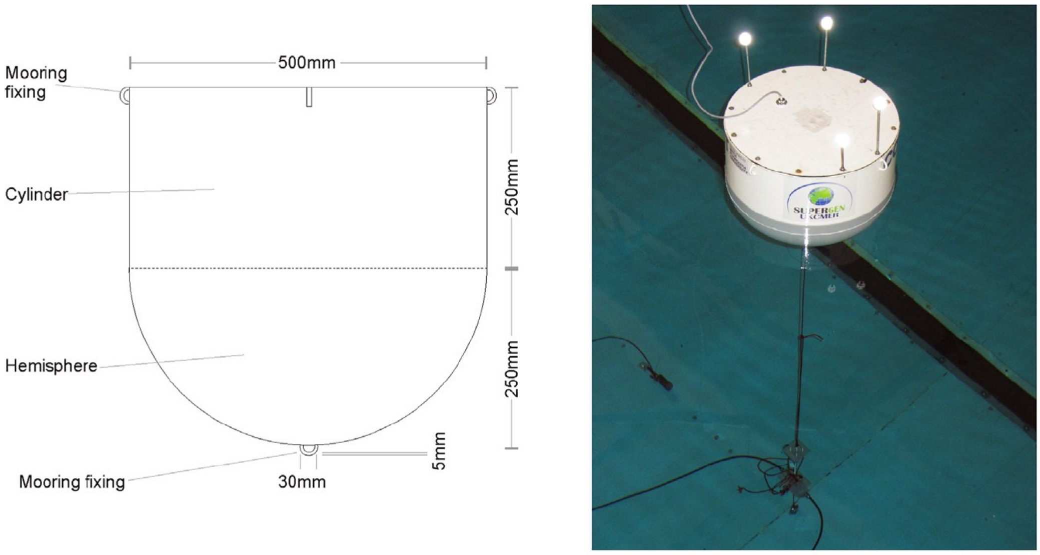

Fig. 3: A sketch of the buoy (left) and a photograph from the COAST laboratory (right) showing the experimental setup. This figure is reprinted from Ransley et al. (2017a), Copyright (2017), with permission from Elsevier.

the region where the buoy was placed. The buoy has a diameter $D = 0.5$ m and consists of a hemispheric at the bottom and a cylinder on the top (see Fig. 3). The buoy has a total mass of 43.2 kg, and its centre of mass is located at 0.181 m from the bottom. The moment of inertia of the buoy is (1.61 1.61 0.5) kgm^2. The motion of the buoy was restrained by a single point mooring, of which one end was attached to the bottom of the buoy and the other was fixed at the basin floor. The mooring can be modelled as a linear spring, having a stiffness $k = 67$ Nm^{-1} and a rest length of 2.18 m. Note that in this case the buoy can move in all 6 degrees of freedom. The focused wave was generated using the NewWave theory based on the Pierson-Moskowitz (PM) spectrum ($f_p = 0.356$ Hz) and wave gauges throughout the basin were used to measure the generated wave. For more details of the experimental setup, the reader is referred to Ransley et al. (2017a).

3.1.2. Numerical results: free decay test

Free decay tests of the buoy with and without the mooring were first used to validate the present solver for simulating the motion of the buoy and mooring force. The buoy was initially lifted up for a small distance away from its equilibrium position and then released, leading to a decay of the heave motion. In the numerical simulation, the buoy was placed at the centre of a 6 m $\times$ 6 m square domain, with the water depth being 2.8 m. A cylindrical relaxation zone centred at the buoy with an inner radius of 1 m and an outer radius of approximately 3 m was used to absorb the radiated waves away from the buoy. The grid size was set at $\Delta x = \Delta y = \Delta z = D/20$ for the case without mooring. For the case with mooring, three grid sizes were used in order to conduct a grid refinement study (note that the buoy is moored when interacting with the focused wave). The grid sizes were $\Delta x = \Delta y = \Delta z = D/15$, $D/20$ and $D/25$. Note that the grid size $D/25$ leads to approximately 15.3 million grid cells and 100.8 million particles; it took approximately 8.3 hours for 5 seconds of simulated time with 64 cores at the University of Bath High Performance Computing System (HPCS).

Fig. 4 shows the numerical results, in comparison with the experimental data, for the free decay test. All of the experimental data used for validation purposes in this test case are digitised from Ransley et al. (2017a). From Fig. 4(left), it is seen that the three grid sizes produce similar results, which indicates that the heave motion of the buoy in this case is not sensitive to the grid sizes used. The numerical results in general match well with the experimental data, although it is seen that the numerical results are less damped than that of the experiment especially towards the end of the time history. This is also seen in Fig. 4(right) which shows the comparison for the case without mooring. This may be because the grid size is not fine enough when the buoy motion is relatively small. However, the grid refinement study does not suggest great potential for a significant improvement if the grid size is further reduced, while maintaining feasible grid and particle resolution. Another concern is that in Eq. 9 the friction-related force is not considered, which may result in an underestimation of the damping force. As shown in the recent work of Gu et al. (2018), in the case of forced heave motion of a similar hemispherical base structure, the contribution of shear force to the drag coefficient may be of the same magnitude as that of pressure. However, due to the limited grid resolution in 3D modelling, the calculation of friction-related force (even if it is included in the current solver) is likely to be inaccurate as the boundary layer would not be fully resolved (Nematbakhsh et al., 2013). One solution may be to include a coupled dynamic adaptive grid and particle merging/splitting approach in the solver, such that the grid resolution around the structure could be sufficiently fine while the overall resolution is still feasible. Overall, the agreement between numerical and experimental results is reasonably good. In particular, the result of the case with mooring is as good as that without mooring, which provides confidence in the numerical solver for predicting the motion of the moored buoy under wave action. Note that based on the results the grid size $D/20$ was chosen for all the other simulations in this test case.

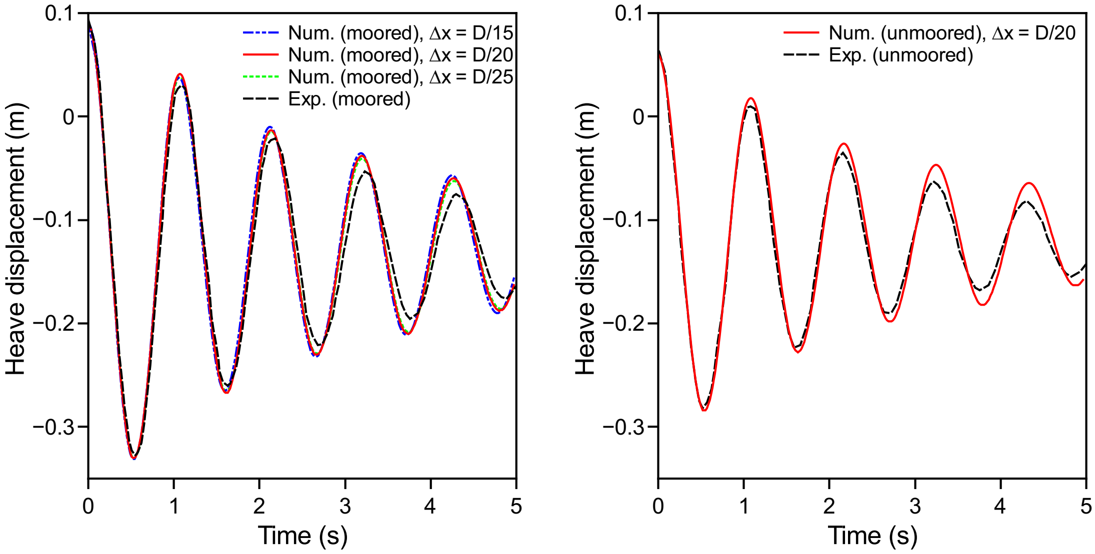

Fig. 4: Comparison between the numerical and experimental data for the heave displacement of the buoy during the free decay test. Left figure: with mooring; right figure: without mooring. The experimental data are digitised from Ransley et al. (2017a).

3.1.3. Numerical results: focused wave generation

In the numerical simulation, a piston-type wave paddle based on the first-order wave-maker theory was employed for wave generation and a relaxation zone approach was used for wave absorption (see Section 2.4.2). For the focused wave generation, the motion and velocity of the paddle were determined via the NewWave theory, based on the PM spectrum. In total, 100 wave components were used, with the frequency ranging from 0.2 Hz to 1.61 Hz. Fig. 5 shows a scheme of the setup of the NWT. The wave paddle in the current simulation was placed at the same location as that of the inlet boundary of the OpenFOAM® model by Ransley et al. (2017a). Note that this location is 8 m forward from the wave paddle used in the experiment. Ransley et al. (2017a) employed the wave gauge measurement in the experiment at this location to derive their expression based boundary conditions for the inlet boundary. However, this experimental measurement was not reported in Ransley et al. (2017a). Therefore, a trial and error process, adjusting the input theoretical focused location and focused wave amplitude, was used in the current wave paddle based simulation to generate the desired waves. The input focused location and wave amplitude were determined to be 5.2 m (from the numerical wave paddle) and 0.25 m respectively for this test case. Note that the input focused location in the experiment is expected to be much larger than 5.2 m. It is also worth noting that in the current simulations the generated focused wave amplitude was usually slightly larger than the input value, which is consistent with

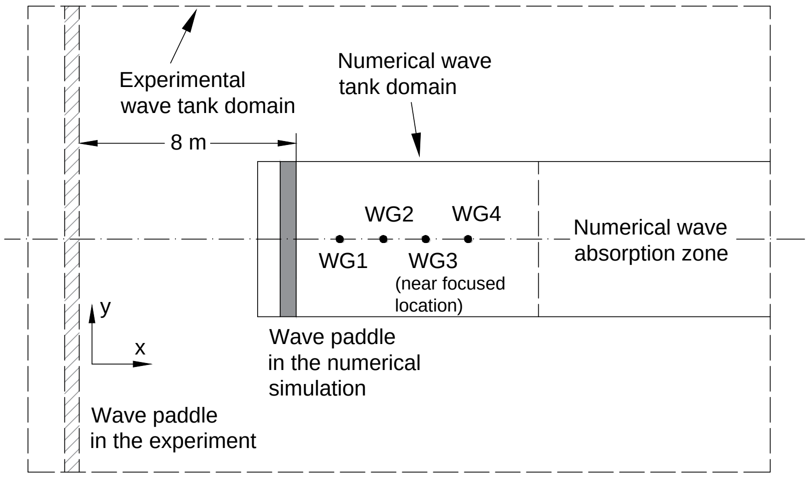

Fig. 5: Scheme (top view) showing the setup of the NWT. WG: wave gauge.

the experimental findings presented in Hann et al. (2015). This is, as also noted in Hann et al. (2015), due most likely to the nonlinear effects related to wave-wave interaction, which could also contribute to a shift of the actual focused location.

As the focused wave used in this test case was uni-directional (in the x-direction), to reduce the CPU effort the numerical domain was set at 0.5 m wide (in the y-direction) and 30 m long (in the x-direction), with 22.5 m dedicated to the relaxation zone. The water depth was set at 2.8 m. Note that the relaxation zone is relatively long because the peak frequency of the PM spectrum used is small: $f_p = 0.356$ Hz, which leads to a wavelength of approximately 11.27 m. The relaxation zone is thus set to nearly two times this wavelength in order to achieve the most cost-effective performance within the present PIC framework (Chen, 2017). The free-surface elevations at four locations, wave gauges 1 to 4 (see Fig. 5), along the x-direction centre line of the NWT were extracted to compare with the experimental data, and their distances to the wave paddle were (in metres): 1.96, 3.87, 5.60 and 6.33. Note that wave gauge 3 is close to the focused location.

Fig. 6 shows the numerical results of the generated focused wave compared to the experimental data. In general, it is seen that the agreement between the numerical and experimental data is quite good. In particular, the main crest and troughs are predicted well by the numerical model. This proves that in this test case by placing the numerical wave paddle closer to the actual focused location in the experiment and using a smaller input focused location, the present NWT can generate the desired focused wave, which provides a foundation to meaningful comparisons in the wave-structure interaction shown in the following section. However, it is not believed that the numerical wave paddle can be placed

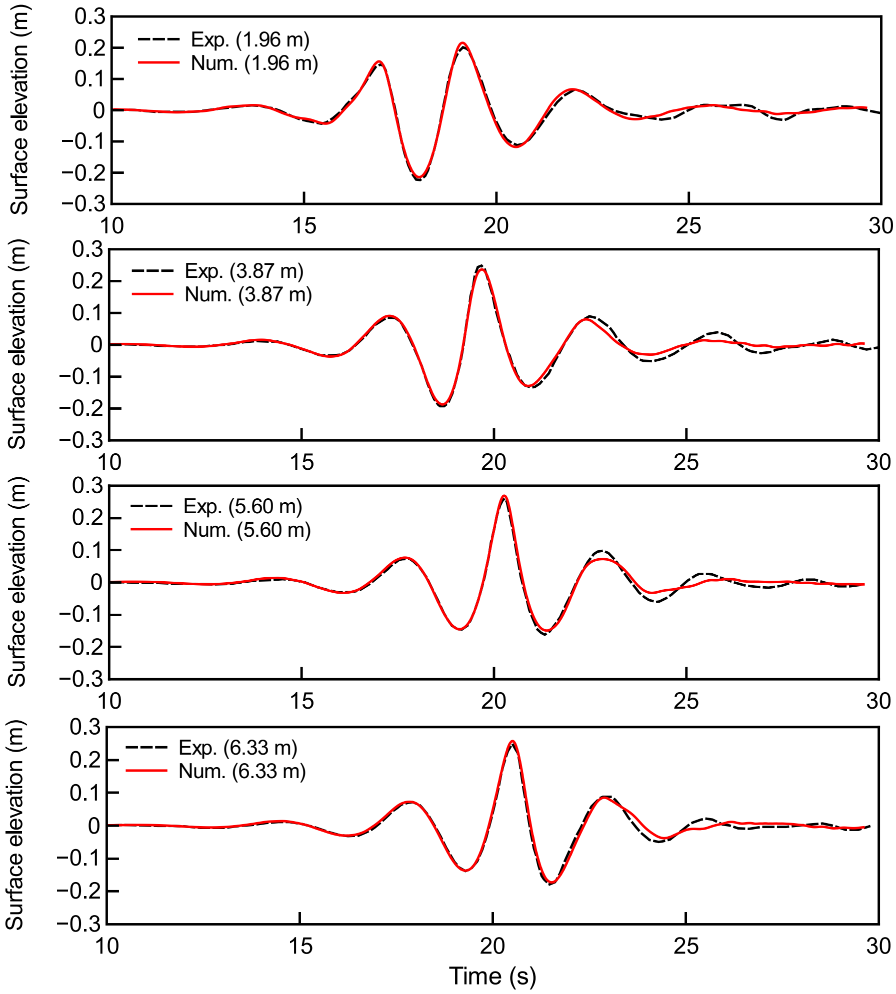

Fig. 6: Comparison between the numerical and experimental results for the free-surface elevations of the generated focused wave at four difference locations differing in the distance to the wave paddle. The experimental data are digitised from Ransley et al. (2017a).

too close to the actual focused location in the experiment, as the development of wave-wave interaction requires both space and time.

3.1.4. Numerical results: wave-structure interaction test

In this section, the focused wave described in Section 3.1.3 is used to interact with the moored buoy. The numerical domain is 6 m wide and 30 m long, with 22.5 dedicated to the relaxation zone. The water depth was set at 2.8 m. The buoy was placed at a distance of 5.49 m from the wave paddle on the centre line of the NWT.

Fig. 7 shows the snapshots of the numerical results at various times close to the focused time of the generated focused wave. In the snapshots, the width of the numerical domain is reduced and the mooring line is not shown to aid visualisation.

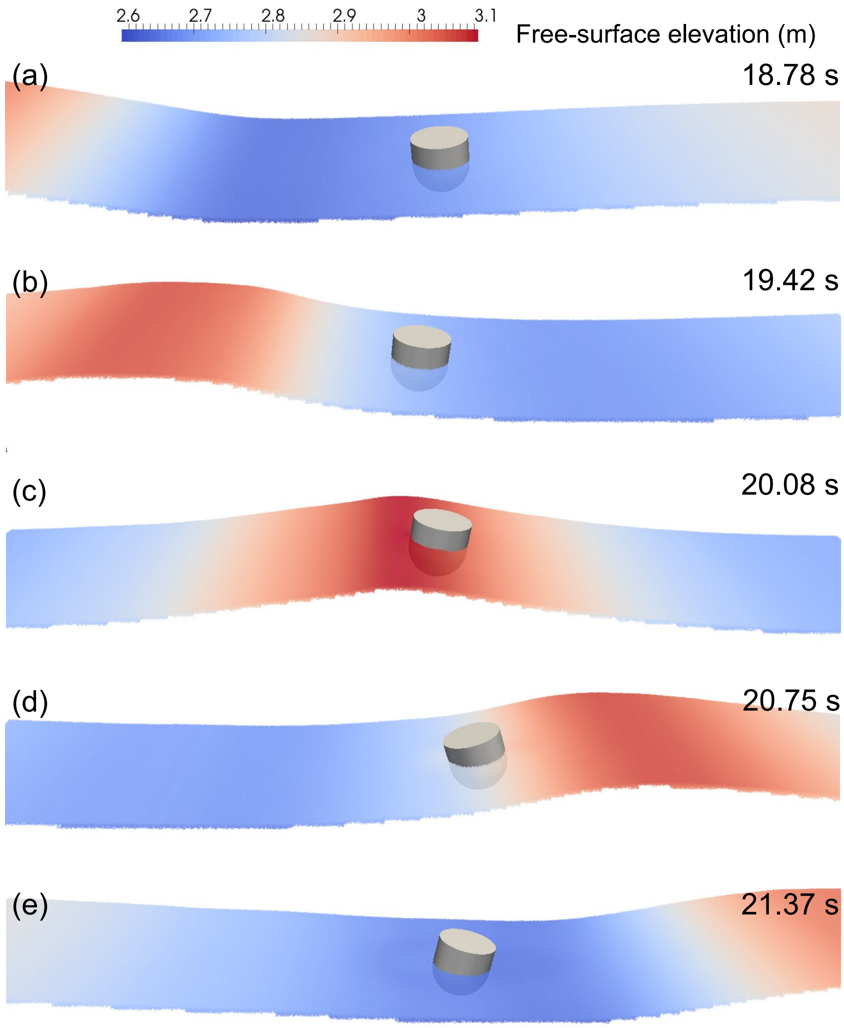

Fig. 7: Snapshots of the numerical results for the focused wave interaction with the buoy at different time instants. The mooring line is not shown but it is used in the simulation.

Fig. 8 presents the numerical results of the present PIC model for the surge displacement, heave displacement and pitch angle of the moored buoy under focused wave action compared to the numerical results of an OpenFOAM® model (digitised from Ransley et al. (2017a)) and the experimental measurements. In general, it is seen that very good agreement between the numerical and experimental results has been achieved, particularly during the period when the main crests and troughs of the focused wave move past the buoy. Also, it is noticed that around the third peaks of the surge and heave displacements, the present PIC model produces better results than the OpenFOAM® model of Ransley et al. (2017a). This could be due to a slightly better reproduction of the incident wave around the third

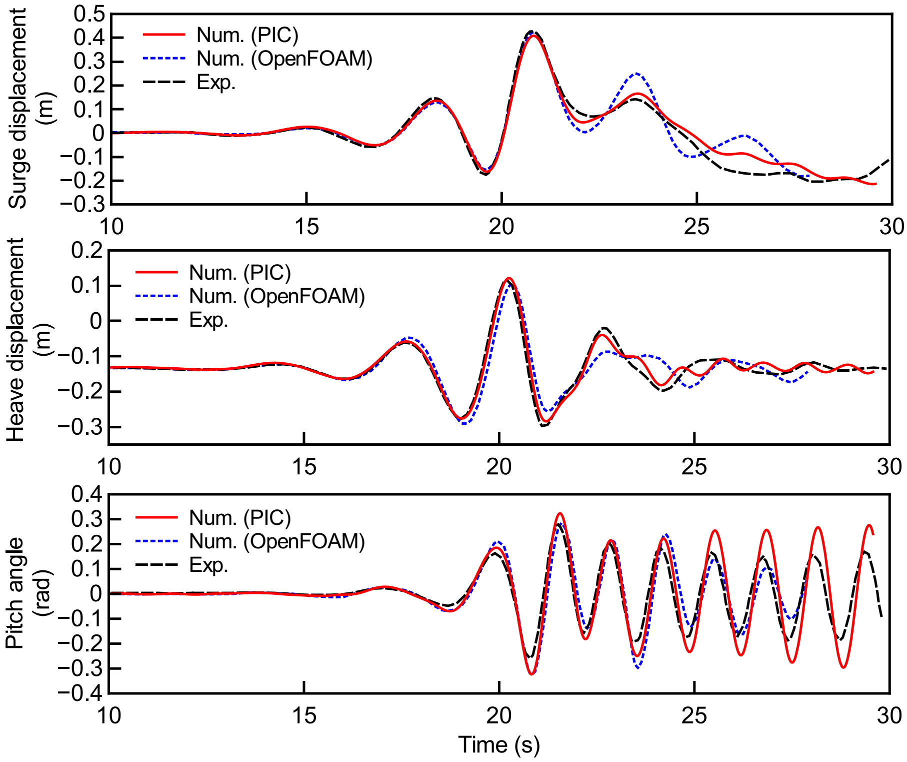

Fig. 8: Comparison between the numerical results of the present PIC model, the numerical results of the OpenFOAM® model of Ransley et al. (2017a) and the experimental data for the surge displacement (top), heave displacement (middle) and pitch angle (bottom) of the moored buoy under focused wave action. The experimental and OpenFOAM® data are digitised from Ransley et al. (2017a).

peak in terms of the wave shape (see Fig. 6 at 5.6 m and c.f. Fig.2(c) of Ransley et al. (2017a)). Another reason could be that after the main wave has passed (resulting in large buoy motion), the quality of the dynamic mesh used in the OpenFOAM® model is not as good as the initial one, which however is not the case for the current cut cell method where the underlying mesh is fixed and unchanged during the simulation. The comparison for the pitch amplitude, however, shows a slightly less satisfying agreement during the period of free oscillation of the buoy after the main wave has passed. This may be more evidence that the present PIC model predicts less damping effects when the buoy motion is small as discussed in Section 3.1.2. Nevertheless, the generally very good reproduction of the motion of the buoy clearly demonstrates the capability of the present PIC model as well as the two-way strong fluid-solid coupling algorithm for handling full 3D scenarios of wave interaction with floating bodies.

Fig. 9 shows the comparison for the mooring force. Again, very good agreement between the numerical and experimental results has been achieved, as a result of the good

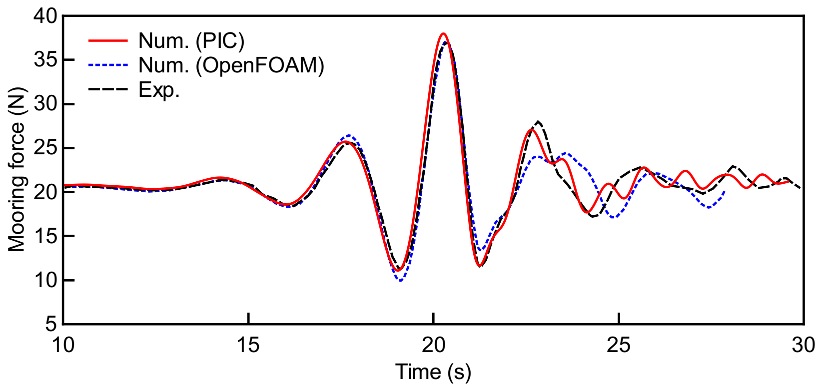

Fig. 9: Comparison between the numerical results of the present PIC model, the numerical results of the OpenFOAM® model of Ransley et al. (2017a) and the experimental data for the mooring force on the moored buoy under focused wave action. The experimental and OpenFOAM® data are digitised from Ransley et al. (2017a).

reproduction of the motion of the buoy.

Following the validation test, the motion of the buoy, without the mooring, under the same focused wave action was also investigated using the PIC model. This case could be considered as a situation when the mooring fails. Although there is no experimental data to compare with, the PIC model should be capable of predicting useful results since it has been validated using the above moored case. The results are plotted in Fig. 10, in comparison with those of the moored buoy. It can be observed that the surge displacement of the buoy is greatly affected by the mooring after the main focused wave has passed. Without the mooring the buoy tends to be shifted in the wave direction, rather than being pulled back as is the case when the mooring is attached. Similarly, the mooring appears to play an important role on the pitch motion of the buoy. In the unmoored case, the amplitude and the period of the pitch motion of the buoy are both larger. This is most likely because of the missing restoring forces on the buoy due to the mooring. Finally, it is seen from the middle panel that the heave displacement of the buoy is less affected by the mooring, compared to the surge displacement and the pitch angle. To understand this, the wave forces on the buoy with and without the mooring are examined. Fig. 11 shows the present numerical results for the wave forces on the buoy with and without the mooring. As can be seen, the wave forces in the surge direction (F_x) and heave direction (F_z) are less affected by the mooring than the torque in the pitch direction (M_y). However, the magnitude of the wave force in the heave direction is one order greater than that of the wave force in the surge direction, and the latter is in the same order as the magnitude of the mooring force (see Fig. 9). Therefore, the

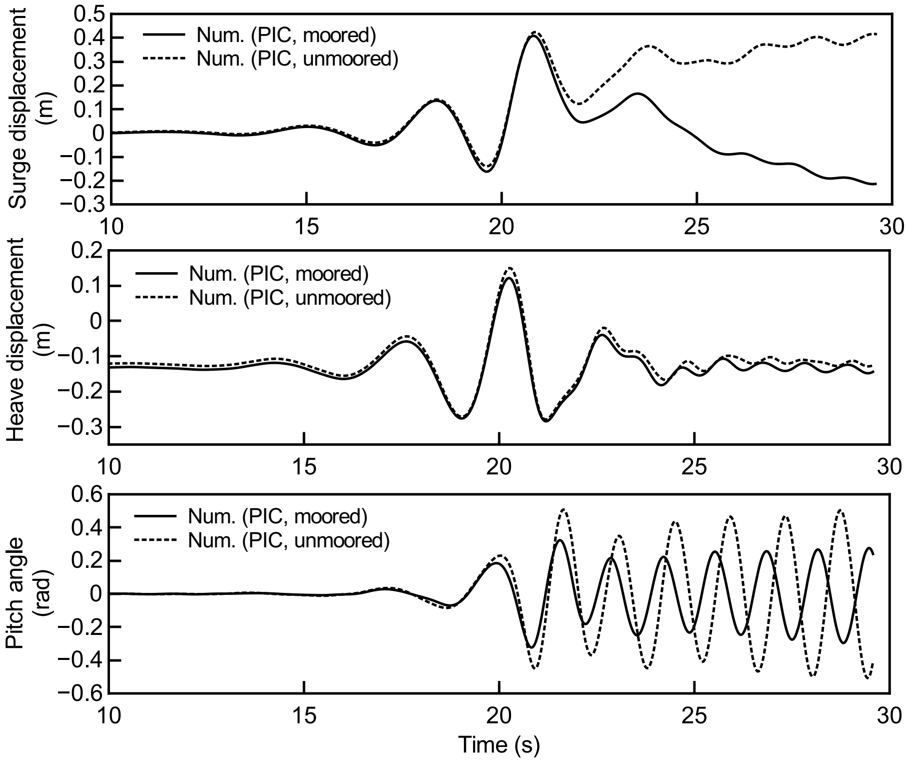

Fig. 10: Comparison between the numerical results of the motion of the buoy, with and without the mooring, under the focused wave action: surge displacement (top), heave displacement (middle) and pitch angle (bottom).

mooring has a smaller effect on the heave displacement of the buoy, as the mooring force is relatively small compared to the wave force in the heave direction. It is noticed that before the arrival of the main wave group, the wave force in the heave direction is greater in the moored case than that in the unmoored case. This is because that in the moored case the mooring is pretensioned, resulting in a larger draft of the buoy and hence larger hydrostatic force than those in the unmoored case.

Finally, in terms of the CPU cost on simulating the moored buoy case, it took approximately 32.9 hours for 30 s of simulated time using 160 cores at the University of Bath HPCS to run the present PIC model, while it took almost 500 hours of CPU time for 28 s simulation running on 6 processors for the OpenFOAM® model of Ransley et al. (2017a). As a very rough comparison using a coefficient $\epsilon = \frac{\text{Total CPU time}}{\text{simulated time}}$, the values of ϵ for the PIC model and the OpenFOAM® model are 176 and 107, respectively. So, the hybrid Eulerian-Lagrangian PIC model achieves a CPU efficiency of the same magnitude as the state-of-the-art OpenFOAM® model. It may be worth mentioning that for the PIC sim-

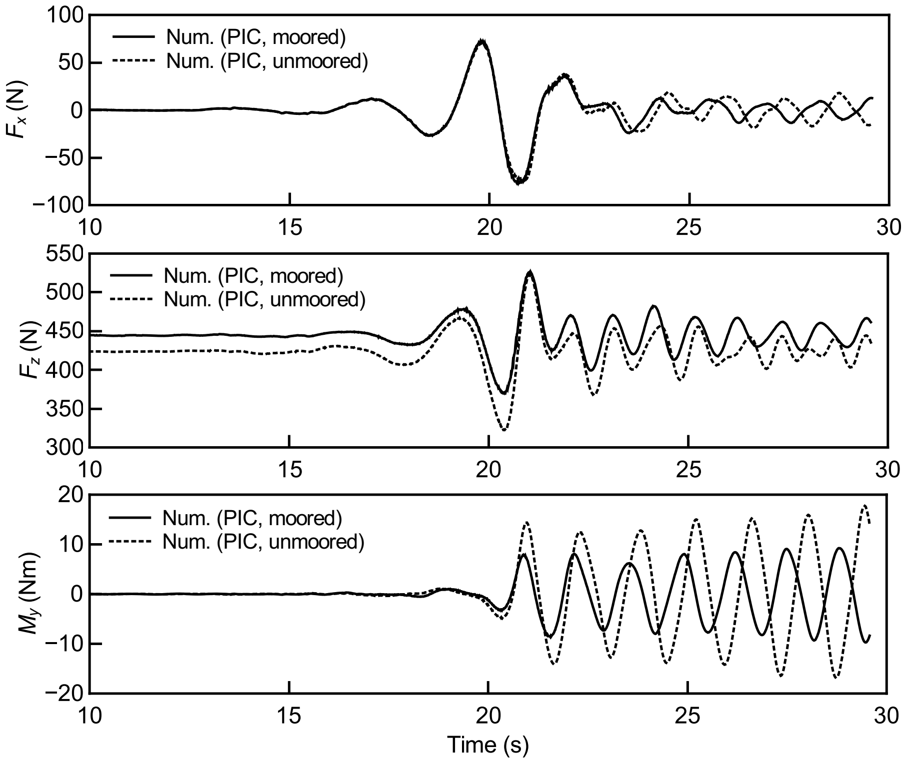

Fig. 11: Comparison between the numerical results of the wave forces on the buoy, with and without the mooring, under the focused wave action: wave force in the surge direction F_x (top), wave force in the heave direction F_z (middle) and pitch torque M_y (bottom).

ulation, approximately 31.72 million grid cells and 253.75 million particles were used to accommodate the water area only.

3.2. Test case 2: mooring configuration with snatch loads

3.2.1. Experimental setup

The experiment of Hann et al. (2015) is used for validation purpose in this test case. This experiment was also conducted in the Ocean Basin at Plymouth University's COAST laboratory. The same water depth (2.8 m) was used at the working section of the basin. Also, the same buoy was used, but with a different mooring setup that encourages snatch loads, which are transient but very large mooring forces experienced in extreme wave conditions (Lind et al., 2016). Here, the mooring system was composed of a spring (k =0.064 N/mm) in series with a very stiff long Dyneema® rope (spring constant, $k \approx 35$ N/mm). In addition, the maximum length of the spring was limited by another four short Dyneema® ropes in a parallel arrangement. So, the mooring force could encounter two phases: at first the mooring force is determined only by the spring extension, and then after the spring reaches its maximum length the snatch load occurs due to further extensions in the ropes. The rest and the maximum lengths of the spring were 0.152 m and 0.406 m respectively, and in still water the spring was extended to 0.257 m. The focused wave used in this test case was generated in the manner as that in the previous test case. Both breaking and non-breaking waves were tested in the experiment, although only a non-breaking wave case is used for the current numerical validation, namely case ST1 in Hann et al. (2015) with peak frequency $f_p = 0.356$ Hz and measured crest amplitude $A = 0.285$ m. To simulate the breaking wave cases, the numerical model would need further inclusions of an air phase (for effects like air cushioning) and a turbulence model (to handle the flow during and post wave breaking); the method presented in Kamath et al. (2016) for numerical modelling of breaking wave interaction with a vertical cylinder should be referred to. For full details of the experimental setup, the reader is referred to Hann et al. (2015).

3.2.2. Numerical results

In the experiment of Hann et al. (2015), the wave group of the chosen case (ST1) was focused at 18.51 m from the wave paddle, which was also located at the front face of the buoy in its initial rest location. In the current numerical simulation, to save on CPU cost the focused location was shifted to be 5.6 m from the wave paddle so that exactly the same NWT as that in the previous test case can be used, with only the buoy being placed at 5.85 m (= 5.6 + $D/2$) from the paddle. Also, the same grid size (0.025 m) was used in this test case. Fig. 12 shows the comparison between the numerical result and the experimental measurement for the time history of the surface elevation at the focused locations. It can be seen that the focused wave is well reproduced in the numerical simulation, demonstrating that the setup of the NWT is acceptable for this test case. Note that the surface elevations are both normalised by the theoretical crest value (0.267 m) used in the experiment, and both data series have been shifted in time so that the main crest occurs at $t = 0$ s. All of the experimental data used in this test case for validation purposes are digitised from Hann et al. (2015).

In the current simulation, once the spring reaches its maximum length the snatch mooring load is calculated following Lind et al. (2016):

$$\boldsymbol{F}_m = -k_{eq}\boldsymbol{x}_m - c\dot{\boldsymbol{x}}_m, \tag{21}$$

where k_{eq} is the equivalent spring constant for the mooring system and is set to $k_{eq} =$ 28 N/mm following Lind et al. (2016), $\boldsymbol{x}_m$ and $\dot{\boldsymbol{x}}_m$ are the mooring extension and rate of

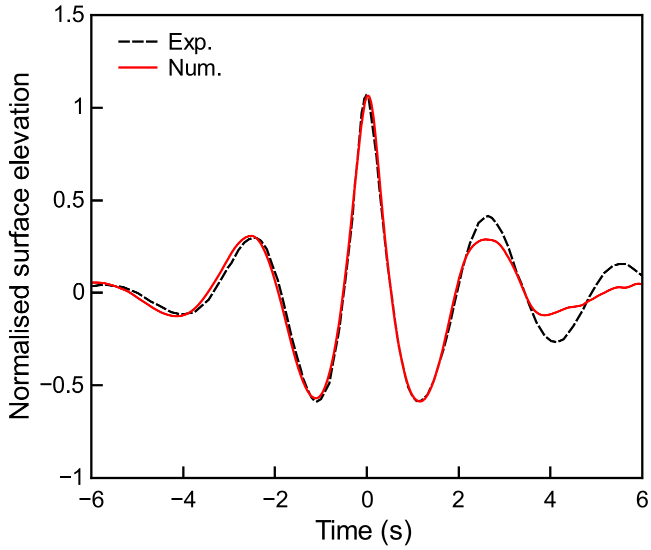

Fig. 12: Comparison between the numerical and experimental results for the free-surface elevation history at the focused locations (experiment: 18.51 m from the wave paddle; simulation: 5.6 m from the wave paddle). Experimental data are digitised from Hann et al. (2015).

extension respectively and $c = 2\zeta\sqrt{k_{eq}m}$, where ζ is the damping ratio and m is the mass of the buoy. The damping ratio has been determined numerically to be approximately $\zeta = 0.25$, although Lind et al. (2016) suggested using $\zeta = 0.175$ according to their SPH modelling on the same case. It will be seen in what follows that the simulated snatch loads are sensitive to the damping ratio.

Fig. 13 shows the comparison between experimental and numerical results for the mooring load, surge, heave and pitch motion of the buoy. Note that the mooring loads are normalised by the force required to reach the maximum length of the spring (9.4 N), and the surge and heave displacements of the buoy are normalised by the diameter of the buoy. Moreover, all of the numerical data series have been shifted in time so that the peak of the first snatch load occurs at $t = 0$ as is the case of the experimental data. It can be seen from Fig. 13(a) that the duration and occurring time of the snatch loads are well predicted by the PIC model. Furthermore, the PIC model well predicts the peak of the first snatch load but over-predicts the peak of the second snatch load by approximately 55%. In fact, as seen from Fig. 14, the second snatch load is more sensitive to the damping ratio ζ (see Eq. 21) than the first snatch load. The peak value of the second snatch load decreases and occurs earlier as the damping ratio increases. In addition, when the damp ratio is set to zero the second snatch load is larger than the first one, and when the damping ratio goes very high (e.g. 0.45) an unphysical third snatch load happens. These results are consistent with the findings of Lind et al. (2016), and similar results from their SPH simulation can also be seen in Fig. 13(a).

From Fig. 13(b) and (c) it can be seen that the current PIC model well reproduces the surge and the heave responses of the buoy. In particular, the double peaks occurring in phase with the snatch loads in the heave motion are also predicted very well. However, as can be seen from Fig. 13(d) there is a large discrepancy between the experimental and numerical results for the pitch motion after the occurrence of the snatch loads. While the numerical result follows the same trend as seen in the previous elastic-spring mooring case (see Fig. 8), the experimental data exhibit a relatively small pitch motion. The reason for this large discrepancy remains unclear at the time of writing. In general, the performance of the current PIC model is reasonably good in such a complex wave-structure interaction scenario involving extreme snatch mooring loads.

4. Conclusions

This paper extends the 3D parallel PIC model proposed in Chen et al. (2018) to simulate extreme wave interaction with floating bodies, using the Cartesian cut cell based two-way strong fluid-solid coupling algorithm proposed in Chen et al. (2016b). The PIC model solves the incompressible Navier-Stokes equations for free-surface flows. The novelty of this model lies in the fact that both Lagrangian particles and Eulerian grid are employed; the particles carry the fluid material information such as mass and momentum, and are used to solve the nonlinear advection term and track the free surface, while the grid is employed for computational convenience in solving all the non-advection terms. This makes the model both flexible on handling large free-surface deformations and efficient in terms of CPU cost. The two-way strong fluid-solid coupling algorithm features the fact that the velocity of the structure is represented by the fluid pressures in cells immediately surrounding the structure and any velocity integral along the structure surface due to the cut cell method can be integrated into the procedure of solving the PPE with a suitably amended coefficient matrix. This technique can resolve fluid interaction with floating bodies both stably and efficiently.

The present PIC model is validated against two existing physical experiments of focused wave interaction with a floating, hemispherical-bottomed, cylindrical buoy with either a linearly-elastic mooring or a more complex mooring configuration that encourages extreme snatch loads. Although both test cases involve extreme wave-structure interaction, the waves do not break and the structure has a smooth geometry that tends to cause less turbulences so that the lack of a turbulence model in the numerical simulations is acceptable. This is

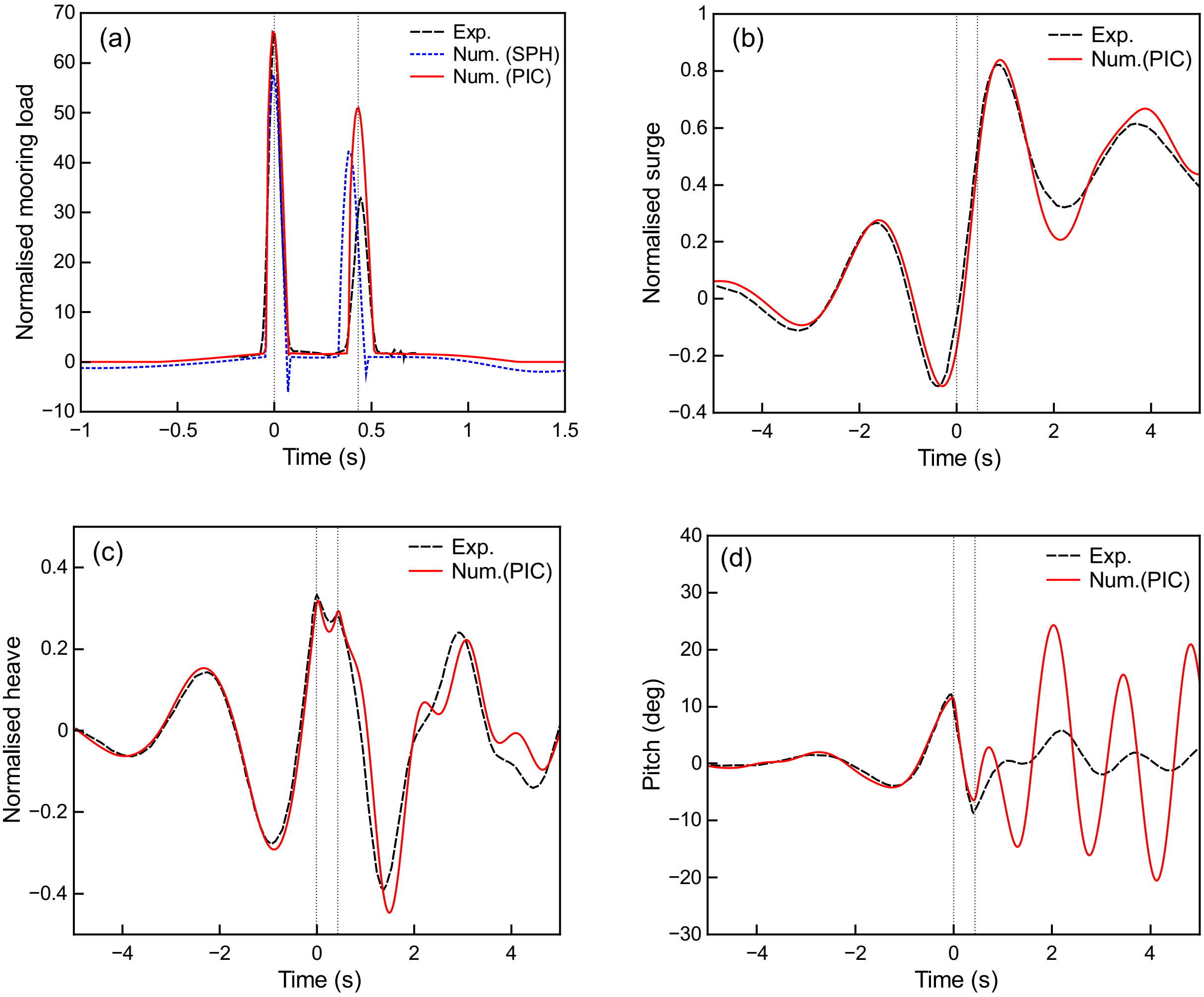

Fig. 13: Comparison between experimental and numerical results for (a) mooring load, (b) surge, (c) heave and (d) pitch motion of the buoy. The experimental results are digitised from Hann et al. (2015) and the SPH result is provided by Lind et al. (2016) (time shifted by -14.438 s).

confirmed at least in the first test case where both the laminar OpenFOAM® model and PIC model have achieved good results compared with the experiment. It is demonstrated through the comparisons with the experimental data that the PIC model can satisfactorily predict the motion of the moored buoy and the mooring force in such extreme wave-structure interaction scenarios. Also, as demonstrated in the first test case, the PIC model achieves a CPU efficiency of the same magnitude as that of the state-of-the-art OpenFOAM® model.

Nevertheless, it is seen that the memory storage requirement is demanding for the PIC model due to the double grid system. Also, the PIC model may predict inaccurate damping effects when the buoy motion is small, due likely to the limited grid resolution in 3D modelling. This situation may be improved by including in the solver a dynamic adaptive

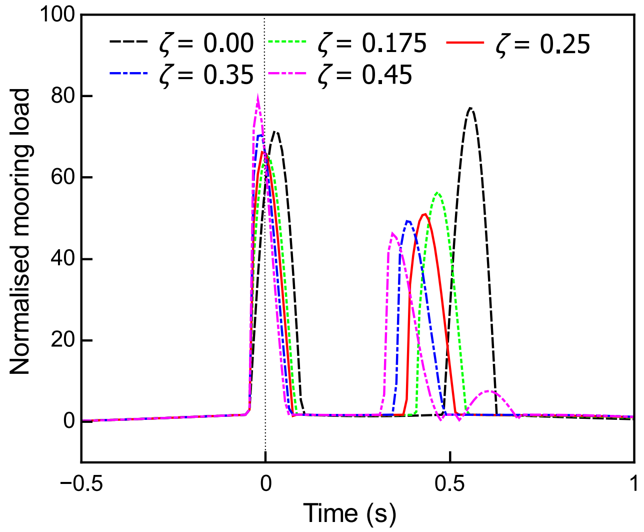

Fig. 14: Numerical results of the mooring load run with different damping ratios ζ.

grid combined with particle merging/splitting, such that the grid could be sufficiently fine around the structure while maintaining a feasible overall grid resolution.

Acknowledgements

This work is supported by the UK-China joint projects ResIn (EPSRC Grant No. EP/R007519/1) and UK-CIAPP (Grant No. UK-CIAPP/73), and the Open Funding of the State Key Laboratory of Coastal and Offshore Engineering at Dalian University of Technology (Grant No. LP1803). This work made use of the Balena High Performance Computing (HPC) Service at the University of Bath. This work is an extension of the PhD study of the first author, which was supported by the University of Bath and HR Wallingford.

References

Batty, C., Bertails, F., Bridson, R., 2007. A fast variational framework for accurate solid–fluid coupling. ACM Transactions on Graphics (TOG) 26 (3), 100.

Brackbill, J. U., Kothe, D. B., Ruppel, H. M., 1988. FLIP: A low-dissipation, Particle-In-Cell method for fluid flow. Computer Physics Communications 48 (1), 25–38.

Brackbill, J. U., Ruppel, H. M., 1986. FLIP: A method for adaptively zoned, Particle-In-Cell calculations of fluid flows in two dimensions. J. Comp. Phys. 65, 314–343.

Causon, D. M., Ingram, D. M., Mingham, C. G., 2001. A Cartesian cut cell method for shallow water flows with moving boundaries. Advances in Water Resources 24 (8), 899 – 911. URL http://www.sciencedirect.com/science/article/pii/S0309170801000100

Causon, D. M., Ingram, D. M., Mingham, C. G., Yang, G., Pearson, R. V., 2000. Calculation of shallow water flows using a Cartesian cut cell approach. Advances in Water Resources 23 (5), 545 – 562.
URL http://www.sciencedirect.com/science/article/pii/S0309170899000366

Chen, Q., 2017. Development of a full particle pic method for simulating nonlinear wave-structure interaction. Ph.D. thesis, University of Bath.

Chen, Q., Kelly, D. M., Dimakopoulos, A. S., Zang, J., 2016a. Validation of the PICIN solver for 2D coastal flows. Coastal Engineering 112, 87 – 98.
URL http://www.sciencedirect.com/science/article/pii/S0378383916300321

Chen, Q., Zang, J., Dimakopoulos, A. S., Kelly, D. M., Williams, C. J., 2016b. A Cartesian cut cell based two-way strong fluid-solid coupling algorithm for 2D floating bodies. Journal of Fluids and Structures 62, 252 – 271.
URL http://www.sciencedirect.com/science/article/pii/S0889974616000153

Chen, Q., Zang, J., Kelly, D. M., Dimakopoulos, A. S., 2017. A 3-D numerical study of solitary wave interaction with vertical cylinders using a parallelised particle-in-cell solver. Journal of Hydrodynamics, Ser. B 29 (5), 790 – 799.
URL http://www.sciencedirect.com/science/article/pii/S1001605816607904

Chen, Q., Zang, J., Kelly, D. M., Dimakopoulos, A. S., 2018. A 3D parallel particle-in-cell solver for wave interaction with vertical cylinders. Ocean Engineering 147, 165 – 180.

Chorin, A. J., 1968. Numerical solution of the Navier–Stokes equations. Math. Comput. 22, 745–762.

Edwards, E., Bridson, R., 2012. A high-order accurate Particle–In–Cell method. International Journal for Numerical Methods in Engineering 90 (9), 1073–1088.
URL http://dx.doi.org/10.1002/nme.3356

Gu, H., Stansby, P., Stallard, T., Moreno, E. C., 2018. Drag, added mass and radiation damping of oscillating vertical cylindrical bodies in heave and surge in still water. Journal of Fluids and Structures 82, 343 – 356.
URL http://www.sciencedirect.com/science/article/pii/S0889974617306552

Gunn, D. F., Rudman, M., Cohen, R. C. Z., 2018. Wave interaction with a tethered buoy: SPH simulation and experimental validation. Ocean Engineering 156, 306 – 317.
URL http://www.sciencedirect.com/science/article/pii/S0029801818302324

Hann, M., Greaves, D., Raby, A., 2015. Snatch loading of a single taut moored floating wave energy converter due to focussed wave groups. Ocean Engineering 96, 258 – 271.
URL http://www.sciencedirect.com/science/article/pii/S0029801814004235

Harlow, F. H., 1955. A machine calculation method for hydrodynamic problems. Technical Report LAMS-1956, Los Alamos Scientific Laboratory, Los Alamos.

Harlow, F. H., 1964. The Particle–In–Cell computing method for fluid dynamics. In: Alder, B. (Ed.), Methods in Computational Physics. Academic Press, New York, pp. 319–343.

Harlow, F. H., Welch, J. E., 1965. Numerical calculation of time–dependent viscous incompressible flow of fluid with free surface. Physics of Fluids 8, 2182–2189.

Jiang, C., Schroeder, C., Selle, A., Teran, J., Stomakhin, A., Jul. 2015. The Affine Particle-in-cell Method. ACM Trans. Graph. 34 (4), 51:1–51:10.
URL http://doi.acm.org/10.1145/2766996

Kamath, A., Chella, M. A., Bihs, H., Arntsen, Ø. A., 2016. Breaking wave interaction with a vertical cylinder and the effect of breaker location. Ocean Engineering 128, 105 – 115.
URL http://www.sciencedirect.com/science/article/pii/S0029801816304590

Kelly, D. M., 2012. Full particle PIC modelling of the surf and swash zones. In: Proc. 33rd Int. Conf. Coast. Eng. A.S.C.E., Santander, pp. 77–92.

Kelly, D. M., Chen, Q., Zang, J., 2015. PICIN: A Particle–In–Cell solver for incompressible free surface flows with two-way fluid-solid coupling. SIAM Journal on Scientific Computing 37 (3), B403–B424.
URL http://dx.doi.org/10.1137/140976911

Lind, S. J., Stansby, P. K., Rogers, B. D., 2016. Fixed and moored bodies in steep and breaking waves using SPH with the Froude–Krylov approximation. Journal of Ocean Engineering and Marine Energy 2 (3), 331–354.
URL http://dx.doi.org/10.1007/s40722-016-0056-4

Maljaars, J. M., Labeur, R. J., Möller, M., 2018. A hybridized discontinuous galerkin framework for high-order particlemesh operator splitting of the incompressible navierstokes equations. Journal of Computational Physics 358, 150 – 172.
URL http://www.sciencedirect.com/science/article/pii/S0021999117309300

Nematbakhsh, A., Olinger, D. J., Tryggvason, G., 2013. A nonlinear computational model of floating wind turbines. Journal of Fluids Engineering 135 (12), 121103.

Ng, Y. T., Min, C., Gibou, F., 2009. An efficient fluid–solid coupling algorithm for single–phase flows. Journal of Computational Physics 228 (23), 8807–8829.

Nishiguchi, A., Yabe, T., 1982. Finite-sized fluid particle in a nonuniform moving grid. Journal of Computational Physics 47 (2), 297 – 302.
URL http://www.sciencedirect.com/science/article/pii/002199918290081X

Nishiguchi, A., Yabe, T., 1983. Second-order fluid particle scheme. Journal of Computational Physics 52 (2), 390 – 413.
URL http://www.sciencedirect.com/science/article/pii/0021999183900372

Omidvar, P., Stansby, P. K., Rogers, B. D., 2013. SPH for 3D floating bodies using variable mass particle distribution. International Journal for Numerical Methods in Fluids 72 (4), 427–452.

Press, W., Flannery, B., Teukolsky, S., Vetterling, W., 1992. Numerical Recipes: The Art of Scientific Computing (second edition). Cambridge Univ. Press, New York.

Qian, L., Causon, D. M., Mingham, C. G., Ingram, D. M., 2006. A free-surface capturing method for two fluid flows with moving bodies. Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences 462 (2065), 21–42.
URL http://rspa.royalsocietypublishing.org/content/462/2065/21

Ralston, A., 1962. Runge–Kutta methods with minimum error bound. Mathematics of Computation 16:80, 431–437.

Ransley, E. J., Greaves, D., Raby, A., Simmonds, D., Hann, M., 2017a. Survivability of wave energy converters using CFD. Renewable Energy 109, 235 – 247.
URL http://www.sciencedirect.com/science/article/pii/S0960148117301799

Ransley, E. J., Greaves, D. M., Raby, A., Simmonds, D., Jakobsen, M. M., Kramer, M., 2017b. RANS-VOF modelling of the Wavestar point absorber. Renewable Energy 109, 49 – 65.

URL http://www.sciencedirect.com/science/article/pii/S0960148117301659

Wang, W., Kelly, D. M., 2018. A high-order PIC method for advection-dominated flow with application to shallow water waves. International Journal for Numerical Methods in Fluids 87 (11), 583–600.

Yang, G., Causon, D. M., Ingram, D. M., Saunders, R., Battent, P., 1997. A cartesian cut cell method for compressible flows Part A: static body problems. The Aeronautical Journal (1968) 101 (1002), 4756.

Zhu, Y., Bridson, R., 2005. Animating sand as a fluid. ACM Trans. Graph. 24 (3), 965–972. URL http://doi.acm.org/10.1145/1073204.1073298