# Improved numerical wave generation for modelling ocean and coastal engineering problems

Pedro J. Martínez-Ferrer[a,*], Ling Qian[a], Zhihua Ma[a], Derek M. Causon[a], Clive G. Mingham[a]

[a] *Centre for Mathematical Modelling and Flow Analysis,*
*Manchester Metropolitan University,*
*Chester Street, Manchester M1 5GD, United Kingdom*

**Abstract**

We introduce a dynamic-boundary numerical wave generation procedure developed for wave structure interaction (WSI) simulations typical of ocean and coastal engineering problems. This implementation relies on a dynamic mesh which deforms in order to replicate the motion of the wave-maker, and it is integrated in wsiFoam: a multi-region coupling strategy applied to two-phase Navier-Stokes solvers developed in our previous work [Martínez Ferrer et al. A multi-region coupling scheme for compressible and incompressible flow solvers for two-phase flow in a numerical wave tank. Computer & Fluids 125 (2016) 116–129]. The combination of the dynamic-boundary method with a multi-region mesh counteracts the increase in computational cost, which is intrinsic to simulations featuring dynamic domains. This approach results in a high performance computing wave generation strategy that can be utilised in a numerical wave tank to carry out accurate and efficient simulations of wave generation, propagation and interaction with fixed structures and floating bodies.

We conduct a series of benchmarks to verify the implementation of this wave generation method and the capabilities of the solver wsiFoam to deal with wave structure interaction problems. These benchmarks include regular and focused waves, wave interaction with a floating body and the modelling of a wave energy converter, using different wave-maker geometries: piston, flap and plunger. The results gathered in this work agree well with experimental data measured in the laboratory and other numerical simulations.

*Keywords:* wave generation, wave structure interaction, floating bodies, coupling

## 1. Introduction

Numerical wave tanks (NWTs) constitute an essential tool for design and analysis in ocean and coastal engineering problems. NWTs must be validated against experiments conducted in the laboratory, from wave

*Corresponding author
*Email address:* pedro.martinez-ferrer@bsc.es (Pedro J. Martínez-Ferrer)

generation to the evaluation of wave impacts on fixed or floating objects, offshore structures, performance and survivability of wave energy converters, etc. The main objectives of a NWT are to complement experiments, e.g. retrieving useful data which otherwise would be difficult to measure experimentally, to simulate full scale geometries in open and real sea state conditions, or even to explore and assess new designs of coastal defence systems, offshore platforms and marine vessels. Computational Fluid Dynamics (CFD) has been extensively used in NWTs with a large variety of simplified and detailed models depending on the degree of physics required or the computational resources available. In general, fully non-linear potential flow models [1, 2] have been widely adopted due to the simplicity of their equations and the good accuracy achieved on wave propagation. With the increasing power and efficiency of computational resources and the development of high performance computing (HPC), Navier-Stokes models in NWTs are experiencing a growing demand [3, 4] because they allow for a detailed analysis of the flow physics accounting for vorticity, viscosity and air entrainment/entrapment effects, at the expense of higher computational costs. Moreover, special attention has been given recently to compressibility effects in water-air mixtures characteristic of violent wave impacts [5, 6, 7], which are simulated with expensive numerical methods such as compressible smooth particle hydrodynamics (SPH) [8, 9] and the volume of fluid (VOF) method [10, 11].

Carrying out very detailed simulations, e.g. based on the resolution of the Navier-Stokes equations, in the entire computational domain, which can feature an extension of several hundreds of metres in typical ocean engineering problems, remains impractical today even with the current state of the art in HPC, e.g. parallel heterogeneous computing (CPU+GPU). Therefore, coupled simulations in which specialised numerical solvers work in different regions of the NWT become a good strategy to overcome the current challenges in the numerical modelling of ocean and coastal engineering problems [12]. Such a *multi-region* wave tank may be principally composed of: (i) a relatively quick fully non-linear potential (FNLP) solver for wave propagation in large extensions of the mesh, (ii) incompressible and compressible Navier-Stokes solvers to study rotational, viscous and complex flows, with associated effects of compressibility in some cases, and (iii) a computational structural dynamics (CSD) solver for the WSI on rigid and deforming bodies. There have been some efforts in this direction and, more specifically, in the coupling between irrotational and viscous flows, see for instance references [13, 14, 15]. However, the diversity in the coupling solutions and the increasing complexity in numerical modelling have prevented these coupled strategies from becoming popular, specially in HPC where the domain decomposition needs to be properly handled between different regions [16]. A recent work on multi-region coupling for incompressible and compressible two-phase flow solvers in a numerical wave tank has been proposed in [12], where the coupling between regions is treated as another boundary condition to simplify the numerical modelling and facilitate programming on HPC architectures.

One of the essential components of a wave tank is the realistic generation of waves. In the laboratory, the *physical modelling* aims to replicate waves found in nature by using different wave generating mechanisms [17]. Thus, wave generators are generally classified in three main categories: (i) pistons, for the physical modelling of shallow water waves, (ii) flaps and (iii) plungers utilised for deeper water waves. The relation between the wave generator motions and the dynamics of the generated waves have been studied, both theoretically and experimentally, using first-order linearised hydrodynamic equations for pistons and flaps [18, 19] as well as plungers [20], see also [21]. More recently, linear theory was revisited with a fully second-order wave-maker theory to correctly reproduce in the laboratory the lower and higher harmonic wave components found in irregular sea states [22]. On the other hand, the *numerical modelling* of waves in NWTs has been traditionally limited to static-boundary wave generation, where the velocity and free surface wave profile are specified at the boundaries, e.g. Dirichlet or Neumann boundary conditions, based on different wave theories for regular and irregular types of wave [23]. Static-boundary wave generation methods have been applied to both potential [24, 25] and Navier-Stokes flows, see for instance [26, 27]. These methods are relatively easy to implement and offer an attractive computational cost as they do not require moving mesh components for numerical wave generation. On the other hand, wave generation methods based on dynamic-boundaries can replicate the exact motion of the paddles used in the experiments and, consequently, close the gap between the physical modelling in the laboratory and the numerical modelling in NWTs. Dynamic-boundary methods are more popular among potential flows solved with the boundary element method (BEM) [2, 28] but they are not commonly found in Navier-Stokes Eulerian flow solvers [29], as the introduction of a deforming mesh to accommodate the motion of the paddles contributes to a significant increase in the computational cost, which is already elevated in Navier-Stokes solvers. Another recent example can be found in the literature [30], where a piston-type wave-maker was implemented within the VOF method. However, it was found that the compelling increase in terms of computational cost compared to static-boundary methods questions the suitability of such methods for large scale applications.

The aforementioned low efficiency problem related to dynamic-boundary methods needs to be carefully addressed through further development, as this approach proves to be necessary in cases dealing with confined geometries susceptible of wave reflection [29], or in cases where static-boundary methods cannot provide accurate predictions [31]. The aim of this paper is therefore to develop an efficient and versatile dynamic-boundary numerical wave generation method in a multi-region NWT in order to replicate the most common wave-makers found in laboratories, including pistons, flaps as well as plungers. Special emphasis is given in the comparison of this approach against other well established static-boundary numerical wave generation methods not only in terms of accuracy, but also and more specifically in terms of computational efficiency. The rest of the paper is organised as follows: Section 2 describes the two-phase incompressible Navier-Stokes solver used to carry out our numerical investigations, the dynamic-boundary wave-maker implementation and its integration in a multi-region computational domain. Results and discussions are presented in Section 3 and Section 4 is dedicated to conclusions and future work.

## 2. Numerical procedures

We utilise the open-source CFD library OpenFOAM [32] to carry out the simulations presented in this work. OpenFOAM numerical solvers rely on a cell-centered, co-located finite-volume method. This library is widely employed in research and industry and it offers the possibility to read, improve and modify the available code for free.

In this paper we present a new dynamic mesh algorithm which mimics a physical wave-maker such as those employed in experimental wave tanks. We apply this algorithm in conjunction with a novel multi-region coupling strategy presented in [12], namely "wsiFoam", in order to increase the efficiency of the method. A description of the solvers, the dynamic mesh and the coupling strategy is detailed below.

### *2.1. The numerical solver*

The simulations performed in this work are conducted with a slightly modified version of "interFoam", which is an incompressible two-phase pressure-based solver [33] that has already been successfully applied in a wide variety of naval and ocean engineering applications, see for instance [4, 34]. It is based on the VOF method to describe an incompressible two-phase flow mixture, i.e. air and water, wherein each phase is assumed to be homogeneous and in mechanical equilibrium: identical velocity and pressure. Furthermore, this solver makes special emphasis on maintaining a sharp water free surface (interface-capturing) by using artificial compression terms.

The mass balance equation for the incompressible ($\boldsymbol{\nabla} \cdot \mathbf{U} = 0$) two-phase flow mixture can be reduced to the mass balance equation for the water volume fraction $\alpha \in [0, 1]$:

$$\frac{\partial \alpha}{\partial t} + \boldsymbol{\nabla} \cdot \mathbf{U}\alpha + \boldsymbol{\nabla} \cdot \mathbf{U}_c \alpha (1 - \alpha) = 0, \tag{1}$$

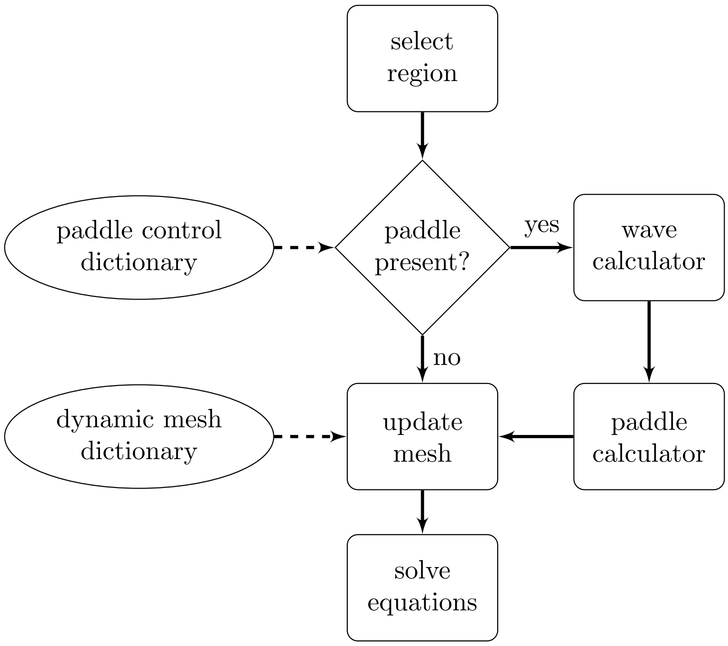


Figure 1: Wave-maker implementation flow chart.

where $\mathbf{U}$ is the mixture velocity vector and $\mathbf{U}_{\mathrm{c}} = \min[\mathbf{U}, \max(\mathbf{U})]$. Herein the density of the mixture is $\rho = \alpha\rho_{\mathrm{w}} + (1-\alpha)\rho_{\mathrm{a}}$ with constant partial densities $\rho_{\mathrm{w}} = 1000\,\mathrm{kg/m^3}$ and $\rho_{\mathrm{a}} = 1.1586\,\mathrm{kg/m^3}$. The third term in eq. (1) is an artificial compression quantity that sharpens the interface and guarantees bounded values of $\alpha$ by using the MULES procedure [33, 35].

The single momentum equation for the homogeneous mixture is given by

$$\frac{\partial \rho \mathbf{U}}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \mathbf{U}\mathbf{U}) - \boldsymbol{\nabla} \cdot (\mu \boldsymbol{\nabla} \mathbf{U}) = \sigma\kappa\boldsymbol{\nabla}\alpha - \mathbf{g} \cdot \mathbf{x} \boldsymbol{\nabla}\rho - \boldsymbol{\nabla} p_{\mathrm{d}}, \tag{2}$$

where denotes the surface tension coefficient and $\kappa = \boldsymbol{\nabla} \cdot (\boldsymbol{\nabla}\alpha / |\boldsymbol{\nabla}\alpha|)$ represents the curvature of the interface. The mixture viscosity is given by $\mu = \alpha\mu_{\mathrm{w}} + (1-\alpha)\mu_{\mathrm{a}}$. The dynamic pressure is calculated as $p_{\mathrm{d}} = p - \rho\mathbf{g} \cdot \mathbf{x}$ with $\mathbf{g}$ and $\mathbf{x}$ the gravity and position vectors, respectively.

The governing equations (1)–(2) are linearized and integrated over each control volume to determine $\alpha$ and $\mathbf{U}$, respectively, and a pressure corrector linearized equation is solved for $p_{\mathrm{d}}$. This solution procedure relies on the segregated projection algorithm PIMPLE [36], derived from the PISO procedure [37, 38], which allows for equation under-relaxation to guarantee convergence of the solutions at each time step.

### 2.2. Wave-maker implementation

We utilise the dynamic mesh handling library provided by OpenFOAM [39] to reproduce the wave-maker motion. This library allows for automatic mesh motion to accommodate prescribed boundary deformation by changing the positions of mesh points [40]. Examples of dynamic mesh applications with this open-source library includes six-degree-of-freedom (6-DOF) floating bodies [41], fluid-structure interaction (FSI) simulations [42] and turbomachinery applications [43] to mention but a few. The wave-maker implementation flow chart is shown in Fig. 1. A new paddle class is implemented on top of OpenFOAM dynamic mesh class and it is only called if a paddle is present in the corresponding region of the computational domain. The approach retained in Fig. 1 allows us to simulate a moving paddle (controlled by the paddle control dictionary) together with an additional moving object such as a 6-DOF floating body (controlled by OpenFOAM dynamic mesh dictionary), within the same region. This concept can be further generalised when multiple regions are present in the computational domain (see section 2.3).

The wave-maker class is divided in two main parts. Firstly, the wave calculator allows to generate regular and irregular waves but the user can also specify a prescribed type of wave given as a time series. Secondly, the paddle calculator generates *theoretically* the appropriate paddle motion depending on the type of geometry of the wave-maker, e.g. piston, flap or plunger, via a transfer function. Following the first-order wave-maker theory [21], the transfer function defined as the wave to paddle stroke ratio, $T = H/S$, of a flap-type wave-maker is given by

$$T = 4\sinh(kh)\frac{(kh-kd)\sinh(kh)-\cosh(kh)+1}{(kh-kd)(\sinh(2kh)+2kh)}, \tag{3}$$

where $k$ is the wave number, $h$ the water depth and $d$ the hinged distance from the sea bed. The transfer function for a piston-type wave-maker, $d \to -\infty$ in Eq. (3), is $T = (2\cosh(2kh)-2)/(\sinh(2kh)+2kh)$. Finally, for a plunger-type wave-maker featuring a wedge angle $\theta$ we follow reference [20] and solve iteratively, i.e. using the Newton-Raphson method, the stroke value $S$ of a *simplified* plunger transfer function

$$T = \frac{2\tan\theta\sinh(kh)}{\sinh(kh)\cosh(kh)+kh}\left(\sinh(kh)+\frac{\cosh(kh-kh_2)-\cosh(kh-kh_1)}{kS}\right), \tag{4}$$

where $S = h_2 - h_1$ with $h_2 = h + S/2$ and $h_1 = h_2 - S$. Additionally, the developed wave-maker library offers the possibility of specifying the *experimental* transfer function value of a given paddle or to replicate the wave-maker displacement signal.

The new positions of the mesh boundary points defining the paddle are updated every time step based

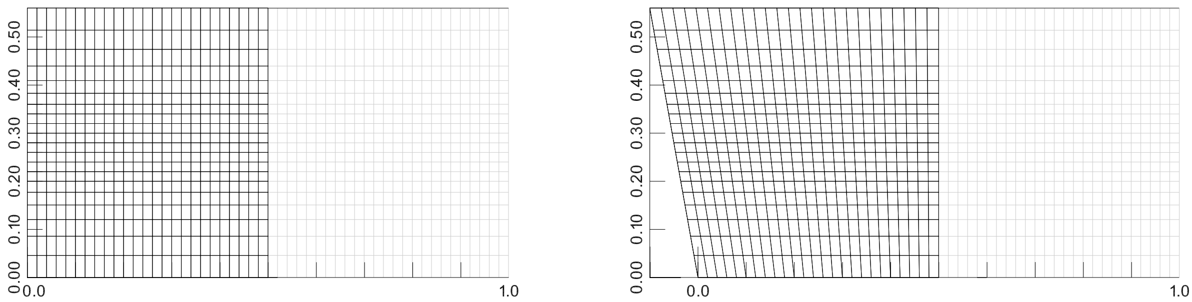


Figure 2: Deformation of the computational mesh for $x < 0.5$ and a flap-type wave-maker: original position (left) and left-most position (right).

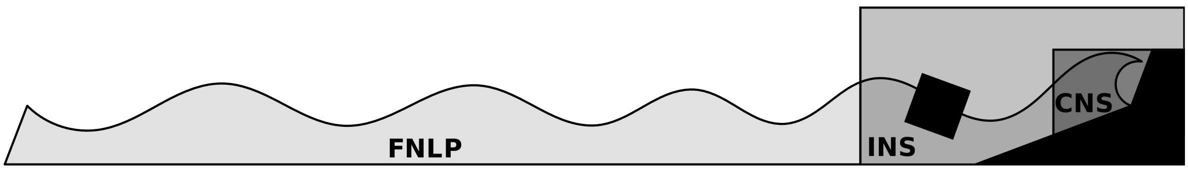


Figure 3: Multi-region virtual wave structure interaction (WSI) simulation environment featuring fully non-linear potential (FNLP), incompressible and compressible Navier-Stokes (INS and CNS, respectively) solvers.

on the initial, undeformed mesh. We allow for a small region of the inner domain to deform smoothly, i.e. without changing the topology of the mesh, in the horizontal direction following $\mathbf{x}^* = \mathbf{x}(l_e - x_0)/(l_e - l_s)$ where $l_s$ and $l_e$ indicate the starting and ending horizontal distances, respectively, and $x_0$ is the initial horizontal coordinate of the mesh points. Fig. 2 illustrates the deformation of the mesh induced by a flap-type wave-maker where only the points with $x < 0.5$ are allowed to move while the rest of the mesh remains unchanged. The new mesh positions calculated from the paddle movement are added in cache before the OpenFOAM's update mesh class is executed, see Fig. 1. This class can specify an additional mesh motion, e.g. a deforming structure or a 6-DOF floating body, and hence the motion from the dynamic mesh dictionary is added up to the stored paddle motion before the mesh gets updated.

### 2.3. Multi-region coupling

The wave-maker introduced in the previous section has been implemented on top of a multi-region solution procedure for NWTs described in [12]. The aim of the multi-region coupling is to build a *general purpose* numerical wave tank to model a wide variety of ocean and coastal engineering problems using several specialised numerical methods. Fig. 3 shows a schematic view of a "multi-region virtual wave structure interaction (WSI) simulation environment", gathering fully non-linear potential (FNLP), incompressible Navier-Stokes (INS) and compressible Navier-Stokes (CNS) solvers, which exchange information through fixed coupling interfaces separating the regions where they are defined. This results in high performance computing coupled simulations in which the coupled system is generally superior to either solver alone.

The wave-maker library is integrated in wsiFoam, where different solvers are coupled together to achieve the aforementioned WSI environment. The multi-region solution procedure of wsiFoam is detailed in Algorithm 1 [12]. When the simulation begins, the minimum time step among the regions is used to advance the solution. For each region present in the simulation, the equations are discretised and the boundary conditions applied: the coupling interfaces between regions act indeed as boundary conditions. Before solving the system of equations, the mesh is updated and thus the motion of the paddle, and another moving object that may be defined in the computational domain, is taken into account. Once the system of equations is solved in the new mesh, the simulation time is updated, and the entire process is repeated again until the end of the simulation.

**Algorithm 1:** wsiFoam multi-region solution procedure.

```
begin time advancement
    calculate the time step for each region;
    find the global (minimum) time step;
    for each region do
        discretise the system of equations;
        apply boundary and coupling conditions;
        update the dynamic mesh;
        solve the system;
    end
    update the simulation time;
end
```

The framework described in Algorithm 1 offers a lot of possibilities and flexibility for wave tank simulations, e.g. multiple wave-makers can be defined in separate regions of the computational domain, each one featuring its own independent motion. Furthermore, the paddle can interact with different solvers ranging from non-linear potential to fully compressible Navier-Stokes depending on the particular application and the degree of physics required. Equally compelling is to use this multi-region strategy in order to speed up the simulations and thus increase the efficiency of the standard, i.e. single-region, approach. For example, the use of a paddle acting at one of the boundaries of the computational domain increases the CPU time of the simulation as the entire mesh must be declared as a *dynamic mesh*: every time step cell volumes and face areas must be recalculated and relative fluxes must be corrected. However, in a multi-region approach the wave-maker can be defined in one small dynamic region of the computational domain whereas the rest of the simulation can be carried out in a *static mesh*. For this reason, this moving-boundary strategy to generate waves becomes an interesting alternative to other static-boundary wave generation methods based on the calculation of the wave velocity profile [26, 27]. Furthermore, it has been verified in [12] that wsiFoam is on a par with OpenFOAM native solvers and the coupling between regions is completely transparent to the parallelisation algorithms (based on MPI domain decomposition), which makes this utility suitable for high performance computing of ocean and coastal engineering applications.

For the sake of simplicity, we only present in this work the numerical simulations carried out with the incompressible Navier-Stokes solver, one single paddle and a maximum of three regions within the same computational domain.

## 3. Results and discussions

In the following we use the same finite-volume discretisation schemes: linear interpolation of quantities from cell centres to face centres, second-order schemes applied to the spatial derivatives and temporal derivatives discretised with the first-order implicit Euler scheme.

It is worth mentioning that no grid convergence studies are included in this work and the solutions presented below correspond to converged cases. These solutions were obtained by gradually refining the cell size until they converged by comparison to previous experimental data and simulations. As a rule of thumb, wsiFoam was able to propagate moderate waves accurately, with convergence of the solutions, for cell sizes $\Delta \lesssim 1\,\mathrm{cm}$. The time step was calculated based on a Courant number equal to 0.5 to guarantee stable and relatively fast computations.

Based on previous simulations, the position of the coupling interface does not have a measurable impact on the solutions, e.g. when comparing the obtained results in graphs against data provided by single-region simulations or experiments. In theory, the coupling interface allows to separate two regions composed of only one cell layer thickness (although such an extreme case has not been tested). In practice, the size of each region, and hence the position of the interface between them, depends on:

- the characteristic length of the phenomenon being studied. For instance, we recommend reading our pseudocavitation test case (Section 3.4 of reference [12]). In that particular case, different solvers are utilised in the incompressible and compressible regions separated by the coupling interface;
- and/or the maximum allowed cell distortion if one of the regions features a dynamic mesh. In Section 3.3, the movement of the floating body stretches cells significantly and therefore an inner domain of about two times the size of the floating body must be defined to handle the amount of deformation near the coupling interface.

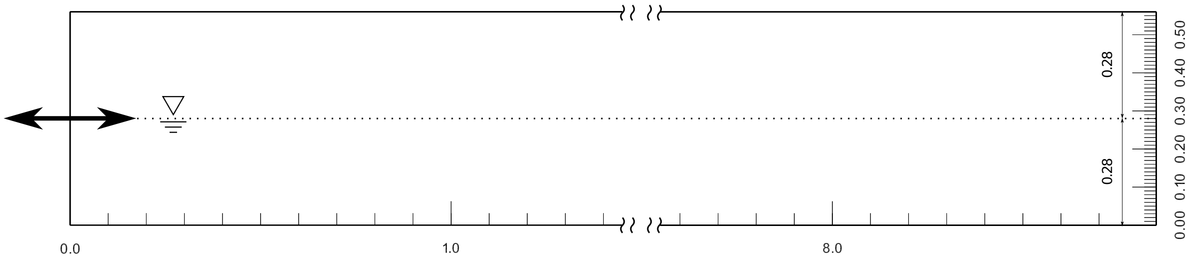


Figure 4: Regular waves generated by a piston-type wave-maker: computational domain. SI units.

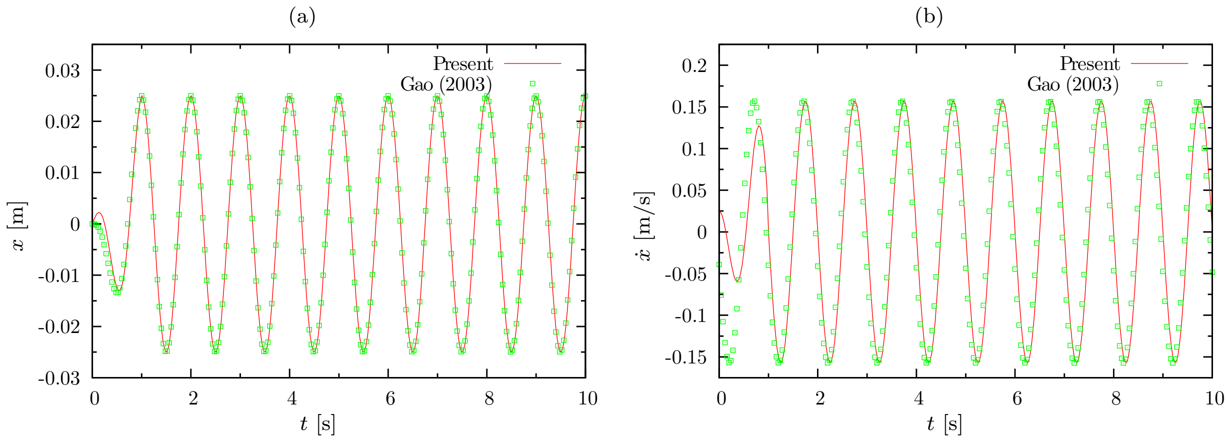


Figure 5: Regular waves generated by a piston-type wave-maker: (a) piston displacement and (b) piston velocity.

### *3.1. Regular waves generated by a piston-type wave-maker*

This experiment was originally proposed in reference [44] to validate numerical models for the generation of waves in shallow water flow regimes, see for instance [45, 46]. In this test, regular waves are generated by a piston-type moving paddle installed in a 8.85 m long flume tank with a still water depth of 0.28 m. Three wave gauges are used to record the free surface elevation at distances 0.55 m, 3.55 m and 5.45 m from the initial position of the paddle.

The 2D computational domain of dimensions $8.85 \times 0.56\,\mathrm{m}^2$ is depicted in Fig. 4. The mesh is uniformly divided into 885 cells ($\Delta x = 1\,\mathrm{cm}$) following the horizontal direction. In the vertical direction, the cell size is kept constant ($\Delta y = 1\,\mathrm{cm}$) within a 16 cm-wide region centred at the initial position of the free surface. A constant vertical stretching is applied to the cells outside that region in order to save computational time. All the walls of the numerical wave tank share a non-slip boundary condition including the moving paddle boundary: the velocity transmitted to the fluids, i.e. water and air, corresponds exactly to the velocity of the paddle. The top boundary remains open to the atmosphere ($p = 1\,\mathrm{bar}$). Finally, the simulation is run up to 12 s with a Courant number set to 0.5.

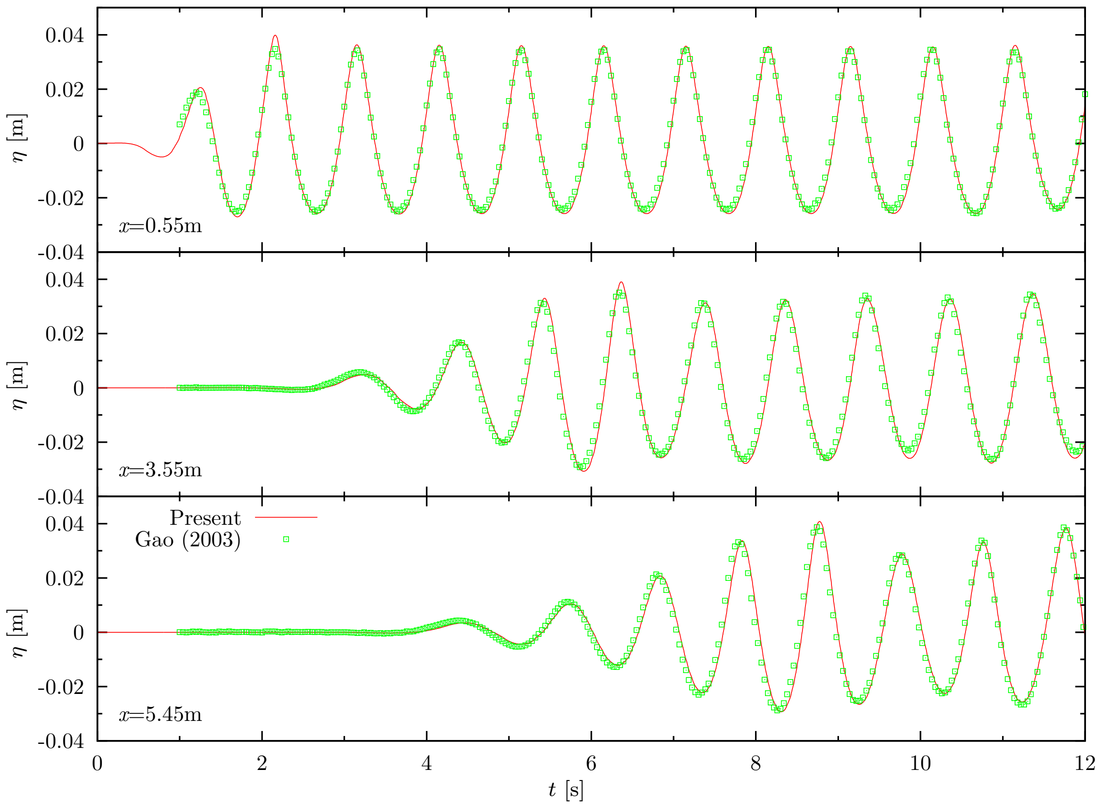


Figure 6: Regular waves generated by a piston-type wave-maker: water elevation measured as a function of the time at locations 0.55 m, 3.55 m and 5.45 m from the initial position of the wave-maker.

Before presenting our numerical results, it is worth mentioning that the target wave is not known in this experiment, but the piston displacement and velocity recorded in the laboratory [44]. For this reason, we have not attempted to simulate this benchmark using a static-boundary wave generation method based on a particular wave theory. Figure 5(a) shows the displacement recorded in the laboratory as well as the prescribed motion used in the simulation, which is given by the analytical expression $x = 0.025\cos(2\pi t)$ and a linear ramp of one second of duration. The piston displacement is reproduced satisfactorily with the analytical expression whereas the velocity, see Fig. 5(b), is underestimated during the ramp time.

Fig. 6 shows the water elevation measured as a function of the time at three different locations in the numerical wave tank. Please note that the experimental data is only available after the first second. In general, the agreement between experiments and numerical solutions for the three gauges is very good in terms of wave height and phase. Moreover, the accuracy of the present solutions is superior in comparison to previous investigations [45, 46] using a similar cell size (not shown here for the sake of conciseness). Therefore, it can be concluded that the wave-maker developed in this work is able to accurately simulate regular waves in an empty tank using a piston-type paddle with a prescribed motion.

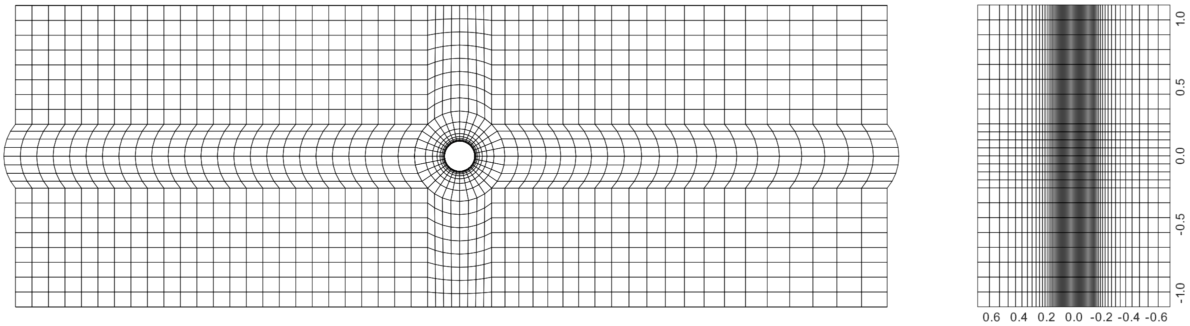


Figure 7: Focused wave generated by a piston-type wave-maker: top view of the computational mesh near the fixed cylinder (left) and mesh refinement around the initial position of the free surface (right). SI units.

Table 1: Focused wave generated by a piston-type wave-maker: wave properties.

| Property | $N$ [–] | $f_1$ [s$^{-1}$] | $f_P$ [s$^{-1}$] | $f_N$ [s$^{-1}$] | $t_{\mathrm{P}}$ [s] |
|---|---|---|---|---|---|
| Value | 32 | 0.34 | 0.68 | 1.02 | 39 |

### *3.2. Focused wave generated by a piston-type wave-maker*

The aim of this benchmark is to simulate irregular waves in a three-dimensional wave tank. The corresponding experiment was initially proposed in [47] to study the interaction between a focused wave and a fixed cylinder, including pressure distribution and vortex shedding around the structure. We already conducted a previous work using a three-dimensional mesh with 32M cells to assess the wave structure interaction [31]. In that work non-negligible discrepancies can be found between experiments and numerical solutions when static-boundary wave generation methods are employed. Therefore, we will focus here on the wave generation by comparing different tools available in OpenFOAM and widely used in the scientific community.

Our simplified three-dimensional computational domain, 50 m long, 2.2 m wide and 1.4 m high, is discretised with $291 \times 24 \times 72$ cells, respectively, and has a water depth of 0.7 m. The mesh is refined near the cylinder of 0.22 m diameter, which is installed in the centre of the tank ($x = 25$ m). Another refinement layer of cells is applied around the initial position of the water free surface ($z = 0$ m), as shown in Fig. 7. Furthermore, a constant cell stretching is applied between the cylinder and the end of the computational domain in order to save CPU time. The walls of the tank and the cylinder are defined with the non-slip condition, the top boundary is open to the atmosphere (1 bar) and the time step is updated every iteration based on a Courant number of 0.5. Additionally, when the motion of the paddle is prescribed to that of the experiments, the time step is fixed to 0.01 s, which is the sampling frequency of the paddle displacement signal measured in the laboratory.

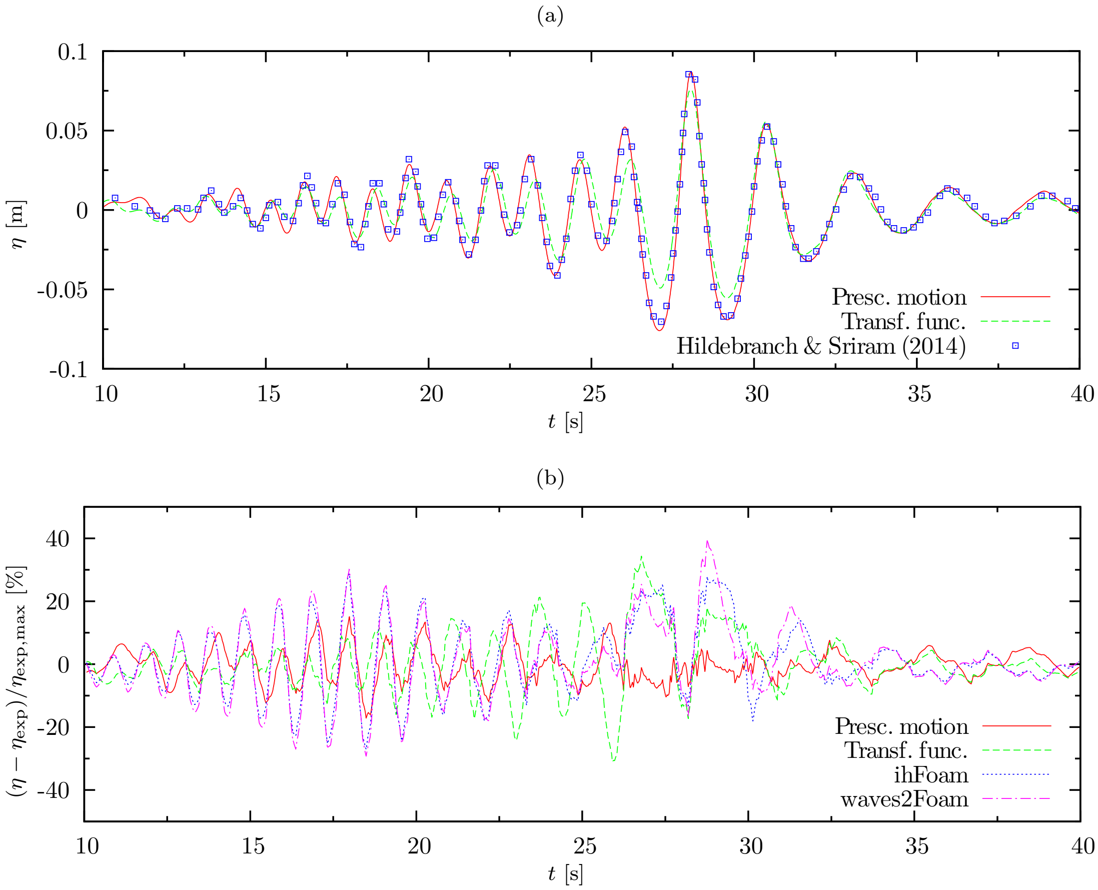


Figure 8: Focused wave generated by a piston-type wave-maker: time history of the (a) water elevation at 4.98 m from the initial position of the wave-maker and (b) normalised differences between water elevation for various wave generation methods.

A piston-type wave-maker generates a focused wave, whose characteristics are gathered in Table 1. $N$ is the number of wave frequency components, $f_1$, $f_P$ and $f_N$ are the starting, peak and last frequencies of the wave spectrum, respectively. Finally, $t_{\mathrm{P}}$ is the focusing time and the focus coordinate corresponds to the centre of the cylinder. We calculate the characteristics of the target wave, i.e. heights, periods, phases and wave numbers, with a preprocessing utility and use this data with two static-boundary numerical wave generation tools, namely waves2Foam [26] and ihFoam [27], altogether with our present dynamic-boundary method using both a theoretical transfer function and the paddle displacement signal from experiments.

Fig. 8(a) displays the water elevation measured at 4.98 m from the initial position of the wave-maker. The experimental data is compared against our present results using the laboratory paddle displacement signal and the theoretical transfer function for a piston-type paddle. It can be readily seen that the maximum water elevation is clearly underestimated when the transfer function is employed and wave crests and troughs adjacent to the focusing point become poorly predicted. On the other hand, when the paddle motion

Table 2: Focused wave generated by a piston-type wave-maker: simulation times normalised by $t_{\mathrm{ref}} = 5218\,\mathrm{s}$.

| ihFoam | waves2Foam | wsiFoam (I) | wsiFoam (I-I) |
|---|---|---|---|
| 1.00 | 1.11 | 2.00 | 1.08 |

matches that of the experiments, the numerical prediction shows good agreement with the laboratory data, specially near the focusing point. This is in agreement with previous numerical results obtained with the same prescribed motion to generate the focused wave, see reference [31]. Fig. 8(b) shows the normalised differences between water elevation for various wave generation methods. The prescribed motion method gives the best agreement with experiments. The maximum error is found before the formation of the focused wave and remains below 15%. As expected, ihFoam and waves2Foam static-boundary wave generation methods give very similar results and small differences may be attributed to the fact that the latter uses an additional relaxation zone to impose the target wave. Both methods underestimate the peak elevation but the largest discrepancies are observed in the adjacent wave troughs, where the error reaches almost 40%. Moreover, non-negligible errors can be clearly spotted between 13 s and 20 s. These differences confirm the limitation of first-order static-boundary methods and theoretical transfer functions to predict accurately certain experiments conducted in the laboratory, such as the one studied here. We conducted additional second-order simulations following reference [48], which lead to similar results (not shown here for the sake of conciseness).

Another parameter of interest apart from the accuracy of our numerical solutions is their efficiency as the use of a dynamic mesh to accommodate the motion of the wave-maker will increase the CPU time due to additional operations such as flux recalculations, determination of new point positions, etc. Table 2 shows the simulation times for different wave generation methods, normalised by a reference time corresponding to the fastest one. All the simulations were carried out on the Neumann cluster installed in Manchester Metropolitan University, using one single node featuring 16 Intel Xeon E5-2620 cores (approximately 30000 cells per core). ihFoam is the fastest method while waves2Foam is 11% slower due to the use of relaxation zones. However, wsiFoam utilises a dynamic mesh and is two times slower than ihFoam in a single incompressible region configuration (denoted as I), which may compromise the use of this technology in time-constrained situations. In order to increase the efficiency of the simulation, a multi-region mesh (denoted as I-I) is defined: the dynamic mesh includes the wave-maker boundary and extends 2 m inside the inner domain following the horizontal dimension whereas the rest of the computational domain remains static. In this scenario, wsiFoam is only 8% slower than ihFoam and outperforms waves2Foam, which ren-

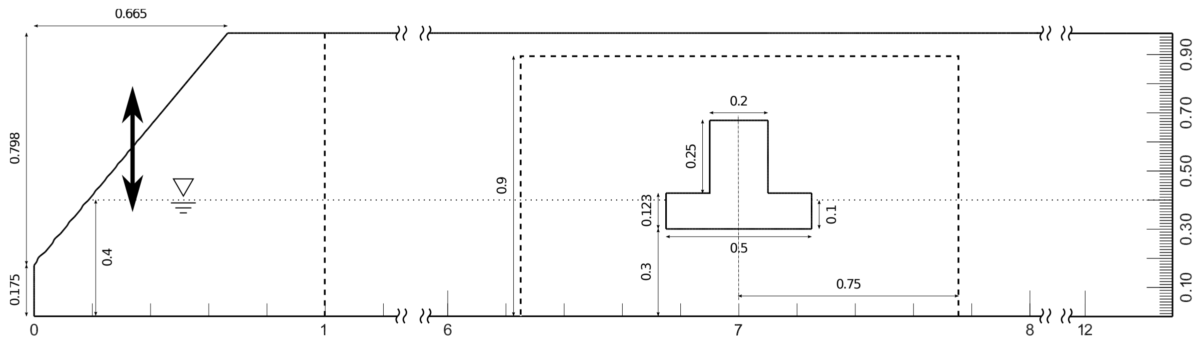


Figure 9: Wave interaction with a floating body: computational domain. Dashed lines represent virtual interfaces to couple different regions. SI units.

Table 3: Wave interaction with a floating body: mechanical properties of the floating object. The centre of gravity (COG) and centre of rotation (COR) are measured from the bottom of the object.

| Property | Length [m] | Breadth [m] | Draft [m] | Mass [kg] | M. Inertia [kgm$^2$] | COG [m] | COR [m] |
|---|---|---|---|---|---|---|---|
| Value | 0.5 | 0.3 | 0.1 | 15.0 | 0.3417 | 0.0796 | 0.1 |

ders this multi-region wave-maker a very interesting candidate for *accurate and efficient* wave generation in three-dimensional wave tanks.

### *3.3. Wave interaction with a floating body*

This benchmark is used to validate the wave-maker in conjunction with wave structure interaction with an additional floating body present in the numerical wave tank. The corresponding experiment was proposed in [49] to study extreme wave conditions on floating objects. The wave tank is 18 m long, 0.3 m wide and 0.7 m high and is equipped with a plunger-type wave-maker. Following [49], we have simplified our numerical wave tank to a $12.3 \times 0.3 \times 0.973\,\text{m}^3$ two-dimensional mesh (1 cell in the spanwise direction) as shown in Fig. 9. The water depth is 0.4 m and a floating body, i.e. a simple box-shaped geometry with a higher superstructure, is placed at a distance of 7 m from the wave-maker. During the experiments, the body is free to move in heave and roll around its centre of rotation but sway is restrained. The mechanical properties of the floating object are summarised in Table 3.

The computational domain is discretised with approximately 255000 cells ($\Delta \approx 0.5\,\text{cm}$) with a progressive cell stretching in the horizontal direction between the floating body and the end of the numerical wave tank. Walls are treated with a non-slip boundary condition and the top boundary has atmospheric pressure (1 bar). In order to absorb waves behind the floating object and avoid reflection at the right boundary, we adopt a sponge layer method based on the application of an additional viscous damping term, $\mu_d = 0.5(1+\cos(\pi x/d))$ with $d = 5\,\text{m}$, in the momentum equation (2). Finally, a Courant number of 0.5 is used to run the simulations.

In a first campaign of experiments, the floating body is removed and regular waves with $H = 0.072\,\mathrm{m}$ and $T = 1\,\mathrm{s}$ and a linear ramp time of 5 s are generated with the plunger-type paddle. The measured experimental transfer function value of this plunger is 0.72, see reference [49]. We solved numerically Eq. (4) to determine the theoretical transfer function of a plunger-type paddle with the same geometry [20] and obtained a value of 1.063, which does not reproduce satisfactorily the target wave. Although theoretical transfer functions are widely used in the literature for flap- and piston-type paddles, the discrepancy of results obtained here questions the use of a theoretical transfer function for a plunger wave-maker. Therefore, in the following we have rescinded the use of the theoretical value and used instead the experimental transfer function to calculate the motion of the paddle. Figure 10 shows the water elevation measured as a function of the time at three different locations. We compare our present results against ihFoam [27] solutions and experimental data from [49]. Overall, differences exist between the simulations and experiments during wave transition, especially when using ihFoam as it does not use a physical paddle to generate the waves. However, when the problem reaches a steady state in gauge 1, wsiFoam overestimates the wave crests although wave troughs are well predicted. On the other hand, ihFoam seems to systematically underestimate the water elevation at the same gauge 1. We believe that this might be due to the influence of the wave generator, which is relatively close (3 m) to gauge 1. In fact, such differences are progressively eliminated in gauges 2 and 3 located at 5.1 m and 7 m from the wave generator, respectively, and the region of main interest for this experiment is precisely located near gauge 3 (see paragraph below).

During a second campaign of experiments, the interaction between extreme waves and a floating body is studied in detail. The main interests are extreme wave generation, floating body motion, free surface variation and green water impact pressure. For this, a pressure gauge is installed on the superstructure at a height of 0.01 m from the deck, at the seaward face, to record the green water impact pressure. Regular waves with $H = 0.062\,\mathrm{m}$ are generated similarly to the previous experiment. Fig. 11 shows the time history of the water elevation as well as the heave and roll of the floating body. We compare our present numerical results against the experimental data and numerical solutions by Zhao & Hu [49]. The water elevation predicted by wsiFoam at $x = 5.1\,\mathrm{m}$ shows a better agreement with experiments compared to Zhao & Hu numerical results, which clearly underestimate this magnitude. Nevertheless, heave and roll numerical values do not show large differences and, in general, the simulations agree qualitatively well with the experiments since both capture non-zero values of the average heave and roll motions due to the presence of water on deck, see Fig. 12. Overall, the position of the floating body and the water free surface are in good agreement with the experimental snapshots. It can be seen in the simulation that the free surface shows more disturbance

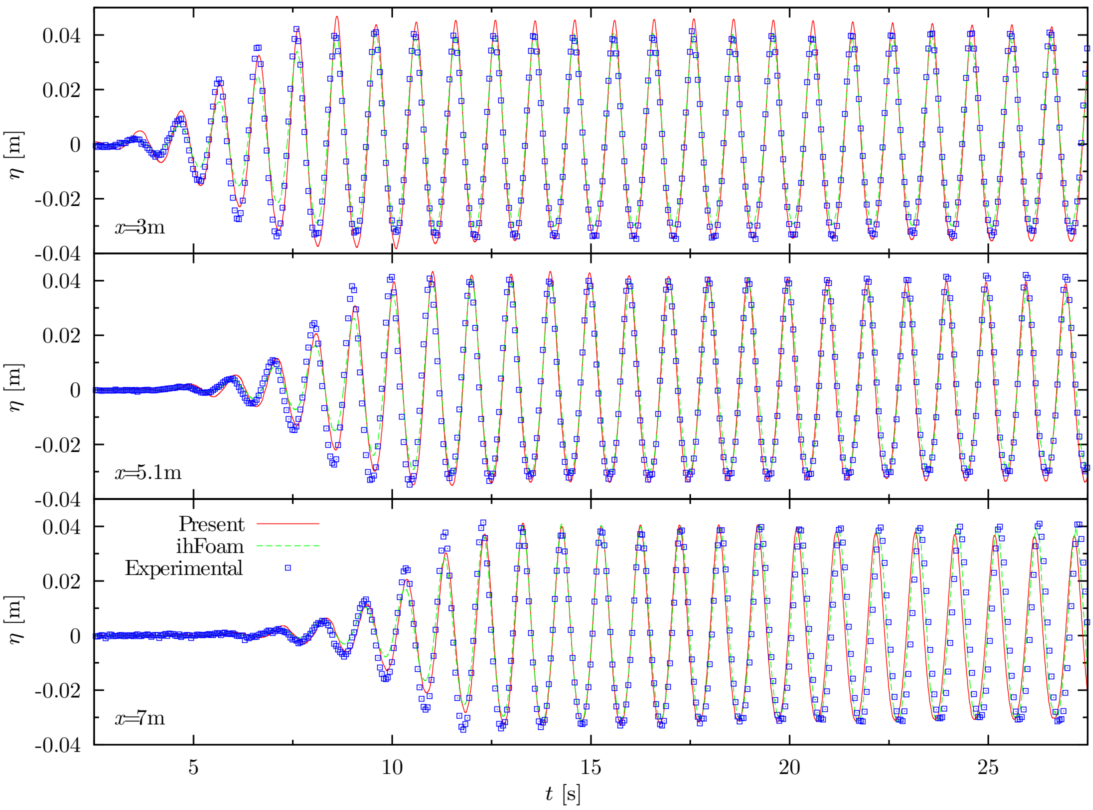


Figure 10: Wave interaction with a floating body: water elevation measured as a function of the time at locations 0.3 m, 5.1 m and 7 m from the initial position of the wave-maker.

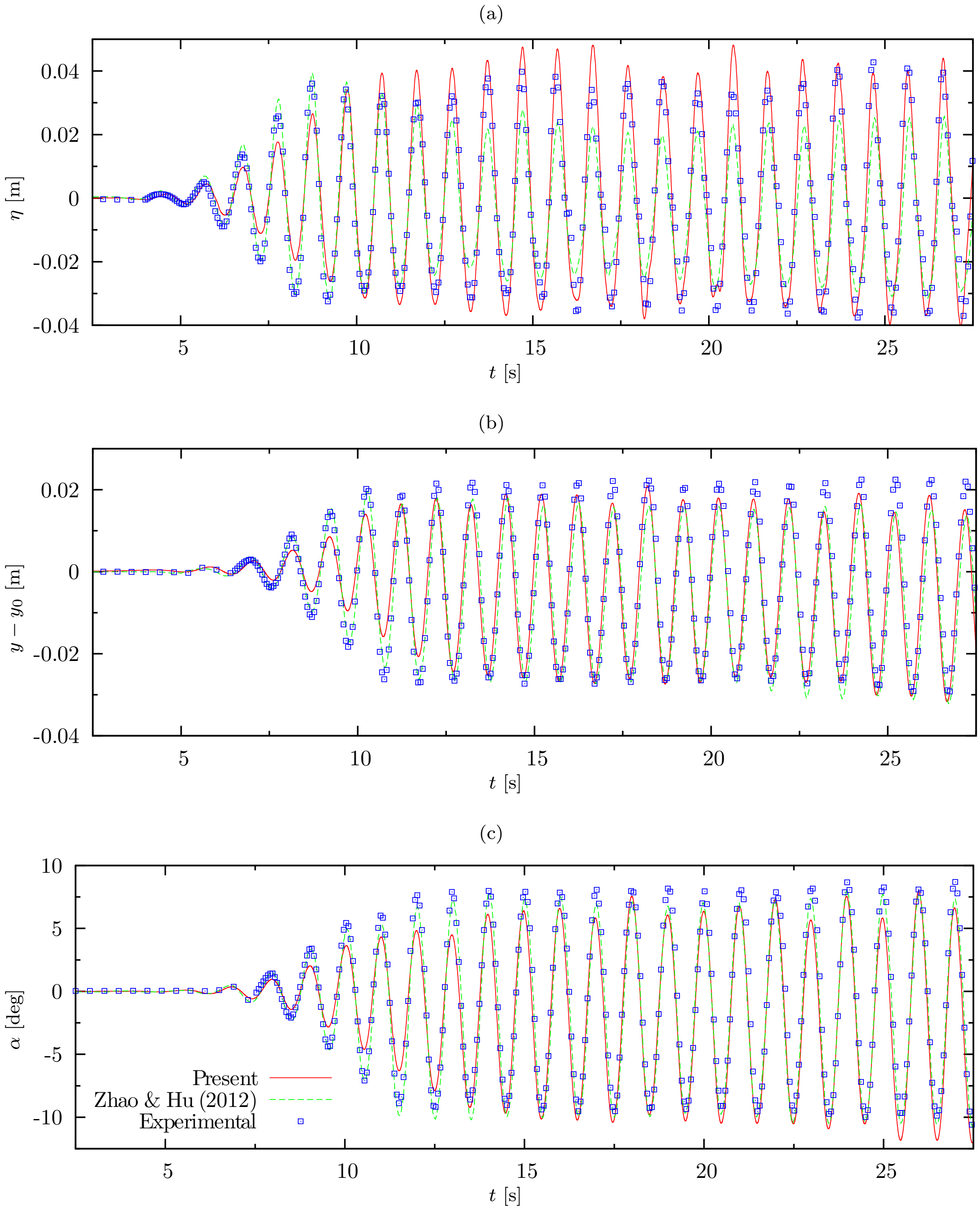


Figure 11: Wave interaction with a floating body: time history of the (a) water elevation at 5.1 m from the initial position of the wave-maker, (b) heave and (c) rotation angle of the floating body.

compared to the experiment while there is an increase in the velocity magnitude in that region. This is due to the presence of spurious currents at the interface, which have been already acknowledged for this interface capturing solver [50]. On the one hand, such currents can pose serious concerns in the computation of capillary flows and several methods have been proposed to suppress them, see for instance reference [51]. On the other hand, spurious interface velocities are not a major issue for inertia-dominated flows like the ones presented in this work and thus no special treatment has been applied. However, a new ghost fluid method has recently been published to deal with spurious currents in inertia-dominated flows [52]. Finally, Fig. 13 shows the pressure as a function of time measured above the deck. The maximum pressure peaks between 300 Pa and 400 Pa captured by our numerical simulations are within the same order of magnitude as those previously reported in the literature. It is worth mentioning that the pressure recorded in the experiment presents an offset probably due to abrupt changes in temperature when the water enters in contact with the gauge.

As previously discussed, the use of a dynamic mesh to accommodate the motion of the wave-maker and the floating body will inherently increase the computational time. Table 4 shows the simulation times for different configurations, normalised by a reference time corresponding to the fastest one. All the computations were carried out in parallel on 16 Intel Xeon E5-2620 cores (approximately 16000 cells per core). Without the presence of a floating body, ihFoam is about three times faster than our current wave-maker implementation using a single incompressible region (I). This is due exclusively to the dynamic mesh library handling the deformation of the mesh to replicate the wave-maker motion. If the domain is split in two regions (I-I), i.e. with a vertical interface at $x = 1\,\mathrm{m}$ as shown in Fig. 9, only the smaller portion of the mesh containing the wave-maker requires to be dynamic. This multi-region simulation only takes 30% more CPU time compared to ihFoam. It is even more interesting to analyse the simulation times in the presence of a floating body. It can be readily seen that the ihFoam simulation time increases almost one order of magnitude compared to the previous case. This is due to (i) the application of the 6-DOF dynamic mesh motion library to the floating body and (ii) the increasing number of subiterations in the PIMPLE procedure [36] to obtain convergent fields near the floating object. On the other hand, wsiFoam is only 5% slower than ihFoam in a single-region setup. In this scenario, the mesh can be divided into three regions (I-I-I): a small dynamic region for the wave-maker, another dynamic region of $1.5 \times 0.3 \times 0.9\,\mathrm{m}^3$ to accommodate the motion of the floating object and a third region (the largest one), which is static and only serves to propagate the waves, see Fig. 9. The simulation time is reduced to less than half of the time employed by the previous approaches relying on a single-region mesh. Therefore, this multi-region wave-maker methodology

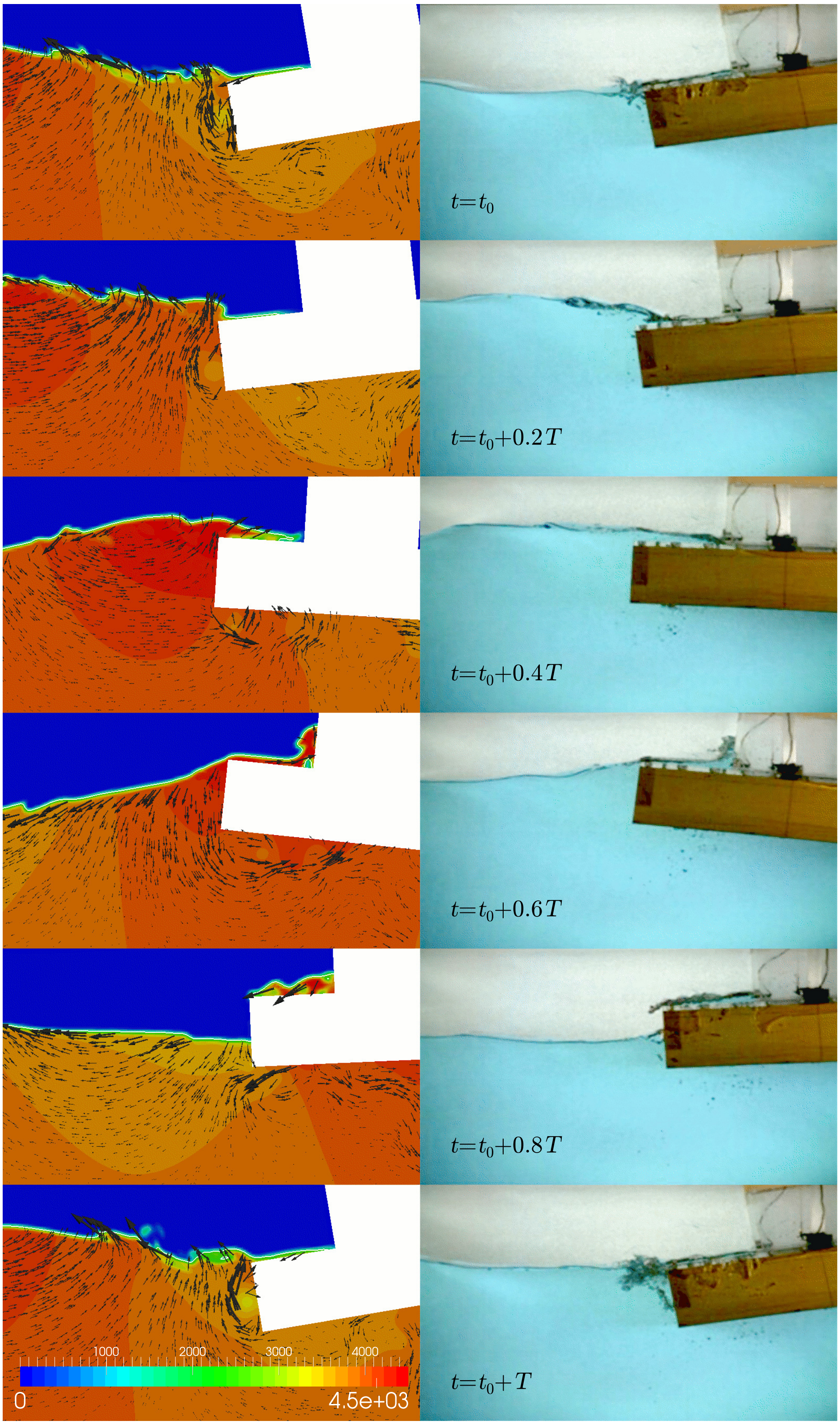


Figure 12: Wave interaction with a floating body: comparison between numerical simulations (left) and experiments (right) during a wave period of $T = 1\,\mathrm{s}$; $t_0 = 19.8\,\mathrm{s}$ in the present simulation and the velocity vectors are superimposed on the dynamic pressure contours ($p - p_{\mathrm{atm}} - \rho\mathbf{g}{\cdot}\mathbf{x}$). SI units.

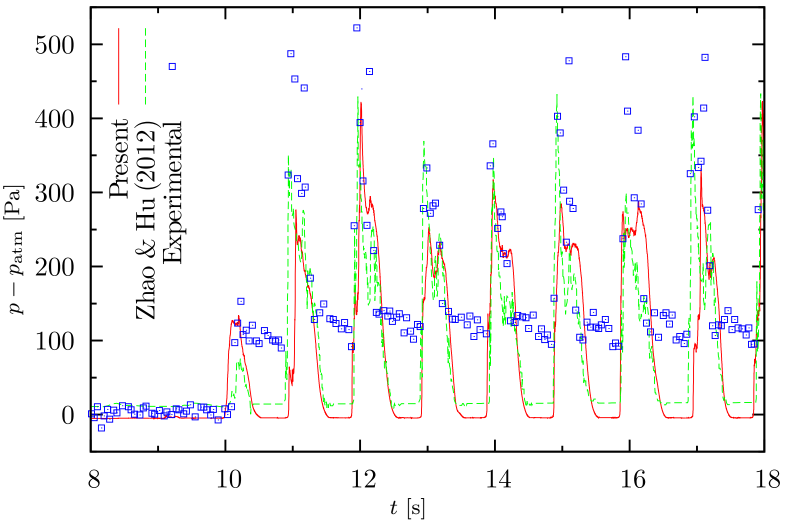


Figure 13: Wave interaction with a floating body: time history of the pressure measured above the deck of the floating body.

Table 4: Wave interaction with a floating body: simulation times normalised by $t_{\mathrm{ref}} = 11547\,\mathrm{s}$.

| Case | ihFoam | wsiFoam (I) | wsiFoam (I-I) | wsiFoam (I-I-I) |
|---|---|---|---|---|
| Without F.O. | 1.00 | 3.02 | 1.30 | – |
| With F.O. | 9.03 | 9.47 | – | 4.05 |

constitutes a *significantly* efficient alternative to other widely used tools employed in complex simulations involving numerical wave generation and interaction with floating bodies.

### *3.4. Modelling of a wave energy converter*

The "Oyster" Wave Energy Converter (WEC) developed by AquaMarine Power Limited and Queen's University Belfast (QUB) is an oscillating wave surge converter device consisting of a flap hinged at the seabed and driven back and forth by the action of waves [53, 54]. Oscillating WECs are typically installed in nearshore regions in order to transform the kinematic energy of ocean waves into electricity. Nearshore locations offer the advantage of filtering most extreme wave heights through the process of depth-induced wave breaking. Nevertheless, a renewable wave energy device such as the Oyster is conceived to operate in an aggressive environment for 25 years and, therefore, it must be designed to survive infrequent *but* large extreme waves.

A first campaign of experiments was carried out elsewhere to assess the characteristics of wave impacts on the Oyster [55]. Slamming phenomenon, identified as significantly large pressure peaks at the flap wall surface, were observed in the experiments and numerical simulations using commercial software [29] and in-house numerical codes [56]. Recent investigations were performed in the laboratory with slight changes

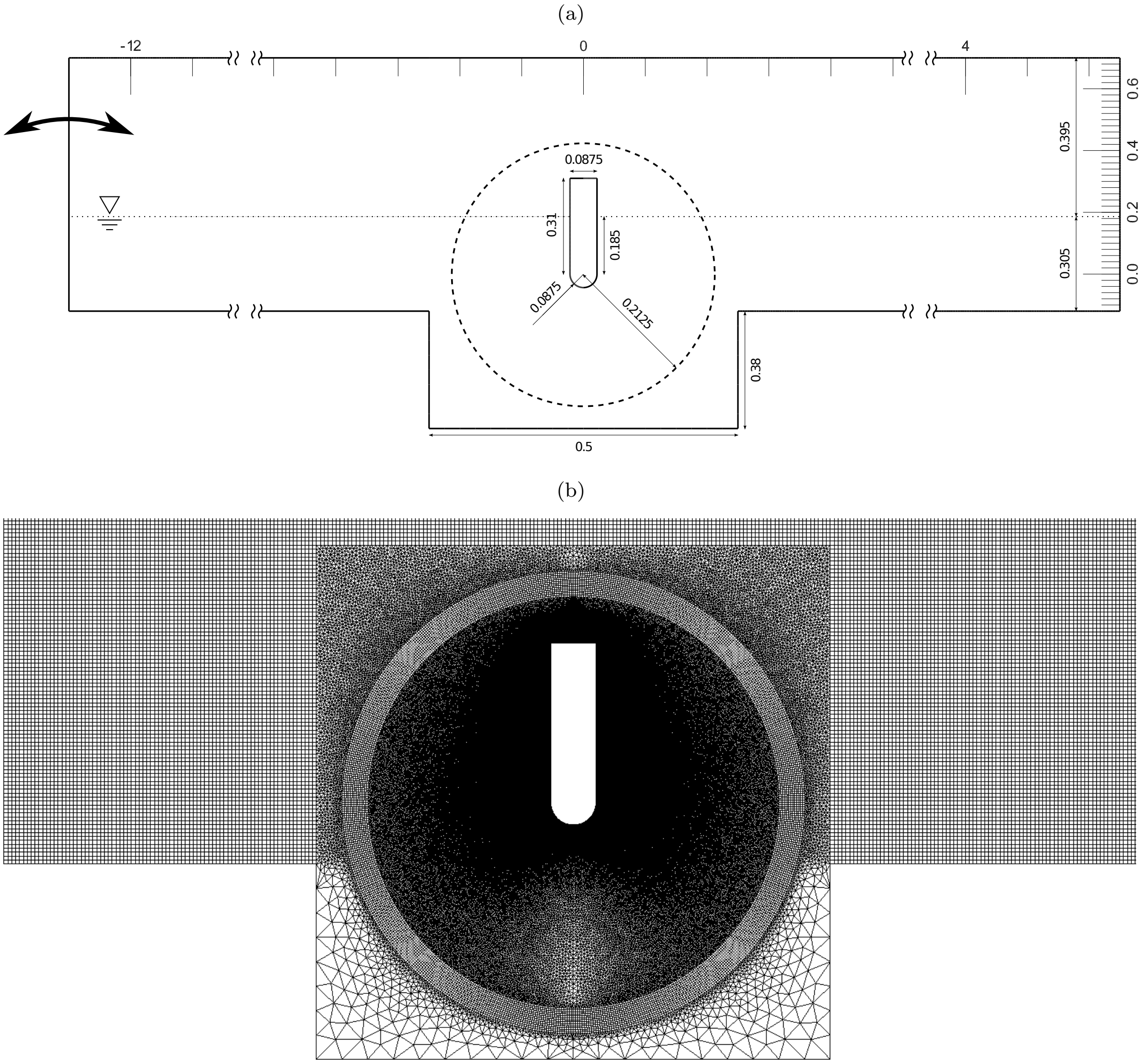


Figure 14: Modelling of a wave energy converter: (a) entire computational domain and (b) mesh structure near the Oyster. The dashed circumference represents the arbitrary mesh interface (AMI) to accommodate the WEC motion. SI units.

Table 5: Modelling of a wave energy converter: mechanical properties of the Oyster. The centre of gravity (COG) is measured from the hinge.

| Property | Thickness [m] | Span [m] | Height [m] | Mass [kg] | M. Inertia [kgm$^2$] | COG [m] |
|---|---|---|---|---|---|---|
| Value | 0.0875 | 0.646 | 0.31 | 4.35 | 0.1147 | 0.1324 |

in the wave tank and Oyster original geometries [57]. These modifications revealed a significant reduction of pressure peaks compared to the first campaign of experiments. This reduction was mainly attributed to the re-reflection phenomenon between the wave-maker and the Oyster within the experimental wave tank, which contribute to a further increase or decrease of wave height and, consequently, to a change in the impact pressure [58]. The aim of this section is to conduct a numerical investigation of wave slamming on the Oyster by replicating the first campaign of experiments. For this purpose, we (i) generate the waves with a flap-type wave-maker similar to the one used in the laboratory and (ii) replicate the wave tank and Oyster geometries. Similarly to all previous experiments and simulations, power take off (PTO) systems for this WEC device are not considered and hence the Oyster is allowed to rotate freely about its hinge.

Fig. 14(a) illustrates the computational domain used to carry out the simulations. The physical dimensions of this two-dimensional mesh are $16.77 \times 0.646 \times 0.7\,\mathrm{m}^3$ and correspond to the wave flume dimensions used in the experiments [29]. It is discretised with approximately 503000 cells ($\Delta \approx 0.8\,\mathrm{mm}$ at the WEC surface), see Fig. 14(b), in order to guarantee the convergence of the results [59]. The numerical wave tank is filled to a water depth of 0.305 m and the centre of coordinates of the mesh coincides with the Oyster's hinge, i.e. axis of rotation, see Fig. 14. This device is installed 12.2 m away from the wave-maker, on a supporting platform, so its hinge is elevated 0.12 m above the seabed. The reader is referred to Table 5 for more details about the mechanical properties of the Oyster. The 1-DOF rotation of the WEC is achieved with the combination of the dynamic mesh and arbitrary mesh interface (AMI) libraries included in OpenFOAM [40, 60]. The AMI region is a cylinder of diameter 0.425 m containing the Oyster and rotating around its hinge, interchanging information with the external mesh through the interface, depicted with a dashed line in Fig. 14. The artificial mesh below the seabed necessary to accommodate the flap rotation, including the platform between the flap and the sea bottom, is handled by porosity terms in order to guarantee a zero velocity flux in that region. The walls of the tank and the Oyster share a non-slip condition, the top boundary remains open to the atmosphere (1 bar) and the right boundary uses ihFoam active wave absorption to dissipate the incoming waves. Finally the simulations are run up to 60 s driven by a Courant number of 0.5.

A flap-type paddle hinged 0.5 m below the seabed generates regular waves with $H = 0.1\,\mathrm{m}$ and $T = 1.9\,\mathrm{s}$ and a linear ramp time of 5 s. In the present simulation, the wave-maker moves according to the

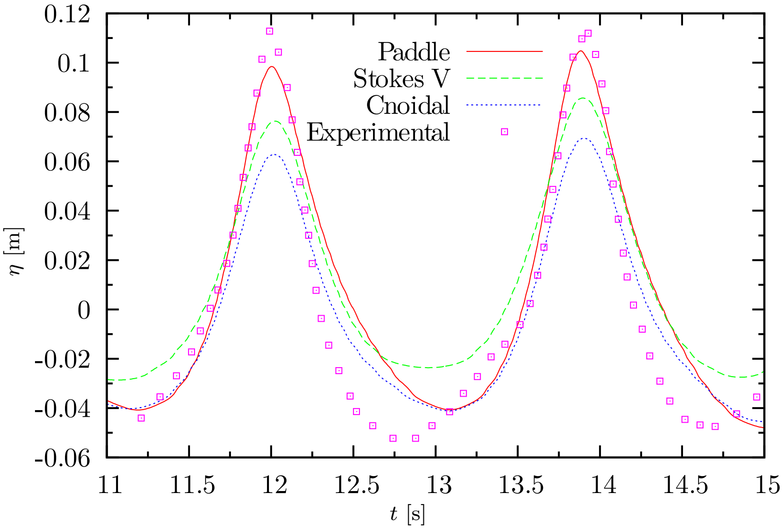


Figure 15: Modelling of a wave energy converter: time history of the water elevation for different wave generation methods at 5.178 m from the initial position of the wave-maker.

theoretical transfer function given by Eq. (3). Fig. 15 compares the time history of the water elevation for different wave generation methods at a distance of 5.178 m from the initial position of the wave-maker. This preliminary study shows that there is a general disagreement between numerical simulations and experimental data measured by QUB [29]. ihFoam cnoidal wave theory gives the less accurate results, followed by Stokes V theory. However, it is noteworthy that our current numerical implementation does not correctly predict the waves measured in the experiment, which may be attributed to the use of a theoretical transfer function. Unfortunately, neither the experimental transfer function nor the paddle displacement signal could be retrieved from the original experiment. In the following we present the rest of the numerical results based on our current dynamic-boundary numerical method as it remains the best approximation to experiments.

Figs. 16(a)–(b) report on the rotation angle and angular velocity of the Oyster, respectively. Negative rotation angles correspond to the flap pitched seaward, i.e. towards the wave-maker. Overall, our present results are on a par with Fluent simulations carried out by University College Dublin (UCD) [29] and agree qualitatively well with the non-linear variation of the angular velocity measured in experiments. However, the two numerical codes struggle to predict the minimum rotation angles of the Oyster and underestimate the angular velocities, specially when the flap is pitching seaward as shown in Fig. 17. This figure also evidences that the aeration level, i.e. the presence of bubbles, is significantly less important in the computation. Fig. 16(c) shows the time history of the *dynamic* pressure measured by a sensor mounted 0.21 m above the hinge of the Oyster, in the seaward wall. Four consecutive pressure peaks, each one of them corresponding

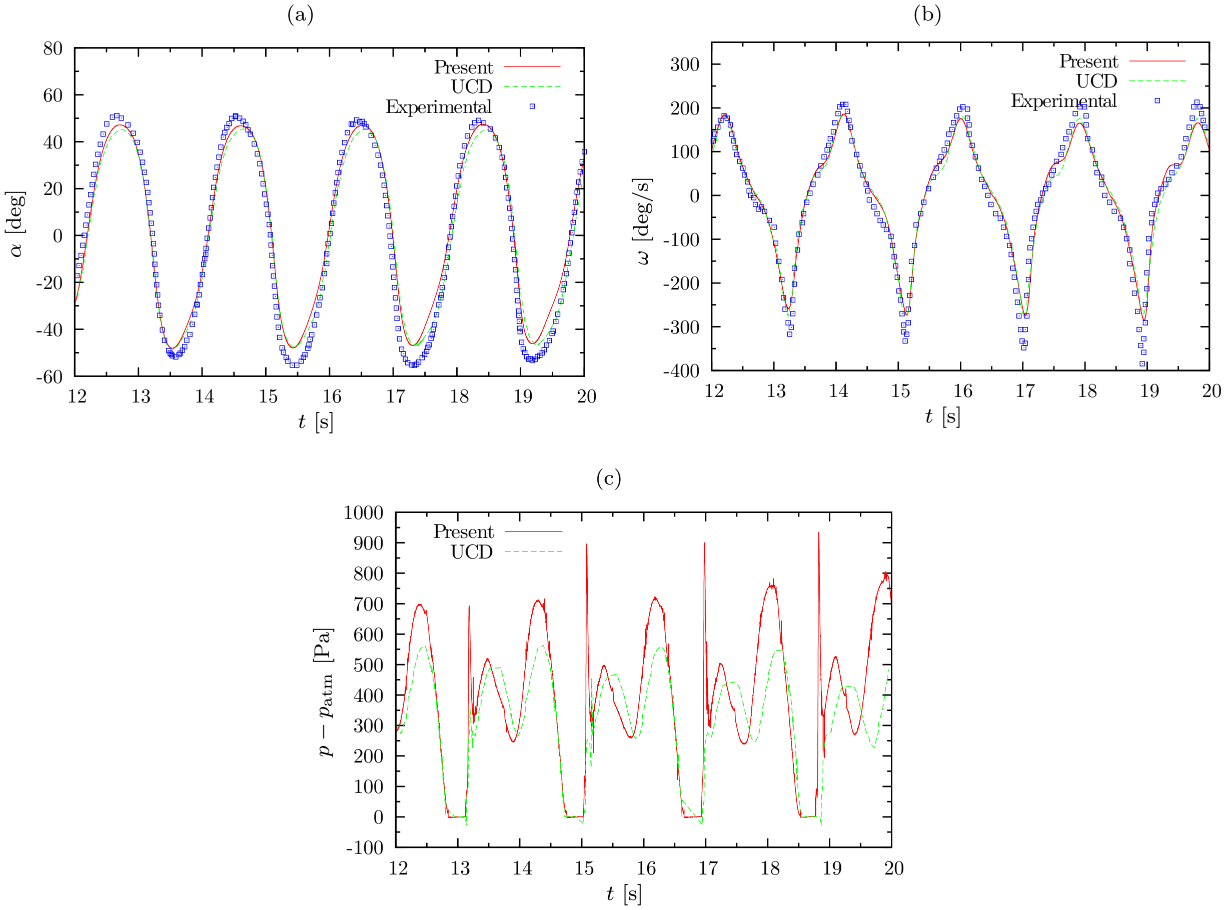


Figure 16: Modelling of a wave energy converter: time history of the (a) rotation angle, (b) angular velocity and (c) dynamic pressure measured 0.21 m above the hinge of the WEC.

to a new wave cycle, can be identified in our present simulations at approximately 13 s, 15 s, 17 s and 19 s, indicating the time instant at which the slam pressure arrives at the gauge. Slamming is produced by the formation of a jet travelling up the seaward face of the Oyster when it pitches seaward reducing the gap with the water surface, see Fig. 17: the increasing values of velocity and pressure take place during the formation of the jet as the flap pitches seaward from its vertical position. Our maximum slamming pressure values below 1 kPa are not as high as those measured in the laboratory that range from 5 kPa to 10 kPa. Note that the UCD results further underestimate the slamming pressure. As pointed out in [29], two critical parameters controlling the slamming intensity are the relative velocity and the contact angle between the WEC and the water surface. The former is underestimated and the latter overestimated in the numerical simulations shown before, which may be a direct consequence of the differences between the waves generated in the numerical wave tank and in the laboratory.

Finally, Fig. 18 shows the dynamic pressure distribution along the surface of the Oyster as a function of the time to give a better illustration of the slamming phenomena. The vertical axis represents the perimeter of the Oyster where $l = 0\,\mathrm{m}$ indicates the lowest point of the flap (0.0875 m below the hinge), increasing values of the perimeter correspond to the seaward wall while negative values belong to the opposite wall. Three consecutive pressure bands at higher coordinates of the seaward wall indicate the presence of slamming events. The maximum pressure values are associated with the root of the jet formed, similar to a classic water entry problem [61, 62]. These peaks with maximum values slightly below 2 kPa are preceded by negative values of the dynamic pressure around $-2\,\mathrm{kPa}$ due to the drop of the water line, i.e. hydro-static pressure, as the flap pitches seaward from its vertical position. In general, our results agree qualitatively well with previous numerical investigations [58], capturing the dynamics and the slamming events observed in the experiments. Nevertheless, all the simulations underestimate the high pressure peaks recorded in the laboratory, which may be explained by the local and stochastic nature characteristic of wave impacts, where experimental results can be very sensitive to free-surface instabilities [63].

## 4. Conclusions

We have presented an efficient and accurate numerical wave generation method for ocean and coastal engineering applications named as wsiFoam and based on the open-source CFD library OpenFOAM. This method has been successfully applied to four different and relevant benchmarks and validated against experimental data and other commonly used numerical tools. This implementation utilises the dynamic mesh library provided by OpenFOAM and is built on top of a multi-region approach, where different regions

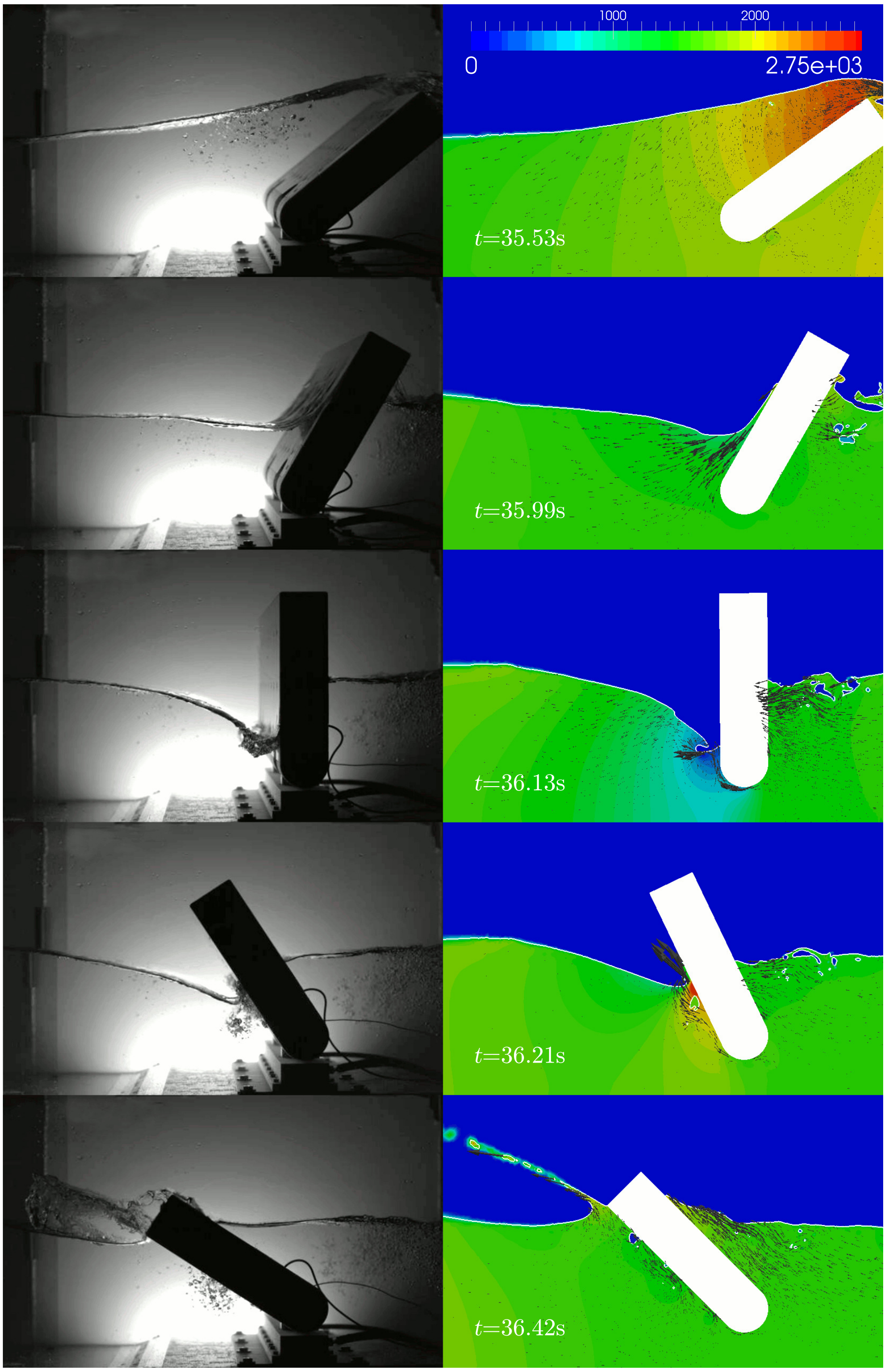


Figure 17: Modelling of a wave energy converter: comparison between experiments (left) and numerical simulations (right) during a wave slamming event; the velocity vectors are superimposed on the dynamic pressure contours ($p - p_{\mathrm{atm}} - \rho\mathbf{g}\cdot\mathbf{x}$). SI units.

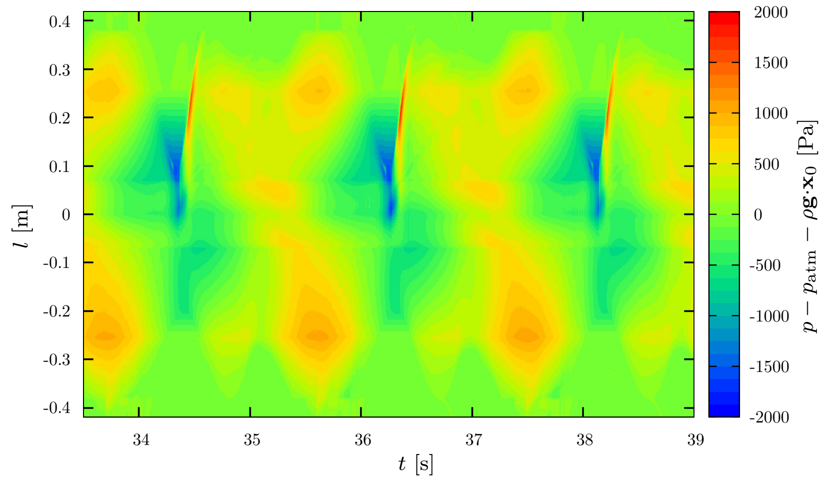


Figure 18: Modelling of a wave energy converter: dynamic pressure distribution along the surface of the WEC as a function of the time.

of the computational domain can be specified, thus providing an accurate and, more importantly, efficient dynamic-boundary numerical wave-maker. Our present results demonstrate that the current implementation is on a par with other wave generation tools while it can provide more accurate solutions in cases where first-order static-boundary methods and theoretical transfer functions underpredict experimental measurements. Furthermore, this strategy can reduce the computational time substantially, compared to standard methods, in practical cases where floating bodies are present in the numerical wave tank. These features render the current numerical wave generation method an attractive open-source candidate for the simulation of ocean and coastal engineering problems in the scientific and industry communities.

Future work remains to extend the functionality of the current numerical wave-maker to other solvers such as a fully non-linear potential method in order to reduce further the computational cost. More complex wave-maker geometries, including arrays of segmented paddles and dynamic-boundary wave absorbers may also be envisaged in the future. Our ultimate goal is to construct a general purpose open-source numerical wave tank for the simulation of wave structure interaction problems, including deforming bodies and therefore the use of CSD solvers, which will allow us to study the complete physics characteristic of ocean and coastal engineering problems.

**Acknowledgements**

The authors acknowledge with gratitude financial support from the Engineering and Physical Sciences Research Council (EPSRC) under the Software for the Future (SoFT) initiative and related research grants EP/K037889/1, EP/K038168/1 and EP/K038303/1, EP/M022382/1 and EP/N008839/1.

## References


[1] S. T. Grilli, J. Horrillo, Numerical generation and absorption of fully nonlinear periodic waves, Journal of Engineering Mechanics 123 (10) (1997) 1060–1069. doi:10.1061/(asce)0733-9399(1997)123:10(1060).
URL http://dx.doi.org/10.1061/(ASCE)0733-9399(1997)123:10(1060)

[2] Q. Ma, S. Yan, Quasi ALE finite element method for nonlinear water waves, Journal of Computational Physics 212 (2006) 52–72. doi:10.1016/j.jcp.2005.06.014.
URL http://dx.doi.org/10.1016/j.jcp.2005.06.014

[3] P. Lubin, S. Vincent, S. Abadie, J.-P. Caltagirone, Three-dimensional large eddy simulation of air entrainment under plunging breaking waves, Coastal Engineering 53 (2006) 631–655. doi:10.1016/j.coastaleng.2006.01.001.
URL http://dx.doi.org/10.1016/j.coastaleng.2006.01.001

[4] P. Higuera, J. L. Lara, I. J. Losada, Simulating coastal engineering processes with OpenFOAM, Coastal Engineering 71 (2013) 119–134. doi:10.1016/j.coastaleng.2012.06.002.
URL http://dx.doi.org/10.1016/j.coastaleng.2012.06.002

[5] D. Peregrine, H. Bredmose, D. Dunn, T. Somerville, G. Bullock, C. Obhrai, G. Wolters, Compressibility of entrained and trapped air in violent water wave impacts, in: Proceedings of the 21st International Workshop on Water Waves and Floating Bodies, 2006.

[6] H. Bredmose, D. H. Peregrine, G. N. Bullock, Violent breaking wave impacts. Part 2: modelling the effect of air, Journal of Fluid Mechanics 641 (2009) 389–430. doi:10.1017/s0022112009991571.
URL http://dx.doi.org/10.1017/S0022112009991571

[7] C. Lugni, M. Brocchini, O. M. Faltinsen, Evolution of the air cavity during a depressurized wave impact. II. The dynamic field, Phys. Fluids 22 (2010) 056102. doi:10.1063/1.3409491.
URL http://dx.doi.org/10.1063/1.3409491

[8] A. Colagrossi, M. Landrini, Numerical simulation of interfacial flows by smoothed particle hydrodynamics, Journal of Computational Physics 191 (2003) 448–475. doi:10.1016/s0021-9991(03)00324-3.
URL http://dx.doi.org/10.1016/S0021-9991(03)00324-3

[9] P.-M. Guilcher, N. Couty, L. Brosset, D. Le Touzé, Simulations of breaking wave impacts on a rigid wall at two different scales with a two-phase fluid compressible SPH model, International Journal of Offshore and Polar Engineering 23 (2013) 241–253.

[10] Z. H. Ma, D. M. Causon, L. Qian, C. G. Mingham, H. B. Gu, P. M. Ferrer, A compressible multiphase flow model for violent aerated wave impact problems, Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences 470 (2014) . arXiv:http://rspa.royalsocietypublishing.org/content/470/2172/20140542.full.pdf, doi:10.1098/rspa.2014.0542.
URL http://rspa.royalsocietypublishing.org/content/470/2172/20140542

[11] Z. Ma, D. Causon, L. Qian, H. Gu, C. Mingham, P. M. Ferrer, A GPU based compressible multiphase hydrocode for modelling violent hydrodynamic impact problems, Computers & Fluids 120 (2015) 1–23. doi:10.1016/j.compfluid.2015.07.010.
URL http://dx.doi.org/10.1016/j.compfluid.2015.07.010

[12] P. Martínez Ferrer, D. Causon, L. Qian, C. Mingham, Z. Ma, A multi-region coupling scheme for compressible and incompressible flow solvers for two-phase flow in a numerical wave tank, Computers & Fluids 125 (2016) 116–129. doi:

http://dx.doi.org/10.1016/j.compfluid.2015.11.005.
URL http://www.sciencedirect.com/science/article/pii/S004579301500376X

[13] A. Iafrati, E. F. Campana, A domain decomposition approach to compute wave breaking (wave-breaking flows), Int. J. Numer. Meth. Fluids 41 (2003) 419–445. doi:10.1002/fld.448.
URL http://dx.doi.org/10.1002/fld.448

[14] V. Sriram, Q. Ma, T. Schlurmann, Numerical simulation of breaking waves using hybrid coupling of FNPF (FEM) and NS (IMLPGR) solver, in: Proceedings of the 21st International Offshore and Polar Engineering Conference, 2012.

[15] Y. Zhang, M. Peszynska, S. Yim, Coupling of viscous and potential flow models with free surface for near and far field wave propagation, International Journal of Numerical Analysis and Modeling, Series B 4 (2013) 256–282.

[16] J. Schluter, X. Wu, E. van der Weide, S. Hahn, J. Alonso, H. Pitsch, Multi-code simulations: a generalized coupling approach, 17th AIAA Computational Fluid Dynamics Conference (2005) doi:10.2514/6.2005-4997.
URL http://dx.doi.org/10.2514/6.2005-4997

[17] I. Svendsen, Physical modelling of water waves, Physical Modelling in Coastal Engineering. AA Balkema, Rotterdam (1985) 13–48.

[18] F. Biesel, F. Suquet, Laboratory wave-generating apparatus, St. Anthony Falls Hydraulic Laboratories, 1954.

[19] F. Ursell, R. G. Dean, Y. S. Yu, Forced small-amplitude water waves: a comparison of theory and experiment, Journal of Fluid Mechanics 7 (01) (1960) 33. doi:10.1017/s0022112060000037.
URL http://dx.doi.org/10.1017/S0022112060000037

[20] J. M. Hyun, Simplified analysis of a plunger-type wavemaker performance, Journal of Hydronautics 10 (3) (1976) 89–94. doi:10.2514/3.63056.
URL http://dx.doi.org/10.2514/3.63056

[21] R. G. Dean, R. A. Dalrymple, Water wave mechanics for scientists and engineers, World Scientific, Advanced Series on Ocean Engineering 2 (1991) .

[22] H. A. Schaffer, Second-order wavemaker theory for irregular waves, Ocean Engineering 23 (1996) 47–88. doi:10.1016/0029-8018(95)00013-b.
URL http://dx.doi.org/10.1016/0029-8018(95)00013-B

[23] B. Le Mehaute, Introduction to hydrodynamics and water waves, Springer-Verlag, New York, 1976.

[24] G. Wei, J. T. Kirby, Time-dependent numerical code for extended Boussinesq equations, Journal of Waterway, Port, Coastal, and Ocean Engineering 121 (1995) 251–261. doi:10.1061/(asce)0733-950x(1995)121:5(251).
URL http://dx.doi.org/10.1061/(ASCE)0733-950X(1995)121:5(251)

[25] A. Mehmood, D. I. Graham, K. Langfeld, D. M. Greaves, Numerical simulation of nonlinear water waves based on fully nonlinear potential flow theory in OpenFOAM-Extend, in: Proceedings of the 26th International Ocean and Polar Engineering Conference, 2016, pp. 672–677.

[26] N. G. Jacobsen, D. R. Fuhrman, J. Fredsoe, A wave generation toolbox for the open-source CFD library: OpenFOAM, Int. J. Numer. Meth. Fluids 70 (2012) 1073–1088. doi:10.1002/fld.2726.
URL http://dx.doi.org/10.1002/fld.2726

[27] P. Higuera, J. L. Lara, I. J. Losada, Realistic wave generation and active wave absorption for Navier-Stokes models, Coastal Engineering 71 (2013) 102–118. doi:10.1016/j.coastaleng.2012.07.002.
URL http://dx.doi.org/10.1016/j.coastaleng.2012.07.002

[28] S. T. Grilli, S. Vogelmann, P. Watts, Development of a 3D numerical wave tank for modeling tsunami generation by underwater landslides, Engineering Analysis with Boundary Elements 26 (2002) 301–313. `doi:10.1016/s0955-7997(01)00113-8`.
URL `http://dx.doi.org/10.1016/S0955-7997(01)00113-8`

[29] A. Henry, O. Kimmoun, J. Nicholson, G. Dupont, Y. Wei, F. Dias, A two dimensional experimental investigation of slamming of an oscillating wave surge converter, in: Proceedings of the 24th International Ocean and Polar Engineering Conference, 2014.

[30] P. Higuera, I. J. Losada, J. L. Lara, Three-dimensional numerical wave generation with moving boundaries, Coastal Engineering 101 (2015) 35–47. `doi:10.1016/j.coastaleng.2015.04.003`.
URL `http://dx.doi.org/10.1016/j.coastaleng.2015.04.003`

[31] S. Yan, Q. Ma, V. Sriram, L. Qian, P. Ferrer, T. Schlurmann, Numerical and experimental studies of moving cylinder in uni-directional focusing waves, in: Proceedings of the 25th International Offshore and Polar Engineering Conference, 2015.

[32] H. Jasak, Error analysis and estimation for the finite volume method with applications to fluid flows, Ph.D. thesis, University of London (1996).

[33] H. Rusche, Computational fluid dynamics of dispersed two-phase flows at high phase fractions, Ph.D. thesis, University of London (2002).

[34] Z. Ma, D. Causon, L. Qian, C. Mingham, P. Martínez Ferrer, Numerical investigation of air enclosed wave impacts in a depressurised tank, Ocean Engineering 123 (2016) 15–27. `doi:10.1016/j.oceaneng.2016.06.044`.
URL `http://dx.doi.org/10.1016/j.oceaneng.2016.06.044`

[35] H. Weller, Derivation, modelling and solution of the conditionally averaged two-phase flow equations, Nabla Ltd, No Technical Report TR/HGW/02 (2002) .

[36] K. Kissling, J. Springer, J. Jasak, S. Schutz, K. Urban, M. Piesche, A coupled pressure based solution algorithm based on the volume-of-fluid approach for two or more immiscible fluids, in: Proceedings of the V European Conference on Computational Fluid Dynamics, ECCOMAS CFD, 2010.

[37] R. Issa, Solution of the implicitly discretised fluid flow equations by operator-splitting, Journal of Computational Physics 62 (1986) 40–65. `doi:10.1016/0021-9991(86)90099-9`.
URL `http://dx.doi.org/10.1016/0021-9991(86)90099-9`

[38] R. Issa, A. Gosman, A. Watkins, The computation of compressible and incompressible recirculating flows by a non-iterative implicit scheme, Journal of Computational Physics 62 (1986) 66–82. `doi:10.1016/0021-9991(86)90100-2`.
URL `http://dx.doi.org/10.1016/0021-9991(86)90100-2`

[39] H. Jasak, Dynamic mesh handling in OpenFOAM, in: Proceedings of the 47th Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, 2009, AIAA 2009-341.

[40] H. Jasak, Z. Tukovic, Automatic mesh motion for the unstructured finite volume method, Transactions of FAMENA 30 (2006) 1–20.

[41] H. Jasak, OpenFOAM: open source CFD in research and industry, International Journal of Naval Architecture and Ocean Engineering 1 (2009) 89–94. `doi:http://dx.doi.org/10.2478/IJNAOE-2013-0011`.
URL `http://www.sciencedirect.com/science/article/pii/S2092678216303879`

[42] H. Jasak, Z. Tukovic, Dynamic mesh handling in OpenFOAM applied to fluid-structure interaction simulations, in: Pro-

ceedings of the V European Conference Computational Fluid Dynamics, 2010, pp. 14–17.

[43] H. Jasak, M. Beaudoin, OpenFOAM turbo tools: from general purpose CFD to turbomachinery simulations, in: Proceedings of the ASME-JSME-KSME 2011 Joint Fluids Engineering Conference, American Society of Mechanical Engineers, 2011, pp. 1801–1812.

[44] F. Gao, An efficient finite element technique for free surface flow, Ph.D. thesis, Brighton University, U.K. (2003).

[45] L. Qian, D. Causon, C. Mingham, D. Ingram, A free-surface capturing method for two fluid flows with moving bodies, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences 462 (2005) 21–42. doi:10.1098/rspa.2005.1528.
URL http://dx.doi.org/10.1098/rspa.2005.1528

[46] H. Gu, D. Causon, C. Mingham, L. Qian, Development of a free surface flow solver for the simulation of wave/body interactions, European Journal of Mechanics - B/Fluids 38 (2013) 1–17. doi:10.1016/j.euromechflu.2012.09.010.
URL http://dx.doi.org/10.1016/j.euromechflu.2012.09.010

[47] A. Hildebrandt, V. Sriram, Pressure distribution and vortex shedding around a cylinder due to a steep wave at the onset of breaking from physical and numerical modeling, in: Proceedings of the 24th International Ocean and Polar Engineering Conference, 2014.

[48] V. Sriram, T. Schlurmann, S. Schimmels, Focused wave evolution using linear and second order wave maker theory in intermediate water depth, in: Proceedings of the 23rd International Offshore and Polar Engineering Conference, 2013.

[49] X. Zhao, C. Hu, Numerical and experimental study on a 2-D floating body under extreme wave conditions, Applied Ocean Research 35 (2012) 1–13. doi:10.1016/j.apor.2012.01.001.
URL http://dx.doi.org/10.1016/j.apor.2012.01.001

[50] S. S. Deshpande, L. Anumolu, M. F. Trujillo, Evaluating the performance of the two-phase flow solver interFoam, Comput. Sci. Disc. 5. doi:10.1088/1749-4699/5/1/014016.
URL http://dx.doi.org/10.1088/1749-4699/5/1/014016

[51] Z. Pan, J. A. Weibel, S. V. Garimella, Spurious current suppression in VOF-CSF simulation of slug flow through small channels, Numerical Heat Transfer, Part A: Applications 67 (2015) 1–12. doi:10.1080/10407782.2014.916109.
URL http://dx.doi.org/10.1080/10407782.2014.916109

[52] V. Vukcevic, H. Jasak, I. Gatin, Implementation of the ghost fluid method for free surface flows in polyhedral finite volume framework, Computers & Fluids 153 (2017) 1–19. doi:10.1016/j.compfluid.2017.05.003.
URL https://doi.org/10.1016/j.compfluid.2017.05.003

[53] M. Folley, T. Whittaker, A. Henry, The effect of water depth on the performance of a small surging wave energy converter, Ocean Engineering 34 (2007) 1265–1274. doi:10.1016/j.oceaneng.2006.05.015.
URL http://dx.doi.org/10.1016/j.oceaneng.2006.05.015

[54] T. Whittaker, M. Folley, Nearshore oscillating wave surge converters and the development of Oyster, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 370 (2011) 345–364. doi:10.1098/rsta.2011.0152.
URL http://dx.doi.org/10.1098/rsta.2011.0152

[55] A. Henry, A. Rafiee, P. Schmitt, F. Dias, T. Whittaker, The characteristics of wave impacts on an oscillating wave surge converter, in: Proceedings of the 23rd International Offshore and Polar Engineering Conference, 2013, pp. 566–573.

[56] T. Abadie, F. Dias, Numerical study of the evolution of slam pressure on an oscillating wave surge converter, in: Proceed-

ings of the 3rd International Conference on Violent Flows, 2016.

[57] A. Henry, T. Abadie, J. Nicholson, A. McKinley, O. Kimmoun, F. Dias, The vertical distribution and evolution of slam pressure on an oscillating wave surge converter, in: Proceedings of the 34th International Conference on Ocean, Offshore and Arctic Engineering, 2015.

[58] Y. Wei, T. Abadie, A. Henry, F. Dias, Wave interaction with an oscillating wave surge converter. Part II: Slamming, Ocean Engineering 113 (2016) 319–334. doi:10.1016/j.oceaneng.2015.12.041.
URL http://dx.doi.org/10.1016/j.oceaneng.2015.12.041

[59] P. J. Martínez Ferrer, D. M. Causon, L. Qian, C. G. Mingham, Z. H. Ma, Numerical simulation of wave slamming on a flap type oscillating wave energy device, in: Proceedings of the 26th International Ocean and Polar Engineering Conference, 2016, pp. 672–677.

[60] P. Farrell, J. Maddison, Conservative interpolation between volume meshes by local Galerkin projection, Computer Methods in Applied Mechanics and Engineering 200 (2011) 89–100. doi:10.1016/j.cma.2010.07.015.
URL http://dx.doi.org/10.1016/j.cma.2010.07.015

[61] H. Wagner, Phenomena associated with impacts and sliding on liquid surfaces, Z. Angew. Math. Mech. 12 (1932) 193–215.

[62] H. Gu, L. Qian, D. Causon, C. Mingham, P. Lin, Numerical simulation of water impact of solid bodies with vertical and oblique entries, Ocean Engineering 75 (2014) 128137. doi:10.1016/j.oceaneng.2013.11.021.
URL http://dx.doi.org/10.1016/j.oceaneng.2013.11.021

[63] P. Guilcher, Y. Jus, N. Couty, L. Brosset, Y. Scolan, D. Le Touzé, 2D simulations of breaking wave impacts on a flat rigid wall–Part 1: influence of the wave shape, in: Proceedings of the 24th International Ocean and Polar Engineering Conference, 2014.