Consiglio Nazionale delle Ricerche

Istituto di Matematica Applicata e Tecnologie Informatiche

PUBBLICAZIONI

Enrico Bertolazzi, Gianmarco Manzini

A MIXED FINITE ELEMENT METHOD FOR LIQUID-SOLID AND LIQUID-LIQUID IMPACTS

N. 3-PV 2003

A Mixed Finite Element Method for Liquid-Solid and Liquid-Liquid Impacts

Enrico Bertolazzi[1] and Gianmarco Manzini[2]

[1] *Dip. Ingegneria Meccanica e Strutturale, Università di Trento, via Mesiano 77, I – 38050 Trento, Italy*
Fax : +39 0461 882599 Voice : +39 0461 882660

E-mail: Enrico.Bertolazzi@ing.unitn.it

[2] *Istituto di Matematica Applicata e Tecnologie Informatiche– CNR, via Ferrata 1, I – 37100 Pavia, Italy*
Fax : +39 0382 548300 Voice : +39 0382 548215

E-mail: Marco.Manzini@ian.pv.cnr.it

A mathematical model is formulated in the framework of the potential theory to describe the impact of a liquid jet on a rigid wall and on a liquid surface initially at rest. The solution of this free-surface flow problem is approximated numerically by a tracking-method of new conception. Basically, the free surface is tracked in time by numerically solving the evolutive equation based on Bernoulli's law assuming the 2-D free surface as a streamline. The steady mixed Dirichlet-Neumann Laplacian problem is solved by the RT_0 mixed finite element method on a computational domain whose shape evolves in time and at each time step the mesh is fully remeshed. The accuracy of the potential flow solver has been checked by simulating liquid-solid and liquid-liquid impacts.

Key Words: Free Surface Simulations, Air entrapment process, Mixed Finite Elements

1. INTRODUCTION

Liquid–liquid and liquid–solid impact is a broad subject which includes relevant parts of fluid mechanics; interest in this topic arose in the early decades of this century – see, for instance, the review by [13].

Examples of such collision phenomena are wavemaker problem; sea–waves or tsunami impacting on breakwaters; hydrodynamic pressure on dams due to seismic waves; formation of glassy metals by a fluid jet; heat transfer from a jet fluid to a surface and the impact of debris–flows on check–dams.

These flows have important industrial applications as well as fundamental fluid mechanics interest. Among these many examples, we are concerned with two applications that have motivated the present work: the bore impact of a liquid mass onto a rigid wall and the impact of a liquid column

onto a plane liquid surface. In the former case, we are interested in modelling the hydrodynamic loading due to the impact of a bore on a wall that is crucial in the rational design of protection structures. In the latter case, we are interested in the investigation of the air entrapment process occurring when a column-shaped liquid jet penetrates into a liquid pool initially at rest.

In the present paper we address a new numerical method to investigate 2-D free-surface flow configurations. The free surface is an interface betweeen two fluids with very different physical properties, typically a gas and a liquid like, for example, air and water. Its shape and location continously evolve in time and the dynamics is mainly governed by the inertia of the liquid, because of the large difference in the densities of the two fluids: the ratio for water and air is about 10^3. Under this assumption, the gas-liquid interface is not constrained, but freely moving and it makes sense to introduce the wording of *free-surface flows*.

As both the interface shape and location can arbitrarly change in time, the numerical simulation of free-surface flows is a challenging problem. In the last decades several approximation methods have been proposed in the literature to treat numerically free-surface flows. For convenience's sake, we can group these methods in the following main families: *(i) the Lagrangian grid methods*, *(ii) the Eulerian grid methods*, and *(iii) the hybrid Eulerian-Lagrangian methods*.

The Lagrangian grid methods basically define and track a free surface by a grid which is imbedded in and moves with the fluid. As grid and fluid move together, the grid automatically tracks free surfaces. The main limitation of Lagrangian methods is that it is difficult to track surfaces that break apart or intersect. Even large amplitude surface motions can be difficult to track without introducing regridding techniques such as the *Arbitrary-Lagrangian-Eulerian* (ALE) method. References [11, 10] may be consulted for early examples of these approaches.

A different approach is instead devised by Eulerian methods, where the computing grid is kept fixed and fluid volumes instead of free surfaces are tracked in time. Free surfaces may thus appear, merge or disappear as the fluid volumes break apart or coalesce. Among the most popular methods that follow this approach, it is worth mentioning the Marker-And-Cell (MAC) method [9] and the Volume-Of-Fluid (VOF) method [17, 18] and all their many variants and improvements. Fluid volumes are tracked in the MAC method by a set of fluid marker particles, while a special indicator function, the volume fraction field, is used in the VOF method.

The marker particles of the MAC method have no volume, no mass or other significant physical properties but move attached to the fluid and identify the grid cells that are filled, those that are empty, and the "surface cells". In particular, the latter ones are defined on the basis of a simple rule: surface cells must contain at least one fluid marker particle and also have at least one neighboring grid cell that is empty. The set of surface cells operatively defines the boundary of the volumes filled by the fluid, and a portion of the free surface is assumed to be present within any surface cell. The marker particles in the MAC methods are usually moved by locally-interpolated fluid velocities. These velocities are determined by taking into account external free-surface boundary conditions such as the gas pressure and physical constraints such as the fluid incompressibility and the zero surface shear stress.

The volume fraction function of the VOF method is a step function having a value of either one or zero and is used to locate the position of the fluid on the underlying Eulerian grid. Shape and location of surfaces, as well as surface slopes and surface curvatures, are reconstructed by using the volume fraction of a given cell and the one of its neighbors. Surfaces lie in cells partially filled with fluid or between filled and empty cells. The volume fraction function is updated by solving a time-dependent convection equation whose numerical discretization applies standard shock-capturing techniques to control numerical diffusion and dispersion and preserve the step function nature of the indicator.

The *Level Set* method has been originally developed to solve moving interface problems [21] and then applied to free surface fluid problems [29], particularly to the investigation of the motion of air and water bubbles. The zero level set of a suitable scalar variable is used to determine the position of the interface. This variable is continuous, smooth and monotonic in the direction normal to the interface. Its value is updated in time by solving the advection equation that predict theoretically the interface shape evolution. Numerical diffusion can occur due to the discretization, thus resulting in progressively worse recovery of the zero level set. Several modifications to the original method have been envisaged recently to overcome this difficulty and improve its effectiveness.

The new method that we discuss in this paper can be considered as an interface tracking method. Basically, the new method is capable of following in the framework of the potential theory of section 2 the interface position on the computational domain over a long period of time and determining how its movement affects the flow configuration. As the interface equation is derived from Bernoulli's law, the new method assumes that the gas-liquid interface is an equipotential surface for the liquid fluid, and that the potential theory applies. Shape and location of the free surface in the computational domain have been defined operatively by introducing a set of surface nodes. The position of these free-surface nodes is calculated in time by the suitable non-linear algorithm presented in section 3. The interface velocity field that must be properly applied to the free surface nodes to update their position is given by solving a steady mixed Dirichlet-Neumann Laplacian problem on a domain whose shape is continously evolving in time. The mixed Finite Element method that is shortly described in section 4 approximates numerically the solution to this Laplacian problem in the lowest order Raviart-Thomas (RT_0) space. The new method can be assimilated to an hybrid Lagrangian-Eulerian method where a set of special marker points are located at the interface to track explicitly the free-surface movements. However, despite of the standard Eulerian methods, there is not a fixed underlying mesh. Instead, the computational domain is re-meshed at any time step to take into account the position and shape evolution of the free surface.

We feel this latter issue as a crucial aspect of this new method that deserves a deeper discussion. We first note that potential formulations and their numerical approximations have been investigated in the past decades mainly using the *Boundary Element Method* (BEM). The pioneering work of Reference [14] illustrated how an integral equation formulation can be successful in treating numerically transient non-linear problems. Other early applications are found in References [7, 8], which proposed a time integration scheme based on a Taylor series development, and References [19, 20, 33] which investigate liquid-liquid impact problems and are very close to the application problems that we are interested on. We also mention References [15] and [3] among the many more recent papers using BEMs applied to potential theory-based model. In the former paper accuracy and stability are investigated and assessed when using a BEM based on a B-spline implementation. In the latter one a BEM is applied to the resolution of non-linear free-surface problems in two-dimensional and axisymmetric configurations where time integration is given by a truncated forward-time Taylor series development. In all these works BEM has proved accurate and computationally efficient, and the authors of these papers generally claim that computational efficiency is mainly due to the avoidance of regridding the flow region at each time step during the simulation. Such a regridding is not required in BEM but is instead needed to follow the evolution of the free surface in other integration methods such as the Lagrangian ones presented above.

Nonetheless, it is our experience that such a regridding is not so expensive in terms of CPU time as it might appear at a first glance when compared to the cost of other parts of the numerical scheme, such as the resolution of the Laplace's problem involved in the potential theory formulation. To assess this issue, we extensively carried out a set of CPU time measurements on a range of

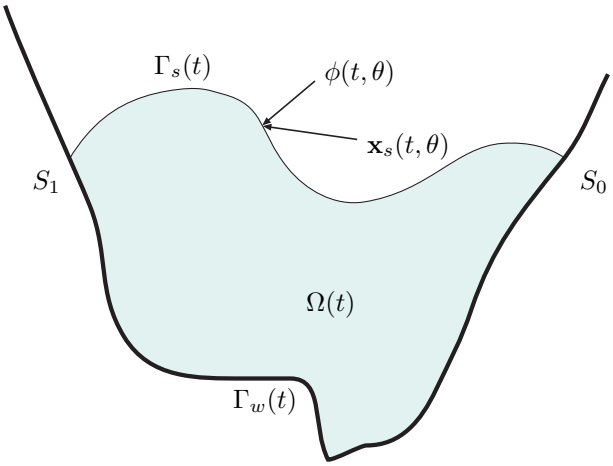

FIG. 1. Sketch of the model problem.

significant test cases as reported in sections 6 and 7. Therein, results of numerical simulations are given to illustrate both the capability of the method in predicting the behavior of free flow configurations on liquid-solid and liquid-liquid collision problems and its computational efficiency.

We stress that the software that has been utilized in our solver implementation is public domain available and absolutely general purpose, that is non particularly optimized to treat these kind of problems. The generation of the new mesh performed at each time step is obtained by calling the public domain mesh generator TRIANGLE [26, 27, 28] incorporated in our C^{++} solver as a library sub-function. All the geometrical and topological data related to the mesh are obtained by a software developed on P2MESH [1], a publicly available collection of C^{++} classes suited to this purpose.

On the basis of our experience, we claim that the generation of a new mesh and related data usually takes only some percents of the total CPU time required by a complete numerical simulation, the great part of the time-consuming calculations being devoted to the resolution of the Laplace's problem.

Finally, as stated in the section 8, it is the authors' feeling that the new method makes possible to develop both a qualitative and a quantitative understanding of the physics involved in these impact processes.

2. THE ANALYTICAL FORMULATION OF THE PROBLEM

The mathematical model formulated in this section to study liquid-solid and liquid-liquid impact phenomena is based on the potential theory under the assumptions that the fluid is incompressible and the flow is irrotational.

The computational liquid domain $\Omega(t)$, as sketched in Fig. 1, is assumed to be simply connected and time dependent.

Its boundary is assumed to be piecewise Lipschitz continuous and is defined as

$$\partial\Omega(t) = \Gamma^s(t) \cup \Gamma^w(t),$$

including the *moving boundary line*

$$\Gamma^s(t) = \left\{ \mathbf{x} \mid \mathbf{x} = \mathbf{x}^s(t,\theta), t \geq 0,\, \theta \in [0,\, 1] \right\},$$

and the rigid impermeable *wall boundary line*

$$\Gamma^w(t) = \left\{ \mathbf{x} \mid \mathbf{x} = \mathbf{x}^w(t,\theta), t \geq 0,\, \theta \in [0,\, 1] \right\}.$$

Both curves $\mathbf{x}^s(t,\theta)$ and $\mathbf{x}^w(t,\theta)$ are suitably parameterized by the scalar variable θ running in the closed interval $[0,1]$. These two boundary curves intersect at the two distinct moving nodes $\mathbf{S}_0(t,\theta)$ and $\mathbf{S}_1(t,\theta)$ such that

$$\begin{aligned} \mathbf{S}_0(t,\theta) &= \mathbf{x}^s(t,0) = \mathbf{x}^w(t,1), \\ \mathbf{S}_1(t,\theta) &= \mathbf{x}^w(t,0) = \mathbf{x}^s(t,1). \end{aligned}$$

The fluid motion is described by the velocity potential field $\Phi = \Phi(t,\mathbf{x})$ and the velocity field $\mathbf{v}(t,\mathbf{x})$, that are the solution of the Laplace's boundary value problem

$$\begin{aligned} \nabla\Phi(t,\mathbf{x}) &= \mathbf{v}(t,\mathbf{x}), && \mathbf{x} \in \Omega(t), \\ \nabla \cdot \mathbf{v}(t,\mathbf{x}) &= 0, && \mathbf{x} \in \Omega(t), \end{aligned} \tag{1}$$

at any instant $t \geq 0$, with mixed Dirichlet and homogeneous Neumann boundary conditions at respectively $\Gamma^s(t)$ and $\Gamma^w(t)$

$$\begin{aligned} \mathbf{v}(t,\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) &= 0, && \mathbf{x} \in \Gamma^w(t), \\ \Phi(t,\mathbf{x}^s(t,\theta)) &= \phi(t,\theta), && \theta \in [0,1], \end{aligned} \tag{2}$$

where $\phi(t,\theta)$ is the potential field at $\Gamma^s(t)$.

The fluid energy conservation expressed by the Bernoulli's law requires that

$$\frac{\partial \phi}{\partial t}(t,\theta) = \nabla\Phi(t,\mathbf{x}^s(t,\theta)) \cdot \mathbf{v}^s(t,\theta) - \frac{1}{2}\left|\nabla\Phi(t,\mathbf{x})\right|^2 - g\mathbf{x} \cdot \hat{\mathbf{z}}, \tag{3}$$

where $\mathbf{v}^s(t,\theta)$ is the velocity of the points at the free moving boundary line $\Gamma^s(t)$, g is the scalar gravity constant and $\hat{\mathbf{z}}$ is the versor along the vertical direction taken positive upward. Clearly, there holds

$$\frac{\partial \mathbf{x}^s}{\partial t}(t,\theta) = \mathbf{v}^s(t,\theta),$$

for any $t \geq 0$, and $\theta \in [0,1]$.

The closure of this model requires the knowledge of $\mathbf{v}^s(t,\theta)$ and $\nabla\Phi(t,\mathbf{x})$. Assuming that the curve $\mathbf{x}^s$ in the definition of Γ^s is a free streamline and introducing a suitable parameterization, we obtain the following model equation

$$\mathbf{v}^s(t,\theta) = \nabla\Phi(t,\mathbf{x}^s(t,\theta)) + s(t,\theta)\frac{\partial \mathbf{x}^s}{\partial \theta}(t,\theta). \tag{4}$$

The scalar field $s(t,\theta)$ in equation (4) is representative of the chosen parameterization, and is used to control numerical instabilities of the free-boundary line. This purpose has been pursued in this work by setting in the evolution equation (4)

$$s(t,\theta) = \varepsilon \left| \frac{\partial^2 \mathbf{x}^s}{\partial s^2} \right| \left| \frac{\partial \mathbf{x}^s}{\partial \theta} \right|^{-1} \tag{5}$$

where $s(\theta) = \displaystyle\int_0^\theta |d\mathbf{x}^s/d\theta|\, d\theta$ is the arc length, and ε a coefficient whose value is a decreasing function of the mesh size and will be specified in the next section. The appendix gives the major derivation details about equations (3) and (4).

As the initial fluid state is known, we obtain the value of the potential field at $t = 0$, that is

$$\Phi(0,\mathbf{x}) = \Phi_0(\mathbf{x}), \quad \mathbf{x} \text{ in } \Omega(0).$$

The initial state of the free boundary line $\Gamma^s(t)$ is similarly given by

$$\begin{aligned} \mathbf{x}^s(0,\theta) &= \mathbf{x}_0^s(\theta), & \theta \text{ in } [0,1], \\ \phi(0,\theta) &= \Phi_0(\mathbf{x}^s(\theta)), & \theta \text{ in } [0,1], \end{aligned}$$

where $\mathbf{x}_0^s(\theta)$ is a suitable parametric form of a free streamline taken at $t = 0$. The potential of the initial flow field will be specified in the sections dealing with the numerical simulations.

3. THE SOLUTION ALGORITHM

The domain of definition of θ is divided into N sub-intervals $[\theta_i, \theta_{i+1}]$, with $\theta_i = i/N$ for $i = 0, \ldots, N$. For the sake of conciseness, in all the formulae of this paper when no otherwise indicated the index i labeling θ_i is running over the whole node set, i.e. $i = 0, \ldots, N$. These definitions are also useful:

$$\begin{aligned} \mathbf{x}_i(t) &= \mathbf{x}^s(t,\theta_i), & \mathbf{x}_i^n &= \mathbf{x}^s(t^n,\theta_i), \\ \phi_i(t) &= \phi(t,\theta_i), & \phi_i^n &= \phi(t^n,\theta_i), \\ s_i(t) &= s(t,\theta_i), & s_i^n &= s(t^n,\theta_i), \end{aligned}$$

where t^n is the n-th time level. The mesh size dependent coefficient in the definition of $s(t,\theta_i)$ given in (5) is $\varepsilon = \dfrac{1}{2N}$. The discrete formulation for the free boundary line location and potential read as

$$\begin{aligned} \frac{\partial \mathbf{x}_i}{\partial t} &= \nabla\Phi(t,\mathbf{x}_i) + s_i(t) \frac{\mathbf{x}_{i+1}(t) - \mathbf{x}_{i-1}(t)}{\theta_{i+1} - \theta_{i-1}}, \\ \frac{\partial \phi_i}{\partial t} &= \frac{1}{2} \left|\nabla\Phi(t,\mathbf{x}_i)\right|^2 + s_i(t) \frac{\mathbf{x}_{i+1}(t) - \mathbf{x}_{i-1}(t)}{\theta_{i+1} - \theta_{i-1}} \cdot \nabla\Phi(t,\mathbf{x}_i) - g\mathbf{x}_i \cdot \hat{\mathbf{z}}. \end{aligned}$$

It is assumed that the free boundary line location $\mathbf{x}_i^n$ and the potential along the free boundary line ϕ_i^n are known at time t^n. The time marching is performed by the Crank-Nicholson method [6] by evaluating node positions at the free boundary line and the boundary potentials at time t^{n+1}

at the node labeled by i:

$$\begin{aligned}\mathbf{x}_i^{n+1} &= \mathbf{x}_i^n + \frac{\Delta t}{2}\left[\nabla\Phi(t^{n+1},\mathbf{x}_i^{n+1}) + \nabla\Phi(t^n,\mathbf{x}_i^n)\right] \\ &\quad + \frac{\Delta t}{2}\left[s_i^{n+1}\frac{\mathbf{x}_{i+1}^{n+1}-\mathbf{x}_{i-1}^{n+1}}{\theta_{i+1}-\theta_{i-1}} + s_i^n\frac{\mathbf{x}_{i+1}^n-\mathbf{x}_{i-1}^n}{\theta_{i+1}-\theta_{i-1}}\right], \qquad (6a)\\ \phi_i^{n+1} &= \phi_i^n + \frac{\Delta t}{4}\left[\left|\nabla\Phi(t^{n+1},\mathbf{x}_i^{n+1})\right|^2 + |\nabla\Phi(t^n,\mathbf{x}_i^n)|^2\right] \\ &\quad + \frac{\Delta t}{2}\left[s_i^{n+1}\frac{\mathbf{x}_{i+1}^{n+1}-\mathbf{x}_{i-1}^{n+1}}{\theta_{i+1}-\theta_{i-1}}\cdot\nabla\Phi(t^{n+1},\mathbf{x}_i^{n+1}) + s_i^n\frac{\mathbf{x}_{i+1}^n-\mathbf{x}_{i-1}^n}{\theta_{i+1}-\theta_{i-1}}\cdot\nabla\Phi(t^n,\mathbf{x}_i^n)\right] \\ &\quad - g\frac{\Delta t}{2}\left[\mathbf{x}_i^{n+1}+\mathbf{x}_i^n\right]\cdot\hat{\mathbf{z}}. \qquad (6b)\end{aligned}$$

The implicit nature of the Crank-Nicholson scheme calls for an iterative procedure to be started up with the initial values $\mathbf{x}_i^{n+(0)} = \mathbf{x}_i^n$ and $\phi_i^{n+(0)} = \phi_i^n$. The iteration proceeds until convergence is reached using the following fixed-point procedure:

$$\begin{aligned}\mathbf{x}_i^{n+(\ell+1)/L} &= \alpha\mathbf{x}_i^n + (1-\alpha)\mathbf{x}_i^{n+\ell/L} \\ &\quad + \alpha\frac{\Delta t}{2}\left[\nabla\Phi(t^{n+\ell/L},\mathbf{x}_i^{n+\ell/L}) + \nabla\Phi(t^n,\mathbf{x}_i^n)\right] \\ &\quad + \alpha\frac{\Delta t}{2}\left[s_i^{n+\ell/L}\frac{\mathbf{x}_{i+1}^{n+\ell/L}-\mathbf{x}_{i-1}^{n+\ell/L}}{\theta_{i+1}-\theta_{i-1}} + s_i^n\frac{\mathbf{x}_{i+1}^n-\mathbf{x}_{i-1}^n}{\theta_{i+1}-\theta_{i-1}}\right], \qquad (7a)\\ \phi_i^{n+(\ell+1)/L} &= \alpha\phi_i^n + (1-\alpha)\phi_i^{n+\ell/L} \\ &\quad + \alpha\frac{\Delta t}{4}\left[\left|\nabla\Phi(t^{n+\ell/L},\mathbf{x}_i^{n+\ell/L})\right|^2 + |\nabla\Phi(t^n,\mathbf{x}_i^n)|^2\right] \\ &\quad + \alpha\frac{\Delta t}{2}\left[s_i^{n+\ell/L}\frac{\mathbf{x}_{i+1}^{n+\ell/L}-\mathbf{x}_{i-1}^{n+\ell/L}}{\theta_{i+1}-\theta_{i-1}}\cdot\nabla\Phi(t^{n+\ell/L},\mathbf{x}_i^{n+\ell/L})\right. \\ &\quad \left. + s_i^n\frac{\mathbf{x}_{i+1}^n-\mathbf{x}_{i-1}^n}{\theta_{i+1}-\theta_{i-1}}\cdot\nabla\Phi(t^n,\mathbf{x}_i^n)\right] - \alpha g\frac{\Delta t}{2}\left[\mathbf{x}_i^{n+\ell/L}+\mathbf{x}_i^n\right]\cdot\hat{\mathbf{z}}. \qquad (7b)\end{aligned}$$

where $\ell = 0, 1, \ldots, L-1$ is the sub-iteration index at each time step, L is the number of iterations required to achieve convergence, $\alpha = 1/2$ is the relaxation parameter, and $\Phi(t^{n+\ell/L},\mathbf{x})$ is the solution of the mixed boundary value Laplace's problem (1), with Dirichlet condition at $\Gamma^s(t^{n+\ell/L})$ given by

$$\Phi(t^{n+\ell/L},\mathbf{x}^s(t^{n+\ell/L},\theta_i)) = \phi_i^{n+\ell/L}.$$

The solution $\Phi(t^{n+\ell/L},\mathbf{x})$ is numerically approximated at $t^{n+\ell/L}$ on the mesh triangulation $\Omega_h(t^{n+\ell/L})$ covering the computational domain $\Omega(t^{n+\ell/L})$. The number of iteration L to achieve convergence in the fixed-point iterative algorithm is generally dependent on the time-step iterative level n. At any time step t^n, the complete re-meshing of $\Omega_h(t^n)$ is performed by the mesh generator TRIANGLE, while the meshing of $\Omega_h(t^{n+\ell/L})$ for $\ell > 0$ is done by stretching the whole mesh

computed at the time t^n. The mesh manager useful to construct the MFE approximation has been performed by means of P2MESH [1], a free software package conceived for the fast development of Finite Volume and Finite Element codes on 2–D *unstructured* mesh. Convergence is achieved for L typically less than 10.

4. A MIXED FINITE-ELEMENT METHOD FOR LAPLACE'S EQUATION.

The coupled system of equations (1) in the potential Φ and the velocity field $\mathbf{v}$ has been approximated by a mixed finite-element approach. In this section, we shortly review some basic ideas underlying the numerical method, referring to the literature for a detailed exposition.

The weak formulation of problem (1) for any fixed $t > 0$ in the domain $\Omega(t)$ reads as

$$\begin{aligned} &\textit{Find } \mathbf{v} \in \mathcal{V} \textit{ and } \Phi \in L^2(\Omega) \textit{ such that} \\ &\int_\Omega \mathbf{v}\cdot\mathbf{w}\, d\mathbf{x} + \int_\Omega \Phi\, \nabla\cdot\mathbf{w}\, d\mathbf{x} \;=\; \int_{\Gamma^s} \phi\mathbf{w}\cdot\mathbf{n}\, ds, \qquad \forall \mathbf{w}\in\mathcal{V}, \\ &\int_\Omega (\nabla\cdot\mathbf{v})\psi\, d\mathbf{x} \;=\; 0, \qquad \forall \psi \in L^2(\Omega), \end{aligned} \tag{8}$$

where the test functions $\mathbf{w}$ are taken in the functional space

$$\mathcal{V} = \{\mathbf{q}\,|\,\mathbf{q}\in \left(L^2(\Omega)\right)^2,\, \nabla\cdot\mathbf{q}\in L^2(\Omega),\, \mathbf{q}\cdot\mathbf{n}|_{\Gamma^w(t)} = 0\}$$

and ψ in $L^2(\Omega)$, which is the usual space of square-integrable functions.

The mixed finite element discretization is given by re-formulating (8) on the lowest-order Raviart-Thomas space $V_h \subseteq \mathcal{V}$

$$V_h = \left\{ \mathbf{w}\in \left(L^2(\Omega)\right)^2,\; \mathbf{w}(\mathbf{x})|_T = \gamma\mathbf{x} + \boldsymbol{\delta},\, \forall T\in\Omega_h\, \alpha\in\mathbb{R}, \boldsymbol{\delta}\in\mathbb{R}^2,\, \int_{e_{ij}\cap\Gamma^s(t)} \mathbf{w}\cdot\mathbf{n} = 0 \right\},$$

and the piecewise-constant space $Q_h \subseteq L^2(\Omega)$

$$Q_h = \{\psi(\mathbf{x}) : \Omega \mapsto \mathbb{R},\, \psi(\mathbf{x})|_T = \text{const},\, \forall T\in\Omega_h\}.$$

These finite dimensional spaces are defined on a suitable mesh Ω_h, that is a collection of disjoint non-empty and non-overlapping triangles $\{T_k\}$ whose union for $k = 1\ldots N_T$ covers internally $\Omega(t)$. The parameter

$$h = \max_{T\in T_h} \text{diam}(T)$$

is the *mesh size*. The sequence of meshes for $h \mapsto 0$ forms a family of triangulations of $\Omega(t)$, that is assumed *conformal* and *regular* in the sense of Ciarlet [4, page 132], i.e. triangles do not degenerate as $h \mapsto 0$. It turns out that the space sequences of V_h and Q_h for $h \mapsto 0$ are respectively dense in $\mathcal{V}$ and $L^2(\Omega)$. We refer to [2] for a detailed presentation of the theoretical properties of this approximation framework.

Let $\{e_j\}_{j=1,\ldots,N_e}$ be the set of the edges of T, where we exclude the edges lying on $\Gamma^w(t)$. The basis functions $\mathbf{w}_j \in V_h$ are defined by the relations [2]:

$$\int_{e_i} \mathbf{w}_j\cdot\mathbf{n}_i\, ds = \begin{cases} 1, & \text{for } i = j \\ 0, & \text{for } i \neq j \end{cases}$$

where $\mathbf{n}_i$ is the unit vector orthogonal to the edge e_i. Thus, we have N_e degrees of freedom, which can be interpreted as the lowest-order momentum of the normal component of $\mathbf{v}$. The basis functions $\{\psi_k\}_{k=1,N_T}$ for Q_h are such that $\psi_k = 1$ on T_k and $\psi_k = 0$ on $\Omega(t) \setminus T_k$. The number of degrees of freedom of Q_h is equal to the number of triangles of the mesh N_T.

The mixed finite element approximation results from substituting $\mathbf{v}(\cdot, \mathbf{x})$ and $\Phi(\cdot, \mathbf{x})$ by the expressions for $\mathbf{v}_h(\mathbf{x})$ and $\Phi_h(\mathbf{x})$ as linear combination of the basis function $\{\mathbf{w}_j\}$ and of $\{\psi_k\}$:

$$\Phi_h(\mathbf{x}) = \sum_{k=1}^{N_T} \zeta_k \psi_k(\mathbf{x}), \qquad \mathbf{v}_h(\mathbf{x}) = \sum_{j=1}^{N_e} u_j \mathbf{w}_j(\mathbf{x}),$$

where $\{u_j\}_{j=1\dots N_e}$ and $\{\zeta_k\}_{k=1\dots N_T}$ are the discrete unknown vectors associated with the velocity and the potential respectively. These unknown vectors are the solution of the augmented linear system

$$\begin{bmatrix} \mathbf{M} & \mathbf{A} \\ \mathbf{A}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \{u_j\} \\ \{\zeta_k\} \end{bmatrix} = \begin{bmatrix} \{q_j\} \\ \mathbf{0} \end{bmatrix} \tag{9}$$

where $\mathbf{M}$, $\mathbf{A}$, and $\mathbf{q}$ are defined as follows

$$\begin{aligned} M_{ij} &= \int_{\Omega(t)} \mathbf{w}_i \cdot \mathbf{w}_j \, d\mathbf{x}, \\ A_{ij} &= \int_{\Omega(t)} (\nabla \cdot \mathbf{w}_i) \psi_j \, d\mathbf{x}, \\ q_j &= \int_{\Gamma^s(t)} \phi \mathbf{w}_j \cdot \mathbf{n} \, ds. \end{aligned}$$

The linear algebraic problem (9) is solved by applying the routine *MA48* of the *HSL-2000* library [12].

5. LEAST SQUARE RECONSTRUCTION OF BOUNDARY VELOCITIES AND POTENTIALS

Let us introduce the sets σ_i and $\mathcal{V}_i$ that are respectively the set of the triangular cells incident the i-th boundary node and of the mesh vertices directly connected to $\mathbf{x}_i$.

On the mesh patch $\cup_{j\in\sigma_i} T_j$ we locally reconstruct the potential solution assuming that there hold a linear dependence on $\mathbf{x}$

$$\phi(t, \mathbf{x}) = \phi_i(t) + \mathbf{v}_i(t) \cdot (\mathbf{x} - \mathbf{x}_i), \qquad \mathbf{x} \in \cup_{j\in\sigma_i} T_j,$$

where $\phi_i(t)$ and $\mathbf{v}_i(t)$ are the minimizers of the quadratic functional

$$\begin{aligned} \mathcal{E}(t) = \sum_{j\in\sigma_i} |T_j| \left[\phi_i(t) + \mathbf{v}_i(t) \cdot (\mathbf{x}_j - \mathbf{x}_i) - \phi_j(t)\right]^2 + \\ + \sum_{k\in\mathcal{V}_i} |e_{ik}| \left[(\mathbf{v}_i(t) - \mathbf{v}_{ik}(t)) \cdot \mathbf{n}_{ik}\right]^2. \end{aligned}$$

6. IMPACT OF A LIQUID JET ON A RIGID WALL

A clear water bore hitting upon a rigid, vertical, plane wall is considered herein. The surge is generated by a dam break and propagates over a dry horizontal bed. A sketch is given in Figure 2.

This impact phenomenon has been exthensively investigated and experimental results have been collected in References [25, 31]. The interested reader can also found a classification for bore impact in Reference [30].

The aim of this test case is to verify whether the new numerical approach based on a mixed finite element potential flow solver can describe the complex dynamics of the bore impact; and, if so, to what extent can a potential model be accurate in obtaining wall forces. In fact the knowledge of the dynamic loading is crucial in the rational design of transversal structures.

Such a liquid-solid impact problem can typically be formulated by assuming that at the initial time $t = 0$ the liquid mass meets the wall; at this moment, the wall location, the liquid domain and the flow field are assumed to be known. For $t > 0$ the flow field together with liquid actions on the wall have to be determined.

The model presented in section 2 holds true under some simplifying hypotheses to be discussed below.

The first one concerns the compressibility of pure water that does not play a significant role in this kind of impact [22]: owing to air entrainment, the velocity of sound in air–water mixtures can be even one order of magnitude less than that in pure water; even so the compressibility of the mixture is expected to be unimportant in actual prototype collisions.

Also, at least in the initial stage of the impact, inertia forces are by far dominant as compared to surface tension, viscosity and gravity forces at almost any stage of the impact process. The dynamic interaction of a structure with a liquid jet should be solved in principle as a unified hydro–elastic system; however, the elastic response of the structure would pose additional complexities in the computations and therefore the wall has been regarded as a rigid body in the present simulation. After all, calculated pressures would be on the safety side; in fact, a pressure overestimation of only $3\% \div 6\%$, as compared to the more realistic case of elastic wall, was claimed by [32].

Hence, assuming further that the flow is irrotational and time–dependent in a simply connected domain, bounded by impervious walls and a free streamline, it is clear that the essential features involved in the collision process are described by the simplified approach based on the potential flow theory described in section 2.

The code has been at first validated by means of the steady state solution [16]; fig. 3 shows how the unsteady free surface profile is clearly evolving towards the steady jet.

The isobars are illustrated in fig. 4, where the symmetry about the bisectrix of the axes is reached by the numerical liquid jet. This symmetry feature is embodied by the steady state analytical solution. The computed wall pressure evolves towards a *pseudo* steady state.

The complete simulation of the dam-break surge, starting from the removal of the gate and proceeding towards the impact wall, would clearly provide the suitable flow field to start the calculation of the impact; but this is beyond the scope of the present work, which is simply to demonstrate the feasibility of a potential flow model to reproduce the bore impact and the new numerical method to approximate its solutions.

The laboratory experimental impact chosen to test the numerical model is the clear water bore, detailed by [25] and by [31]. The initial condition for the whole flow field is not known from the physical experiments; instead, only the toe velocity $U = 2.77$ m/s, was measured, apparently a rough estimate of the bulk velocity. The measured velocity is the picture of an instant of time and can hardly represent a highly unsteady phenomenon. It is to be recalled that the measurements of *instantaneous* toe velocities were performed by tracing the location of the most advanced part

of the ever–breaking front rushing downstream; therefore toe velocities do embody huge turbulent streamwise fluctuations.

The 1–D Ritter's solution [24] was then adopted as the initial condition for the flow field; this 1–D analytical solution provides a parabolic shape for the free surface and a linear law for the longitudinal velocity. The front region has been simulated in this work by a trapezoidal domain moving at constant velocity $U = 2.77$ m/s (Fig. 5); then the Ritter's solution has been implemented in the upstream region.

This initial computational domain is a good approximation of the mildly elongated physical toe, as it was observed in the video pictures of the experiments.

By trial and error a convenient temporal step was found to be 10^{-4} seconds. The typical number of generated triangles was 1000. The typical CPU time for a complete simulation, involving $3.5 \cdot 10^3$ time steps, was half an hour in an ordinary $500\,Mhz$ processor.

The model predicts the moving free surface while verifying closely the mass conservation, the maximum error being well less than 1%. It has to be pointed out that the accurate modeling of the free surface is not important as far as the calculation of the wall force is concerned; in fact it was observed in the present numerical simulation that different schematizations of the impacting liquid shape affect the wall force evolution by a small amount. This result is not new in literature: the authors of Reference [5] found a relative insensitivity to the shape of the incident free boundary, which is a result of relevant value in practical circumstances; this implies that even simple schematizations of the free surface can be effective, since wall pressure is not much affected both by the jet shape and by the liquid body further away from the rigid surface.

The free surface comparison (Fig. 6) shows a clear discrepancy between the numerical and the physical profile due to the huge air entrainment, phenomenon which has not been accounted for in the numerical simulation. Nevertheless it will be shown that the wall force is predicted very well.

The knowledge of the wall force is normally the most relevant quantity for design. The predicted force has been compared against the experimental force, (Fig. 7), which was obtained by integrating the pressure diagrams, [31]. The tiny oscillations of the numerical force diagram are not physical: they are due to the nodes of the moving mesh (which is rebuilt at any time step) passing through the numerical gauge's location.

7. IMPACT OF A LIQUID JET ON A LIQUID PLANE SURFACE

In this section we apply the new numerical method to the investigation of the air entrapment process due to the impact of a liquid mass on a liquid surface. Typical examples are rain drops or splashes hitting the sea surface. Basically, there are a number of different processes by which air is entrained when a liquid mass hits a liquid surface, depending on fluid viscosity, level of surface disturbances, flow rate, geometry, and many others. We refer the interested reader to the paper of Reference [19] for a review of these issues.

In this section, we investigate the performance of the new method in predicting the entrapment of an air bubble as documented in the papers of References [19, 20, 23].

The early stages of the impact can show a very complex behavior due to liquid fragmentation and entrainment of very small air bubbles. Furthermore, the liquid-liquid contact occurs at a number of places in addition to the point of the initial contact.

To avoid the simulation of the early stages of the impact process that, as we mentioned, can be extremely complicated, we start the calculation from a configuration in which the initial column-shaped jet has a non-vanishing area of contact with the receiving liquid.

The computational domain is taken to be the (x, z) plane, with the z-axis directed upward against gravity and the initial liquid surface at rest at the line $z = \text{const}$.

At $t = 0$ a column of liquid of radius a impacts the undisturbed surface with velocity $\mathbf{v}$. The aspect ratio r, which is the ratio between the column height h and a, is kept as parameter. Since the details of how the impacting jet column is terminated do not affect significantly the successive development of the flow [23], the top of the column is rounded with a hemisphere to avoid sharp corners on the free surface.

The initial condition is that the liquid in the jet column has a unit dimensionless velocity, while the receiving liquid is at rest, i.e.

$$\phi = z, \qquad 0 \leq z \leq 2,$$

$$\phi = 0, \qquad z \leq 0.$$

Both the impinging jet and the receiving liquid are composed by the same fluid (water), whose kinematic viscosity is indicated by ν.

Let us shortly discuss to what extent the impacting process is simplified by applying the mathematical model of section 2. As the impact velocities are of the order of a few metres per second and the initial vertical jet is column shaped with a radii of a few millimiters, the typical Reynolds numbers $\mathsf{Re} = a\,|\mathbf{v}|\,/\nu$ are of the order of 10^3 or higher. Consequently, viscous effects can be neglected in these highly transient processes. Compressibility effects can also be neglected and the liquid is considered as incompressible. This is because they are significant on timescales of the order of the acoustic travel time in drop, which can be measured in microseconds, while we are concerned with the evolution of the physical system over much longer time scales. Surface tension effects, related to variations of the Weber number $\mathsf{We} = \rho\,|\mathbf{v}|^2\,a/\sigma$, where σ is the surface tension coefficient and ρ the fluid density, are also neglected by the model because they are likely to be unimportant except for small drops. Weber number effects on some flows of the general type considered here have been investigated in References [19, 20].

As already pointed out by Reference [23], eventually only two parameters turn out to play a role in these flow simulations: the Froude number, defined as $\mathsf{Fr} = |\mathbf{v}|^2\,/(ag)$, and the aspect ratio $\mathsf{R} = h/a$ of the impacting column. We present the results for impacts with $\mathsf{Fr} = 2$, 8, 32 and aspect ratios of 10 and 20. These calculations can be compared with the ones presented in Reference [23] and using the same parameters. In all the pictures discussed below, times and lenghts are in dimensionless units.

The air entrapment process basically occurs as follows. When the liquid mass column penetrates into the liquid initially at rest, a cavity forms with a downward-moving lower boundary. At the later moments of the penetration process, the downward motion of the cavity bottom starts decreasing and can eventually reverse. This can produce an upward moving jet that can try to escape from the cavity. This phenomenon is in competition with the collapse of the cavity walls that can enclose an air bubble.

The evolution of the successive free surface configurations are summarized in Figure 8, where successive profiles are shown at the same instants for simulations with the same aspect ratio number. Figure 9 depicts the velocity fields at the final instants of each simulation, except for $\mathsf{Fr} = 32$ and $\mathsf{R} = 10$, where for comparison's purposes the picture illustrates the velocity field at $T = 40$ for the run with $\mathsf{Fr} = 32$ and $\mathsf{R} = 20$.

The sequence of pictures in Figure 10 depicts how the free surface and the computational mesh generated by the method evolves from the starting time $T = 0$ to the final time $T = 8$ for the simulation with the lowest Froude number $\mathsf{Fr} = 2$ and aspect ratio $\mathsf{R} = 10$. When the aspect ratio is increased to $\mathsf{R} = 20$, the cavity walls collapse against the lateral sides of the jet column well before the latter has fully penetrated within the initially resting fluid. The sequence of pictures in

TABLE 1
Details on the resolution of the liquid-liquid impact test cases.

Fr	R	# Equations			# Iterations			Time
		Min	Max	Avg	Min	Max	Avg	(sec)
2	10	5672	6016	5875	8	11	9.3	136.2
2	20	6986	7166	7075	9	12	10.2	148.5
8	10	5752	6046	5901	6	13	7.3	232.7
8	20	6936	7256	7114	6	10	8.1	286.9
32	10	5593	6163	5862	5	20	6.1	1184.5
32	20	6926	7260	7095	5	10	6.3	554.2

TABLE 2
Details on the mesh generation of the liquid-liquid impact test cases.

Fr	R	# Triangles			# Vertices			# Edges			Time
		Min	Max	Avg	Min	Max	Avg	Min	Max	Avg	(sec)
2	10	2166	2304	2248	3506	3712	3627	1341	1409	1380	1.91
2	20	2660	2732	2696	4326	4434	4380	1667	1703	1685	2.15
8	10	2198	2316	2258	3554	3730	3643	1357	1415	1386	4.17
8	20	2640	2768	2711	4296	4488	4403	1657	1721	1693	5.00
32	10	2134	2362	2242	3459	3801	3620	1326	1440	1379	24.34
32	20	2636	2768	2703	4290	4492	4392	1655	1725	1690	12.31

Figure 11 illustrates the process. The simulation has been arrested at the instant $T = 7$ because at later times the cavity walls collide against the jet column. Two small bubbles are entrained within the fluid at the sides of the jet. For both the aspect ratios, the bottom cavity is again moving downward when the cavity closure occurs, as reveals the inspection of the velocity fields (top row) in Figure 9.

When the simulation is considered at the intermediate Froude number $\mathsf{Fr} = 8$, the velocity of the impacting jet is enough to complete the absorption before the cavity walls collapse and enclose an air bubble. The situation depicted in Figure 12 for the low aspect ratio $\mathsf{R} = 10$ is similar to the one depicted in Figure 13 for the larger value $\mathsf{R} = 20$, even if in this latter case the cavity formation takes a longer time. In both cases, this is coeherent with the velocity fields depicted in Figure 9, intermediate row.

Finally, in the highest Froude number simulations $\mathsf{Fr} = 32$, we can notice by the sequence of pictures in Figures 14 and 14 that at the low aspect ratio $\mathsf{R} = 10$ the bottom of the cavity reverses its motion and a "fluid tongue" is ejected before the cavity walls enclose a bubble of gas.

The reversal in motion of the cavity bottom is evident from the velocity fields depicted in Figure 9, bottom row. Again, the aspect ratio value is critical, and in facts this phenomenon disappear when the simulation is run with the larger value $r = 20$, as shown in Figure 16.

7.1. Performance of the method

In this section we discuss the computational cost and the efficiency of the method in the simulation of liquid-liquid impacts.

Table 1 displays the details about the coupled potential-Laplacian solver. For each test case specified by the Froude number and the aspect ratio of the initial free surface configuration the table reports the minimum (subcolumn Min), maximum (subcolumn Max) and average (subcolumn Avg) number of equations (Equations) forming the linear system (9) as well as of the iterations required to solve the coupled solver. The last column on the right (Time) displays the total CPU time (in seconds) required to build and solve (9) in each simulation. The three sub-columns of Equations in Table1 illustrates the fact that the number of equations does not change dramatically.

Similarly, the three sub-columns of Iterations in Table1 illustrates the fact that the number of iterations is always in the range [5, 20] and the average number per time step is rarely over 10.

Table 2 displays the details about the mesh generated by the method at each time step and used to solve the Laplace's problem (1-2). For each test case specified by the Froude number and the aspect ratio of the initial free surface configuration the table reports the minimum (subcolumn Min), maximum (subcolumn Max) and average (subcolumn Avg) number of triangular cells (Triangle), mesh vertices (Vertices) and edges (Edges). The last column on the right (Time) displays the total CPU time (in seconds) required to generate the mesh.

It is clear from Table 2 that the size of the mesh generated at each time step is stable for what concerns the number of triangular cells, vertices and edges.

In particular, comparing the last columns of Table 1 and Table 2 concerning CPU times we can draw the result that the mesh generation step is globally two orders of magnitude less than the resolution of the algebraic system (9).

We can thus deduce that although the present method asks for a full remeshing at each time step this fact does not have a significant impact on the total cost of our simulations.

8. CONCLUSIONS

A new numerical procedure has been developed and presented for the analysis of the motion of free-surface flow configurations in liquid-solid and liquid-liquid impacts. The method approximate the solution of a mathematical model that is formulated on the basis of potential theory. A mixed finite element scheme using the lowest-order Raviart-Thomas space is implemented for solving the Laplace's equation in the potential model. This scheme is non-linearly coupled with an implicit time-stepping technique for the temporal evolution of the position of nodes at the free surface in accord with Bernoulli's law. A full remeshing of the computational domain is required at each time-step to track the motion of free-surface flows. Despite the appearance, the remeshing step does not cost excessively, in particular when compared to the cost of the numerical resolution of the algebraic system of the Laplace's solver. Thus, the method is actually effective in the problem resolution and computationally efficient.

Numerical results for the modelization of liquid-solid and liquid-liquid impacts are presented. By conjecturing realistic initial conditions in bore jet impacts on rigid walls, the present numerical approach proves successful in obtaining a quantitative evaluation of important physical quantities, such as the maximum force acting on the wall, so that meaningful predictions can be obtained not only from laboratory tests but also from numerical simulations.

The proposed method has also been proved promising in the simulation of collision processes of liquid column-shaped jets onto a liquid surface initially at rest. In particular, the presentation is herein focused about to what extent the impact process of interest is simplified by applying this model which is based on the potential theory.

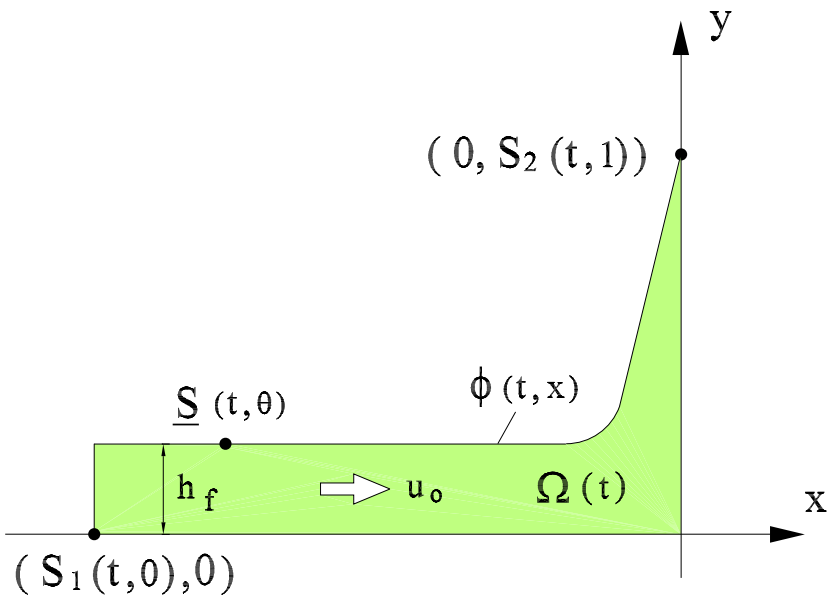

FIG. 2. Jet impact on a rigid wall: sketch of the problem and notation.

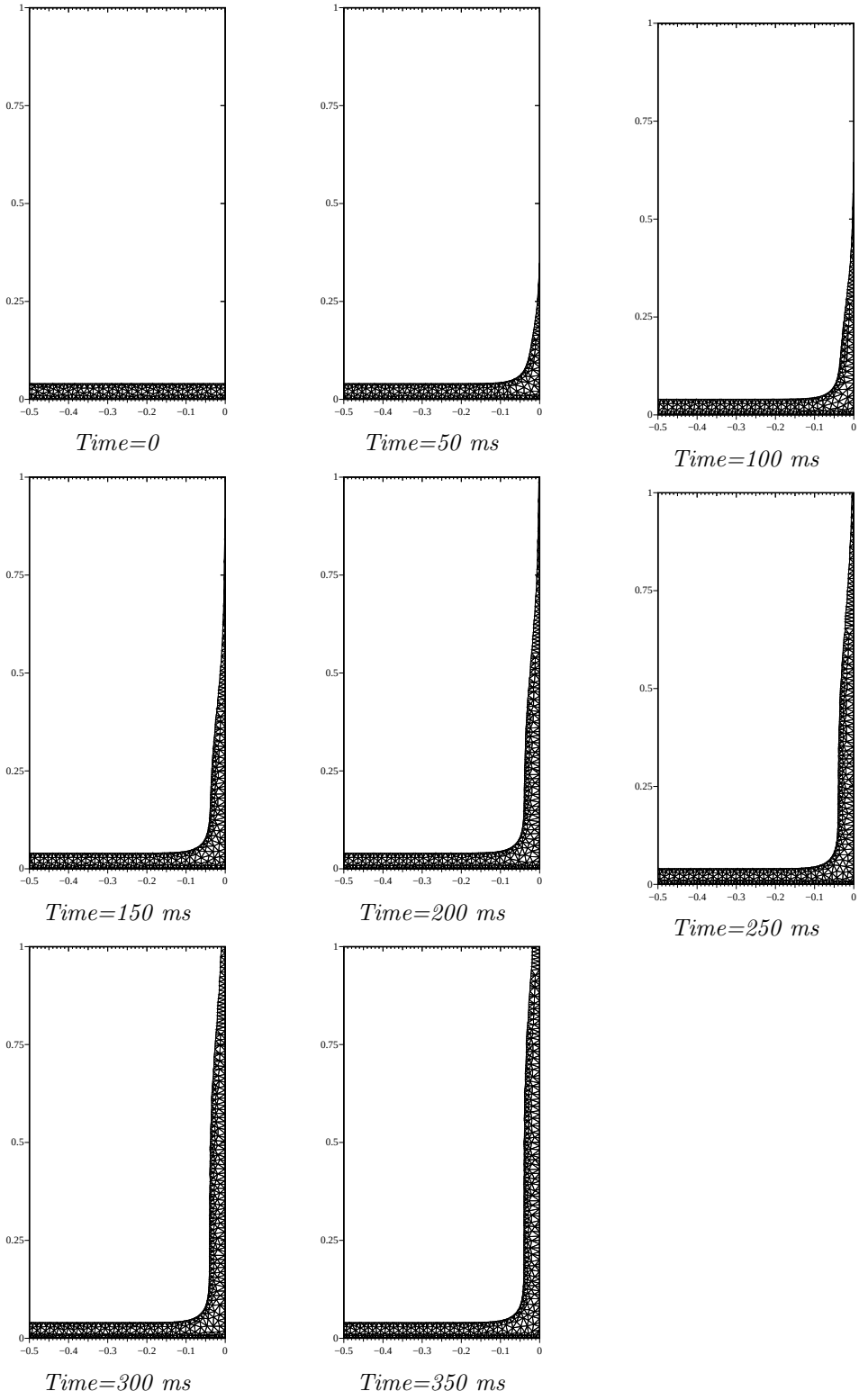

FIG. 3. Test Case 1: computational mesh at different instants.

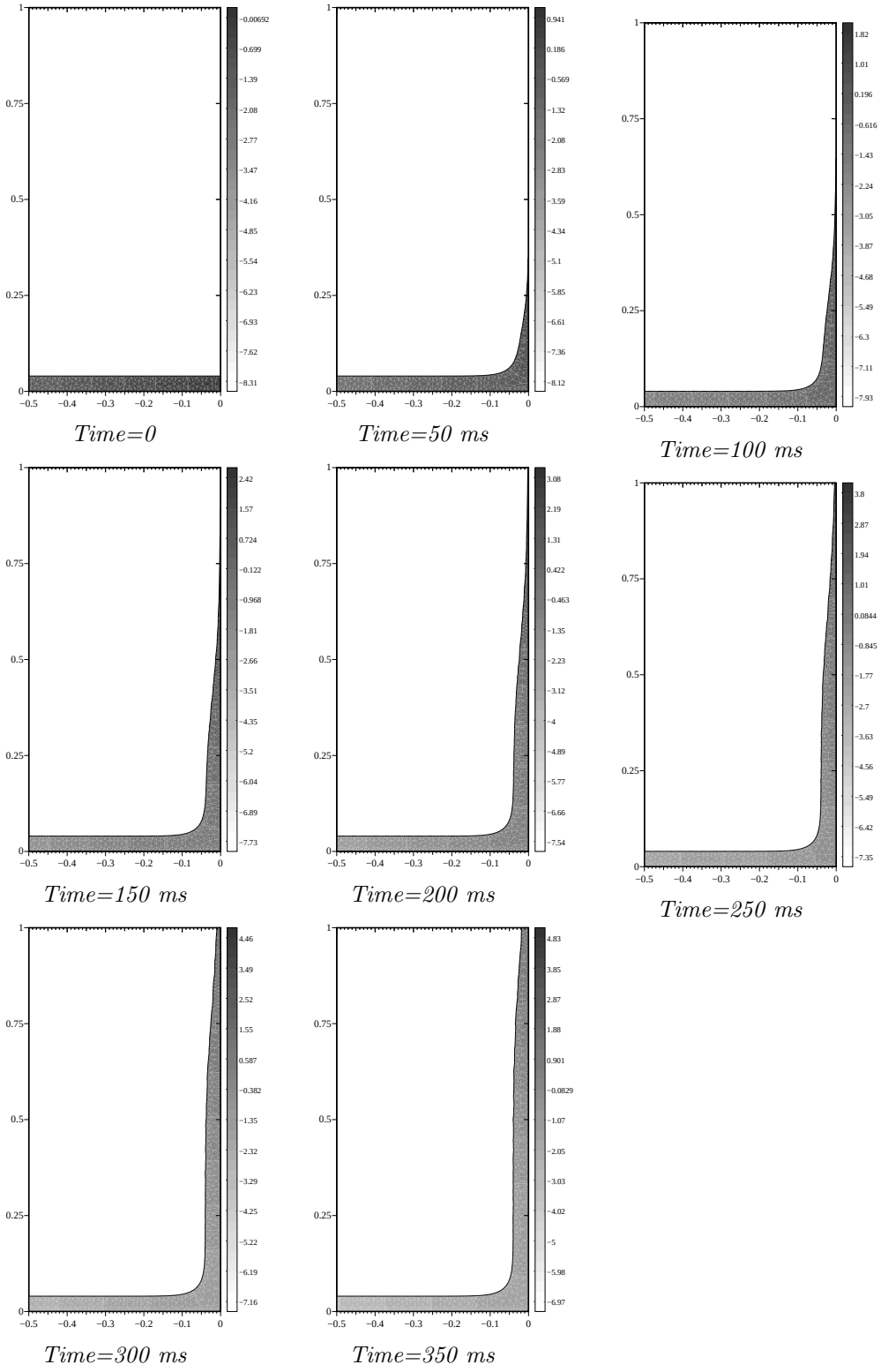

FIG. 4. Test Case 1: velocity potential field at different instants.

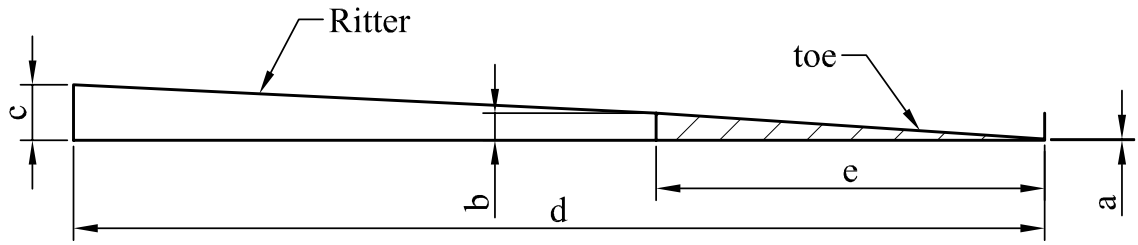

FIG. 5. Initial domain (a=1 mm; b= 2.8 mm; c=6.3 mm; d=1m; e= 0.4cm)

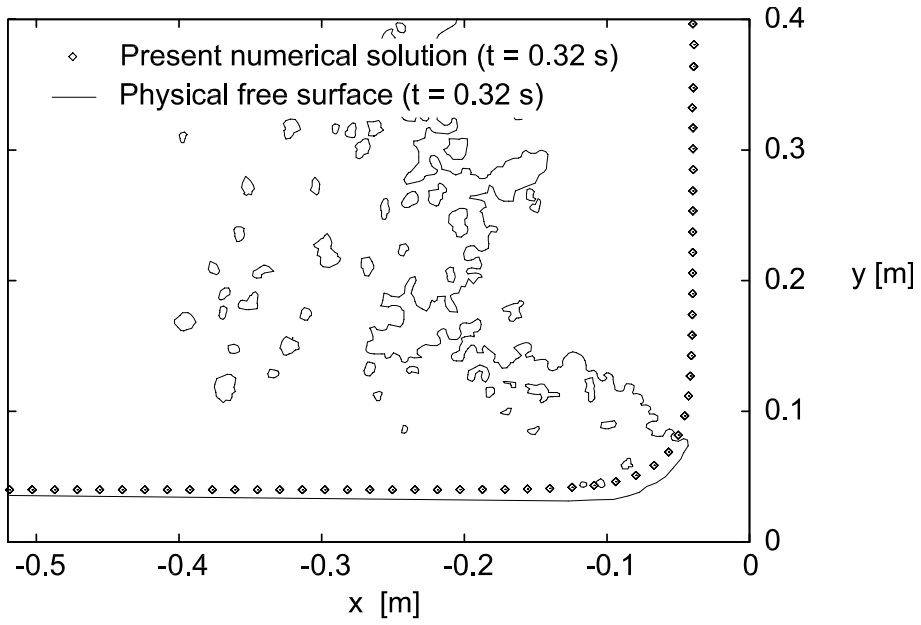

FIG. 6. Measured and computed free surface (flow is from left to right)

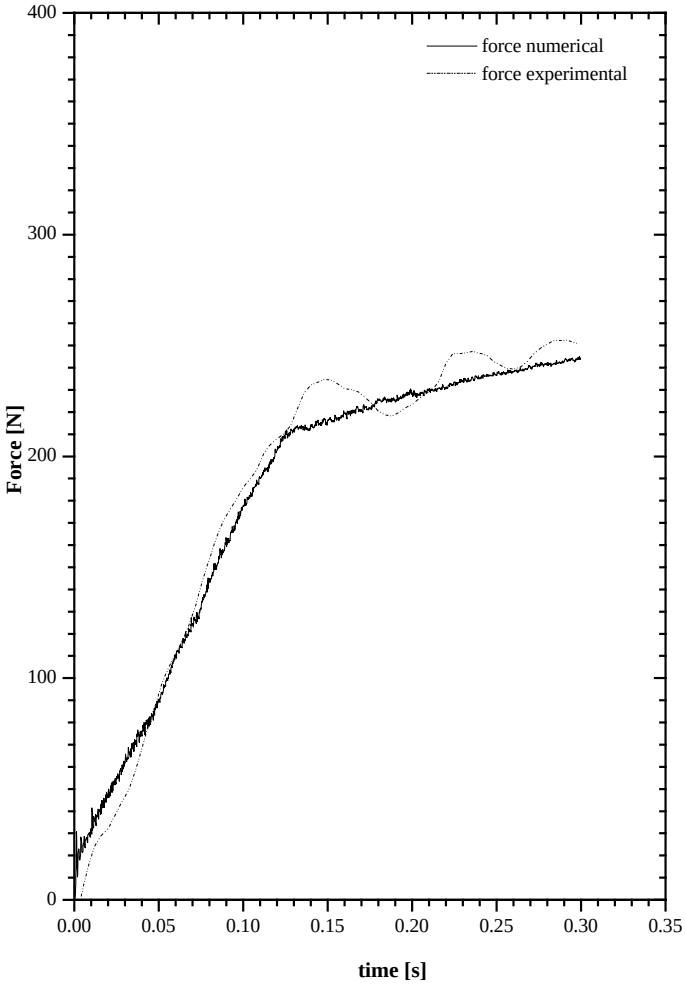

FIG. 7. Temporal evolution of the wall force

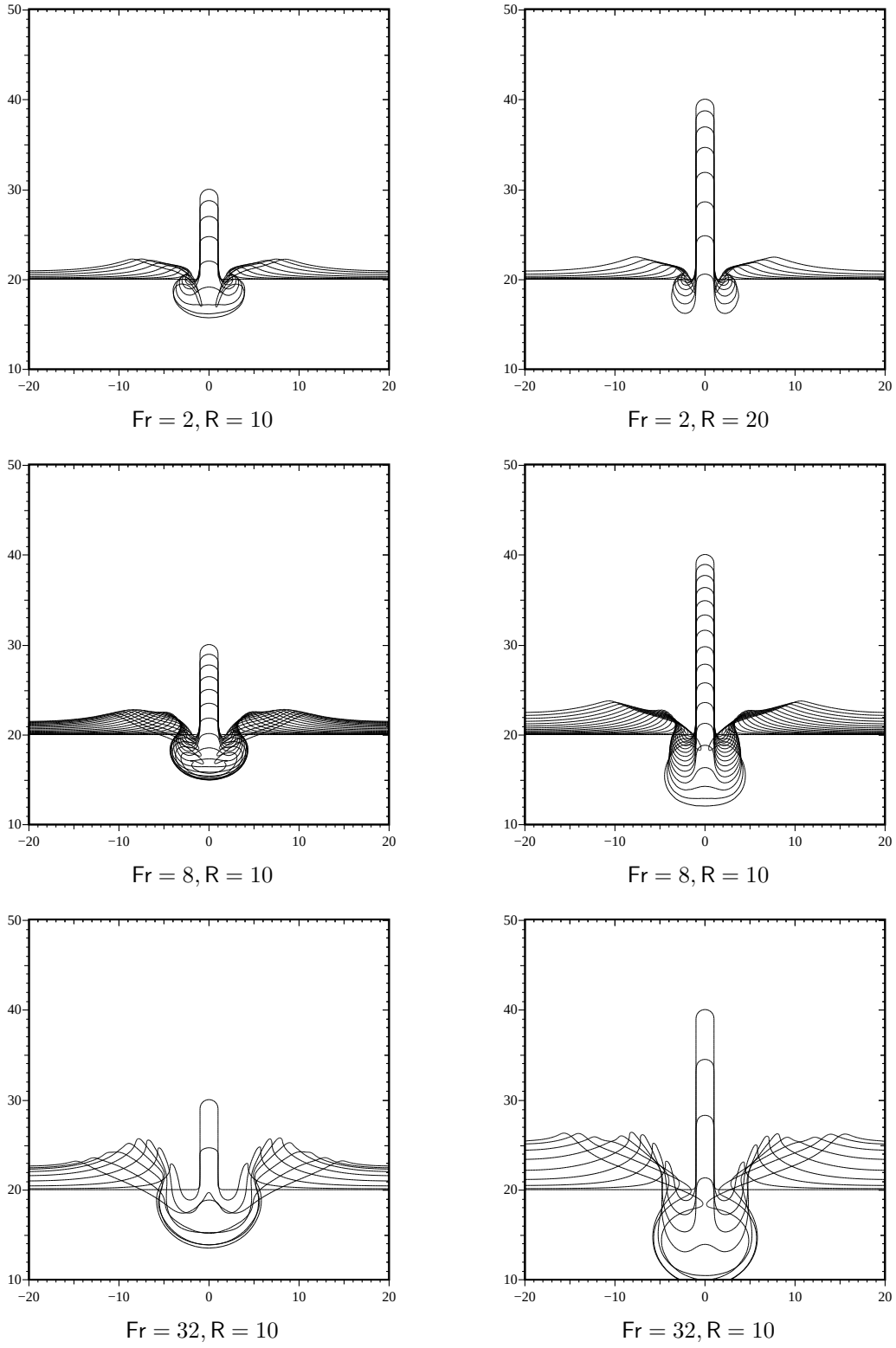

FIG. 8. Evolution of the free surface profiles at different Froude numbers and aspect ratios.

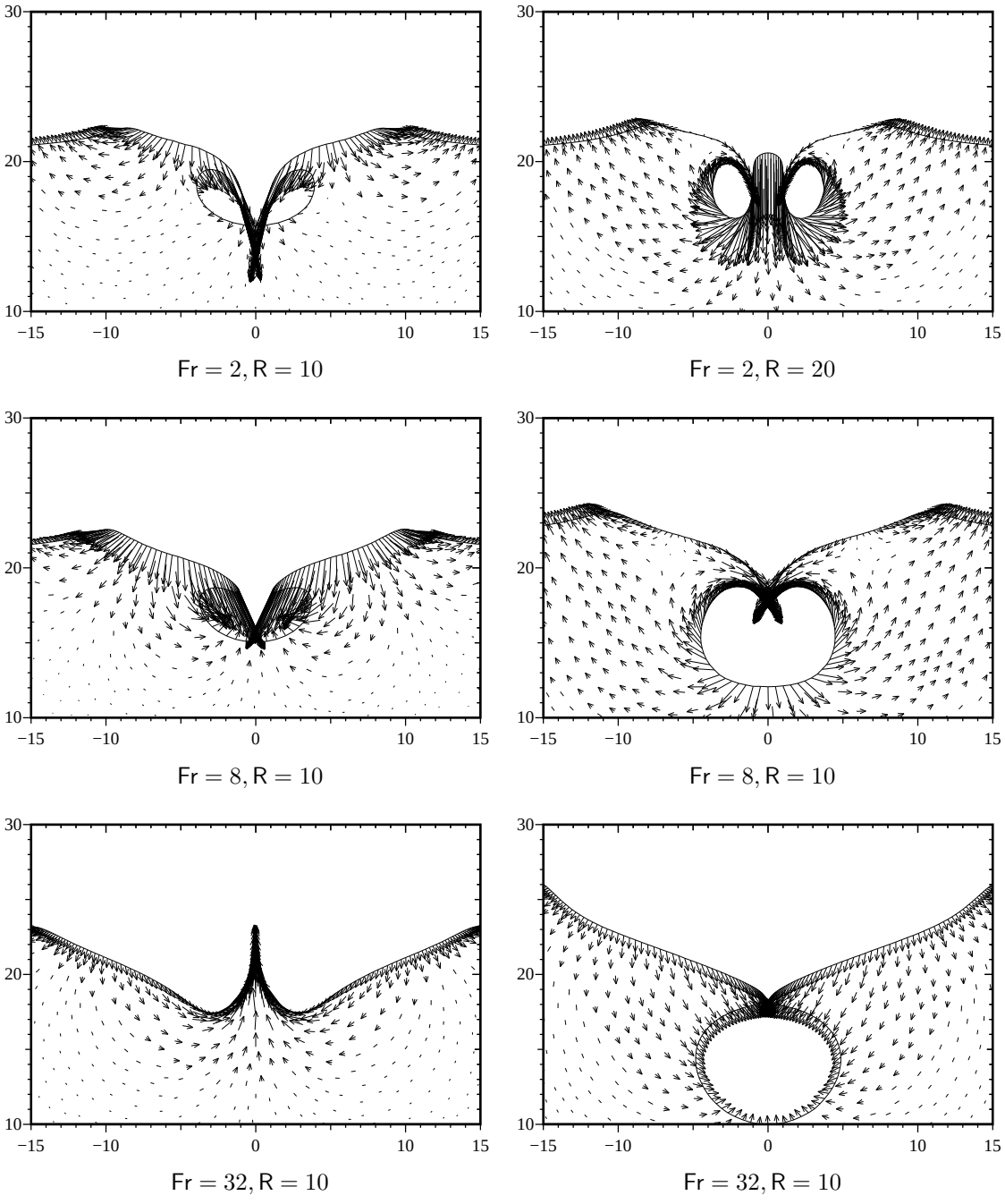

FIG. 9. Velocity fields at the final times at different Froude numbers and aspect ratios.

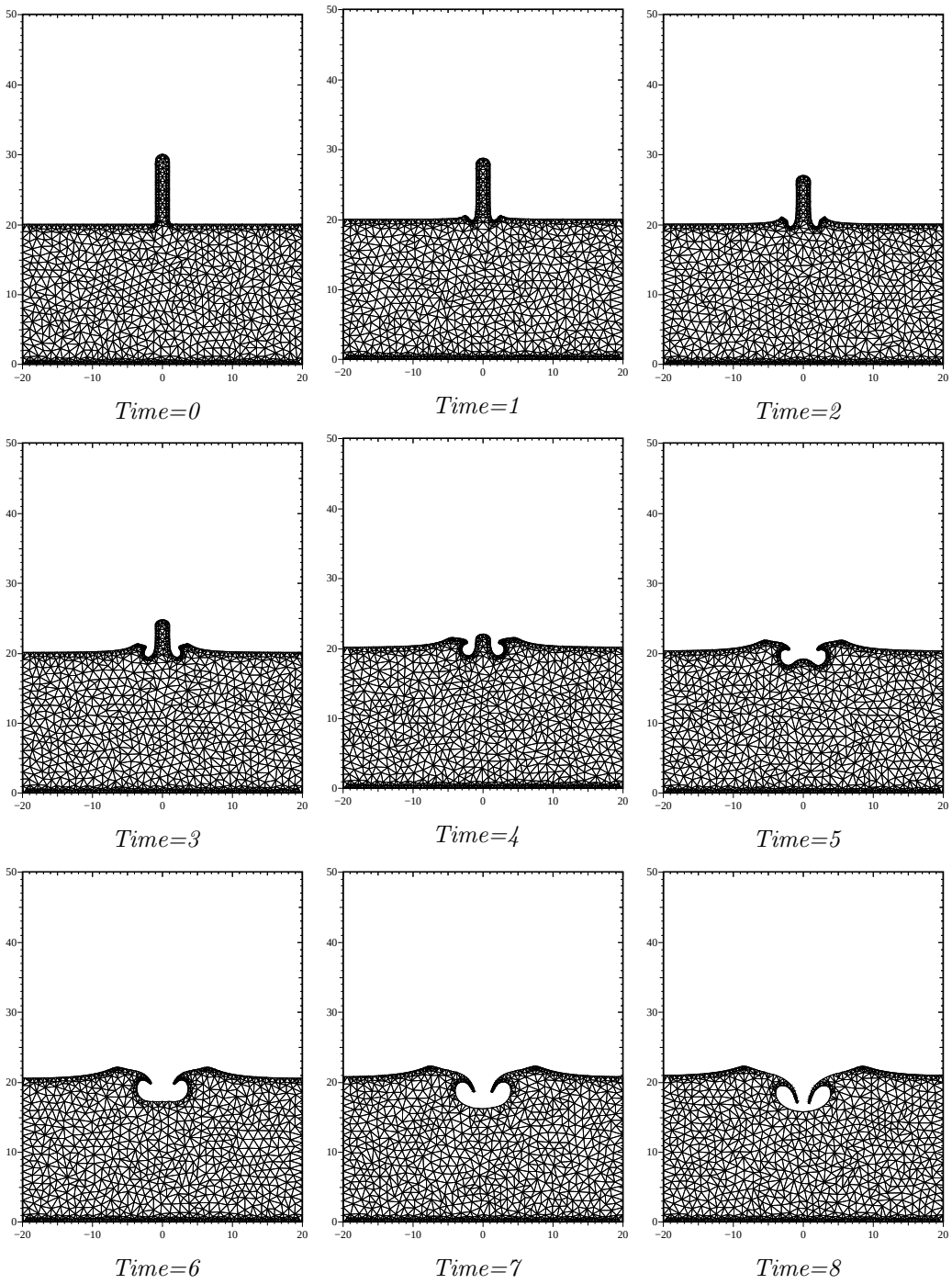

FIG. 10. Froude number $\mathsf{Fr} = 2$, aspect ratio $\mathsf{R} = 10$, computational mesh at different instants.

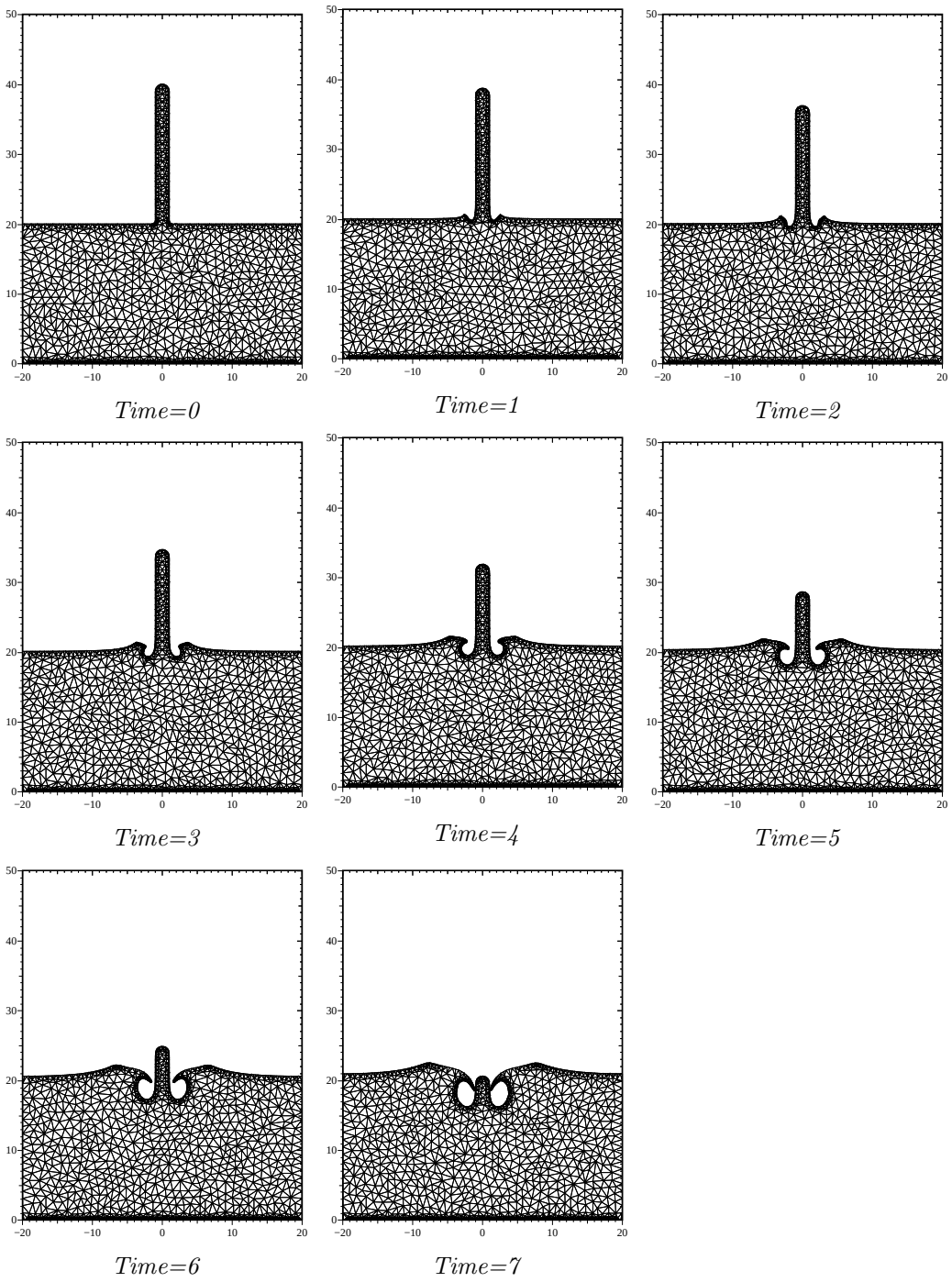

FIG. 11. Froude number $\mathsf{Fr} = 2$, aspect ratio $\mathsf{R} = 20$, computational mesh at different instants.

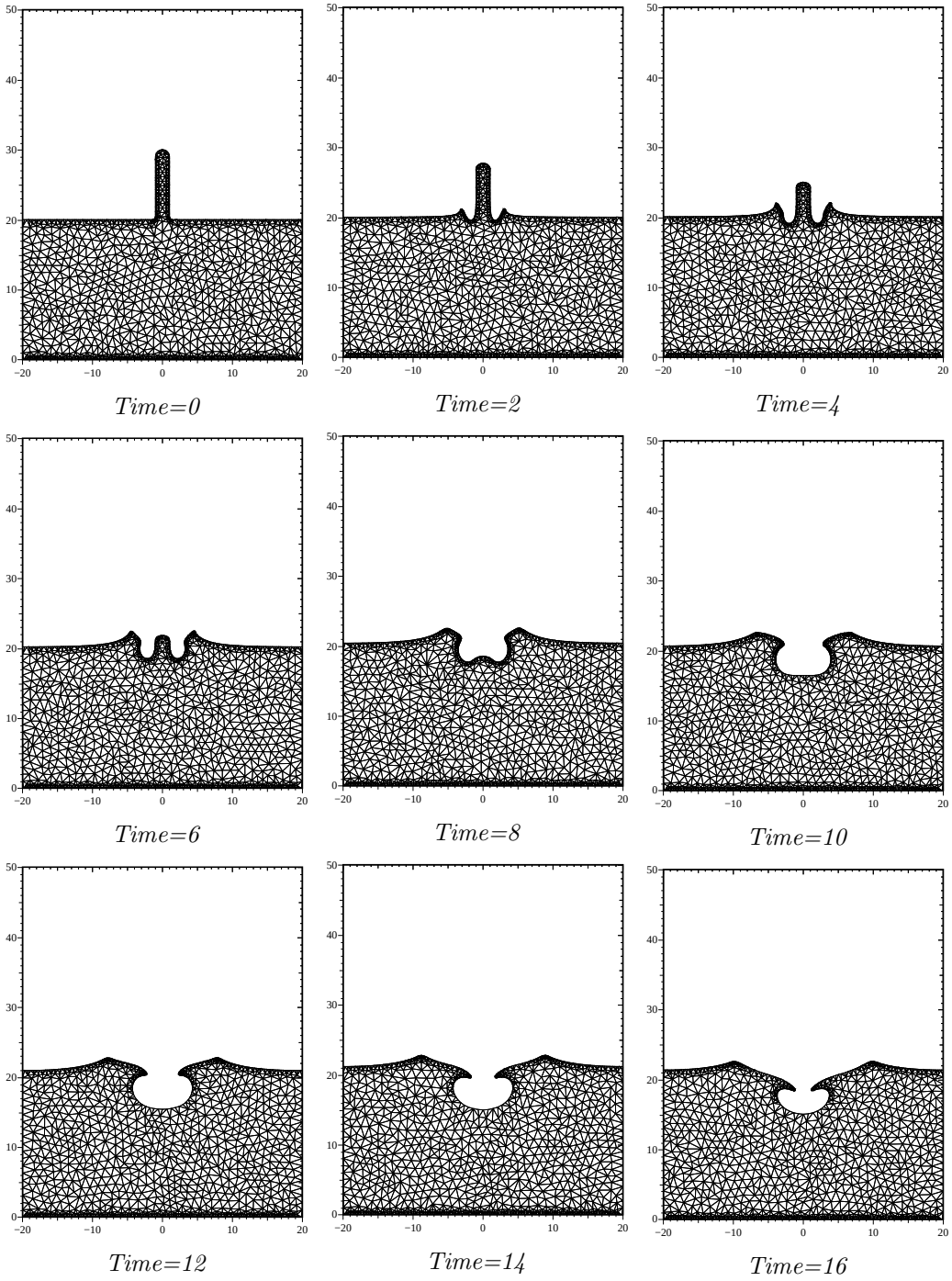

FIG. 12. Froude number $\mathsf{Fr} = 8$, aspect ratio $\mathsf{R} = 10$, computational mesh at different instants.

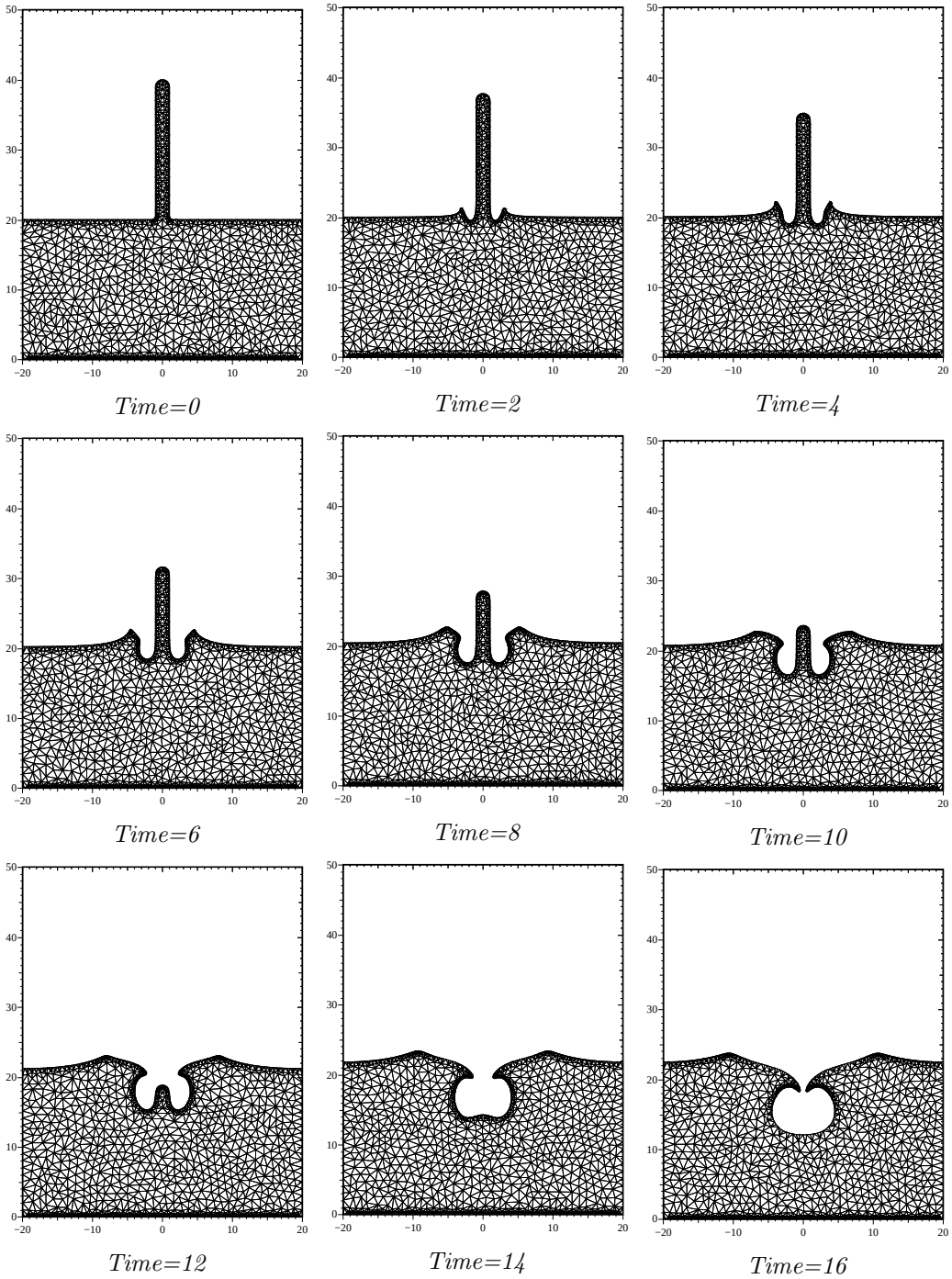

FIG. 13. Froude number $\mathsf{Fr} = 8$, aspect ratio $\mathsf{R} = 20$, computational mesh at different instants.

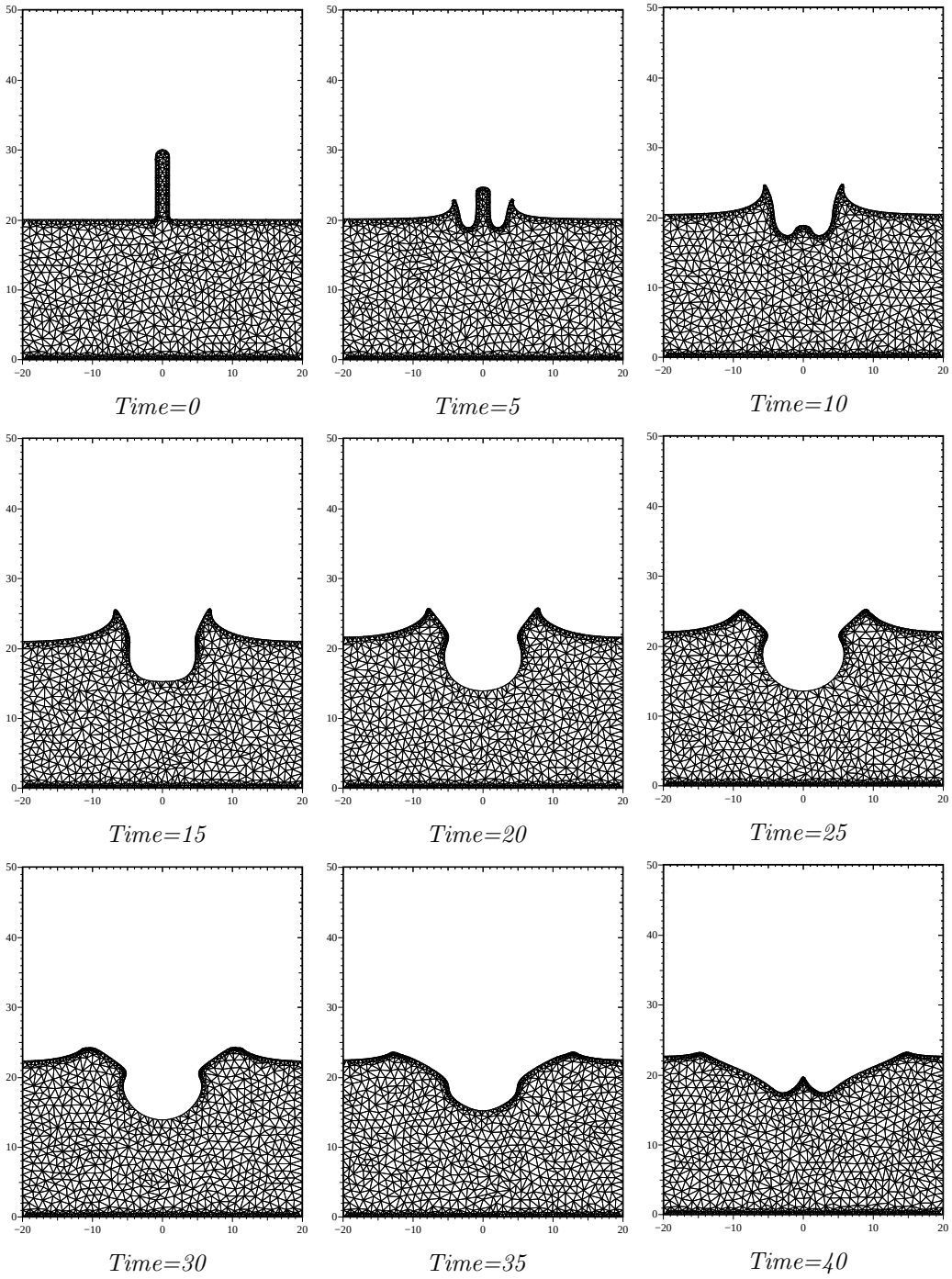

FIG. 14. Froude number $\mathsf{Fr} = 32$, aspect ratio $\mathsf{R} = 10$, computational mesh at different instants.

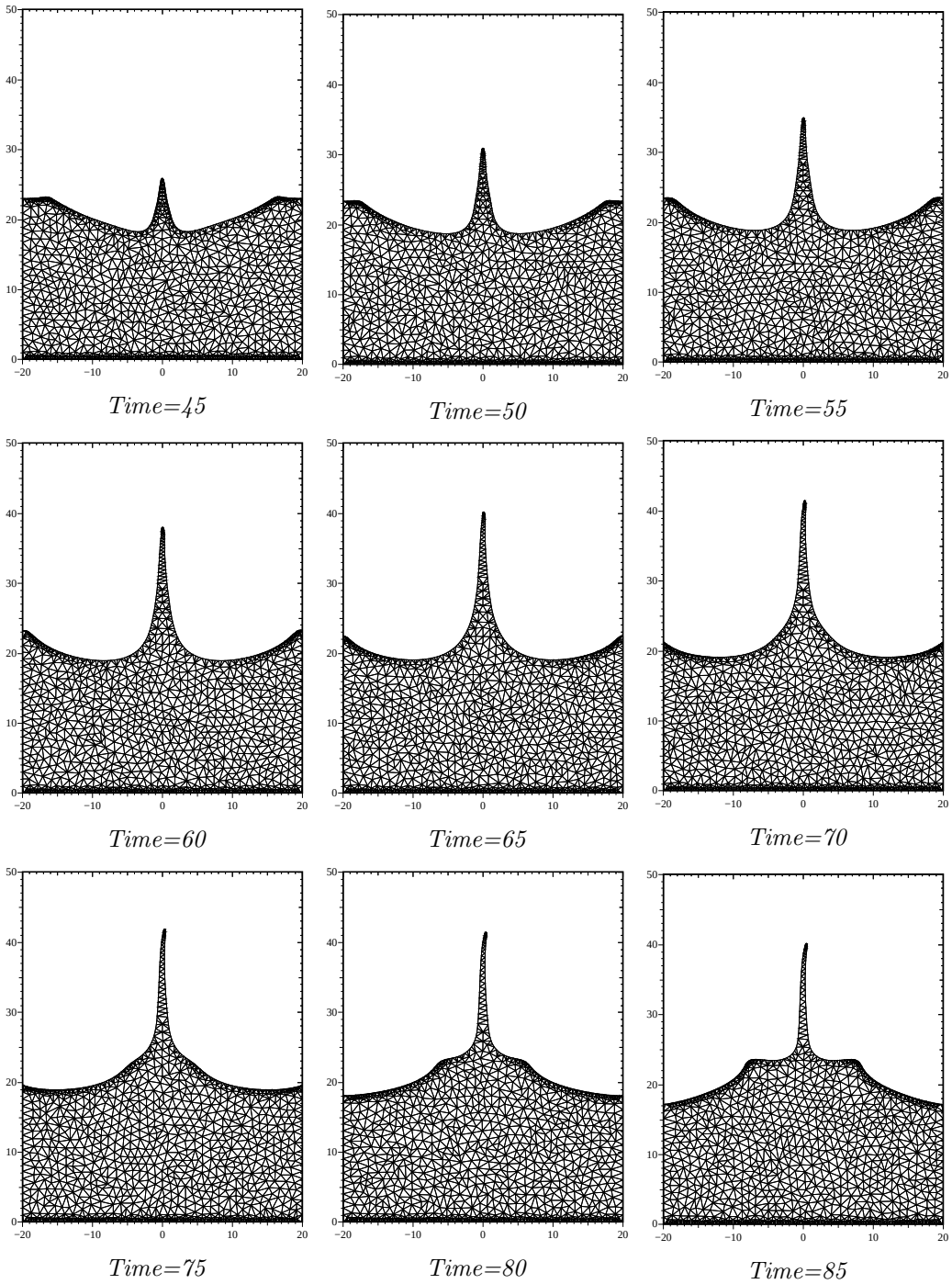

FIG. 15. Froude number $\mathsf{Fr} = 32$, aspect ratio $\mathsf{R} = 10$, computational mesh at different instants.

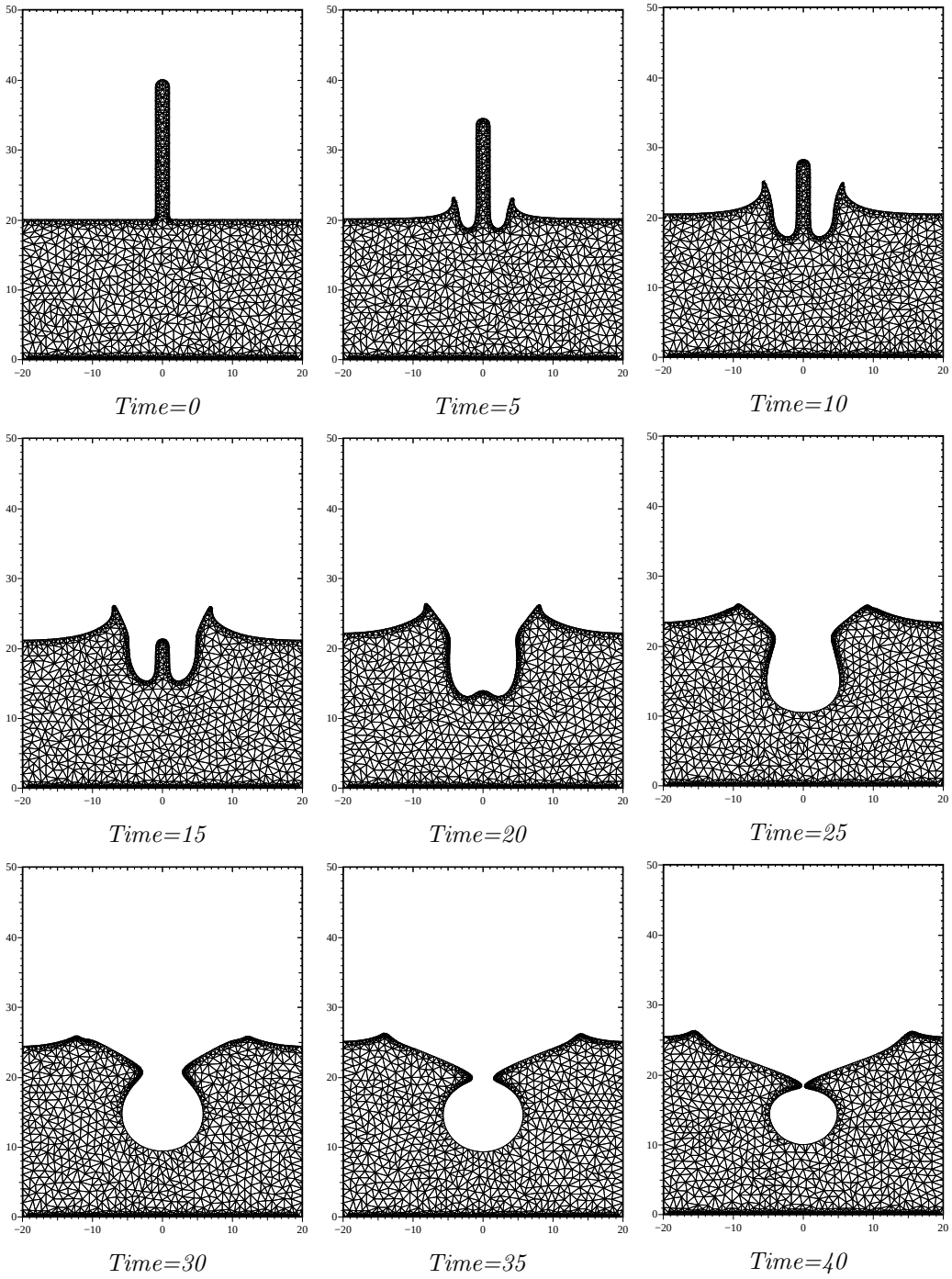

FIG. 16. Froude number $\mathsf{Fr} = 32$, aspect ratio $\mathsf{R} = 10$, computational mesh at different instants.

APPENDIX

A.1. DERIVATION OF EQUATION (3)

In order to describe energy conservation, the velocity potential $\Phi = \Phi(t, \mathbf{x})$ satisfies the Bernoulli's equation

$$\frac{\partial \Phi}{\partial t}(t, \mathbf{x}) + \frac{1}{2} |\nabla \Phi(t, \mathbf{x})|^2 + \frac{p(t, \mathbf{x})}{\rho} + g\mathbf{x} \cdot \hat{\mathbf{z}} = 0, \qquad \text{for } \mathbf{x} \in \Omega(t), \tag{A.1}$$

where $p(t, \mathbf{x})$ is the fluid pressure field and ρ is the fluid density. Notice that $p(t, \mathbf{x}) = \text{const}$ when $\mathbf{x} \in \Gamma^s(t)$ because the streamline $\mathbf{x}^s(t, \theta)$ is an isobaric curve. Equation (3) follows immediately by substituting the relation $\Phi(t, \mathbf{x}^s(t, \theta)) = \phi(t, \theta)$ into (A.1), applying the chain rule of derivation, and taking $p = 0$ at $\Gamma^s(t)$.

A.2. DERIVATION OF EQUATION (4)

Let $\xi(t, \omega)$ be the parametric equation of a fluid streamline, with $\omega \in [0, 1]$ the scalar parameter running over the curve. Clearly,

$$\frac{\partial \xi}{\partial t}(t, \omega) = \nabla \Phi(t, \xi(t, \omega)).$$

Let us introduce a re-parameterization of the streamline curve that takes the form

$$\mathbf{x}^s(t, \theta) = \xi(t, \omega(t, \theta)).$$

Applying the chain rule, we obtain

$$\begin{aligned} \frac{\partial \mathbf{x}^s}{\partial t}(t, \theta) &= \frac{\partial \xi}{\partial t}(t, \omega(t, \theta)) + \frac{\partial \xi}{\partial \omega}(t, \omega(t, \theta)) \frac{\partial \omega}{\partial t}, \\ &= \nabla \Phi(t, \xi(t, \omega)) + \frac{\partial \xi}{\partial \omega}(t, \omega(t, \theta)) \frac{\partial \omega}{\partial t}, \\ &= \nabla \Phi(t, \mathbf{x}^s(t, \theta)) + \frac{\partial \xi}{\partial \omega}(t, \omega(t, \theta)) \frac{\partial \omega}{\partial t}. \end{aligned}$$

Using

$$\frac{\partial \mathbf{x}^s}{\partial \theta}(t, \theta) = \frac{\partial \xi}{\partial \omega}(t, \omega(t, \theta)) \frac{\partial \omega}{\partial \theta}(t, \theta),$$

equation (4) easily follows by introducing the scalar field

$$s(t, \theta) = \left[\frac{\partial \omega}{\partial \theta}(t, \theta) \right]^{-1} \frac{\partial \omega}{\partial t}(t, \omega(t, \theta)).$$

A.3. DERIVATION OF EQUATIONS (7a-7b).

Let us introduce the compact notation

$$\begin{pmatrix} \mathbf{x} \\ \phi \end{pmatrix}_i^{n+1} = \begin{pmatrix} \mathbf{x}_i^{n+1} \\ \phi_i^{n+1} \end{pmatrix}$$

and the mapping

$$\mathcal{F}\left[\begin{pmatrix}\mathbf{x}\\ \phi\end{pmatrix}_i^{n+1}\right] = \left[\begin{array}{l} +\frac{\Delta t}{2}\left[\nabla\Phi(t^{n+1},\mathbf{x}_i^{n+1})+\nabla\Phi(t^n,\mathbf{x}_i^n)\right]+\frac{\Delta t}{2}\left[s_i^{n+1}\frac{\mathbf{x}_{i+1}^{n+1}-\mathbf{x}_{i-1}^{n+1}}{\theta_{i+1}-\theta_{i-1}}+s_i^n\frac{\mathbf{x}_{i+1}^{n}-\mathbf{x}_{i-1}^{n}}{\theta_{i+1}-\theta_{i-1}}\right] \\ \\ +\frac{\Delta t}{4}\left[\left|\nabla\Phi(t^{n+1},\mathbf{x}_i^{n+1})\right|^2+\left|\nabla\Phi(t^n,\mathbf{x}_i^n)\right|^2\right]-\frac{\Delta t}{2}\Big[s_i^{n+1}\frac{\mathbf{x}_{i+1}^{n+1}-\mathbf{x}_{i-1}^{n+1}}{\theta_{i+1}-\theta_{i-1}}\cdot\nabla\Phi(t^{n+1},\mathbf{x}_i^{n+1}) \\ \qquad +s_i^n\frac{\mathbf{x}_{i+1}^{n}-\mathbf{x}_{i-1}^{n}}{\theta_{i+1}-\theta_{i-1}}\cdot\nabla\Phi(t^n,\mathbf{x}_i^n)\Big]-g\frac{\Delta t}{2}\left[\mathbf{x}_i^{n+1}+\mathbf{x}_i^n\right]\cdot\hat{\mathbf{z}}. \end{array}\right] \tag{A.2}$$

that allows equations (6a-6b) to be re-written in the more compact form:

$$\begin{pmatrix}\mathbf{x}\\ \phi\end{pmatrix}_i^{n+1} = \begin{pmatrix}\mathbf{x}\\ \phi\end{pmatrix}_i^{n} + \mathcal{F}\left[\begin{pmatrix}\mathbf{x}\\ \phi\end{pmatrix}_i^{n+1}\right]. \tag{A.3}$$

Let us assume that the solution of (A.3) is known at the iterative step $n+\ell/L$.

The fixed-point algorithm of equations (7a-7b) is built by applying a two-stage relaxed predictor-corrector method. The predictor-stage solution labeled by the super-script "$n+(\ell+1)/L,\star$" is given by

$$\begin{pmatrix}\mathbf{x}\\ \phi\end{pmatrix}_i^{n+(\ell+1)/L,\star} = \begin{pmatrix}\mathbf{x}\\ \phi\end{pmatrix}_i^{n} + \mathcal{F}\left[\begin{pmatrix}\mathbf{x}\\ \phi\end{pmatrix}_i^{n+\ell/L}\right].$$

The corrector-stage solution is finally given by introducing the relaxation parameter α and using the relation

$$\begin{aligned}\begin{pmatrix}\mathbf{x}\\ \phi\end{pmatrix}_i^{n+(\ell+1)/L} &= \begin{pmatrix}\mathbf{x}\\ \phi\end{pmatrix}_i^{n+\ell/L} + \alpha\left[\begin{pmatrix}\mathbf{x}\\ \phi\end{pmatrix}_i^{n+(\ell+1),\star} - \begin{pmatrix}\mathbf{x}\\ \phi\end{pmatrix}_i^{n+\ell/L}\right], \\ &= \alpha\begin{pmatrix}\mathbf{x}\\ \phi\end{pmatrix}_i^{n} + (1-\alpha)\begin{pmatrix}\mathbf{x}\\ \phi\end{pmatrix}_i^{n+\ell/L} + \mathcal{F}\left[\begin{pmatrix}\mathbf{x}\\ \phi\end{pmatrix}_i^{n+\ell/L}\right],\end{aligned}$$

and afterA.2 the fixed-point algorithm of equations (7a-7b) can be clearly identified.

REFERENCES

1. E. Bertolazzi and G. Manzini, *Algorithm 817 P2MESH: generic object-oriented interface between 2-D unstructured meshes and FEM/FVM-based PDE solvers*, ACM TOMS, 28 (2002), pp. 101–132.
2. F. Brezzi and M. Fortin, *Mixed and hybrid finite element methods*, in Springer Series in Computational Mathematics, 15, Springer Verlag, 1991.
3. B. Buchmann, *Accuracy and stability of a set of free-surface time-domain boundary element models based on b-splines*, Int. J. Numer. Meth. Fluids, 33 (2000), pp. 125–155.
4. P. Ciarlet, *The Finite Element Method for Elliptic Problems*, North-Holland, Amsterdam, 1978.
5. M. J. Cooker and D. H. Peregrine, *Pressure impulse theory for liquid impact problems*, Journal of Fluid Mechanics, 297 (1995), pp. 193–214.
6. J. Crank and P. Nicholson, *A practical method for numerical evaluation solutions of partial differential equations of the heat conduction type*, 1947.
7. J. Dold and D. Peregring, *Steep unsteady water waves: an efficient computational scheme*, in Proc. 19th Int. Conf. on Coastal Engineering, Houston, TX, 1984, pp. 955–967.
8. ———, *An efficient boundary-integral method for steep unsteady water waves*, in Numerical Methods for Fluid Dynamics II, K. Morton and M. Baines, eds., Oxford, 1986, Oxford University Press, pp. 671–679.
9. F. Harlow and J. Welch, *Numerical calculation of time-dependent viscous incompressible flow*, Phys. Fluids, 8 (1965), p. 2182.
10. C. Hirt, A. Amsden, and J. Cook, *An arbitrary Lagrangian-Eulerian computing method for all flow speeds*, J. Comp. Phys., 14 (1974), p. 227.
11. C. Hirt, J. Cook, and T. Butler, *A Lagrangian method for calculating the dynamics of an incompressible fluid with free surface*, J. Comp. Phys., 5 (1970), p. 103.
12. *HSL: A collection of Fortran codes for large scale scientific computation*, http://www.cse.clrc.ac.uk/Activity/HSL, 2000.
13. A. A. Korobkin and V. V. Pukhnachov, *Initial stage of water impact*, Annual review of fluid mechanics, 1988, ch. 20, pp. 159–185.
14. M. Longuet-Higgings and E. Cokelet, *The deformation of steep surface waves on water. Part I. A numerical method of computation*, Proc. R. Soc. Lond. A, 350 (1976), pp. 1–26.
15. R. Machane and E. Canot, *High-order schemes in boundary element methods for transient non-linear free surface problems*, Int. J. Numer. Meth. Fluids, 24 (1997), pp. 1049–1072.
16. L. M. Milne-Thompson, *Theoretical Hydrodynamics, 5th Edition*, MacMillan, New York, 1968.
17. B. Nichols and C. Hirt, *Methods for calculating multidimensional, transient free surface flows past bodies*, in First International Conf. On Num. Ship Hydrodynamics, Gaithersburg, ML, Oct. 20-23 1975.
18. ———, *Numerical simulation of BWR vent-clearing hydrodynamics*, Nucl. Sci. Eng., (1980).
19. H. Oguz and A. Prosperetti, *Bubble entrainment by the impact of drops on liquid surfaces*, J. Fluid Mech., 219 (1990), pp. 143–179.
20. H. Oguz, A. Prosperetti, and A. Kolaini, *Air entrapment by a falling water mass*, J. Fluid Mech., 294 (1995), pp. 181–207.
21. S. Osher and J. Sethian, *Fronts propagating with curvature dependent speed: Algorithms based on a Hamilton-Jacobi formulation*, J. Comput. Phys., 79 (1988).
22. D. H. Peregrine and M. E. Topliss, *The impact of water waves upon a wall*, in Proceedings of the IUTAM/ISIMM Symposium on Structure and Dynamics of Nonlinear Waves in Fluids, 17–20 August, 1994, pp. 83–98.
23. A. Prosperetti and H. N. Oguz, *Air entrainment upon liquid impact*, Philos. Trans. Roy. Soc. London Ser. A, 355 (1997), pp. 491–506.
24. A. Ritter, *Die fortpflanzung der wasserwellen*, Zeitschrift des vereines deutscher ingenieure, 36 (1892), pp. 947–954.
25. P. Scotton, *Dynamic impact of debris flow : experimental study*, tech. rep., IDR2, Department of Civil and Environmental Engineering, University of Trento, Italy, 1996.
26. J. Shewchuk, *Triangle homepage. URL:*˜ `http://almond.srv.cs.cmu.% edu/afs/cs/project/quake/public/-www/triangle.html`.
27. ———, *Delaunay Refinement Mesh Generation*, PhD thesis, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania, 1997. Technical Report CMU-CS-97-137.

28. ———, *Delaunay refinement algorithms for triangular mesh generation*, Computational Geometry: Theory and Applications, 22 (2002), pp. 21–74.

29. M. Sussman, P. Smereka, and S. Osher, *A level set approach for computing solutions to incompressible two-phase flow*, J. Comput. Phys., 124 (1994), pp. 146–159.

30. F. Trivellato, *Hydrodynamic forces due to the impact of a water bore on a structure*, in Proceedings, First International Conference on Fluid Structure Interaction, Halkidiki, Greece, 2001.

31. F. Trivellato and P. Scotton, *Bore impact upon a wall (experimental database)*, tech. rep., Department of Civil and Environmental Engineering, University of Trento, Italy, 2001.

32. S. Zhang, D. K. P. Yue, and K. Tanizawa, *Simulation of plunging wave impact on a vertical wall*, Journal of Fluid Mechanics, 327 (1996), pp. 221–254.

33. Y. Zhu, H. Oguz, and A. Prosperetti, *Air entrainment by impinging liquid jets*, J. Fluid Mech., 404 (2000), pp. 151–177.