

A multi-region coupling scheme for compressible and incompressible flow solvers for two-phase flow in a numerical wave tank



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

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

ARTICLE INFO

Article history:

Received 9 February 2015

Revised 2 September 2015

Accepted 18 November 2015

Available online 2 December 2015

Keywords:

Hydrodynamics

Coupling

Compressibility

Aeration

Wave impact

ABSTRACT

We present a multi-region coupling procedure based on the finite-volume method and apply it to two-phase hydrodynamic free surface flow problems. The method combines the features of one incompressible and one compressible two-phase flow solvers to obtain a coupled system which is generally superior to either solver alone. The coupling strategy is based on a partitioned approach in which different solvers, pre-defined in different regions of the computational domain, exchange information through interfaces, i.e. areas separating these regions. The interfaces act as boundary conditions passing the information from one region to the other mimicking the finite-volume cell-to-face interpolation procedures. This results in high performance computing coupled simulations whose functionality can be further extended in order to build a generic numerical wave tank accounting for incompressible flow regions as well as compressibility and aeration effects. We select a series of preliminary benchmarks to verify this coupling procedure which includes the simulation of a hydrodynamic dam break, the propagation and reflection of regular waves, the convection of an inviscid vortex, pseudocavitation, a water column free drop in a closed tank and a plunging wave impact at a vertical wall. The obtained results agree well with exact solutions, laboratory experiments and other numerical data.

© 2015 Elsevier Ltd. All rights reserved.

1. Introduction

Coupled simulations, in which different numerical solvers are working together, are in growing demand and present a challenging task. In recent years, an increasing number of coupled simulations have been employed in the scientific community and industry in very different disciplines such as Fluid-Structure Interaction or FSI [1–4], multi-physics [5,6] and multi-solver [7–13] problems and, in particular, incompressible–compressible coupling [14,15], to mention but a few. The range and complexity of problems are expanding while they vary widely in their nature, both in terms of the included physics and manner of coupling. In general terms, a coupling strategy consists of integrating together individual solvers with very specific features in order to improve the accuracy, reduce computational costs and extend the functionality of the global simulation. Due to the large diversity of coupling solutions, there is no general implementation of these strategies. Moreover, with the increasing demand on High Performing Computing (HPC) and heterogeneous computing, e.g. GPGPU and hybrid parallel programming, some strategies remain inefficient as they cannot easily be implemented in parallel via domain decomposition methods [3,8,13]. In some cases, like the one presented in

Ref. [10], the interface between the solvers is in charge of communicating the relevant information using the message passing interface (MPI) protocol allowing the running of coupled simulations on massively parallel supercomputers.

The approach presented in this work aims to overcome these issues by introducing a procedure based on the finite-volume method. We are particularly interested in the coupling of pre-defined finite-volume meshes, each one containing its specific solver, within a unique global simulation by specifying an appropriate set of boundary conditions at the interface connecting the meshes, also referred to as regions. For this purpose, we rely on a “partitioned approach”, e.g. see Refs. [8,12], in which each region is simulated separately and the coupling is performed at the interface. The main advantages of the partitioned approach compared to other approaches, such as the “monolithic approach” utilized in Ref. [13], are its easy implementation and wide range of coupling options: the individual solvers do not need to be extensively modified allowing, for instance, coupling of finite-element and finite-volume based solvers in a relatively easy manner. For this reason, the partitioned approach has been popularly adopted in FSI simulations [3] and hybrid model coupling [9].

Our ultimate goal is to construct a Virtual Wave Structure Interaction (WSI) Simulation Environment such as the one schematized in Fig. 1. A Numerical Wave Tank (NWT) is used to evaluate the impact of waves on fixed or floating objects, offshore structures as well as to assess the performance of wave energy converters. Traditionally, NWTs

* Corresponding author. Tel.: +44 1612471519.

E-mail address: p.martinez-ferrer@mmu.ac.uk (P.J. Martínez Ferrer).

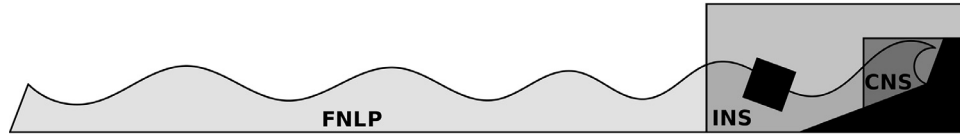


Fig. 1. Virtual Wave Structure Interaction (WSI) Simulation Environment.

have been constructed using potential flow models [16], incompressible Navier-Stokes solvers [17,18] or a combination of both [8,13] for engineering design and analysis. However, in violent wave impacts and slamming events the fluid compressibility needs to be taken into account for the correct prediction of impact loadings [19]. Additionally, aeration effects may become important in hydrodynamics problems involving enclosed air bubbles and air pockets trapped in water bodies [20]. Therefore, the incompressible assumption of the fluids, both air and water, in which the influence of air on waves is ignored because of its small density compared to water, must be revisited. In fact, especial attention has begun to be paid to the compressibility of the water–air mixture in recent years [21–25]. Furthermore, experiments of wave impacts on structures have demonstrated that during the temporal transition of a plunging wave, the air pocket trapped in the water body expands and compresses yielding a strong peak pressure and subsequent pressure oscillations above and below the ambient atmospheric pressure which can potentially damage offshore structures [20,26,27]. Therefore, compressibility effects in the water–air mixture and air pockets as well as cavitation effects, i.e. change of phase, need to be handled properly. Thus, in order to represent the most accurate physics where most necessary within a NWT, specialized numerical solvers must be coupled together within the same simulation framework, e.g. using the scheme given in Fig. 1: (i) a Fully Non-Linear Potential (FNL P) solver generates and propagates accurately the waves from the wave generator source, (ii) an interface-capturing Incompressible Navier–Stokes (INS) solver is used when aeration and compressibility of the water–air mixture are negligible and (iii) an interface-capturing Compressible Navier Stokes (CNS) may be envisaged to evaluate accurately the expected violent wave loads against structures. Fixed structures and floating bodies may be deformable and thus a Computational Structural Dynamics (CSD) code may also be necessary for wave and structure interactions.

In this work we introduce a new coupling procedure based on the finite-volume method and concentrate our efforts on the coupling of the INS and CNS solvers which we validate through a series of numerical benchmarks. Section 2 describes the incompressible and compressible solvers, the coupling procedure and the solution procedure algorithm for the coupled simulations. Results and discussions are provided in Section 3 and Section 4 is dedicated to conclusions and future work.

2. Numerical procedures

The numerical methods used in this work rely on a cell-centered, co-located finite-volume method already implemented in the open-source CFD software OpenFOAM [28]. The reason for using this CFD library is because it is a robust and advanced tool widely employed in research and industry. Moreover, the advantage of being open-source allows to read, improve and modify the available code for free. Nevertheless, we stress the fact that the coupling strategy and numerical methods described below can be applied to any finite-volume solver in different engineering applications such aerodynamics, hydrodynamics, etc. We modify two native, incompressible and compressible, two-phase pressure-based solvers available in OpenFOAM in order to get them working together exchanging information at their interfaces through a new set of appropriate boundary conditions. A description of the solvers, the coupling strategy as well as the solution algorithm is detailed below.

2.1. The incompressible solver “interFoam”

The native OpenFOAM based solver “interFoam” is an incompressible two-phase pressure-based solver [29] which has successfully been applied in a wide variety of naval and coastal engineering applications, see for instance [30,31]. It solves the three-dimensional equations for two incompressible phases, i.e. air and water, using the volume of fluid (VOF) method with especial emphasis on maintaining a sharp free surface (interface-capturing solver).

The first equation to be solved is the mass balance. For an incompressible two-phase flow ($\nabla \cdot \mathbf{U} = 0$), only the mass balance equation for the water volume fraction $\alpha \in [0, 1]$ is considered

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

where \mathbf{U} is the mixture velocity vector and $\mathbf{U}_c = \min[\mathbf{U}, \max(\mathbf{U})]$. The third term in Eq. (1) is an artificial compression term that helps to maintain the interface sharp and bounded between zero and unity using the MULES procedure [29,32].

The single momentum equation for the homogeneous mixture is given by

$$\frac{\partial \rho \mathbf{U}}{\partial t} + \nabla \cdot (\rho \mathbf{U} \mathbf{U}) - \nabla \cdot (\mu_{\text{eff}} \nabla \mathbf{U}) = \sigma \kappa \nabla \alpha - \mathbf{g} \cdot \mathbf{x} \nabla \rho - \nabla p_d, \quad (2)$$

where $\rho = \alpha \rho_w + (1 - \alpha) \rho_a$ is the density of the mixture; note that the sum over the volume fractions of water and air is equal to unity and $\rho_w = 1000 \text{ kg/m}^3$ and $\rho_a = 1.1586 \text{ kg/m}^3$ are constant parameters. The surface tension coefficient is represented with the variable σ and the curvature of the interface is calculated as $\kappa = \nabla \cdot (\nabla \alpha / |\nabla \alpha|)$. Furthermore, the mixture viscosity is given by $4(\alpha \mu_w + (1 - \alpha) \mu_a) / 3$, $p_d = p - \rho \mathbf{g} \cdot \mathbf{x}$ is the piezometric pressure and \mathbf{g} and \mathbf{x} are the gravity and the position vectors, respectively. It can be noticed from Eq. (2) that heat and mass transfer are neglected. Additionally, a pressure correction equation is derived from the momentum and continuity equations to solve for the dynamic pressure and correct the velocity adopting a segregated approach.

2.2. The compressible solver “compressibleInterFoam”

This OpenFOAM native solver is an extension of the previous interface-capturing solver. The water phase is treated as a compressible perfect fluid, $\rho_w = \rho_{w,0} + p/R_w T$ with $\rho_{w,0} = 1000 \text{ kg/m}^3$ and $R_w = 3000 \text{ J/kg K}$, while the air phase is a compressible perfect gas, $\rho_a = p/R_a T$ with $R_a = 287 \text{ J/kg K}$. The water volume fraction transport equation presents an additional source term due to the compressibility of this phase:

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot \mathbf{U} \alpha + \nabla \cdot \mathbf{U}_c \alpha (1 - \alpha) = - \frac{\alpha}{\rho_w} \frac{D \rho_w}{Dt}, \quad (3)$$

which is implicitly calculated from the pressure correction equation at the previous time step. The density of the compressible mixture is evaluated according to

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{U}) = 0. \quad (4)$$

The same single momentum equation for the homogeneous mixture (2) is solved and an additional transport equation for the

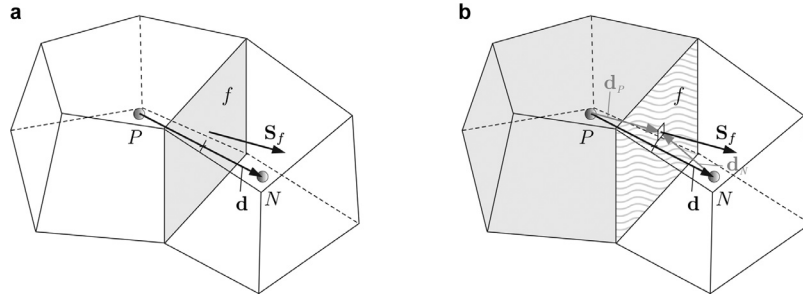


Fig. 2. (a) Two neighbouring cells in a unique finite-volume discretized domain; (b) two neighbouring finite-volume discretized domains (only one cell is illustrated for each domain) located side-by-side without overlapping.

temperature T is deduced from the energy equation

$$\frac{\partial \rho T}{\partial t} + \nabla \cdot (\rho \mathbf{U} T) - \Delta(\mu_{\text{eff}} T) = - \left(\frac{\alpha}{c_{v,w}} + \frac{1-\alpha}{c_{v,a}} \right) \left(\frac{\partial \rho k}{\partial t} + \nabla \cdot (\rho \mathbf{U} k) + \nabla \cdot (\mathbf{U} p) \right), \quad (5)$$

with $k = |\mathbf{U}|^2/2$ the specific kinetic energy and $c_{v,a}$ and $c_{v,w}$ the specific heat capacities at constant volume for the air and water phases, respectively.

Similarly to the incompressible solver, a pressure correction equation is derived from Eq. (2) and the continuity equations of each phase to solve for the dynamic pressure and correct the velocity and partial densities. This is carried out by the means of a segregated algorithm which accounts for the pressure–velocity–density coupling that arises from compressibility effects [33].

2.3. Coupling strategy

The finite-volume method is based on the integral form of the conservation equations over a cell volume V as illustrated in Fig. 2(a). This method evaluates partial differential equations (PDEs) in the form of algebraic equations [34]. Spatial derivative terms are converted into surface integrals over the cell surface S using Gauss's theorem

$$\int_V \nabla \cdot \phi dV = \int_S \mathbf{dS} \cdot \phi, \quad (6)$$

where S is the surface area vector, ϕ is any tensor field and the star notation \star is used to represent any tensor operation. For example, applying Gauss's theorem to common fluid mechanics tensor operators such as the divergence gives

$$\int_V \nabla \cdot \phi dV = \int_S \mathbf{dS} \cdot \phi = \sum_f \mathbf{S}_f \cdot \phi_f, \quad (7)$$

while the gradient operator becomes

$$\int_V \nabla \phi dV = \int_S \mathbf{dS} \phi = \sum_f \mathbf{S}_f \phi_f, \quad (8)$$

and the Laplacian is

$$\int_V \nabla \cdot (\nabla \phi) dV = \int_S \mathbf{dS} \cdot (\nabla \phi) = \sum_f \mathbf{S}_f \cdot (\nabla \phi)_f, \quad (9)$$

with

$$\mathbf{S}_f \cdot (\nabla \phi)_f = \left(|\mathbf{S}_f| \frac{\phi_N - \phi_P}{|\mathbf{d}|} \right) \left(\frac{\mathbf{S}_f \cdot (\nabla \phi)_f}{|\mathbf{S}_f| |\nabla \phi|_f} \right), \quad (10)$$

where the index N denotes neighbouring cell, index P denotes current cell and \mathbf{d} is the vector distance between the two cell centres, see Fig. 2(a). Volume and surface integrals introduced in the above equations need to be linearized using appropriate discretization schemes.

Additionally, values at the cell faces, ϕ_f , must be calculated from values at the adjacent cell centres (co-located configuration), ϕ_P and ϕ_N , using interpolation procedures.

We now consider that the cells N and P belong indeed to independent finite-volume domains in which different PDEs may be applied, as shown in Fig. 2(b). This configuration assumes that cell faces must be conformal between the regions and placed side-by-side without overlapping. In this particular scenario, the surface f becomes an *interface* between the two regions, i.e. the two independent finite-volume domains, and appropriate boundary conditions must be specified at each side of this interface in order to *couple* the information between the regions. In particular, the values of the magnitude of ϕ_f and its gradient $\nabla \phi_f$ (or $\mathbf{S}_f \cdot \nabla \phi_f$) must be defined at the interface, as deduced from Eqs. (7)–(10). It is worth noting that the schematic configurations presented in Fig. 2(a) and (b) remain very similar and, therefore, it seems reasonable to mimic the procedures applied to evaluate face values from cell centre data in order to estimate values at the interface from different regions. Thus, we define the value of the magnitude of ϕ at the interface as

$$\phi_f = \frac{1}{|\mathbf{d}_P| + |\mathbf{d}_N|} (\phi_P |\mathbf{d}_N| + \phi_N |\mathbf{d}_P|), \quad (11)$$

which represents a distance-weighted average between two cells ($\mathbf{d} = \mathbf{d}_P - \mathbf{d}_N$). Additionally, we define the value of the gradient at the interface as

$$\mathbf{S}_f \cdot \nabla \phi_f = \left(|\mathbf{S}_f| \frac{\phi_N - \phi_P}{|\mathbf{d}|} \right) \left(\frac{\mathbf{S}_f \cdot \mathbf{d}}{|\mathbf{S}_f| |\mathbf{d}|} \right), \quad (12)$$

for the region containing the cell P and

$$\mathbf{S}_f \cdot \nabla \phi_f = \left(|\mathbf{S}_f| \frac{\phi_P - \phi_N}{|\mathbf{d}|} \right) \left(- \frac{\mathbf{S}_f \cdot \mathbf{d}}{|\mathbf{S}_f| |\mathbf{d}|} \right), \quad (13)$$

for the region containing the cell N . In other words, the face values derived from Eqs. (11)–(13) are used as boundary conditions in a coupled problem. For example, in the particular case in which both regions remain identical, i.e. they use the same numerical procedures and PDEs and the meshes are orthogonal, the estimation given by Eqs. (11)–(13) sensibly matches the face interpolation procedure required in the not coupled finite-volume method. Furthermore, this strategy only uses information from the two extreme cells pertaining to each region and its parallel implementation using a mesh decomposition technique is straightforward provided that each region can be individually parallelizable, which results in an efficiently parallel coupling implementation. Finally, if the meshes do not match at the interfaces and thus the cell faces are not conformal, the above procedure can still be envisaged provided that an additional interpolation procedure is applied, see [35].

When the above set of boundary conditions is applied in conjunction with a partitioned approach, it results in a *two-way* coupling procedure in which the information shared between the coupled region remains, in principle, symmetrical. Fig. 3 illustrates a schematic

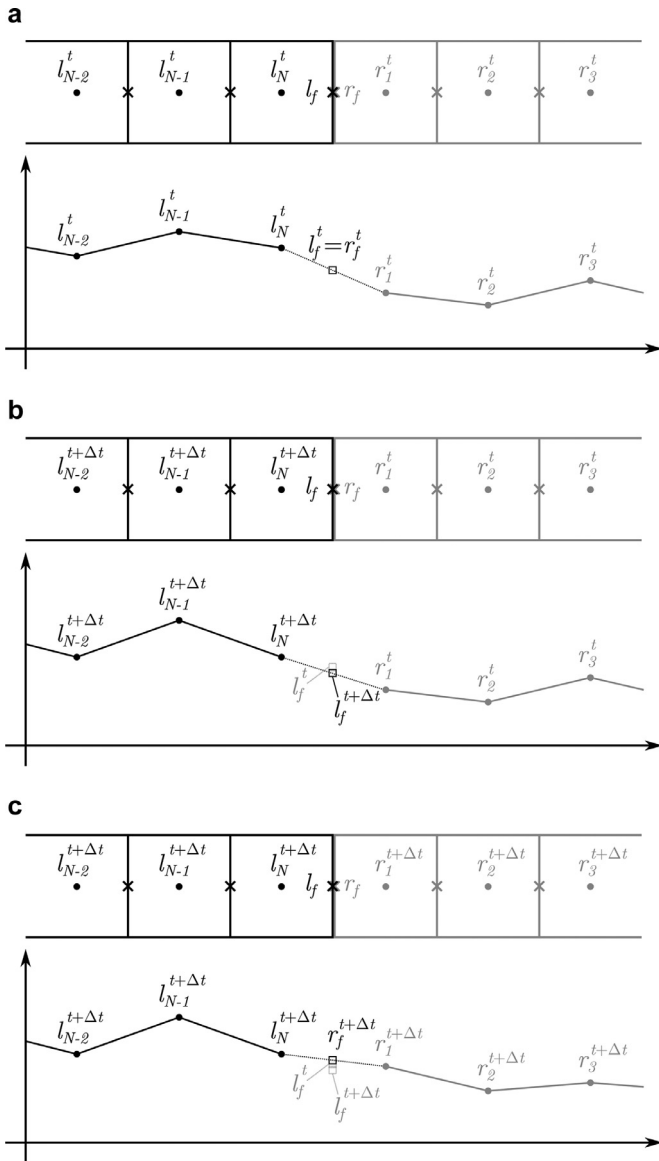


Fig. 3. Partitioned approach between regions I (left) and r (right) using the boundary conditions (11)–(13): (a) previous time step, (b) left hand side region solved and (c) right hand side region solved.

one-dimensional coupled simulation employing orthogonal cells and this coupling strategy. The solution at a previous time step is known in both regions and the boundary conditions (11)–(13) can be specified at each side of the interface, see Fig. 3(a). During the next time step, Fig. 3(b), the solution in the left hand side region is advanced a time step (the minimum found in the two regions) using the previous boundary conditions while the solution in the right hand side region remains unmodified. Finally, the solution in the right hand side domain is advanced the same time step using the new interface values obtained from the previous time step, as illustrated in Fig. 3(c).

This procedure, which introduces a certain delay between the left and right hand side solutions, is referred as “loose coupling”, and has given satisfactory results for all the simulations conducted in this work as will be shown in Section 3. This is a consequence of the broadly similar time steps that apply to the two different pressure-based solvers, i.e. the same value of the Courant number and similar meshes, which permit choosing the minimum time step without penalizing the simulation. Nevertheless, using the minimum time step for all regions may lead to a relatively slow method when individual

time steps remain significantly different. For simulations in which loose coupling may lead to instabilities, e.g. FSI with slender structures, or differences in time steps of several orders of magnitude, a sub-iterative procedure, known as “tight coupling”, can be applied at every time step to achieve the desired coupling between the solutions: Aitken’s adaptive under-relaxation is often employed in FSI applications [36].

2.4. Solution procedure

In the rest of this work we shall refer to the modified version of interFoam as the “incompressible” or “I” solver whereas “compressible” or “C” will be used to refer to the modified version of compressibleInterFoam. For the coupled simulations, we will frequently use the expressions “I–I” or “I–C” to designate incompressible–incompressible or incompressible–compressible coupled simulations, respectively.

The governing equations in the incompressible solver, (1) and (2), are linearized and integrated over each control volume to determine α and \mathbf{U} , respectively. Then, a pressure corrector linearized equation is solved for p_d . The overall solution procedure relies on the segregated projection algorithm PIMPLE [37], derived from the original PISO procedure [38,39], which allows for equation under-relaxation to guarantee convergence of the solutions to all the equations at each time step. Further details are provided in Refs. [28,29]. During the linearization process, neighbouring cell values are gathered to form a matrix of the resulting system of linear algebraic equations [34]. For coupled simulations, cell values and geometrical properties (distance vectors), belonging to neighbouring regions are taken by previously identifying matching cells faces at the two sides of the interface separating the regions. In the case of parallel runs, cells may be available in different cores and thus MPI communications may be necessary to gather this information [10]. Eqs. (1)–(5) in addition to a pressure corrector linearized equation are treated in a similar fashion in the compressible solver. In coupled problems, the I and C solvers are applied sequentially following the partitioned method that we summarize in the following algorithm:

FOR EACH ITERATION IN THE MAIN LOOP, DO

1. Calculate the time steps of the I regions
2. Calculate the time steps of the C regions
3. Find the global (minimum) time step

FOR EACH I REGION, DO

- FOR EACH I TRANSPORT EQUATION, DO
 - i. Linearize
 - ii. Apply boundary and interface conditions
 - iii. Solve
- END I TRANSPORT EQUATION

END I REGION

FOR EACH C REGION, DO

- FOR EACH C TRANSPORT EQUATION, DO
 - i. Linearize
 - ii. Apply boundary and interface conditions
 - iii. Solve
- END C TRANSPORT EQUATION

END C REGION

4. Update the simulation time

END MAIN LOOP

It can be seen that the current algorithm allows coupling of an undetermined number of incompressible and compressible regions while the interface is simply treated as another boundary condition when solving each transport equation. The incompressible regions will gather the necessary neighbouring cell values of the water volume fraction, velocity and pressure when solving the corresponding transport equations. On the other hand, compressible regions will

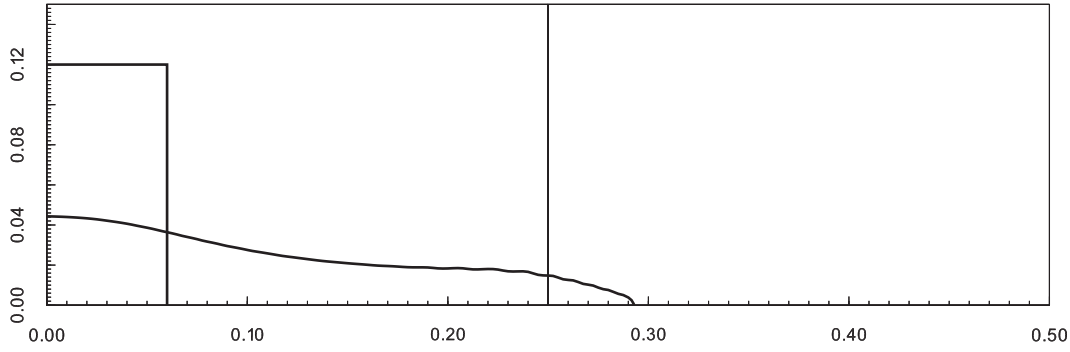


Fig. 4. Water dam break benchmark: free surface contour plotted at $t = 0$ s and $t = 0.2$ s (I-I simulation); the vertical line at $x_1 = 0.25$ m is the interface.

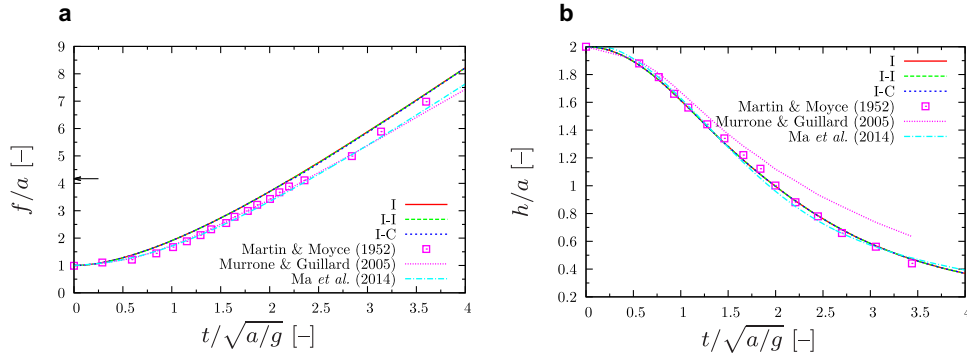


Fig. 5. Water dam break benchmark: time history of the normalized (a) water front position and (b) water height; arrows indicate the position of the interface.

gather the same information but density and temperature must be specified as well. The interface conditions for the density are calculated from the water volume fraction $\rho = \alpha \rho_w + (1 - \alpha) \rho_a(p, T)$ and we assume presently that temperature variations are negligible locally near the interface, which results in a zero-gradient Neumann condition for this variable. Our approach is justified because, in a NWT configuration, increasing values of the temperature of the order of several degrees Kelvin may take place near impact zones, which will be located sufficiently far from the interface. However, a more general temperature interface boundary condition using Eqs. (11)–(13) in order to take into account temperature gradients will require the addition of a novel transport equation for the temperature in the incompressible solver. Nevertheless, the results from Section 3 show that the simple Neumann condition suffices in most cases.

Finally, it is worth mentioning that the solvers, the coupling procedure and additional algorithms are gathered in a utility named by the authors “wsiFoam” (for Virtual Wave Structure Interaction), written entirely in the C++ programming language and employing the OpenFOAM framework.

3. Results and discussions

In the following we use the same finite-volume discretization schemes unless otherwise stated: (i) temporal derivatives are discretized with the first-order implicit Euler scheme, (ii) a first-order upwind scheme is used for the convective terms in the momentum equation and (iii) the van Leer limiter is employed to discretize the convective terms in the water volume transport equation [29]. Moreover, we found that adding a filter at the interface to bound values of the water volume fraction between zero and unity, as done in the MULES procedure [32], is particularly helpful to aid with stabilizing the numerical solutions. Finally gravity acceleration is set to 9.81 m/s^2 when required, $\sigma = 0.07 \text{ kg/s}^2$, $\rho_w = 1000 \text{ kg/m}^3$, $\rho_a = 1.16 \text{ kg/m}^3$ (incompressible solver), $\mu_w = 8.9 \cdot 10^{-4} \text{ kg/ms}$, $\mu_a = 1.6 \cdot 10^{-5} \text{ kg/ms}$, $c_{v,w} = 4195 \text{ m}^2/\text{s}^2 \text{ K}$ and $c_{v,a} = 719.3 \text{ m}^2/\text{s}^2 \text{ K}$.

3.1. Water dam break

This problem has been studied extensively in the past using both laboratory experiments [40] and numerical works [25,41]. Whereas this benchmark has largely been used in previous numerical studies to verify an implementation of a new flow model, here it will serve to verify the implementation of the new interface boundary conditions for coupling solvers as discussed in the previous section. The geometrical configuration is presented in Fig. 4 along with the free surface contour ($\alpha = 0.5$) at two different times: $t = 0$ s and $t = 0.2$ s. The initial conditions for this problem correspond to a water column of $a = 0.06$ m width and $2a = 0.12$ m height placed in a tank of 0.5 m width and 0.15 m height which remains open to the atmosphere ($p = 1$ bar) at its top boundary. The wall boundaries share a no-slip condition. The domain is uniformly divided into 200×60 mesh cells and the Courant number is set to 0.5 . For the coupled simulations, the interface between the left and right regions is placed in the middle of the computational domain ($x_1 = 0.25$ m) as shown in Fig. 4.

As the simulation begins, the water collapses under gravity and the water front position reaches gradually the entire width of the tank. Two quantities of interest are most commonly compared against measurement data: (i) the water front position along the bottom of the tank, f , and (ii) the height of the water column along the left boundary of the tank, h . Fig. 5(a) shows the time history of the normalized water front position for the pure incompressible (I), coupled incompressible–incompressible (I–I) and coupled incompressible–compressible (I–C) solvers. The three solutions are almost superposed even after the water front passes through the interface ($f/a \approx 4.17$). The normalized solutions corresponding to the water column height, depicted in Fig. 5(b), also remain indistinguishable and agree well with previous results [25,40,41]. The interface propagates correctly the information between the two regions without altering significantly the solution. In the light of these results, it can be deduced that the original mass of water is conserved through the interface.

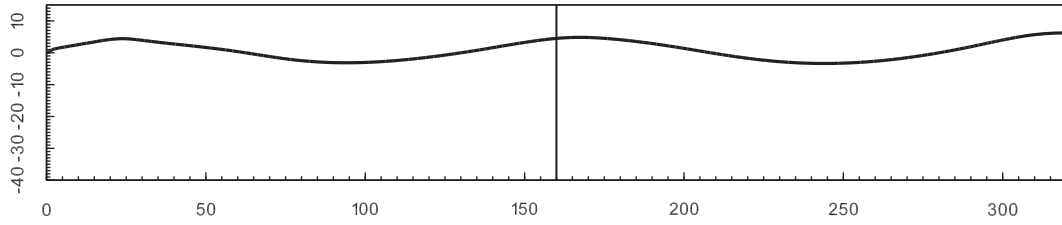


Fig. 6. Wave propagation benchmark: free surface contour plotted at $t = 85$ s (I-I simulation); the vertical line at $x_1 = 160$ m is the interface.

Table 1

Water dam break benchmark: simulation speed up for a mesh of 400×120 cells; times are normalized by $t_{\text{ref}} = 262.93$ s.

Cores	I	C	I-C
1	1.25	1.00	1.10
2	2.10	1.66	1.83
4	3.18	2.52	2.80

Table 2

Wave propagation benchmark: simulation speed up for a mesh of 350×133 cells; times are normalized by $t_{\text{ref}} = 506.32$ s.

Cores	I	C	I-C
1	1.10	1.03	1.00
2	1.87	1.70	1.71
4	3.06	2.46	2.49

It is also interesting to compare the simulation times required to solve this problem using the different solvers: I, C and I-C. Table 1 shows the normalized speed up for an increasing number of computational cores. It can be readily noticed that, when solving this problem with one single core, the pure compressible solver is the fastest while the pure incompressible solver is the slowest by a factor of 25%. The coupled solver speed up remains in between these values for both the sequential and parallel cases. This evidences the good potential parallel capabilities of the coupling strategy, as no significant amount of overhead is added in comparison to the not coupled simulations.

3.2. Wave propagation

This benchmark has been considered in order to verify the correct propagation of waves through an interface separating two different regions. This is a very important feature required for the construction of a NWT built with significantly different specialized solvers interacting altogether. Fig. 6 shows the geometrical configuration consisting of a tank with dimensions 55×320 m² open at the top boundary ($p = 1$ bar) and featuring no-slip conditions at the bottom and side walls. A velocity boundary condition at the left wall generates regular waves thanks to a sinusoidal function with an amplitude $A = 2.5$ m and a period $T = 10$ s. Initially, the free surface is represented by a horizontal line at $x_2 = 0$ m and, as the simulation begins, waves are generated and propagated towards the right wall and reflected back into the tank. Even though wave reflection is often an undesired phenomena to be avoided in NWTs (and laboratory wave tanks), it has been retained in this work as we are also interested in how the interface boundary conditions can handle wave interaction and reflection. The domain is discretized with 250×95 mesh cells with a slight stretching applied far from the free surface in order to reduce computational costs. All the simulations have been conducted up to $t = 100$ s with a fixed value of the CFL equal to 0.5. For the coupled cases, the interface is placed in the middle of the computational domain ($x_1 = 160$ m) to separate the left and right regions.

Fig. 7 shows the history of the free surface height at the $x_1 = 80$ m, $x_1 = 160$ m and $x_1 = 240$ m locations. The arrival of the first wave at the rightmost probe takes place at $t < 20$ s. Therefore, wave reflection and interaction can be expected at the interface at $t \lesssim 100$ s. Additionally, the irregular amplitude of wave crests and valleys captured by all the numerical water depth probes also confirms this reflection (see Fig. 7). Regarding the four different numerical solutions obtained, it can readily be seen that they all remain very close to each other. However, the compressible solutions seem to diverge slightly from the

incompressible ones towards the end of the simulation time. We can confirm that the interface boundary conditions implemented are capable of transmitting the information of ongoing waves between regions in both directions simultaneously: two-way coupling.

Finally, Table 2 shows the speed up associated with the I, C and I-C simulations. On the one hand and contrary to the water dam break benchmark, the coupled simulation (I-C) is now the slowest while the incompressible simulation (I) is the fastest; however, the overhead added in the coupled problem remains particularly small, making the simulation slower by 10% at most using one single core. On the other hand, the scalability up to 4 cores in this relatively coarse two-dimensional mesh is well maintained for the three simulations. Thus, better scalability values should be expected for an increasing number of cells and in three-dimensional configurations.

3.3. Inviscid vortex

This benchmark has been extensively used in the past [42,43] and more recently in Ref. [44] to measure the accuracy of discretization methods implemented in different numerical solvers. In addition, we are particularly interested in the propagation of such a vortex through the interface connecting different or similar numerical solvers and therefore the impact of such interface treatments on the properties of the vortex itself. A similar study has been done previously in Ref. [14]. The version considered here is slightly different from the original one [42] as it consists of a two-dimensional uniform flow of water $(u_{1,0}, u_{2,0})^t = (1, 0)^t$ m/s with a superimposed inviscid vortex defined by

$$(\delta u_1, \delta u_2)^t = \frac{\Gamma}{2\pi} \exp\left(\frac{1-r^2}{2}\right) (x_{2,c} - x_2, x_1 - x_{1,c})^t, \quad (14)$$

where $\Gamma = 5 \text{ s}^{-1}$ is the vortex circulation, $(x_{2,c}, x_{1,c})^t = (5, 5)^t$ m denotes the initial position of the vortex centre and the square of its radius is $r^2 = (x_1 - x_{1,c})^2 + (x_2 - x_{2,c})^2$. The initial pressure is set to $p_0 = 1$ bar. On the one hand, for the not coupled simulation, this vortex is initially placed in the centre of a square domain of $L = 10$ m length. Symmetry plane boundary conditions are applied at the bottom and top boundaries whereas periodic boundary conditions are utilized at the left and right boundaries. On the other hand, for the coupled simulations, a rectangular domain is used instead, see Fig. 8, and the left and right boundary conditions are replaced by inflow and outflow boundary conditions, respectively. This rectangular domain is then divided in half in the horizontal direction by an interface separating the left and right regions as illustrated in Fig. 8. The computational domains are uniformly divided into 159×159 and 318×159

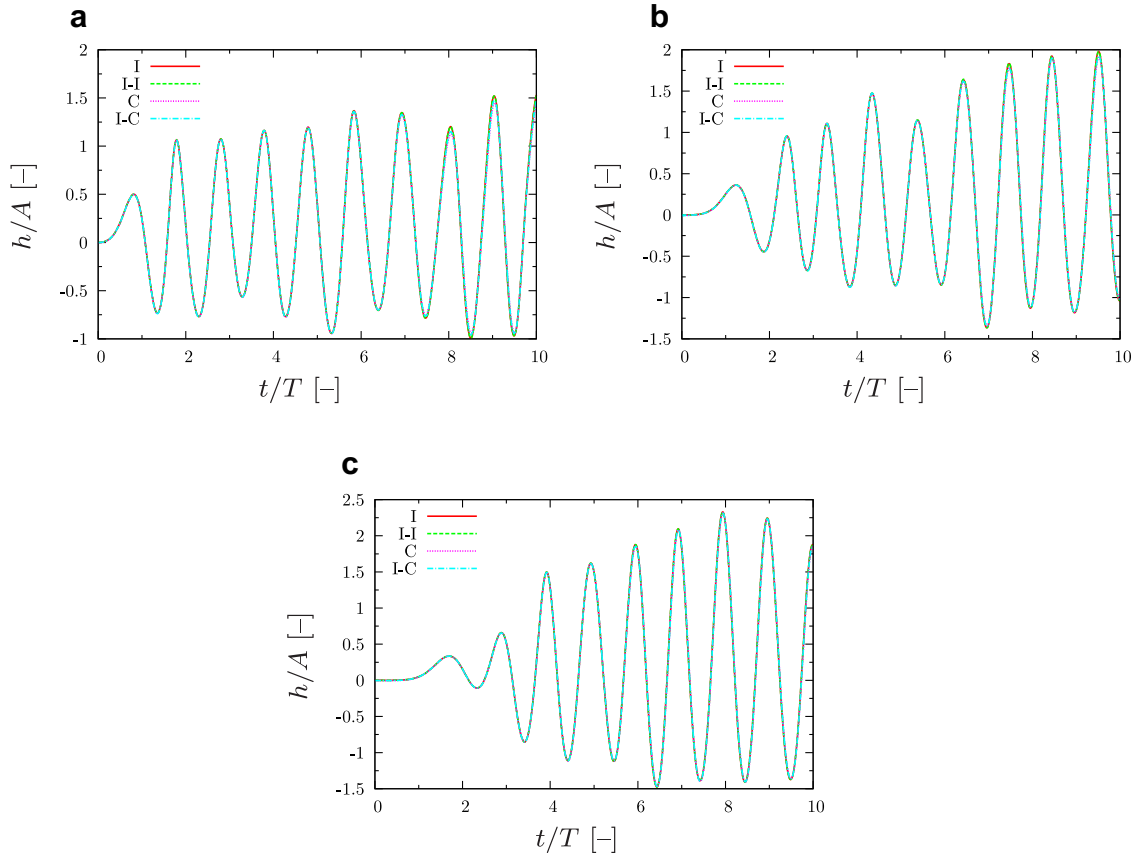


Fig. 7. Wave propagation benchmark: time history of the normalized free surface height at (a) $x_1 = 80$ m, (b) $x_1 = 160$ m and (c) $x_1 = 240$ m.

mesh cells, respectively. In both cases, the simulations are run up to one convective time, i.e. $t = 10$ s, with a constant Courant number of 0.5 and suppressing gravity and viscous terms from the system of equations. Moreover, the discretization schemes are specifically modified in this particular benchmark to guarantee second-order accuracy of the solutions in both space and time.

Fig. 8 shows the contours of the normalized pressure and x_2 -component of the velocity. A tiny asymmetry can be observed in the normalized pressure contours at $t = 5$ s as well as a slight modification of the contour lines. Additionally, new contour lines arise near the interface, within the right region, probably indicating the accumulation of numerical errors in this zone. The normalized velocity contours are less affected by the interface but a clear shift can be noticed, especially at $t = 10$ s. Fig. 9 shows the profiles of the normalized pressure ($\Delta p = \rho \Gamma^2 / (8\pi^2)$) and x_2 -component of the velocity ($\Delta u = \Gamma / (2\pi)$) along the line $x_2 = 5$ m after one convective time. The solutions obtained with the not coupled solver (I) and the two coupled solvers (I-I and I-C) are compared against the exact solution. Firstly, it is worth noting that all the numerical solutions suffer from small but noticeable numerical diffusion and dispersion errors which have been previously identified as the asymmetry and shift of the pressure and velocity contours. Secondly, all three numerical solutions compare well and it is difficult to distinguish between them.

Following Refs. [43,44] we proceed to evaluate the order of accuracy of the numerical solutions. For this purpose, the numerical error is calculated using the L_2 -norm as a relevant metric

$$\mathcal{E}^{\Delta x}(\phi) = \sqrt{\frac{\sum_{i,j,k} (\phi_{i,j,k}^{\Delta x} - \phi_{i,j,k}^e)^2}{N}}, \quad (15)$$

where ϕ can be any relevant field, N is the total number of mesh cells and the index e designates the exact solution. We assume [44] that

Table 3

Inviscid vortex benchmark: numerical error and order of convergence for the pressure (one convective time cycle), S.I. units.

Δx	$\mathcal{E}_1^{\Delta x}(p)$	$s_1^{\Delta x}(p)$	$\mathcal{E}_{1-1}^{\Delta x}(p)$	$s_{1-1}^{\Delta x}(p)$	$\mathcal{E}_{1-C}^{\Delta x}(p)$	$s_{1-C}^{\Delta x}(p)$
0.5263	84.1127	–	99.3406	–	98.5397	–
0.2564	28.7781	1.4915	37.6655	1.3486	35.6438	1.4141
0.1266	9.7613	1.5317	13.6487	1.4380	12.7981	1.4510
0.0629	4.6947	1.0465	6.5256	1.0550	6.2456	1.0236

Table 4

Inviscid vortex benchmark: numerical error and order of convergence for the x_2 -component of the velocity (one convective time cycle), S.I. units.

Δx	$\mathcal{E}_1^{\Delta x}(u_2)$	$s_1^{\Delta x}(u_2)$	$\mathcal{E}_{1-1}^{\Delta x}(u_2)$	$s_{1-1}^{\Delta x}(u_2)$	$\mathcal{E}_{1-C}^{\Delta x}(u_2)$	$s_{1-C}^{\Delta x}(u_2)$
0.5263	0.1620	–	0.1643	–	0.1622	–
0.2564	0.0404	1.9315	0.0555	1.5084	0.0523	1.5737
0.1266	0.0132	1.5858	0.0183	1.5716	0.0171	1.5817
0.0629	0.0059	1.1414	0.0083	1.1242	0.0078	1.1190

Eq. (15) can be written as $\mathcal{E}^{\Delta x} = C(\Delta x)^q$ and thus the order of accuracy can be calculated using the following expression

$$s^{\Delta x}(\phi) = \frac{\log(\mathcal{E}^{\Delta x'}(\phi) / \mathcal{E}^{\Delta x}(\phi))}{\log(\Delta x' / \Delta x)}. \quad (16)$$

Table 3 reports on the numerical error and the order of accuracy for the pressure field. Values of $s^{\Delta x}(p)$ are greater than unity but relatively far from the theoretical order of accuracy of the discretization schemes employed. It is also worth noting that the interface has little or no effect on the order of accuracy of the solutions. On the other hand, Table 4 reports on the numerical error and the order of accuracy for the x_2 -component of the velocity. Overall, the orders of accuracy

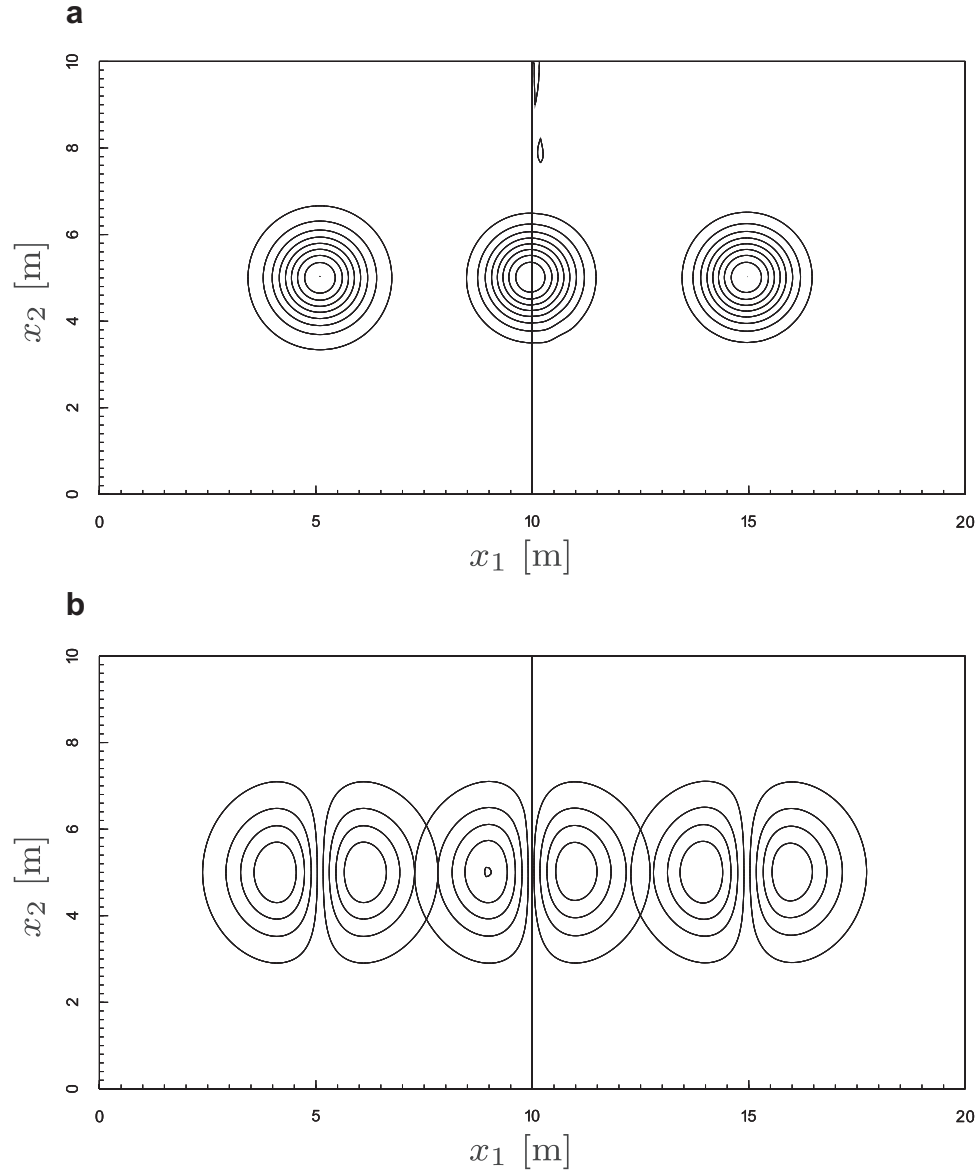


Fig. 8. Inviscid vortex benchmark: normalized contours of the (a) pressure and (b) x_2 -component of the velocity plotted at three different times (I-C simulation): $t = 0$ s, $t = 5$ s and $t = 10$ s; the vertical line at $x_1 = 10$ m is the interface.

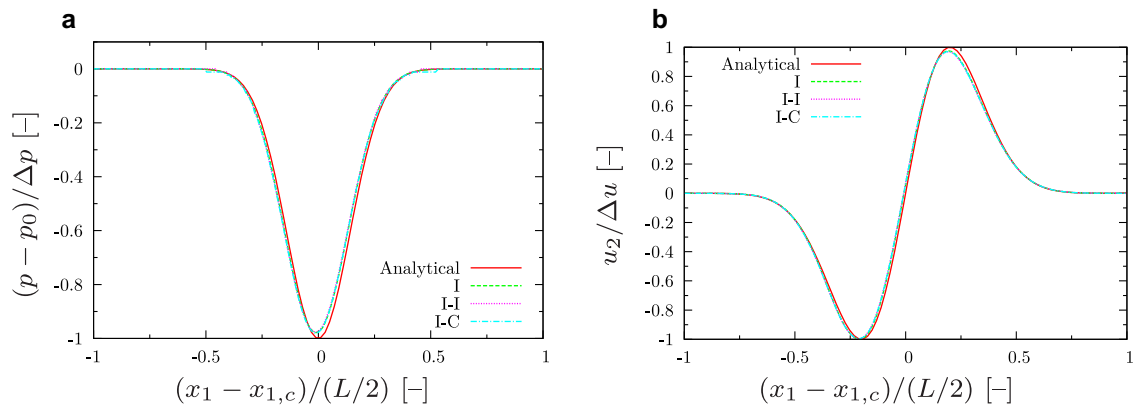


Fig. 9. Inviscid vortex benchmark: normalized distributions of the (a) pressure and (b) x_2 -component of the velocity along the line $x_2 = 5$ m after one convective time ($t = 10$ s).

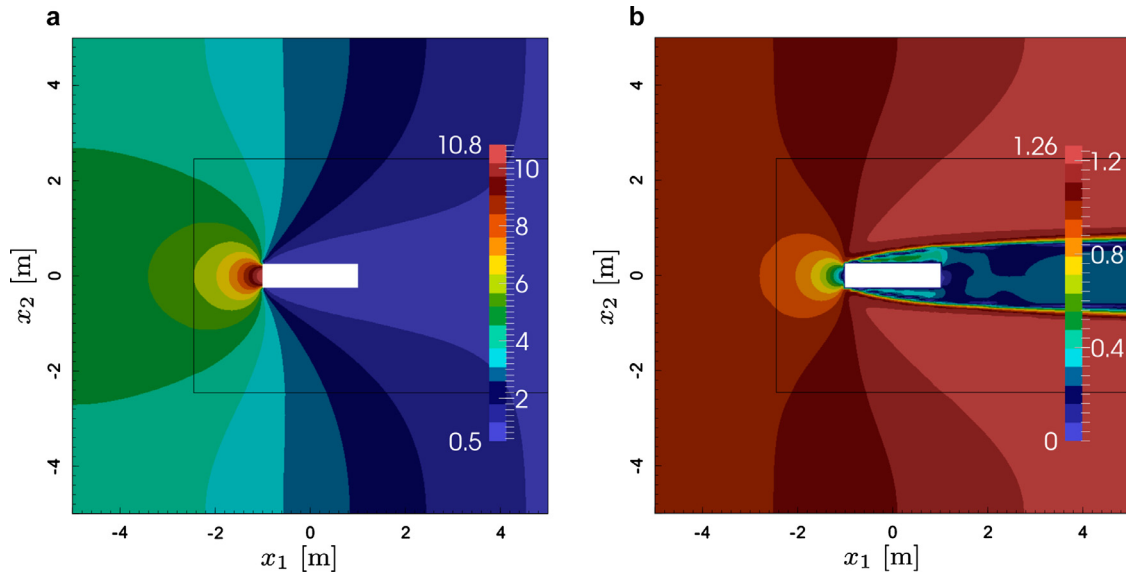


Fig. 10. Pseudocavitation benchmark: (a) normalized pressure and (b) normalized velocity magnitude fields corresponding to the stationary solution (I–C simulation); the interface shown separates the inner and outer regions.

are higher in comparison with the ones given in Table 3. It can be readily seen that the introduction of the interface does not alter significantly the order of accuracy, which remains always greater than unity for all the grids considered.

The results presented in this benchmark confirm that the interface boundary condition presented in this work is capable of propagating the inviscid vortex from one region to the other reasonably well in a coupled solver within one convective time cycle introducing less numerical errors than the intrinsic scheme discretization errors thus conserving the order of accuracy of the original solutions.

3.4. Pseudocavitation

This pseudocavitation benchmark has been selected to verify the coupling between different solvers placed at strategic locations, e.g. places where the describing physics may be different and several numerical methods may be envisaged, across the computational domain. In other words, we are interested to capture the *phase change* due to partial density variations of a compressible mixture by means of our coupling strategy which otherwise would prove to be impossible using solely an incompressible solver. We recall that the cavitation phenomena is not modelled as a mass transfer process, see Eq. (5), but mimicked as a mechanical relaxation process. When the pressure drops, the water volume fraction decreases due to the source term in Eq. (4). Consequently, the air volume fraction increases and this leads to the generation of an air gas pocket.

This problem consists of a square domain of $10 \times 10 \text{ m}^2$ with a rectangular object of dimensions $L = 2 \text{ m}$ width and $L/4 = 0.5 \text{ m}$ height placed at the middle of the computational domain as illustrated in Fig. 10. This object is initially submerged in almost pure water ($\alpha = 0.999$) and, as the simulation begins, it moves to the left at a constant speed of 35 m/s creating a high pressure region in front of it and a low pressure region behind it. If the fluid pressure behind the object is lowered below the vapour pressure of the liquid, a bubble of vapour forms encompassing the object. This phenomena is also known as *supercavitation*. As the solvers used in this work do not explicitly treat cavitation, we refer to this benchmark as *pseudocavitation* instead. Furthermore, we are only interested in the feasibility of coupling an incompressible solver in the outer region of the flow while keeping a compressible solver in the region close to the object:

on the one hand, the incompressible solver assumes a constant density of the water and air phases and thus is unable to reproduce a phase change around the object when the pressure drops below the vapour pressure; on the other hand, the compressible solver does allow for air density changes and, eventually, a mixture phase change when appropriate thermodynamic conditions are met. The computational domain is decomposed into 104×86 mesh cells with a slight stretching applied towards the boundaries. Symmetry plane boundary conditions are set at the bottom and top boundaries while an inlet boundary condition with specified velocity of $u_{1,0} = 35 \text{ m/s}$ is set at the left boundary and an outlet boundary condition with fixed pressure of $p_0 = 1 \text{ bar}$ is specified at the right boundary. Simulations are run up to the stationary solution removing gravitational forces from the system of equations and using a CFL number of 0.5. For the coupled simulations, the interface between the incompressible and compressible solvers is placed at the edges of a rectangle with vertices $(-2.5, -2.5)^T \text{ m}$ and $(5, 2.5)^T \text{ m}$ as shown in Fig. 10.

Fig. 10 also illustrates the pressure and velocity magnitude fields for the I–C simulation. Higher pressure values concentrate in front of the object indicating a compression in this zone while a region of more uniform pressure is progressively created further along the object. Velocity fields denote a high shear flow around and behind the object close to the phase change zone. Fig. 11 compares the solutions obtained with the three different solvers along the vertical line $x_1 = 3 \text{ m}$ situated behind the object. The “twoPhaseEulerFoam” solver (2P) is another compressible two-phase flow OpenFOAM pressure-based solver accounting for two mass balance equations, two momentum equations and two energy equations [29]. Another significant difference of this solver compared to compressibleInterFoam is that the free surface is not handled with any artificial compression term, which can result in smearing of the interface even on relatively fine meshes. Fig. 11(a) shows the water volume fraction profile with the phase change taking place in a cavity around 2 m wide. The same profile corresponding to the 2P simulation appears to exhibit a bit more smoothing and no significant differences can be observed between the C, I–C and 2P results. While the I and I–C solutions for the pressure profiles remain almost identical, see Fig. 11(b), differences arise when they are compared against the 2P solution. The cavity remains at a constant pressure of 1 bar for the three different solutions as specified by the right boundary condition. Finally, Fig. 11(c) shows the velocity magnitude profiles. The 2P solution is

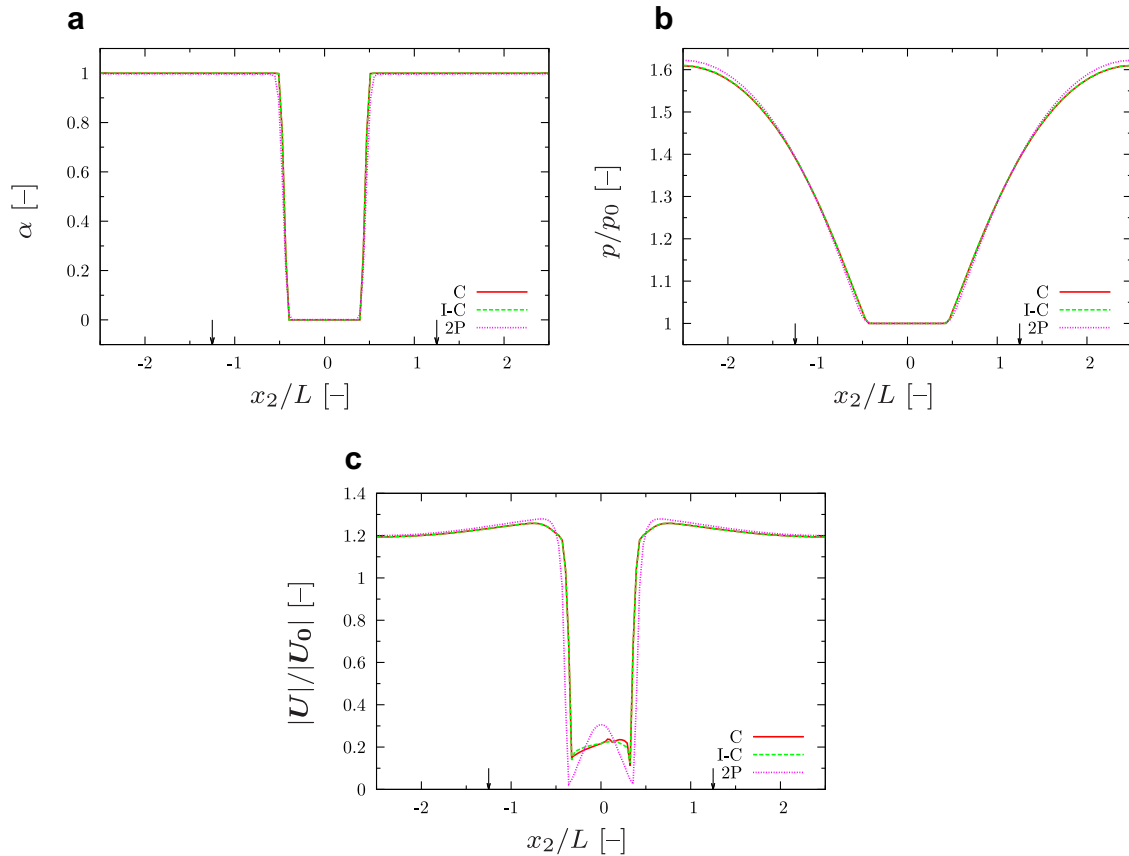


Fig. 11. Pseudocavitation benchmark: normalized distributions of the (a) water volume fraction, (b) pressure and (c) velocity magnitude along the line $x_1 = 3$ m; arrows indicate the position of the interface.

smoother than the other two and conserves symmetry with respect to line $x_2 = 0$ m. The C and I-C solutions agree well within the shear flow region but both remain slightly asymmetrical with respect the horizontal axis.

Fig. 12 displays the previous quantities along the horizontal line $x_2 = 1$ m above the object. While the C and I-C solutions remain almost identical, differences between the 2P solution become overall more pronounced. It can be deduced from the water volume fraction profile, Fig. 12(a), that the 2P solver predicts the formation of a cavity around 1 m earlier than the other two solvers. Contrarily, the pressure profiles do not seem to disagree to such an extent, especially from $x_1 = -1$ m, as shown in Fig. 12(b). Differences in the velocity profiles reported in Fig. 12(c) might arise from the two phase momentum equations integrated in twoPhaseEulerFoam as compared to compressibleInterFoam, which solves for one single mixture momentum equation. This combined with different values of α_w and α_a at $x_1 \geq 0$ m within the shear layer region, see Fig. 12(a), might lead to the differences observed in the velocity field in Fig. 12(c).

This pseudocavitation benchmark, although not focused on the cavitation problem per se, has shed some light on the viability of coupling different solvers to account for different physical phenomena, i.e. cavitation or phase change, within a unique simulation framework. The excellent agreement between the C and I-C solutions indicates that, in principle, it would be possible to couple a more appropriate solver explicitly to handle cavitation. As a matter of fact, we have conducted this simulation with our in-house two-phase compressible density-based solver AMAZON-CW [25]. Even though the simulation showed a developing cavity encompassing the object, it stopped abruptly before reaching the steady state and, consequently, at the present time we were unable to compare the final results against those presented in Figs. 11 and 12.

3.5. Water column free drop in a closed tank

This benchmark is extracted from the dedicated sessions on Sloshing at the ISOPE conferences [45,46] and has also been considered in other numerical studies [23,25]. It consists of a 20×15 m² closed tank which initially contains a rectangular water column with dimensions 10×8 m² surrounded by air at $p_0 = 1$ bar as illustrated in Fig. 13. The four walls of the tank share a no-slip boundary condition. When the simulation begins, the water column drops under gravity and impacts upon the bottom of the tank at $t \approx 0.65$ s. A small amount of air is trapped between the water column and the bottom of the tank at the moment of impact. The air is not only compressed due to the impact, but also expands when the air pressure exceeds the liquid pressure. The prediction of the maximum impact pressure is of importance in liquid sloshing applications and therefore two fine meshes with 800×600 and 1600×1200 uniform cells have been retained in this work. Simulations are run up to $t = 1$ s with CFL = 0.5 and, additionally, an interface has been placed at $x_2 = 2.5$ m to separate the incompressible region (top) and the compressible region (bottom) for the coupled simulations (see Fig. 13).

Fig. 14 shows the results obtained. Firstly, it is worth mentioning that this particular problem remains symmetrical and the highest pressure is found at the bottom centre of the tank. Secondly, the three interface-capturing solvers (I, C and I-C) maintain a sharp free surface and therefore a relatively accurate impact pressure can be predicted with 800×600 mesh cells. However, the 2P solver tends to smooth the free surface and a finer mesh of 1600×1200 cells is required to produce similar peak pressure values. Fig. 14(a) shows the time history of the pressure plotted every time step at the bottom centre of the tank. The incompressible solver predicts the earliest and lowest pressure peak at $t \approx 0.632$ s followed by non-physical

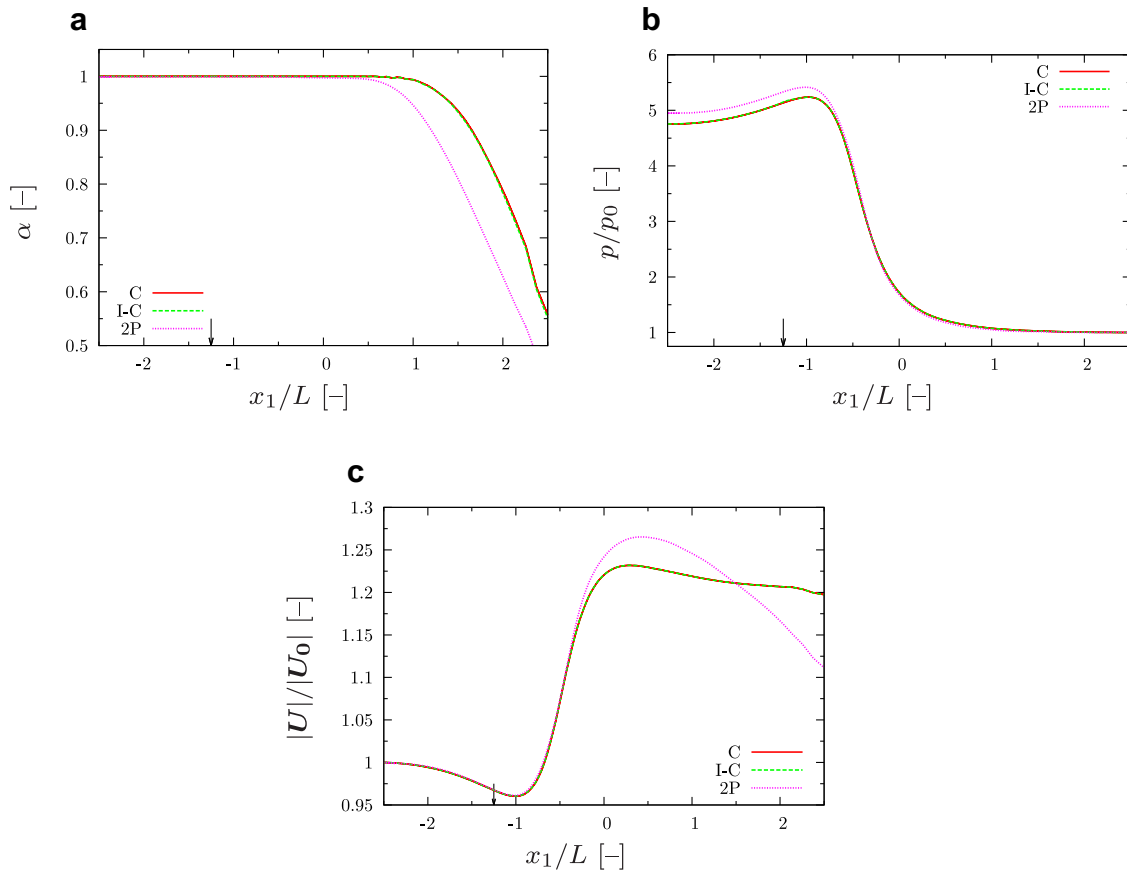


Fig. 12. Pseudocavitation benchmark: normalized distributions of the (a) water volume fraction, (b) pressure and (c) velocity magnitude along the line $x_2 = 1$ m; arrows indicate the position of the interface.

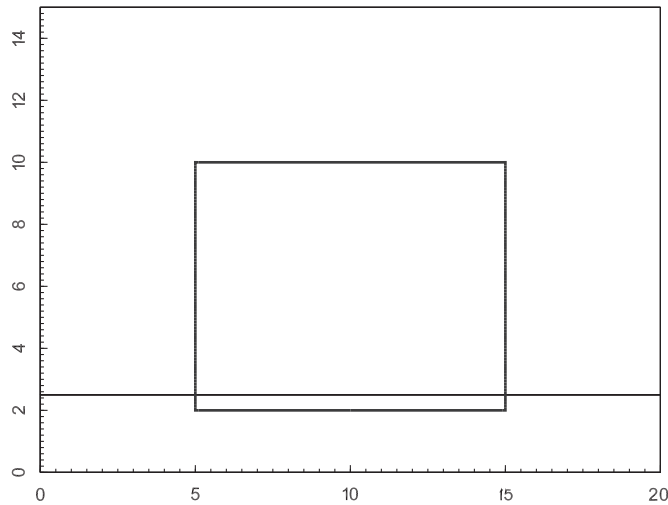


Fig. 13. Water column free drop benchmark: free surface contour plotted at $t = 0$ s (I simulation); the horizontal line at $x_2 = 2.5$ m is the interface.

negative pressure oscillations which render this solver unsuitable for compressible problems like this one. The compressible solver predicts a higher peak pressure around 50 bar at $t \approx 0.65$ s while the incompressible–compressible coupled solution shows the same tendency but with a slightly earlier impact prediction of order 0.5% with respect to the compressible solution. Finally, the 2P pressure history reveals a lower peak pressure of 45 bar taking place around $t = 0.65$ s.

Table 5

Water column free drop benchmark: simulation speed up for a mesh of 300×225 cells; times are normalized by $t_{\text{ref}} = 1253.22$ s.

Cores	I	C	I-C	2P
1	1.17	4.94	2.20	1.00
2	2.22	7.70	4.03	1.55
4	3.50	9.96	5.40	2.00

Fig. 14(b) compares the present results against previous data from the literature [23,45] and the results obtained with our in-house two-phase compressible density-based solver AMAZON-CW [25]. The fact that the present peak pressures are somewhat higher than the previous data can be attributed to the use of finer and square meshes in the present work. Apart from the differences in the peak values, overall, all the solutions predict a maximum pressure at $t \lesssim 0.65$ s. It is interesting to mention that the added compressibility in the air phase in the C and I-C solutions results in predictions of much higher peak pressure. In fact, the peak pressures thus obtained are of the same order of magnitude as the ones calculated with the 2P and AMAZON-CW solvers, in which both the water and air phases are treated as compressible fluids. Moreover, we highlight the fact that the use of a surface-capturing solver maintains a sharp free surface in order to correctly capture the peak pressure with a moderate number of mesh cells.

Finally, Table 5 compares the simulation speed up using different solvers and numbers of cores. The 2P solver is the slowest followed by the I solver which also struggles with both the accuracy of the

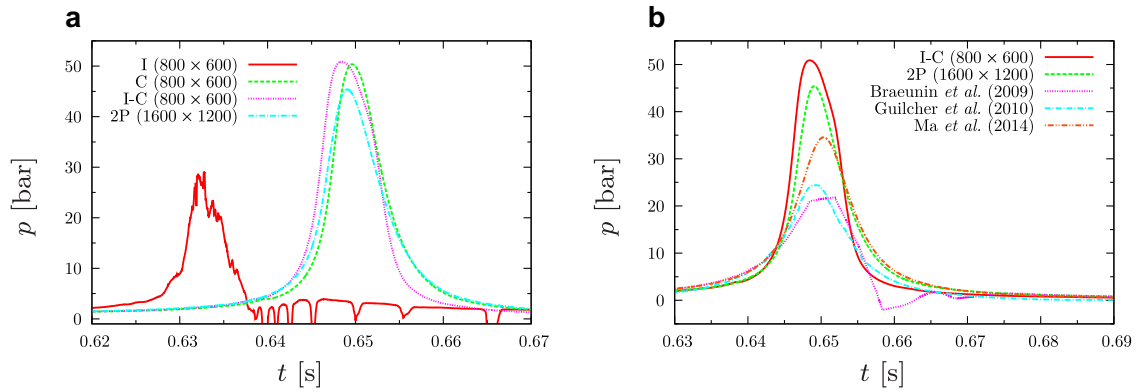


Fig. 14. Water column free drop benchmark: (a) time history of the maximum pressure for the present solvers and (b) comparison against previous results.

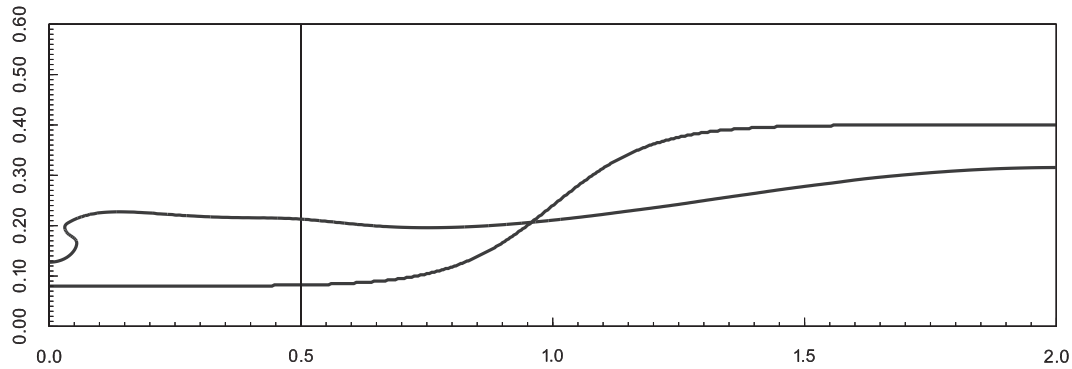


Fig. 15. Wave impact benchmark: free surface contour plotted at $t = 0$ s and $t = 0.58$ s (I-C simulation); the vertical line at $x_1 = 0.5$ m is the interface.

results and the computational times. The C solver is the fastest whereas the speed of the coupled I-C solver remains in between the I and C solvers. Nevertheless, the scalability of the incompressible solver improves with an increasing number of cores reaching almost 3 (in absolute terms) when using 4 cores while it appears to stagnate at 2 in the case of compressible solvers.

3.6. Wave impact

The aim of the present test is to verify the numerical methods in the presence of violent wave impacts involving air pockets and aeration effects. A wave tank of dimensions 2×0.6 m² open to the atmosphere ($p = 1$ bar) at the top boundary initially contains a volume of water whose free surface is defined by $0.24 + 0.16 \tanh(5(x - 1))$ as illustrated in Fig. 15. The use of an analytical expression to define the initial conditions allows us to simplify the treatment of the boundary conditions to generate an appropriate wave [25,47] and focus exclusively on the wave impact against the wall. A no-slip boundary treatment is applied at the walls of the tank. The domain is uniformly divided into 800×240 mesh cells and the Courant number is set to 0.5. However, when using the 2P solver a much finer computational domain of 3200×960 mesh cells is necessary due to the numerical diffusion of the free surface. Finally, for the coupled simulations, the interface separating the incompressible and compressible regions is placed at $x_1 = 0.5$ m, see Fig. 15.

It can be seen from Fig. 15 that the aforementioned initial free surface profile allows us to create an air pocket trapped by the water body near the wall at $t = 0.58$ s. Fig. 16 shows the quantitative results obtained with one incompressible (I) and three compressible two-phase solvers (2P, C and I-C). Several numerical probes distributed along the vertical wall at the left boundary register the pressure every time step. In general terms, it can be deduced that compressibility

becomes important in the presence of an air pocket. The compressible results obtained in this work agree qualitatively with previous experimental and numerical works in similar conditions, e.g. wave impact loads at a vertical wall [24–26,48,49]: the pressure responds as the air pocket is compressed and expands several times reaching values below the ambient pressure, which also can be reproduced by the compressible solvers; however, the incompressible solver cannot simulate such behaviour because of the constant density assumption. The present incompressible result indicates a sharp peak pressure of almost 1.24 bar at $x_2 = 0.15$ m at around 0.59 s without any following oscillations. For the same conditions, the 2P solution shows a peak pressure around 1.3 bar at the highest position of $x_2 = 0.24$ m, which is also accompanied by pressure oscillations reaching values below the ambient pressure. On the other hand, the C and I-C solutions indicate the presence of stronger oscillations along with maximum peak pressures slightly inferior to 1.5 bar. Similarly to the water column free drop case discussed earlier, there is a shift between the peak pressures corresponding to the C and I-C simulations. Being that air is the only compressible fluid component in the C and I-C solutions, it can be deduced that the oscillations shown in Fig. 16(c)–(d) are due to successive compressions and expansions of the air pocket entrapped between the wall and the body of water, which agree qualitatively well with experimental reports in liquid sloshing tanks under similar conditions [19].

Finally Table 6 shows the simulation speed up. The 2P solver is again the slowest while the C solver remains the fastest. The coupled I-C solver speed up lies between the original I and C solvers speed up indicating that no noticeable overhead is added to the simulation, which renders the coupling strategy suitable for HPC. In addition, for this particular problem, the C and I-C interface-capturing solvers perform much better than the 2P solver because of the relatively coarse meshes that are used.

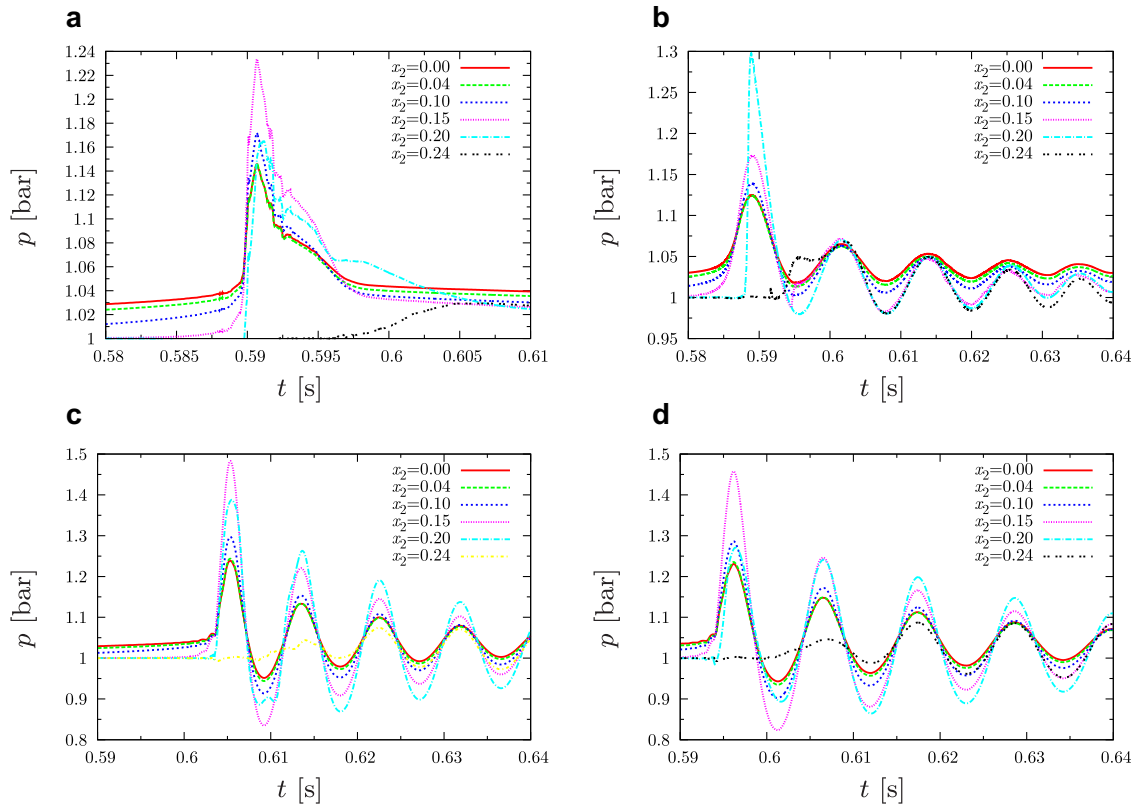


Fig. 16. Pseudocavitation benchmark: time history of the pressure at the wall at different vertical positions for the (a) I, (b) 2P, (c) C and (d) I-C solvers.

Table 6

Wave impact benchmark: simulation speed up for a mesh of 400×120 cells; times are normalized by $t_{ref} = 522.25$ s.

Cores	I	C	I-C	2P
1	1.72	1.97	1.82	1.00
2	2.89	3.26	2.92	1.58
4	4.35	4.73	4.30	2.08

4. Conclusions

We have presented a novel finite-volume coupling strategy that has been successfully applied to two-phase hydrodynamic free surface flows. It relies on a partitioned approach in which an incompressible and compressible two-phase solver, used in different regions of the computational domain, exchange information through a set of appropriate boundary conditions based on face interpolation from neighbouring cell values. This method has been verified to be stable and robust and ready for HPC in a wide variety of preliminary test cases, including naval, offshore and coastal engineering applications. Moreover, the viability of performing NWT simulations using specialized methods, i.e. incompressible and compressible interface-capturing solvers, in order to predict compressibility and aeration effects has also been verified and compared against other independent two-phase solvers through a series of benchmarks. Finally, this coupling strategy has been implemented in the popular open-source software OpenFOAM, but it can also be applied in a similar manner to any finite-volume solver.

Future work remains to extend the functionality of the current coupling procedures to cases in which neighbouring cells may not be conformal and thus further interpolation procedures may be required. Tight coupling as referred to in Section 2.3 may also be

envisaged in the future. It is also our intention to couple a wider range of relevant solvers with the present ones, e.g. a NLFP solver for incident wave propagation and a CSD code for structural interaction and deformation, in order to include the more complex physics characteristic of realistic NWT FSI simulations.

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.

References

- [1] Chahine GL, Kalumuck KM, Hsiao C-T. Simulation of surface piercing body coupled response to underwater bubble dynamics utilizing 3DYNAFS, a three-dimensional BEM code. *Comput Mech* 2003;32:319–26.
- [2] Campbell RL. Towards a stable and accurate coupling of compressible and incompressible flow solvers. The Pennsylvania State University; 2010. [Ph.D. thesis].
- [3] Liao K, Hu C, Sueyoshi M. Numerical simulation of free surface flow impacting on an elastic plane. In: *Proceedings of the 29th international workshop on water waves and floating bodies*; 2014.
- [4] Popko W, Vorpahl F, Zuga A, Kohlmeier M, Jonkman J, Robertson A, et al. Offshore code comparison collaboration continuation (OC4), phase I—results of coupled simulations of an offshore wind turbine with jacket support structure. *J Ocean Wind Energy* 2014;1:1–11.
- [5] Jaure S, Duchaine F, Guicquel LYM. Comparisons of coupling strategies for massively parallel conjugate heat transfer with large eddy simulation. In: *Proceedings of the IV international conference on computational methods for coupled problems in science and engineering*; 2011.
- [6] Keyes DE, McInnes LC, Woodward C, Gropp W, Myra E, Pernicé M, et al. Multi-physics simulations: challenges and opportunities. *Int J High Perform Comput Appl* 2013;27:4–83.
- [7] Sussman M, Puckett EG. A coupled level set and volume-of-fluid method for computing 3D and axisymmetric incompressible two-phase flows. *J Comput Phys* 2000;162:301–37.
- [8] Iafrati A, Campana EF. A domain decomposition approach to compute wave breaking (wave-breaking flows). *Int J Meth Fluids* 2003;41:419–45.

- [9] Schluter JU, Wu X, van der Weide E, Hahn S, Alonso JJ, Pitsch H, et al. AIAA Paper 2005:4997.
- [10] Schluter JU, Wu X, Kim S, Shankaran S, Alonso JJ, Pitsch H. A framework for coupling Reynolds-averaged with large-eddy simulations for gas turbine applications. *J Fluids Eng* 2005;127:806–15.
- [11] Frohlich J, von Terzi D. Hybrid LES/RANS methods for the simulation of turbulent flows. *Prog Aerosp Sci* 2008;44:349–77.
- [12] von Terzi DA, Frohlich J. Segregated coupling of large-eddy simulations with downstream Reynolds-averaged Navier–Stokes calculations. *Comput Fluids* 2010;39:1314–31.
- [13] Zhang Y, Peszyńska M, Yim SC. Coupling of viscous and potential flow models with free surface for near and far field wave propagation. *Int J Num Anal Model Ser B* 2013;4:256–82.
- [14] Mattson K, Iourokina I, Ham F. Towards a stable and accurate coupling of compressible and incompressible flow solvers. Stanford University; 2005. Technical report.
- [15] Schleper V. On the coupling of compressible and incompressible fluids. Numerical methods for hyperbolic equations. Taylor & Francis Group; 2012.
- [16] de Dituviis N, de Socio LM. Impact of floats on water. *J Fluid Mech* 2002;471:365–79.
- [17] Greaves DM. Simulation of viscous water column collapse using adapting hierarchical grids. *Int J Numer Methods Fluids* 2005;50:693–711.
- [18] Lubin P, Vincent S, Abadie S, Caltagirone J-P. Three-dimensional large eddy simulation of air entrainment under plunging breaking waves. *Coast Eng* 2006;53:631–55.
- [19] Lugni C, Brocchini M, Faltinsen OM. Evolution of the air cavity during a depressurized wave impact. II. The dynamic field. *Phys Fluids* 2010;22:056102.
- [20] Bullock GN, Obhrai C, Peregrine DH, Bredmose H. Violent breaking wave impacts. Part 1: results from large regular wave tests on vertical and sloping walls. *Coast Eng* 2007;54:602–17.
- [21] Bredmose H, Peregrine D, Bullock G. Violent breaking wave impacts. Part 2: modelling the effect of air. *J Fluid Mech* 2009;641:389–430.
- [22] Plumerault L-R, Astruc D, Villedieu P, Maron P. A numerical model for aerated-water wave breaking. *Int J Numer Methods Fluids* 2012;69:1851–71.
- [23] Guilcher P-M, Oger G, Brosset L, Jacquin E, Grenier N, Trouz DL. Simulation of liquid impacts with a two-phase parallel SPH model. In: Proceedings of 20th international offshore and polar engineering conference (ISOPE); 2010.
- [24] Guilcher P-M, Brosset L, Couty N, Trouz DL. Simulations of breaking wave impacts on a rigid wall at two different scales with a two phase fluid compressible SPH model. In: Proceedings of 22th international offshore and polar engineering conference (ISOPE); 2012.
- [25] Ma ZH, Causon DM, Qian L, Mingham CG, Gu HB, Ferrer PM. A compressible multiphase flow model for violent aerated wave impact problems. *Proc R Soc A* 2014;470:2172.
- [26] Lugni C, Brocchini M, Faltinsen OM. Wave impact loads: the role of the flip-through. *Phys Fluids* 2006;18:122101.
- [27] Peregrine DH, Bredmose H, Dunn DC, Somerville TA, Bullock GN, Obhrai C, et al. 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.
- [28] Jasak H. Error analysis and estimation for the finite volume method with applications to fluid flows. University of London; 1996. [Ph.D. thesis].
- [29] Rusche H. Computational fluid dynamics of dispersed two-phase flows at high phase fractions. University of London; 2002. [Ph.D. thesis].
- [30] Higuera P, Lara JL, Losada IJ. Realistic wave generation and active wave absorption for Navier–Stokes models. Application to OpenFOAM. *Coast Eng* 2013;71:102–18.
- [31] Higuera P, Lara JL, Losada IJ. Simulating coastal engineering processes with OpenFOAM. *Coast Eng* 2013;71:119–34.
- [32] Weller HG. Derivation, modelling and solution of the conditionally averaged two-phase flow equations. Nabla Ltd.; 2002. Technical report TR/HGW/02.
- [33] Moukalled F, Darwish M. A unified formulation of the segregated class of algorithms for fluid flow at all speeds. *Numer Heat Transf B: Fundam* 2000;37:103–39.
- [34] Leveque RJ. Finite-volume methods for hyperbolic problems. Cambridge University Press; 2002.
- [35] Farrell PE, Maddison JR. Conservative interpolation between volume meshes by local Galerkin projection. *Comput Method Appl M* 2011;200:89–100.
- [36] Turek S, Hron J, Razzaq M, Wobker H, Schafer M. Numerical benchmarking of fluid–structure interaction: a comparison of different discretization and solution approaches. In: Bungartz HJ, Mehl M, Schafer M, editors. Fluid structure interaction II: modeling, simulation and optimization. Springer; 2010.
- [37] Kissling K, Springer J, Jasak J, Schutz S, Urban K, Piesche M. 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; 2014.
- [38] Issa R. Solution of the implicitly discretised fluid flow equations by operator-splitting. *J Comput Phys* 1986;62:40–65.
- [39] Issa R, Gosman A, Watkins A. The computation of compressible and incompressible recirculating flows by a non-iterative implicit scheme. *J Comput Phys* 1986;62:66–82.
- [40] Martin JC, Moyce WJ. Part IV. An experimental study of the collapse of liquid columns on a rigid horizontal plane. *Phil Trans R Soc Lond A* 1952;244:312–24.
- [41] Murrone A, Guillard H. A five equation reduced model for compressible two phase flow problems. *J Comput Phys* 2005;202:664–98.
- [42] Davoudzadeh F, McDonald H, Thompson BE. Accuracy evaluation of unsteady CFD numerical schemes by vortex preservation. *J Comput Phys* 1995;24:883–95.
- [43] Yee HC, Vinokur M, Djomehri MJ. Entropy splitting and numerical dissipation. *J Comput Phys* 2000;162:33–81.
- [44] Ferrer PJM, Buttay R, Lehnasch G, Mura A. A detailed verification procedure for compressible reactive multicomponent Navier–Stokes solvers. *Comput Fluids* 2014;89:88–110.
- [45] Braeunig J-P, Brosset L, Dias F, Ghidaglia J-M. Phenomenological study of liquid impacts through 2D compressible two-fluid numerical simulations. In: Proceedings of 19th international offshore and polar engineering conference (ISOPE); 2009.
- [46] Dias F, Brosset L. Comparative numerical study: description of the calculation case. In: Proceedings of 20th international offshore and polar engineering conference (ISOPE); 2010.
- [47] Qian L, Causon DM, Mingham CG, Ingram DM. A free surface capturing method for two fluid flows with moving bodies. *Proc R Soc Lond A* 2006;462:21–42.
- [48] Verhagen J. The impact of a flat plate on a water surface. *J Ship Res* 1967;11:211–23.
- [49] Yang Q, Qiu W. Numerical simulation of water impact of 2D and 3D bodies. *Ocean Eng* 2012;43:82–9.