Accepted Manuscript

An overset mesh based multiphase flow solver for water entry problems

Z.H. Ma, L. Qian, P.J. Martínez-Ferrer, D.M. Causon, C.G. Mingham, W. Bai

Please cite this article as: Z.H. Ma, L. Qian, P.J. Martínez-Ferrer, D.M. Causon, C.G. Mingham, W. Bai, An overset mesh based multiphase flow solver for water entry problems, *Computers and Fluids* (2018), doi: 10.1016/j.compfluid.2018.01.025

Highlights

- An overset mesh based multiphase flow solver for ship slamming is proposed.

- The method can handle moving bodies with large amplitude motions in air and waves.

- The method can successfully avoid non-physical oscillations on solid boundaries.

5 - The method can accurately predict the slamming loads on structures.

An overset mesh based multiphase flow solver for water entry problems

Z. H. Ma[a,*], L. Qian[a], P. J. Martínez-Ferrer[b,a], D. M. Causon[a], C. G. Mingham[a], W. Bai[a]

[a]*Centre for Mathematical Modelling and Flow Analysis, School of Computing, Mathematics and Digital Technology,
Manchester Metropolitan University, Manchester M1 5GD, United Kingdom*
[b]*Barcelona Supercomputing Center (BSC), C/ Jordi Girona 31, 08034 Barcelona, Spain*

Abstract

This paper extends a recently proposed multi-region based numerical wave tank (Martínez-Ferrer *et al.*, 2016 [1]) to solve water entry problems in naval engineering. The original static linking strategy is developed to enable the dynamic coupling of several moving regions. This permits the method to deal with large-amplitude motions for structures slamming into water waves. A background grid and one or more component meshes are firstly generated to overlay the whole computational domain and the sub-domains surrounding the structures, respectively. During computation, the background mesh is fixed while the small grids move freely or as prescribed without deformation and regeneration. This effectively circumvents the large and often excessive error-prone dynamic deformation of a single-block mesh as well as the complex and time-consuming mesh regeneration. Test cases of dam breaking with and without obstacles are first conducted to verify the developed code by comparing the numerical solution against experimental data. Then the new code is used to solve prescribed and free-fall water entry problems. The obtained results agree well with experimental measurements and other computational results reported in the literature.

Keywords: naval engineering, computational fluid dynamics, slamming

1. Introduction

Launching free-fall lifeboats from large vessels and offshore platforms under high sea states presents a vital challenge to maritime safety. The most crucial problem occurs at the moment when the lifeboat slams into water waves and experiences extremely violent hydrodynamic impact pressures and forces, which could severely damage the vessel and threaten the safety of the crew and passengers on-board [2, 3]. A versatile numerical modelling tool that can accurately and efficiently predict these critical hydrodynamic loadings is urgently needed to assess the potential life-threatening risks in the early stages [4].

---

*Corresponding author. Tel: +44 (0)161-247-1574
Email address: z.ma@mmu.ac.uk (Z. H. Ma)

From the point of view of numerical simulation, modelling the whole process of the launching operation is a most challenging task. The fluid flow is not steady but changes rapidly with respect to the kinematics and dynamics. Obviously, the problem is not single phase but actually involves a complex multi-phase flow involving both air (high wind) and water (large waves) [5]. Although air and water are usually treated as incompressible fluids in many ocean and naval applications, the compressibility of an air and/or water-air mixture could significantly influence the hydrodynamic loadings on a structure under critical conditions. This important effect must be carefully taken into account in modelling complex free surface flows [6–8].

Considering the dropping height of the lifeboat from the mother ship or offshore platform to the sea surface, the amplitude of the motion of the small vessel could be well above its own size. It is very demanding to deal with such a large amplitude motion in computational fluid dynamics by using deforming mesh methods alone, which usually can only handle relatively small amplitude motions. Some researchers have proposed introducing grid regeneration into dynamic deforming meshes when the mesh quality has severely deteriorated [9, 10]. However runtime mesh regeneration for complex problems is extremely time-consuming and error-prone [11].

This limitation obliges scientific researchers to look at other advanced methodologies to overcome the inherent difficulty of deforming mesh methods. Efforts have been devoted to the development of immersed boundary [12] and Lagrangian particle methods [13, 14]. For multiphase flow impact problems, however, non-physical oscillations of computed pressures have been observed without being fully addressed in these works. On the other hand, in the context of single-phase flow problems e.g. aerodynamics, a novel method allowing the use of several overlapping meshes simultaneously has been developed to deal with multiple stationary or moving bodies. This strategy, namely overlapping, overset or Chimera grid methods, has proved to be an effective way to handle moving boundaries with large amplitude motions in aerospace engineering [15–17]. However the investigation and application of overset mesh methods in free surface hydrodynamics is still quite limited. Attempts to introduce overlapping grid techniques into open source CFD codes have been reported in the literature by some researchers [18, 19]. However these authors did not address the crucial impact of pressures and forces on structures, which is a major concern of the present work. Consequently, it is uncertain whether they could properly handle challenging slamming problems for maritime safety. Moreover these works rely on third-party proprietary software, which make it difficult for other researchers and users to access them.

In the present work we aim at taking advantage of the flexibility and robustness of overset mesh methods to solve representative water entry problems for lifeboat launching operations. This paper extends our recently proposed multi-region based numerical wave tank named *wsiFoam* (Ferrer *et al.* , 2016 [1]) to solve water entry problems in naval engineering. The original static interface linking strategy, which exchanges information at the fixed boundaries be-

3

tween different non-overlapping meshes by using a linear interpolation, is developed to permit the dynamic coupling of several moving regions. This enables the method to deal with large-amplitude motions for structures slamming into water waves. A background grid and one or more component meshes are firstly generated to overlay the whole com-
65 putational domain and the sub-domains surrounding the structures, respectively. During computation, the background mesh is fixed while the small grids are able to move freely or as prescribed without deformation or regeneration. This effectively circumvents the large and often excessive error-prone dynamic deformation of a single-block mesh as well as the complex and time-consuming mesh regeneration. The developed computer program utilises the OpenFOAM framework only and it does not depend on any other third-party software like the overset grid library SUGGAR++
70 applied in the reported works [18, 19].

The remainder of the paper is organised as follows. Section 2 describes the free surface flow model and the underlying numerical solution methodology. The procedure to implement the overset mesh technique for solving multiphase flow problems is presented in Section 2.2. Test cases of dam breaking with and without obstacles are firstly conducted to verify the developed code in Section 3. Then the code is applied to solve vertical and oblique
75 water entry of wedges and ship hulls in Section 4. Final conclusions are drawn in Section 5.

2. Multiphase flow model

The numerical code *wsiFoam* developed in our early work [1] consists of several solvers for free surface problems: 1) a standalone incompressible two-phase flow solver *interFoam*, 2) a compressible two-phase flow solver *compressibleInterFoam* and 3) a coupled model linking non-overlapping incompressible and compressible regions through a
80 static interface, at which the flow kinematics and dynamics are exchanged between different regions by applying a linear interpolation. The first two basic flow solvers come with standard releases of the open source library Open-FOAM. The coupling method offers a suitable way to deal with the complex physics of wave-structure-interaction problems. Each of these has been tested and validated through a series of benchmark cases including dam break, wave propagation, inviscid vortex, pseudocavitation and air-enclosed sloshing wave impacts [1, 20]. In addition to these
85 three primary options, the numerical code can be utilised in a flexible way to combine multiple incompressible and/or compressible regions. Moreover in each region either a rigid or deforming mesh can be easily adopted to allow for efficient computation of the whole domain. This is superior to the use of a single-block deforming grid for the whole domain, which could be dramatically slow [21].

The framework of *wsiFoam* is inherited here to treat each small flow region as a sub-domain of the whole field. The
90 present work develops the original static interface linking strategy to permit the dynamic coupling of several moving regions. To avoid over-complicating the development of the new method, we focus on the implementation of the

4

overset mesh capability in the component solver *interFoam*. The aim of the present work is to assess the effectiveness and accuracy of the developed method.

2.1. Governing equations

The underlying component package *interFoam* is an incompressible two-phase pressure-based solver. It solves the three-dimensional equations for two incompressible phases, i.e. air and water, using the volume of fluid (VOF) method with special emphasis on maintaining a sharp free surface. The first equation to be solved is for mass conservation. For an incompressible two-phase flow, where the densities of both fluids are constants, only the mass balance equation for the water volume fraction $\alpha \in [0, 1]$ is considered

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

where the volume fraction for the water component is given by

$$\alpha = \frac{\Omega_{\text{water}}}{\Omega_{\text{water}} + \Omega_{\text{air}}} \tag{2}$$

and $\mathbf{U}$ is the mixture velocity vector and $\mathbf{U}_c$ is a velocity field suitable to compress the interface [22]. The third term in Eq. (1) is a compression term that helps to maintain the interface sharp and bounded between zero and unity (see Section 4.2.1 of the work [22] for detailed formulae calculating the compression term). The air-water mixture is considered as homogeneous in the present work, therefore the momentum equation for the flow is given by

$$\frac{\partial \rho \mathbf{U}}{\partial t} + \nabla \cdot (\rho \mathbf{U}\mathbf{U}) = \nabla \cdot (\mu \nabla \mathbf{U}) - \mathbf{g} \cdot \mathbf{x} \nabla \rho - \nabla p_{\text{d}}, \tag{3}$$

where $\rho = \alpha \rho_{\text{water}} + (1 - \alpha) \rho_{\text{air}}$ is the density of the mixture (note that the sum over the volume fractions of water and air is equal to unity), $p_{\text{d}} = p - \rho \mathbf{g} \cdot \mathbf{x}$ is the dynamic pressure. Furthermore, the mixture viscosity is given by $\mu = \alpha \mu_{\text{water}} + (1 - \alpha) \mu_{\text{air}}$, $\mathbf{g}$ and $\mathbf{x}$ are the gravity and the position vectors, respectively. Heat and mass transfer terms are not considered in the present work, hence they are not included in Eq. (3).

The governing equations (1)–(3) are linearised and integrated over each control volume to determine α and $\mathbf{U}$, respectively, and a pressure corrector linearised equation is solved for p_{d}. This solution procedure relies on the segregated projection algorithm PIMPLE [23], derived from the PISO procedure [24], which allows for equation under-relaxation to guarantee convergence of the solutions at each time step.

5

2.2. Overlapping meshes

Figure 1 illustrates the essential idea of overset mesh methods. Let us suppose that we have a one-dimensional domain of [0, 1] with Dirichlet boundary conditions at the left and right sides. If we split the big domain into two sub-domains A and B, then the overall solution on the whole domain could be obtained by solving the problem in each sub-domain. In the sub-domain A, the solution is fixed at the left end. However the value at its right side (point 6), which needs to be used in computation, is unknown. Similarly, the value at the left side of domain B (point 7) is also needed, but it is not known either. In order to obtain the values at points 6 and 7, which are usually named *fringe nodes*, a proper interpolation of the surrounding values must be applied. For point 6, several donor nodes/cells should be selected from sub-domain B. Here we can choose points 7, 8, 9 and 10 as its donors and suppose values ϕ_i ($i = 7, \ldots, 10$) at these points are already known, then an appropriate explicit interpolation formula given by

$$\phi_6^{n+1} = (\beta_7 \phi_7^n + \beta_8 \phi_8^n + \beta_9 \phi_9^n + \beta_{10}^n \phi_{10}^n)/\beta_6 \tag{4}$$

is used to obtain the information for point 6. In the formula (4), β_i ($i = 6, \ldots, 10$) are interpolation coefficients; the superscripts n and $n + 1$ represent the current and next time steps, respectively. Besides this option, we can also derive an implicit interpolation equation given by

$$\beta_6 \phi_6^{n+1} - \beta_7 \phi_7^{n+1} - \beta_8 \phi_8^{n+1} - \beta_9 \phi_9^{n+1} - \beta_{10} \phi_{10}^{n+1} = 0 \tag{5}$$

where the values ϕ at the donor points 7, 8, 9 and 10 are supposed unknown, and assemble it with the flow field equation. Hence the information at point 6 is implicitly obtained by solving the interpolation and field equations together. Similar strategies can be used to deal with point 7. In the present work, the explicit interpolation formula (4) is utilised to handle all the fringe nodes. The formula could be executed multiple times along with the prediction and correction steps in a time step.
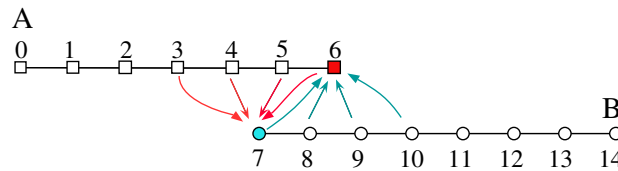

Figure 1: The overlapping of two 1D meshes in a domain. Nodes 6 and 7 are usually named fringe points.

Although the problem presented in figure 1 is relatively simple as there is no structure, the underlying idea is generic and suitable for complex problems with either stationary or moving bodies such as the water entry of a wedge.
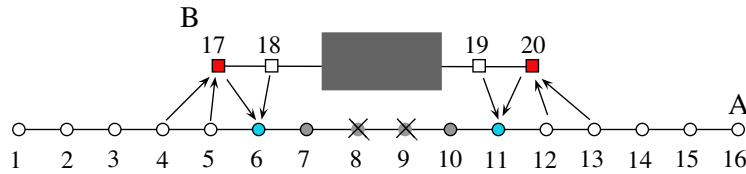
6

Figure 2: Hole cutting of two overlapping meshes in a 1D domain. A body-fitted mesh is used in sub-domain B to deal with the structure (grey rectangular). The crossed grey points on sub-domain A are hole nodes/cells, their adjacent neighbouring (grey) points could also be excluded from computation.

When a structure is present in any sub-region of the whole domain, extra work, namely *hole cutting*, needs to be done to mark and block the mesh points/cells locating in or adjacent to the structure. Figure 2 shows an example of this

135 situation. Firstly a background mesh A is generated to overlay the whole domain without considering the structure. Then a body-fitted small mesh B is generated to deal with the structure (represented by the grey rectangle in the figure). Before carrying out any computation of the flow problem, all the points/cells of mesh A need to be scanned to mark and blank out all the elements located in the structure. Points 8 and 9 are in the structure, therefore they are marked as hole points. Their neighbouring nodes 7 and 10 could also be blanked out to reduce the number of points

140 taking part in computation. In the computer program, these invalid hole points/cells do not need to be removed from memory, but a flag should be set for every point to indicate whether it is valid, invalid (in or close to the structure), or on the fringe that needs interpolation. For mesh B, we can choose points 17 and 20 as fringe nodes. For mesh A, the valid points adjacent to the hole cells could be selected as fringe points, hence points 6 and 11 are chosen. Please note that for the purpose of simplicity, we only use one fringe layer for the examples shown in Figure 1 and 2. In real

145 applications, more layers can be used to ensure a smooth transition of flow information between different regions.

Following the hole cutting procedure, attention should be paid to the *domain connectivity information* (DCI). For every fringe point/cell on a mesh, a donor stencil from another mesh should be selected to aid the interpolation procedure. If we look at the example in Figure 2, for point 17 of mesh B, we can choose nodes 4 and 5 of mesh A as its donor stencil; for point 20, nodes 12 and 13 could be used as the donor stencil. For fringe point 6 of mesh A, we

150 can choose points 17 and 18 of mesh B as its donors. And we can use points 19 and 20 as the donors of node 11. This important information should be calculated and stored before solving the flow problem.

For stationary boundary problems, hole cutting and DCI need to be processed only once and stored in the computer memory before solving the physical problem. For moving boundaries, these steps need to be repeated during flow calculation. Figure 3 presents the whole procedure of the developed overset multiphase flow solver. It is very similar

155 to a routine CFD program, the difference lies in the inclusion of hole cutting, DCI and interpolation.
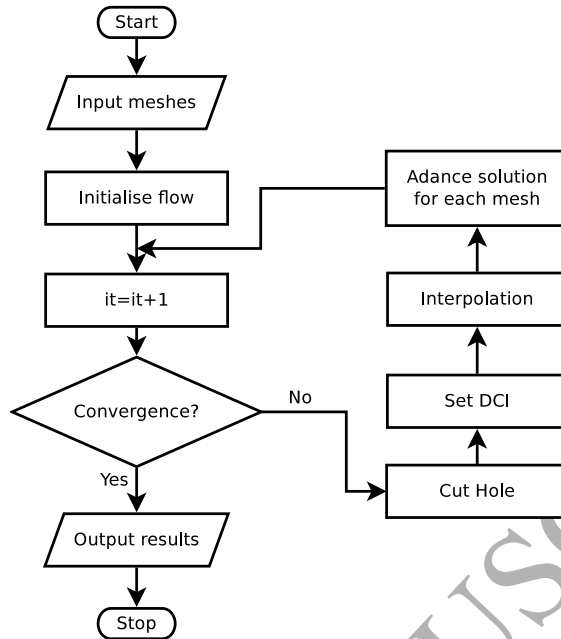
7

Figure 3: The flow chart of an overset multiphase flow solver. DCI: domain connectivity information.

3. Validation

The collapse of a water column, for which all the solid wall boundaries are stationary, is firstly considered to test the developed code. The configurations of the 2D and 3D problems are illustrated in Figure 4 and 5, respectively.

3.1. Collapse of a 2D water column in an empty tank

As shown in the left part of Figure 4, the length of the water tank is 0.5 m, the height is 0.15 m. The water column's height and width are 0.12 m and 0.06 m, respectively. The left, right and bottom sides of the water tank are
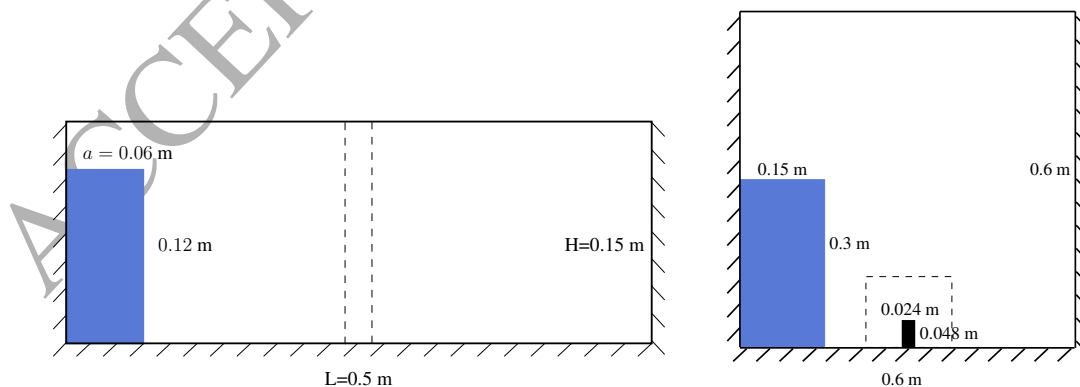
Figure 4: Initial configurations of the 2D water column breaking up without (left) and with (right) a rectangular obstacle. The dashed lines represent the boundaries of the sub-domains.

8
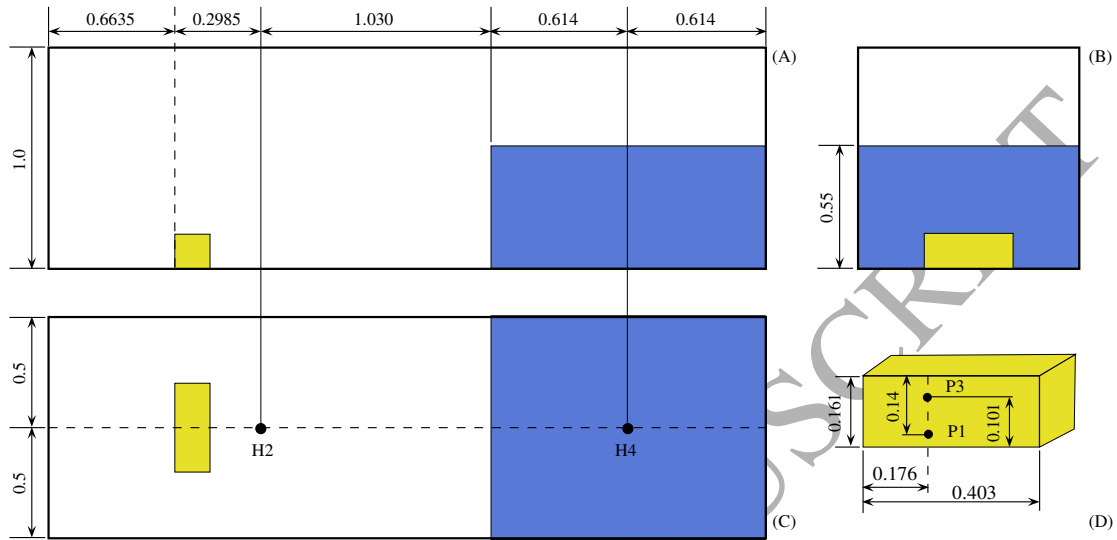
Figure 5: Setup of the 3D dam break with an obstacle (units in metres). (A) Frontal view, (B) Left view, (C) Top view, (D) The obstacle and pressure sensors. H2 and H4 are wave gauges. P1 and P3 are pressure transducers installed on the vertical surface of the obstacle facing the water column.
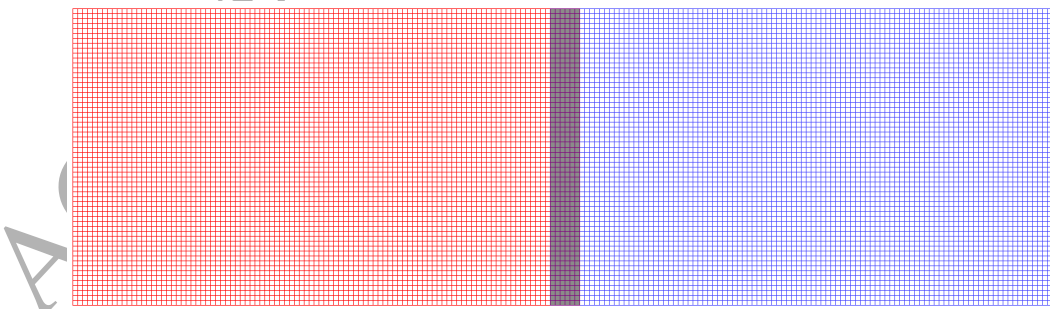

Figure 6: Overlapping meshes for the 2D water column breaking up in an empty tank. The grey area represents the fringe cells.
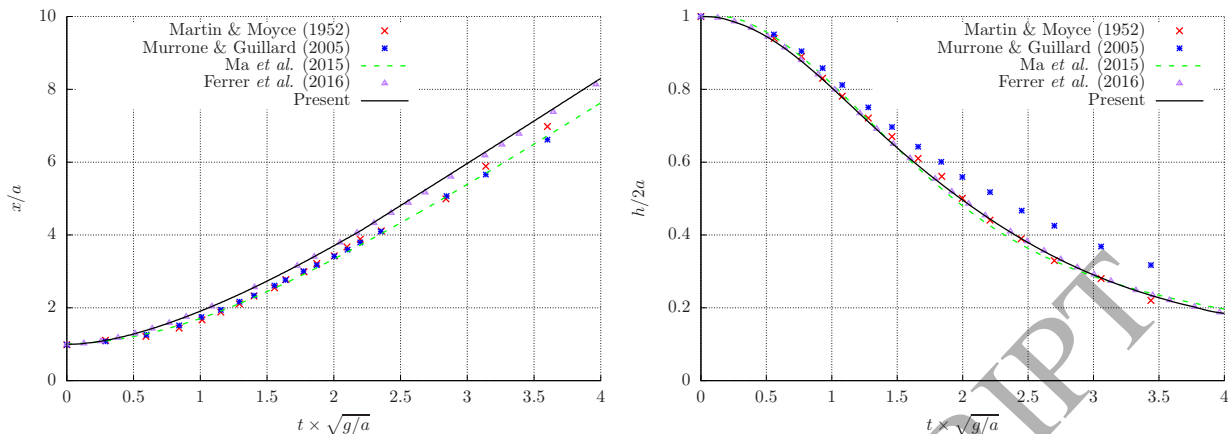
9

Figure 7: Dam break without object. Left: water front position; Right: water height.

rigid walls. The top side of the tank is an open boundary. At the initial state, a barrier is placed at the right side of the water column to keep it still. The water starts to collapse when the barrier is suddenly removed.

Due to the simple geometry of this problem, the whole domain is divided into two halves in the horizontal direction (near the middle). Two uniform meshes with a cell size of 2.5 mm × 2.5 mm are generated for the left and right parts. Figure 6 shows the overlapping meshes for this problem. In the vertical direction, both meshes have 60 cells. In the horizontal direction, the left sub-domain spans from 0 to 0.2575, the right sub-domain spans from 0.2425 to the right end of the water tank. Both parts have 103 cells in the horizontal direction; thus each sub-domain has three layers of fringe cells.

Two important quantities, the position of the water front along the tank bottom and the height of the water column along the left wall, are monitored during the simulation. Figure 7 gives a comparison of the present computed result with the experimental measurements of Martin and Moyce [25] and other reference calculations. Please note that all the data presented in Figure 7 are dimensionless. For the water front position, the present results agree well with the solution of Martínez-Ferrer *et al.* [1], while both are slightly above the experimental data and other calculations. For the water column height, the present solution agrees well with the experimental measurements and all the reference numerical computations [1, 26] except the one calculated by Murrone and Guillard [27], which shows a large discrepancy compared to the other results.

3.2. Collapse of a 2D water column against an obstacle

The configuration of this problem is depicted in the right part of Figure 4. A square water tank has a side length of 0.6 m with a rectangular obstacle installed at the middle of its bottom wall. The width of obstacle is 0.024 m and the height is 0.048 m. The water column has a width of 0.15 m and a height of 0.3 m. The top side of the tank is an open boundary and all the other boundaries are rigid walls. Firstly a uniform mesh with 100 × 100 cells is generated
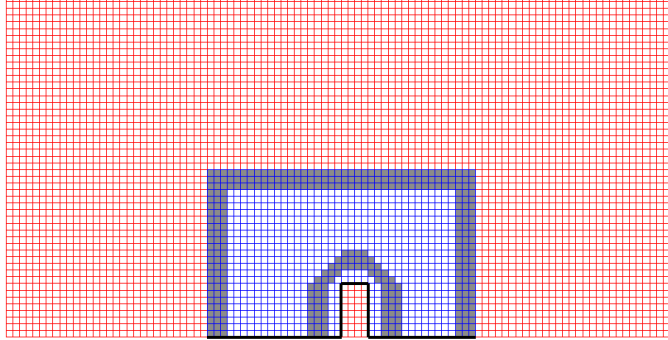
10

Figure 8: Overlapping meshes for the 2D water column breaking up against a rectangular obstacle. Red: uniform background mesh; Blue: body-fitted mesh around the obstacle. Only the lower half of the background mesh is shown here, the upper half is not shown due its simplicity. The grey areas represent the fringe cell regions.

to discretise the whole domain without considering the small obstacle. Then a body-fitted uniform mesh covering the region $[0.18, 0.42] \times [0, 0.15]$ (units in m^2), of which the element size is 6 mm $\times$ 6 mm, is created to deal with the obstacle. For this test case, there is no specific requirement on the size of the sub-domain, hence it is chosen for test purposes only. It actually could be reduced (or increased) as long as the generated mesh can provide proper fringe cells (about three layers in general) to exchange information between grids. Figure 8 shows the generated overlapping meshes. Figure 9 gives snapshots of the free surface at times 0.2, 0.4 and 0.5 seconds. It is shown that the present calculation compares well with the experiments of Martin and Moyce [25] regarding the main features of the collapsing water column.

3.3. Collapse of a 3D water column against an obstacle

Figure 5 shows the set up of this 3D problem. The length of the water tank is 3.22 m, its cross section is a square with side length of 1 m. A cuboid obstacle is installed on the bottom wall of the tank. The length of the obstacle is 0.403 m, its cross section is a square with side length of 0.161 m. The water column is placed in the right part of the tank. The length of the water column is 1.228 m, its height is 0.55 m and its width is 1 m. Two wave gauges H2 and H4 are installed in the tank to measure the free surface elevation. Two pressure transducers P1 and P3 are installed on the obstacle's surface that faces towards the water column.

A background mesh with $267 \times 79 \times 71$ cells is generated to cover the whole domain. The background mesh is stretched towards the bottom wall and the obstacle. A body-fitted mesh with 110, 466 cells is created to handle the obstacle. Figure 10 shows the body-fitted volume mesh around the small cuboid. Several snapshots of the computed free surface at different times are presented in Figure 11. At $t = 0$, the water is still and the barrier holding the water is suddenly removed causing the water to collapse. At $t = 0.4$ s, the water front approaches the small obstacle closely. At $t = 0.6$ s, the water hits the right side of the obstacle causing a strong upward splash, and the water also flows
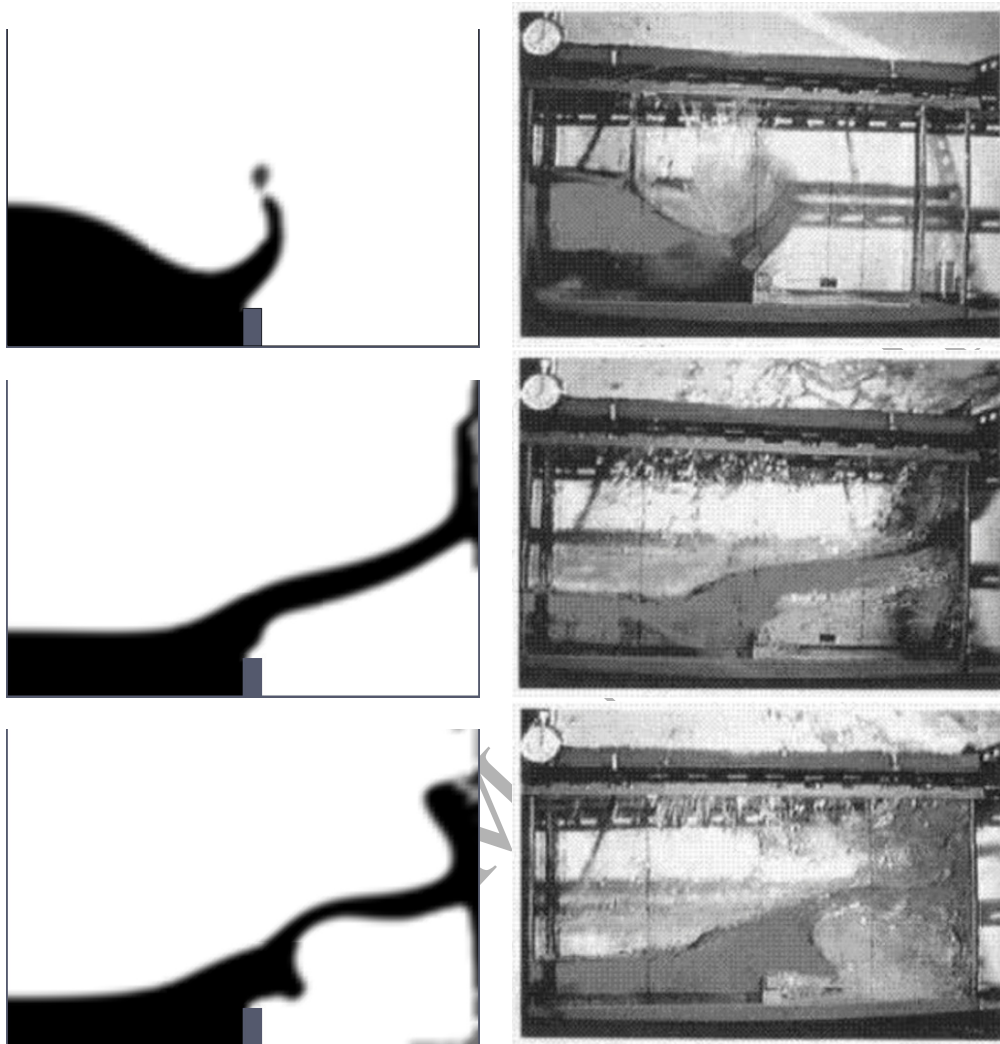
11

Figure 9: Collapsing water column hitting an obstacle. Left: present calculation; Right: experiment of Martin & Moyce (1952). From the top to the bottom, $t = 0.2$, 0.4 and 0.5 s.
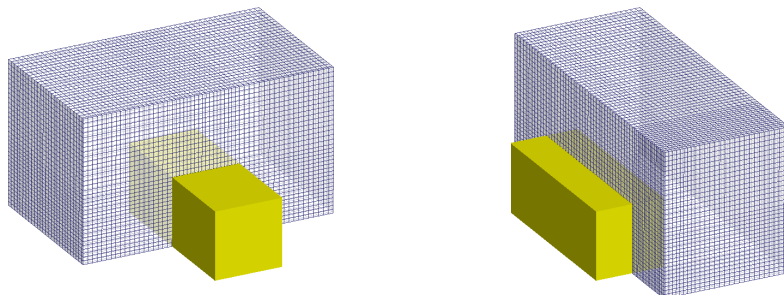

Figure 10: Volume mesh around the obstacle.

through the gaps between the obstacles and the side walls of the tank. The water level in the right part of the tank

²⁰⁵ continues decreasing whilst in the left part it keeps increasing. At $t = 2$ s, the obstacle is fully submerged in the water.

A comparison of the water heights at H2 and H4 computed by the present method and obtained by other researchers is shown in Figure 12. For the wave gauge H2, all the results show two peaks from 1.5 s to 2.5 s. The measured peaks are lower than all the numerical computations. For the wave gauge H4, the measured peak at around $t = 3.8$ s is larger than all the numerical solutions. In general the present results compare well with the experimental data and other

²¹⁰ reference computations.

Figure 13 shows the pressures obtained at P1 and P3. At P1, the measured peak obtained by Kleefsman *et al.* [28] is around 12 kPa. They also carried out simulations by using the ComFLOW code and obtained a peak value about 11.2 kPa. The peak pressure obtained by the present work is around 11.6 kPa, which lies between the measurements and computations of Kleefsman *et al.* [28]. The numerical solution produced by Zhang *et al.* [13], which was calculated

²¹⁵ by using a Lagrangian particle method, is very far from these results; they obtained a peak value of about 28.5 kPa at P1, which is more than twice of the measured peak. Due to the large discrepancy, their result is not fully shown in the figure; please refer to their recent publication for details (see Fig. 17 of [13]). Also, their computational results are very oscillatory near the pressure peak. After the peak, the pressure drops quickly. The experiment of Kleefsman *et al.* [28] and the computation of Zhang *et al.* [13] show oscillatory spikes superimposed on the curves.

²²⁰ The simulation of Kleefsman *et al.* [28] and the present results are relatively smooth after the peak. Surprisingly, the ComFLOW code picks up a relatively large pressure peak of about 6.4 kPa at around $t = 1.3$ s, not found in the experimental measurements or reproduced by other numerical simulations.

For the first pressure peak at P3, the measurement of Kleefsman *et al.* [28] is about 6.5 kPa whilst their computation gives a lower prediction of 4 kPa; the present work gives about 4.7 kPa, and Zhang *et al.* [13] produce the largest

²²⁵ value of 8.7 kPa. At $t = 1.32$ s, the ComFLOW code picks up another pressure peak of 8.5 kPa, which is not captured by other calculations and the experiment. At around $t = 4.8$ s, the pressure peak obtained by the ComFLOW code is almost twice that in the experimental measurement and other computations. Regarding the overall trends and peak values of the pressures, the present solutions can be seen to compare well with other reference results.

4. Water entry

²³⁰ Ship slamming is an important topic in ocean and naval engineering. For frontal and bottom slamming of ships, 2D wedges and curved hull sections are widely used in experimental studies to investigate the hydrodynamic impacts on the structure surface. This is much simpler, cheaper and quicker than 3D model tests, in which full-size ships are scaled down to manufacture the complex and expensive models. Since the objective of the present work is to
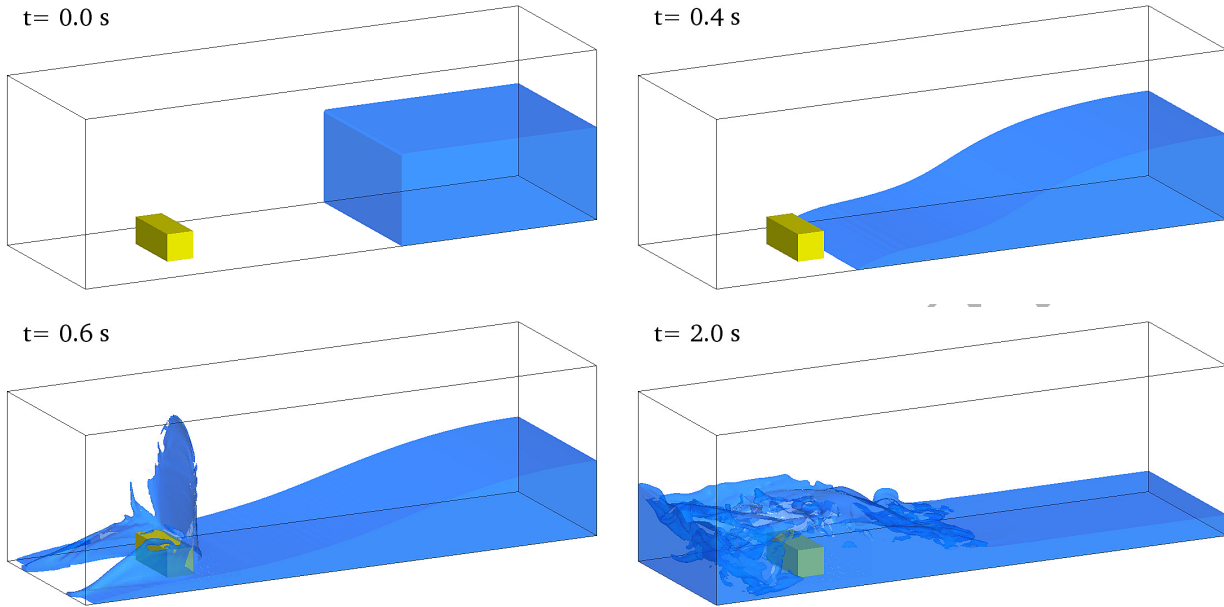
13

Figure 11: Snapshots of the computed free surface in the tank at different times.
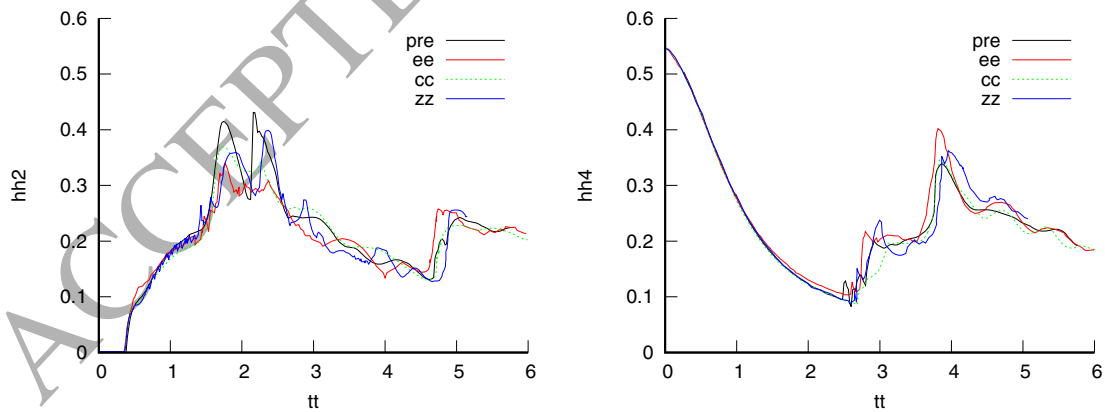

Figure 12: Vertical water heights in the tank at H2 (left) and H4 (right). Black lines: present results; Red lines: experiments of Kleefsman *et al.* [28]; Green lines: computations of Kleefsman *et al.* [28]; Blue lines: calculations of Zhang *et al.* [13].
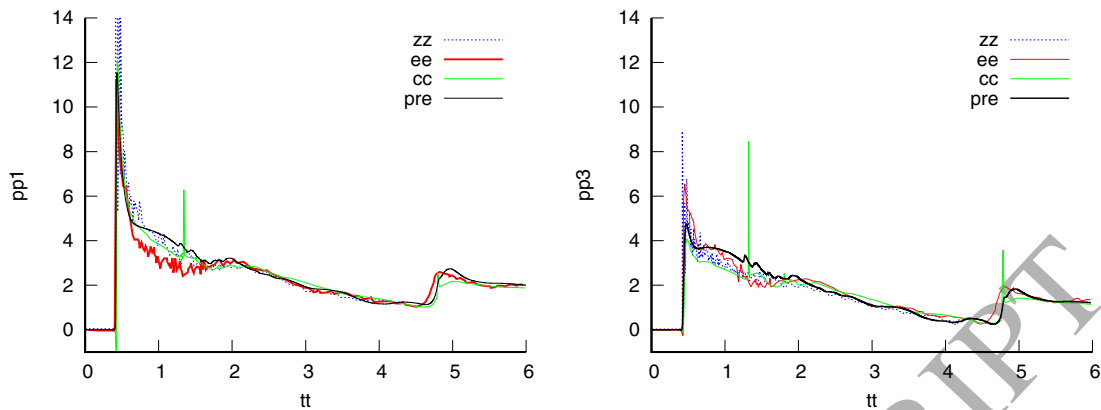
14

Figure 13: Pressure time histories at P1 (left) and P3 (right). Black lines: present results; Red lines: experiments of Kleefsman *et al.* [28]; Green lines: computations of Kleefsman *et al.* [28]; Blue lines: calculations of Zhang *et al.* using a SPH method [13]. The peak value at P1 obtained by Zhang *et al.* [13] is about 28.5 kPa, and it is not full shown here due to its large discrepancy compared to other results.
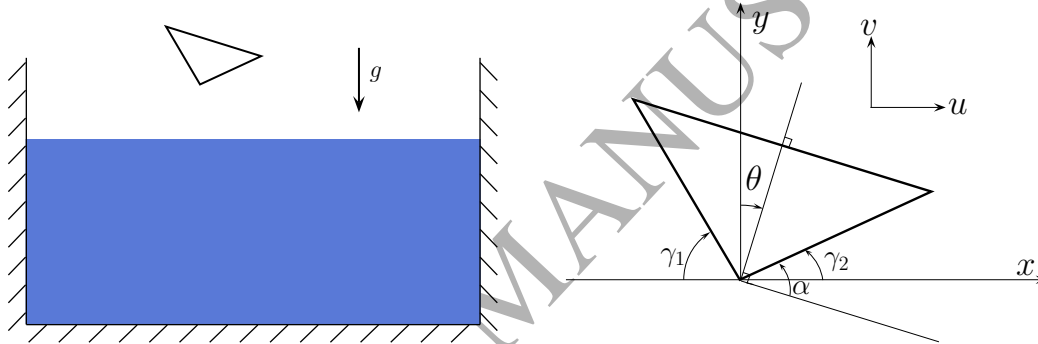

Figure 14: Water entry problem. Left: global view of the experimental set up; Right: close up view of of a 2D wedge with titling angle θ.

develop a numerical tool to predict the hydrodynamic loadings on free-fall lifeboats, at this stage, attention is focused

235 on examining the developed method for wedge and hull section slamming problems.

Figure 14 shows the set up of a water entry problem. The left part gives a global view of the water tank and the structure of interest, which could be a wedge, hull section or other bodies. The right part gives a close up view of a wedge, $\alpha = (\gamma_1 + \gamma_2)/2$ and θ are usually called the deadrise and titling angles, respectively. The assessment of the hydro-code is carried out in three incremental steps: 1) prescribed vertical entry, 2) prescribed oblique entry and 3)

240 free fall. Steps 1 and 2 provide idealised conditions for slamming problems, where the entry velocities of structures are fully controlled as prescribed. In step 3, structures are held still at first and then dropped into water freely.

4.1. Prescribed vertical entry

For the first group of slamming tests, two symmetric wedges with deadrise angles of 30° and 45° are considered. Initially the wedges are held still with the tips placed at the free surface. They are then driven into the water with a
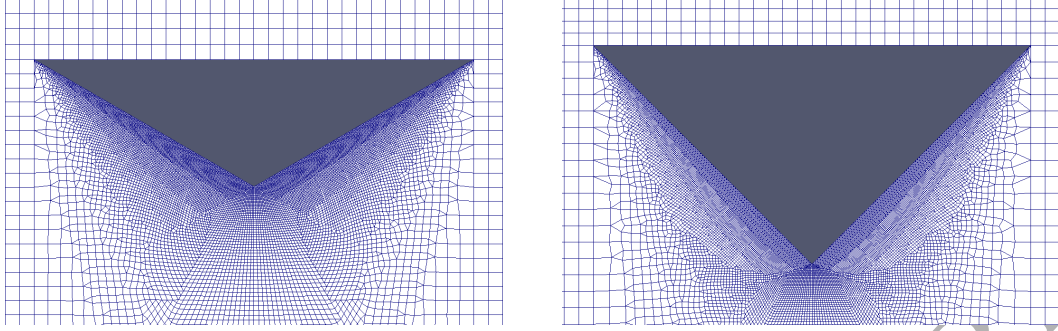
15

Figure 15: Body-fitted meshes around the symmetric wedges. Left: $\alpha = 30°$, Right: $\alpha = 45°$.

245 constant velocity (0 m/s, -1 m/s). A relatively large region of $[-8, 8] \times [-8, 8]$ (units in m^2) is chosen as the background domain. A uniform mesh with 200×200 cells is used to discretise the whole flow field for both cases. Both wedges' top sides have a length of 2.4 m. A region of $[-2.4, 2.4] \times [-2.4, 3.2]$ (units in m^2) enclosing the structure is chosen to generate the small meshes at the initial state. Please note that the domain sizes are chosen only for test purposes; they could be either increased or reduced if required. Figure 15 gives close up views of the generated body-fitted meshes

250 around the two wedges. Near the bottom surfaces of the wedges, small mesh elements are used in order to capture the slamming pressures accurately. For the wedge with a deadrise angle of 30°, both of its left and right bottom surfaces are discretised with 278 mesh cells. The small mesh has 40,550 cells for this wedge. For the wedge with $\alpha = 45°$, both of its left and right bottom surfaces are divided into 340 segments. The small mesh has 45,804 cells for this wedge.

255 Figure 16 exhibits the computed dynamic pressures around the wedges at $t = 0.2$ s. Comparing the left and right parts of the figure, it is shown that highest pressures appear in different places for these two wedges. For the wedge with $\alpha = 30°$, the highest pressures (two red spots) occur symmetrically about its central section and they are away from the tip. For the wedge with $\alpha = 45°$, the highest pressure (one red spot) focuses at the tip.

Figure 17 shows the pressure coefficients on the surfaces of these two wedges. The solid lines are the similarity

260 solutions obtained by Dobrovol'skaya [29]. The dashed lines represent the analytical solutions produced by Mei *et al.* [30]. The dots are the present results computed at 0.1, 0.12 and 0.15 seconds. The dimensionless forms of these calculations at different times are very similar to each other, and all of them compare well with the similarity solution. This confirms that impact pressures at different times are self-similar for a short period, when the inertia is dominant compared to the gravitational effect [30, 31]. Free surface profiles computed at 0.1, 0.12 and 0.15 seconds are plotted

265 in Figure 18. The dimensionless forms are also self-similar to each other within a short impact period for both wedges.
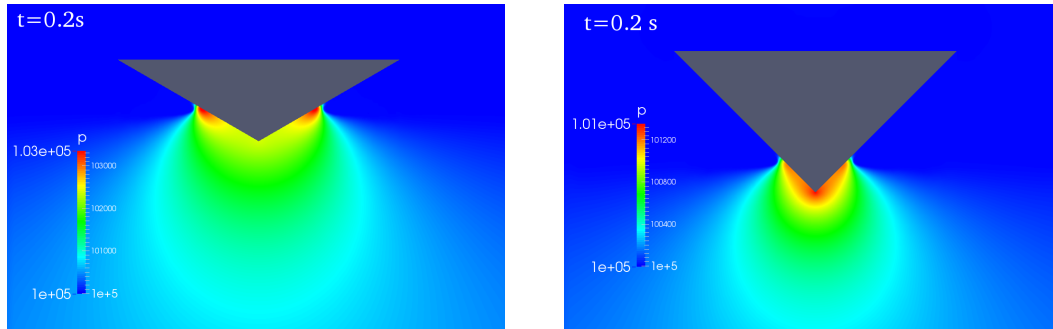
16

Figure 16: Computed pressures at t = 0.2 s. Left: $\alpha = 30°$, Right: $\alpha = 45°$.
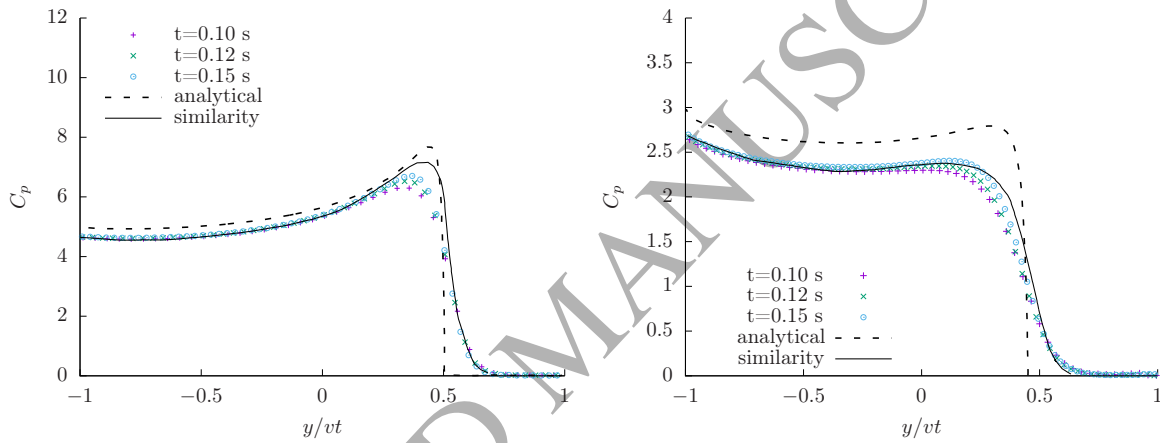

Figure 17: Pressure coefficients on the wedges, $C_p = (p - p_0)/(0.5\rho v^2)$, p_0 = 1 bar. Left: $\alpha = 30°$, Right: $\alpha = 45°$. Dashed line: similarity solution of Dobrovol'skaya [29]; Solid line: analytical solution of Mei *et al.* [30]; Dot: present result.
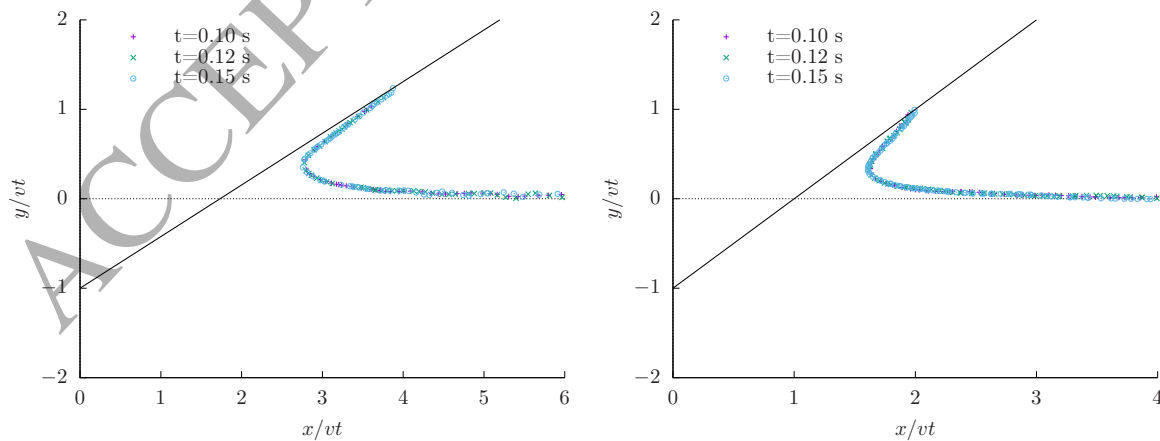

Figure 18: Non-dimensional free surface profiles. Left: $\alpha = 30°$, Right: $\alpha = 45°$.

17

4.2. Prescribed oblique entry

Here the oblique entry of a symmetric wedge and an asymmetric wedge, where the structures have non-zero horizontal velocity u, is considered. The symmetric one has a deadrise angle of $\alpha = 45°$ with zero titling angle, thus $\gamma_1 = \gamma_2 = 45°$. For the asymmetric wedge, $\gamma_1 = 20°$ and $\gamma_2 = 40°$. In the calculations, the vertical entry velocity is

270 set to $v = -1$ m/s, while the horizontal velocity u is set to different values for each case.

For the symmetric wedge, the same mesh shown in the right part of Figure 15 is utilised here. Figure 19 illustrates the results obtained for different horizontal velocities. The ratio $u/|v|$ equals 0, 0.1, 0.3 and 0.5 from the top row to the bottom. The left column represents the pressure coefficients on the wedge surface, and the right part gives the free surface profiles. Please note that all the values here are dimensionless. The solid lines in the figure were

275 calculated by Xu *et al.* [32]. The coloured dots are the present results computed at 0.1, 0.11 and 0.12 seconds. The free surface profiles obtained in the present work agree well with the reference results of Xu *et al.* [32]. For the small velocity ratios 0 and 0.1, the present pressure solutions agree quite well with the work of Xu *et al.* The differences between them grow with the velocity ratio. For $u/|v| = 0.3$ and 0.5, large discrepancies can be seen around the tip of the wedge. This is due to the fact that Xu *et al.* [32] used a potential flow code, which does not account for fluid

280 viscosity and vorticity, to solve the slamming problem. However, the sharp tip of the wedge can induce relatively large vorticity when moving in the horizontal direction. This effect can not be properly handled by potential flow theory [33]. Nevertheless, in general, both the potential flow code and the present two-phase viscous code produce quite close results except in the region adjacent to the wedge tip.

Water entry of the asymmetric wedge is further considered here. The top left of Figure 20 shows the body-

285 fitted mesh around the structure. Before increasing the horizontal velocity, a pure vertical entry problem is solved. The top right of the figure exhibits the instantaneous dynamic pressure at $t = 0.15$ s. The bottom left depicts the non-dimensional free surface profiles, and it is clearly seen that the present calculations are self-similar and agree well with the solution of Xu *et al.* [32]. The bottom right part of the figure exhibits the pressure coefficients on the wedge surface, in which the present solutions agree well with other reference results. It seems that Gu *et al.* [12]

290 over-predicted the pressure peak, and their computation is oscillatory along the wedge surface due to the use of an immersed boundary (partial cell) treatment of the solid object.

Figure 21 shows the pressure coefficients on the lower surfaces of the wedge computed for different horizontal velocities. It is clearly shown that the present results agree well with the potential flow solutions of Xu *et al.* [32]. Though the calculation of Gu *et al.* [12] is generally acceptable, the produced pressure distribution along the wedge

295 surface is rather oscillatory. Comparing the left and right columns of the figure, it is shown that the tip of the wedge has more significant effect in reducing the pressure for forward movements ($u > 0$) than backward motions ($u < 0$).

18
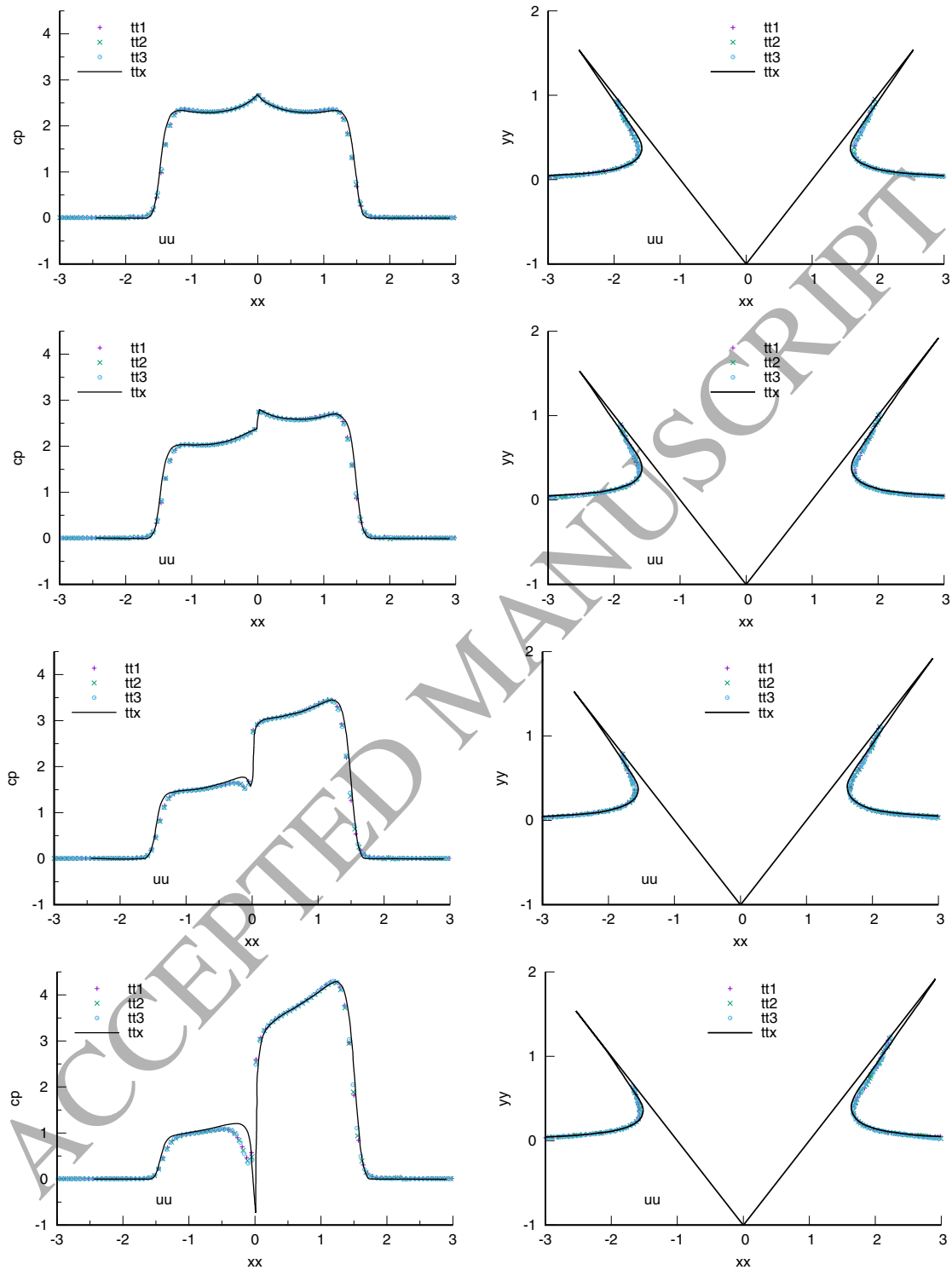
Figure 19: Vertical and oblique entry of a symmetric wedge, $\gamma_1 = \gamma_2 = 45^o$. Left: dimensionless pressure $C_p = (p - p_0)/(0.5\rho v^2)$, $p_0 = 1$ bar; Right: dimensionless free surface profile.
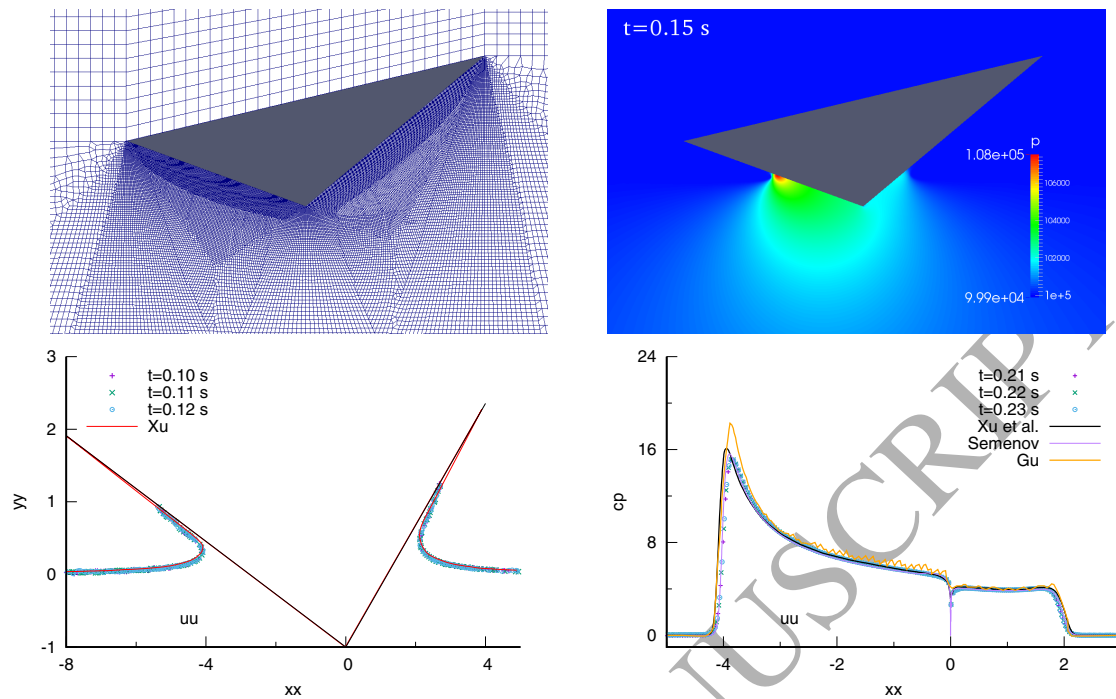
19

Figure 20: Water entry of an asymmetric wedge, $\gamma_1 = 20^o, \gamma_2 = 40^o$. Top left: mesh around the structure; Top right: pressure shaded contours; Bottom left: free surface profile; Bottom right: pressure coefficients on the wedge slamming surface, $C_p = (p - p_0)/(0.5\rho v^2)$, $p_0 = 1$ bar. The immersed boundary method used by Gu *et al.* [12] overpredicts the pressure peak and produces non-physical oscillations along the wedge surface.

The pressures at the tip are significantly lower for the former than the latter.

4.3. Free fall

The prescribed water entry tests carried out so far have demonstrated the effectiveness of the overset multiphase code in dealing with these problems to produce accurate prediction of the pressure loadings on structures. In the real practice of maritime operations, free fall is potentially a favourable option to quickly launch lifeboats. Consequently, it is necessary for numerical tools to be capable of handling these important problems.

In 2016, the International Society of Offshore and Polar Engineers (ISOPE) organised a collaborative experimental and numerical study of ship slamming problems. The Korea Research Institute of Ships & Ocean Engineering conducted the experiments and collected a set of important data. Figure 22 illustrates the experimental set up for the free fall tests of wedges and a hull section. On the right lower surface of the wedge, a pressure transducer is installed at a location 100 mm away from the tip. On the left lower surface of the wedge, a load cell is installed to measure the local force impinging on the small area. Please note that the load cell is not used to measure the total impact force on the structure.

The test conditions for the experiments are listed in Table 1. For the ship hull section, a load cell is installed on its left surface, three pressure sensors are installed on its right surface as shown in Figure 22. The tip of the hull
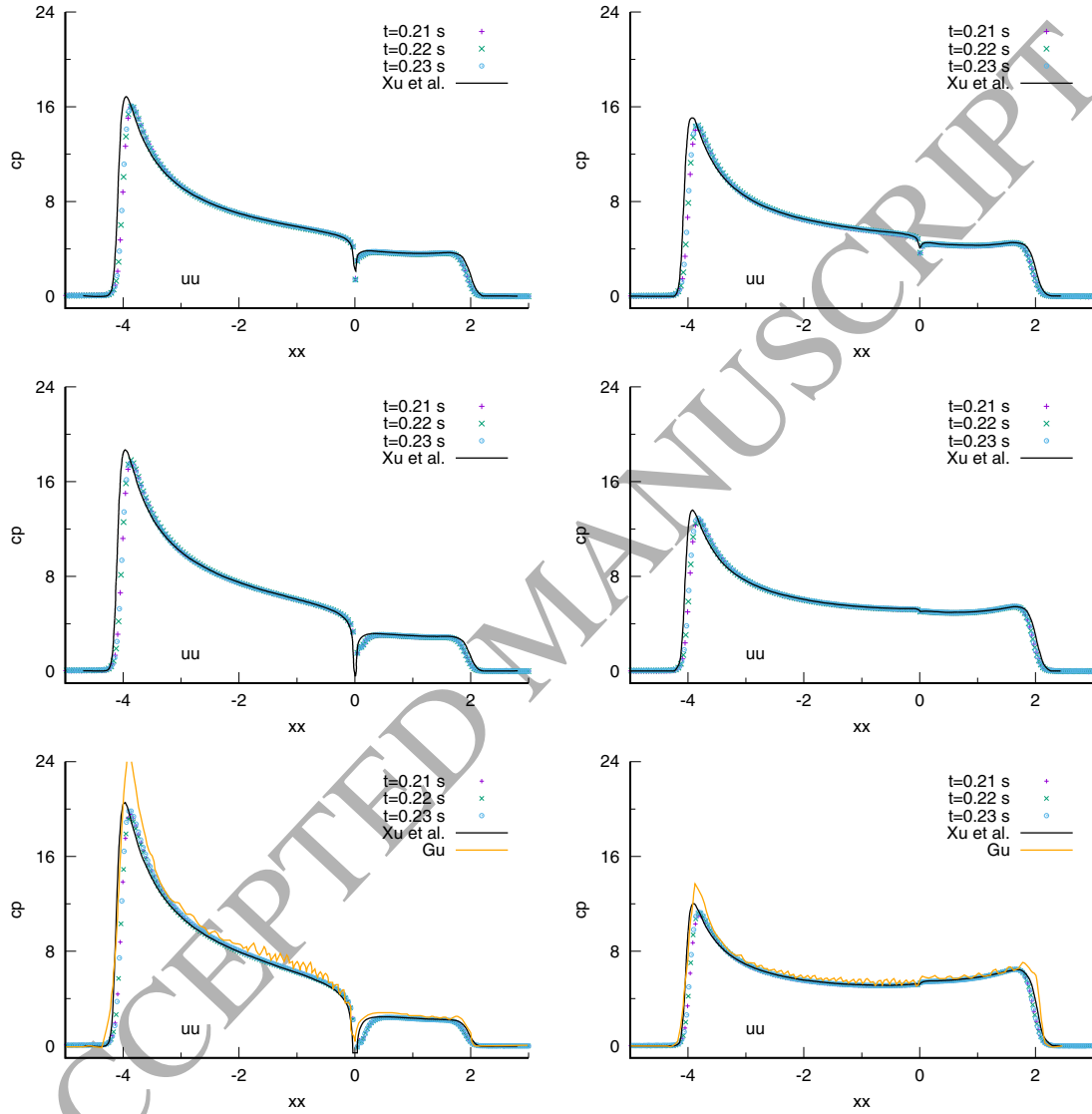
20

Figure 21: Pressure coefficients on the asymmetric wedge with $\gamma_1 = 20^o$ and $\gamma_2 = 40^o$. Left: forward movement with $u > 0$; Right: backward movement with $u < 0$. The vertical velocity is kept constant as -1 m/s. $C_p = (p - p_0)/(0.5\rho v^2)$, $p_0 = 1$. The immersed boundary method used by Gu *et al.* [12] overpredicts the pressure peaks and produces non-physical oscillations along the wedge surface.
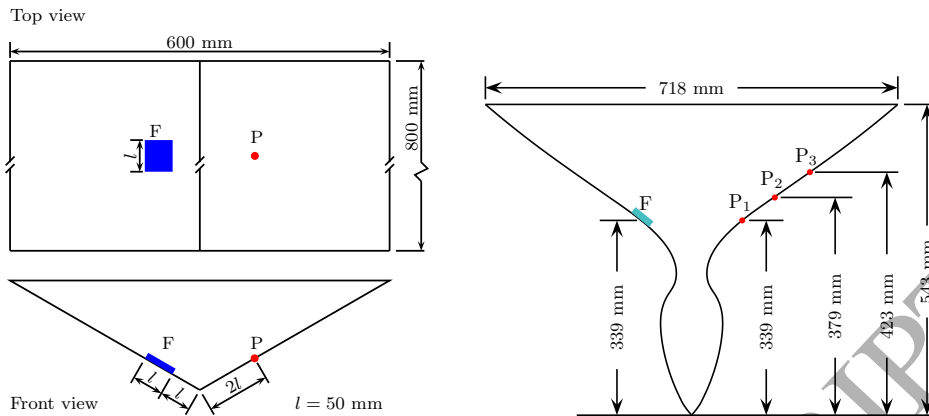
Figure 22: Setup of the free falling objects (Unit: mm). Left: wedge; Right: ship hull.

is initially placed at the free surface. The wedges have different deadrise and tilting angles as well as drop heights. The maximum drop distance is 0.5 m, and it is almost three times the height of the wedge with 30° deadrise angle. This relatively large motion is quite challenging for a single-block dynamic deforming mesh, because the mesh will

315 suffer from excessive stretching, compression and skewness that could cause the computation to diverge, for which an example is shown in Figure 23. Our early attempt to use deforming meshes to deal with free fall problems was unsuccessful for the above mentioned reasons. One way to circumvent the difficulty is to pre-deform the mesh in the upward direction, so that the deformation can reduce with the structure's downward movement to accommodate a large displacement. However this method is cumbersome and not practical for general water entry problems. On the

320 contrary, however, overlapping meshes can effectively deal with this issue.

For each case, we carry out a convergence study by running simulations on successively refined meshes. Figure 24 shows the coarse level body-fitted meshes around the dropping objects. The grids for the three wedges are generated by using the open source software Gmsh. The mesh for the hull section is generated by using the snappyHexMesh tool in OpenFOAM. Many other meshing tools are also applicable to generate grids, which could then be converted

325 to OpenFOAM meshes. Figure 25 presents snapshots of the free surface profiles at $t = 0.5$ s computed on the coarse mesh for each dropping object. The calculated as well as measured time series of point pressure, local force, acceleration and velocity of the wedges for the first three tests are displayed in Figure 26, 27 and 28. These figures demonstrate that satisfactory convergence has been achieved for the computations, and the calculated results agree well the experimental data.

330 Figure 29 presents the obtained results for the hull section. Convergence has also been achieved for local pressures, force and the displacement of the structure. The first spikes are well captured by the present work for the pressures and local force. For the acceleration, the first spike could only be captured on the finest mesh. After the first spike
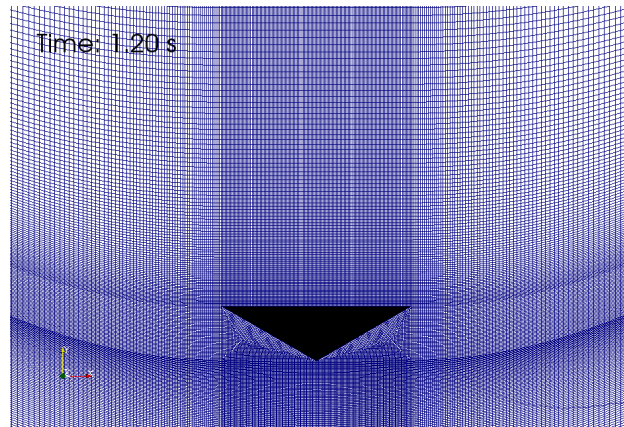
22

Figure 23: Dynamic deforming mesh for a free-fall wedge. The upper part mesh is stretched while the lower part is compressed.
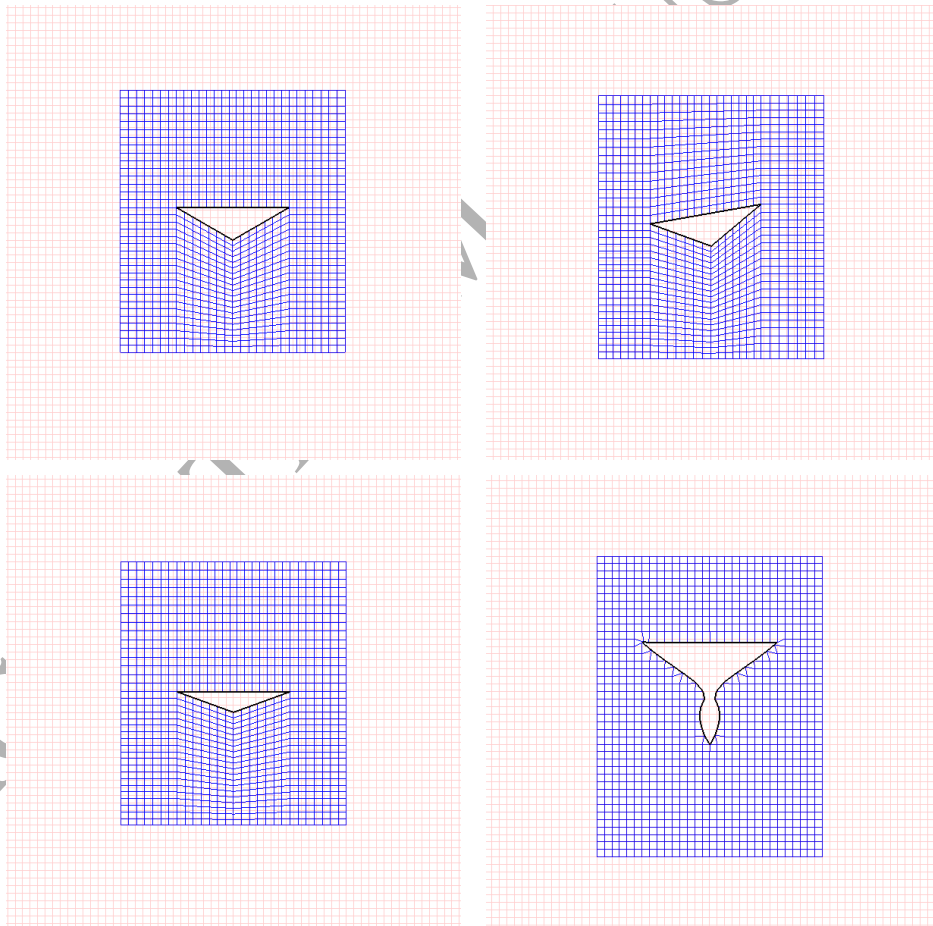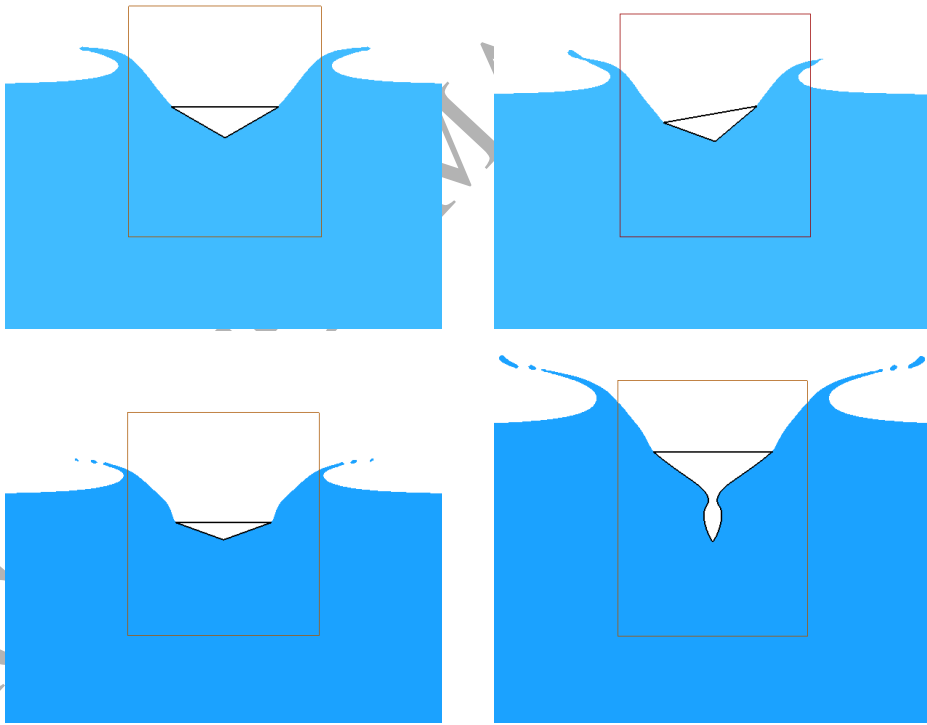

Figure 24: Overlapping meshes for the free falling objects. Light red: background meshes; Blue: body-fitted meshes. For clarity purpose, the meshes shown here have been coarsened in x and y directions.

Structure	Test	$\alpha(^o)$	$\theta(^o)$	Drop height (m)
	I	30	0	0.5
Wedge	II	30	10	0.5
	III	20	0	0.25
Hull	IV	–	–	0

Table 1: Experimental conditions for free fall tests.

Figure 25: Computed free surface profiles for the free falling objects at $t = 0.5$ s. The rectangles represent the boundaries of the small domains surrounding the objects.
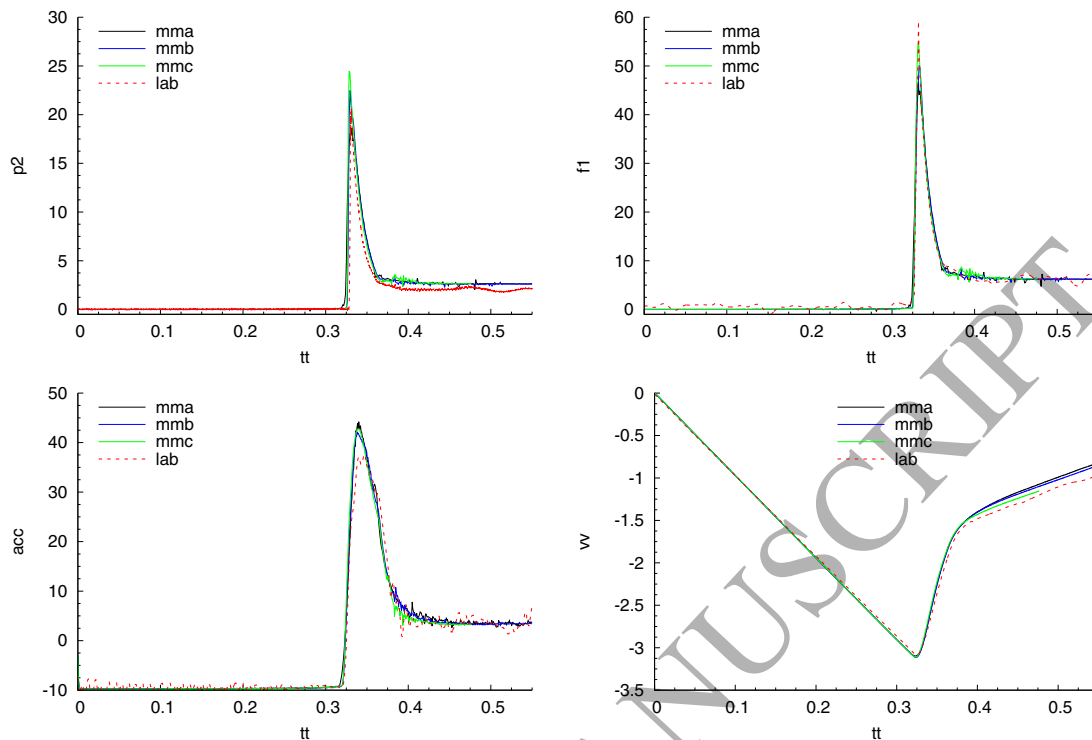
Figure 26: Computed and measured dynamics and kinematics for the free falling wedge 1. Successively refined meshes A, B and C are used in the simulation.

caused by the strong water impact, the computed and measured local loadings drop quickly following a similar trend. Disparity between the simulations and experiments occurs at about $t = 0.45$ s and afterwards. This is due to the use of

335   a spring in the laboratory to slow down and stop the structure in order to prevent it from impacting the tank bottom. Overall, the numerical simulations compare well with the experiments.

5. Conclusions

Numerical simulation of the free fall launch of a lifeboat from a large ship or an offshore platform remains a vital challenge for computational fluid dynamics. A traditional dynamic deforming mesh method can not properly

340   deal with the large-amplitude motions of the moving objects in these operations. Other methods like particle based or immersed boundary methods tend to produce non-physical oscillatory results and/or excessively over-predict the impact pressures. To overcome these difficulties, an overset mesh based multiphase flow model has been proposed in the present work to solve slamming problems in ocean and naval engineering. The developed code does not rely on third-party overset grid libraries but utilises the *wsiFoam* framework, which is based on the open-source

345   software OpenFOAM only, extending it to achieve the dynamic coupling of overlapping mesh regions. The solid wall boundary condition in multiphase flow problems, which is difficult to implement for both Lagrangian particle
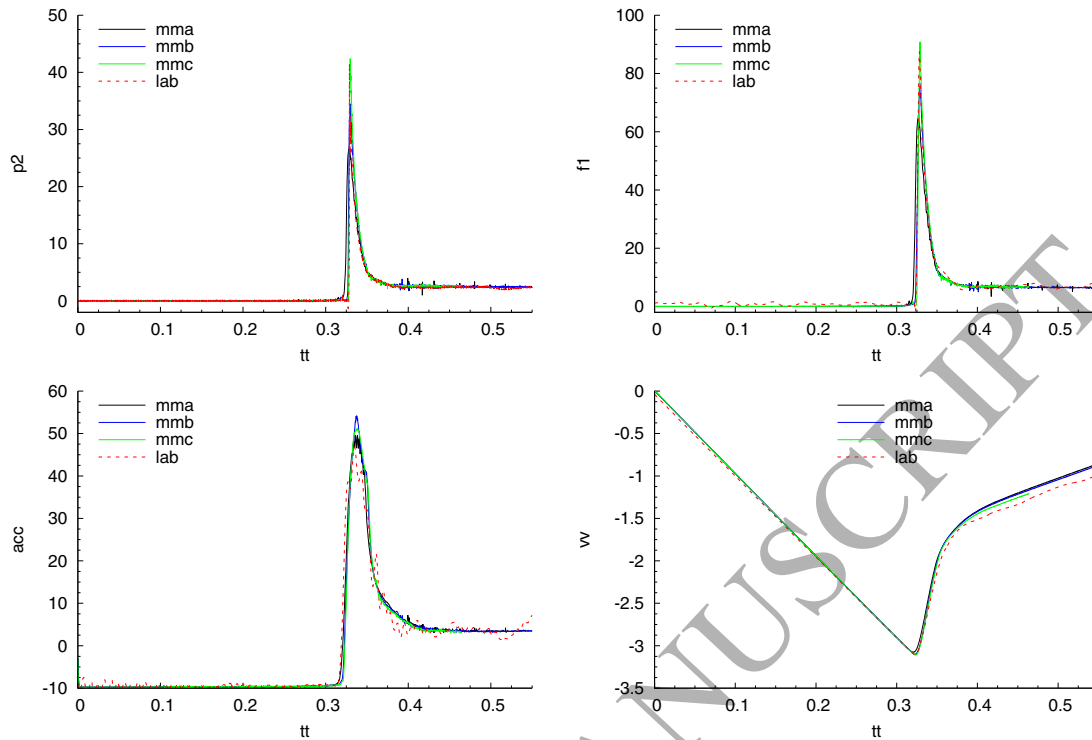
25

Figure 27: Computed and measured dynamics and kinematics for the free falling wedge 2. Successively refined meshes A, B and C are used in the simulation.

methods and immersed boundary methods, can be exactly satisfied using body-fitted overlapping meshes. A series of fixed and moving boundary problems including 2D and 3D dam breaks as well as prescribed and free-falling water entry of various wedges and a hull section have been solved by using the overset code. The obtained results agree well with experimental data and other computations reported in the literature. This demonstrates the effectiveness and accuracy of the developed method for these problems and also shows its potential to be extended to handle more complex problems. Simultaneously, complementary development of a fluid-structure-interaction code for wave slamming problems is under way and will be reported shortly. In future, the overset and structural codes will be coupled and parallelised on HPC facilities to carry out large-scale simulations of real 3D lifeboat launch operations under high sea states.
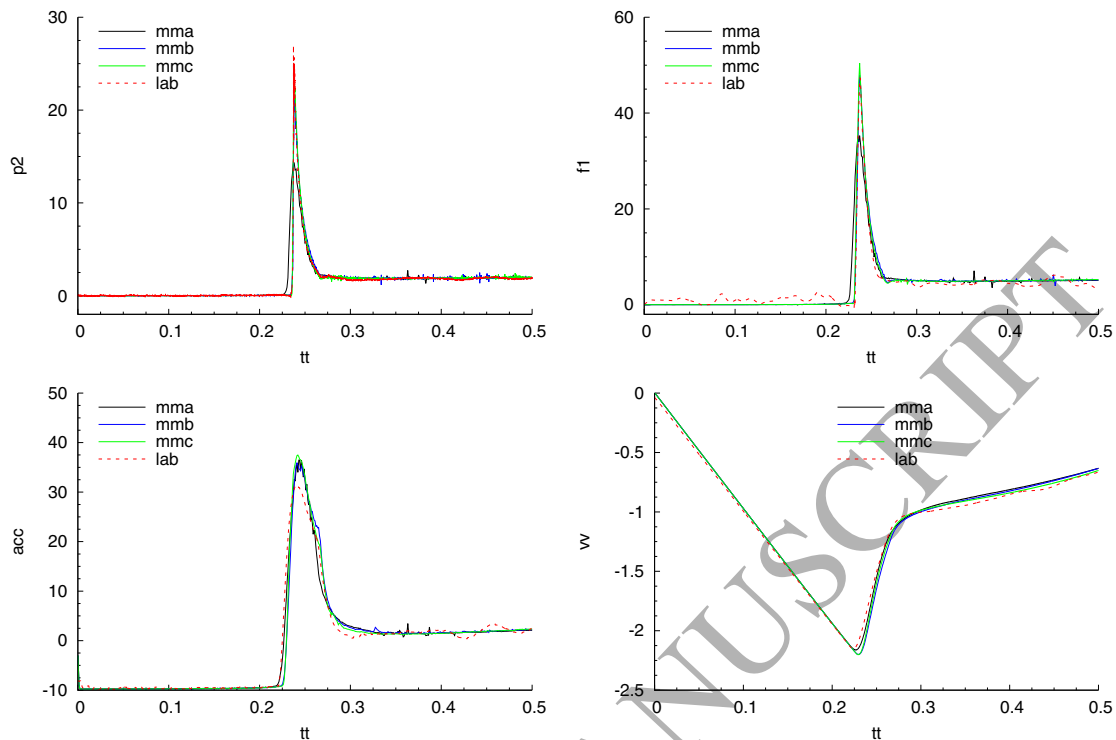
Acknowledgements

Figure 28: Computed and measured dynamics and kinematics for the free falling wedge 3. Successively refined meshes A, B and C are used in the simulation.

References

[1] P. J. Martínez-Ferrer, D. M. Causon, L. Qian, C. G. Mingham, Z. H. Ma, A multi-region coupling scheme for compressible and incompressible flow solvers for two-phase flow in a numerical wave tank, Computers and Fluids 125 (2016) 116–129. doi:10.1016/j.compfluid.2015.11.005.

[2] G.-d. Xu, W.-y. Duan, Review of prediction techniques on hydrodynamic impact of ships, Journal of Marine Science and Application 8 (3) (2009) 204–210. doi:10.1007/s11804-009-8039-7.

[3] J. W. Ringsberg, S. E. Heggelund, P. Lara, B.-S. Jang, S. E. Hirdaris, Structural response analysis of slamming impact on free fall lifeboats, Marine Structures 54 (2017) 112 – 126. doi:http://dx.doi.org/10.1016/j.marstruc.2017.03.004.

[4] M.-S. Kim, H.-S. Park, K.-H. Jung, H.-H. Chun, Air-gap effect on life boat arrangement for a semi-submersible FPU, International Journal of Naval Architecture and Ocean Engineering 8 (5) (2016) 487 – 495. doi:http://dx.doi.org/10.1016/j.ijnaoe.2016.05.006.

[5] H. J. Mørch, S. Enger, M. Peric, E. Schreck, Simulation of lifeboat launching under storm conditions, in: 6th International Conference on CFD in Oil & Gas, Metallurgical and Process Industries, 2008.

[6] V. Tregde, Compressible air effects in CFD simulations of free fall lifeboat drop, in: ASME 2015 34th International Conference on Ocean,
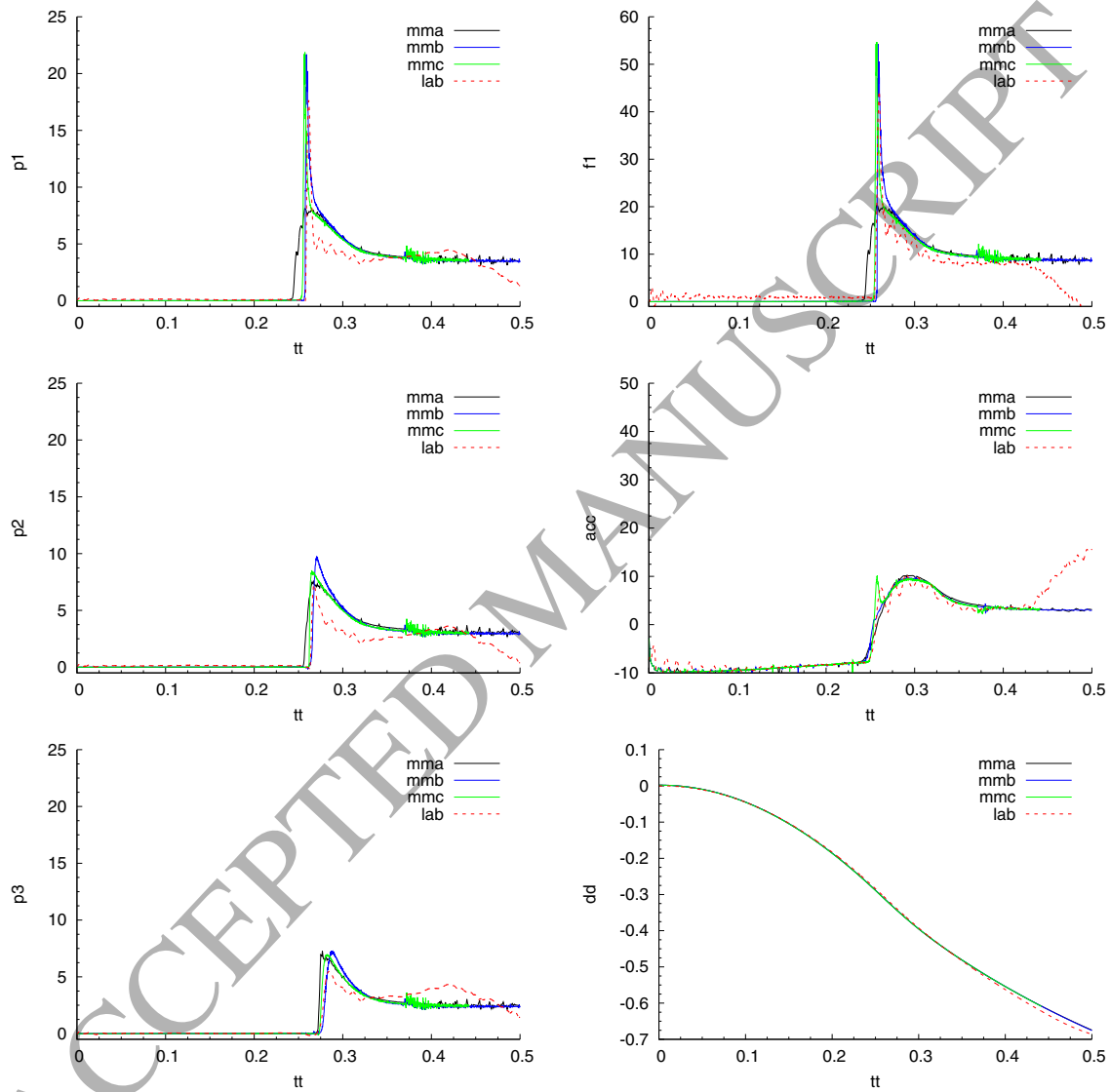
27

Figure 29: Computed and measured dynamics and kinematics for the ship hull. Successively refined meshes A, B and C are used in the simulation.

28

Offshore and Arctic Engineering, American Society of Mechanical Engineers, 2015, p. V002T08A018, OMAE2015-41049. `doi:10.1115/OMAE2015-41049`.

[7] V. Tregde, T. E. Halvorsen, K. O. Underhaug, Simulation of free fall lifeboats impact forces, slamming and accelerations, in: 11th International Conference on Fast Sea Transportation, Honolulu, Hawaii, USA, 2011.

[8] Z. H. Ma, D. M. Causon, L. Qian, C. G. Mingham, T. Mai, D. Greaves, A. Raby, Pure and aerated water entry of a flat plate, Physics of Fluids 28 (2016) 016104. `doi:10.1063/1.4940043`.

[9] D. Wang, M. Randolph, D. White, A dynamic large deformation finite element method based on mesh regeneration, Computers and Geotechnics 54 (2013) 192–201. `doi:10.1016/j.compgeo.2013.07.005`.

[10] F.-L. Yang, C. Chen, D. Young, A novel mesh regeneration algorithm for 2D FEM simulations of flows with moving boundary, Journal of Computational Physics 230 (9) (2011) 3276–3301. `doi:10.1016/j.jcp.2011.01.008`.

[11] A. Johnson, T. Tezduyar, Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and interfaces, Computer Methods in Applied Mechanics and Engineering 119 (1) (1994) 73 – 94. `doi:http://dx.doi.org/10.1016/0045-7825(94)00077-8`.

[12] H. B. Gu, L. Qian, D. M. Causon, C. G. Mingham, P. Lin, Numerical simulation of water impact of solid bodies with vertical and oblique entries, Ocean Engineering 75 (2014) 128–137. `doi:10.1016/j.oceaneng.2013.11.021`.

[13] C. Zhang, X. Hu, N. Adams, A weakly compressible SPH method based on a low-dissipation riemann solver, Journal of Computational Physics 335 (2017) 605–620. `doi:10.1016/j.jcp.2017.01.027`.

[14] A. Khayyer, H. Gotoh, Modified moving particle semi-implicit methods for the prediction of 2d wave impact pressure, Coastal Engineering 56 (4) (2009) 419 – 440. `doi:http://dx.doi.org/10.1016/j.coastaleng.2008.10.004`.

[15] W. M. Chan, Overset grid technology development at NASA Ames research center, Computers & Fluids 38 (3) (2009) 496–503. `doi:10.1016/j.compfluid.2008.06.009`.

[16] P. Fast, W. D. Henshaw, Time-accurate computation of viscous flow around deforming bodies using overset grids, in: 15th AIAA Computational Fluid Dynamics Conference, 2001, AIAA paper 2001-2640.

[17] W. M. Chan, R. L. Meakin, Advances towards automatic surface domain decomposition and grid generation for overset grids, in: 13th Computational Fluid Dynamics Conference, 1997, AIAA 97-1979.

[18] D. Boger, R. Noack, E. Paterson, Foamedover: A library to add a dynamic overset grid capability to openfoam, in: 5th OpenFOAM Workshop, Gothenburg, 2010.

[19] Z. Shen, D. Wan, P. M. Carrica, Dynamic overset grids in OpenFOAM with application to KCS self-propulsion and maneuvering, Ocean Engineering 108 (2015) 287–306. `doi:10.1016/j.oceaneng.2015.07.035`.

[20] Z. H. Ma, D. M. Causon, L. Qian, C. G. 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`.

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

[22] H. Rusche, Computational fluid dynamics of dispersed two-phase flows at high phase fractions, Ph.D. thesis, Imperial College of Science, Technology & Medicine (2002).

[23] 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.

[24] R. Issa, Solution of the implicitly discretised fluid flow equations by operator-splitting, Journal of Computational Physics 62 (1) (1986) 40–65.

29

doi:10.1016/0021-9991(86)90099-9.

[25] J. C. Martin, W. J. Moyce, Part IV. An experimental study of the collapse of liquid columns on a rigid horizontal plane, Philosophical Transactions of the Royal Society A 244 (882) (1952) 312–324. doi:10.1098/rsta.1952.0006.

[26] Z. H. Ma, D. M. Causon, L. Qian, H. B. Gu, C. G. Mingham, P. Martínez-Ferrer, A GPU based compressible multiphase hydrocode for modelling violent hydrodynamic impact problems, Computers and Fluids 120 (2015) 1–23. doi:10.1016/j.compfluid.2015.07.010.

[27] A. Murrone, H. Guillard, A five equation reduced model for compressible two phase flow problems, Journal of Computational Physics 202 (2) (2005) 664–698. doi:10.1016/j.jcp.2004.07.019.

[28] K. Kleefsman, G. Fekken, A. Veldman, B. Iwanowski, B. Buchner, A volume-of-fluid based simulation method for wave impact problems, Journal of Computational Physics 206 (1) (2005) 363 – 393. doi:http://dx.doi.org/10.1016/j.jcp.2004.12.007.

[29] Z. N. Dobrovol'skaya, On some problems of similarity flow of fluid with a free surface, Journal of Fluid Mechanics 36 (4) (1969) 805–829. doi:10.1017/S0022112069001996.

[30] X. Mei, Y. Liu, D. K. Yue, On the water impact of general two-dimensional sections, Applied Ocean Research 21 (1) (1999) 115. doi:10.1016/s0141-1187(98)00034-0.

[31] R. Zhao, O. M. Faltinsen, Water entry of two-dimensional bodies, Journal of Fluid Mechanics 246 (1993) 593–612. doi:10.1017/S002211209300028X.

[32] G. D. Xu, W. Y. Duan, G. X. Wu, Numerical simulation of oblique water entry of an asymmetrical wedge, Ocean Engineering 35 (16) (2008) 1597–1603. doi:10.1016/j.oceaneng.2008.08.002.

[33] X. Feng, W. Bai, X. B. Chen, L. Qian, Z. H. Ma, Numerical investigation of viscous effects on the gap resonance between side-by-side barges, Ocean Engineering 145 (2017) 44 – 58. doi:https://doi.org/10.1016/j.oceaneng.2017.08.060.

[34] Y. A. Semenov, A. Iafrati, On the nonlinear water entry problem of asymmetric wedges, Journal of Fluid Mechanics 547 (2006) 231–256. doi:10.1017/s0022112005007329.