Consiglio Nazionale delle Ricerche

Istituto di Matematica Applicata e Tecnologie Informatiche

PUBBLICAZIONI

Enrico Bertolazzi, Gianmarco Manzini

COMPUTER MODELING OF LIQUID-SOLID IMPACTS

N. 15-PV 2004

Computer Modeling of Liquid-Solid 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 bore on a rigid wall. The solution of the resulting free-interface flow problem is numerically approximated by a tracking-method of new conception. Basically, the free interface separating liquid and air is assumed to be a free fluid line. Its shape and location are tracked in time by numerically solving the evolutive equations of a set of interface node positions and potentials. The evolutive equations are derived from the Bernoulli's law and are integrated by the Crank-Nicholson method. As the shape of the computational domain evolves in time, the domain is fully re-meshed at each time step, and a new steady mixed Dirichlet-Neumann Laplacian problem is formulated and solved by applying the RT_0 mixed finite element method. This potential flow solver has been validated by simulating the liquid-solid impact of a bore against a rigid wall and comparing the numerical results with the available experimental measurements.

Key Words: Partial Differential Equation, Potential Flow Model, Mixed Finite Element Method, Front Tracking Method, Computer Simulation, Free Surface Simulation

1. INTRODUCTION

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 in Reference [18]. Examples of such collision phenomena are wave maker 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 the numerical modeling of the bore impact of a liquid mass onto a rigid wall. In such a case, we are interested in predicting the hydrodynamic loading due to the impact of a bore on a wall because this quantity is crucial in the rational design of protection structures. In particular, we address a new numerical method that is suitable to describe the time evolution of 2-D free-surface flow configurations. The free surface is an interface between 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 arbitrary 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 embedded 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 [14, 15] may be consulted for early examples of these approaches. Among the most pertinent and latest works that are based on the Lagrangian formulation, it is surely worth mentioning those of References [1, 4, 10, 17], and in particular the one of Reference [33], where an enhanced-discretization interface-capturing technique is developed for solving unsteady flows with interface, such as two-fluid and free-surface flows.

A different approach is instead devised by the Eulerian methods. In such methods, the computing grid is kept fixed and the fluid volumes are tracked in time instead of the free surfaces. The surfaces may thus appear, merge and 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 [13] and the Volume-Of-Fluid (VOF) method [22, 23] and all their many variants and improvements. The fluid volumes are tracked in the MAC method by a set of fluid marker particles, while a special indicator function, which is 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 like the gas pressure and physical constraints like 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 preserves the step function nature of the indicator.

The *Level Set* method has been originally developed to solve moving interface problems [26] and then applied to free surface fluid problems [32], 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 predicts 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 address in this paper can be considered as an interface tracking method. Basically, the method is capable of following the interface position on the computational domain over a long period of time and determining how interface movement affects the flow configuration. As the interface equation is derived from the Bernoulli's law, the numerical scheme relies on the assumption that the gas-liquid interface be an equipotential surface for the liquid fluid and that the potential theory apply. 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 a special non-linear algorithm. 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 continuously evolving in time. The numerical approximation of the Laplace's problem is given by using the lowest order Raviart-Thomas (RT_0) space in the framework of the Mixed Finite Element Method.

It is worth noting that the resulting numerical model 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 movement. 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 that this latter fact is a crucial aspect of this new method and for this reason it deserves a thorough discussion.

First, it should be pointed out that the potential formulations and their numerical approximations have been investigated in the past decades by mainly using the *Boundary Element Method* (BEM). The pioneering work of Reference [20] illustrates how an integral equation formulation can be successful in numerically treating time-dependent non-linear problems. Other early applications are found in References [11, 12], where a time integration scheme based on a Taylor series development is proposed. We also cite References [6, 21, 24, 25, 37] among the many more recent papers using BEMs applied to potential theory-based models. In all these works, the BEM has proved accurate and computationally efficient. The authors of these papers generally claim that the 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 the BEM but is instead needed to follow the evolution of the free surface in other integration methods such as the Lagrangian ones cited above.

Nonetheless, it is our experience that the 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 [3]. We

wish to point out that the software utilized in our solver implementation is public domain available and absolutely general purpose; i.e., it has not been specifically designed to treat impact problems with varying or adaptive meshes. In fact, the generation of the new mesh at each time step is carried out by calling the mesh generator TRIANGLE, which has been incorporated in our C^{++} solver as a library sub-function. The mesh generator TRIANGLE is described in References [30, 31] and is distributed on-line at the URL of Reference [29]. All the geometrical and topological data related to the mesh are obtained by using P2MESH [2], a publicly available collection of C^{++} classes suited to this purpose. On the basis of our practical experience, 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. Thus, it is our feeling that the method proposed in this work may be a valid alternative to other numerical techniques (like the BEM) for the numerical treatment of impact models, even if a comparative assessment of performance is beyond the scope of the present paper.

The outline of the paper is as follows. In Section 2 we introduce the analytical formulation of the mathematical model, which is based on the potential theory under the assumptions that the fluid is incompressible and the flow is irrotational. In Section 3 we present the numerical algorithm suitable to the time integration of the model equations. In Section 4, typical results of numerical simulations are discussed to illustrate both the capability of the method in predicting the behavior of free flow configurations on liquid-solid collision problems. Final remarks and conclusions are offered in Section 5.

2. THE ANALYTICAL FORMULATION OF THE PROBLEM

In this section, we formulate the mathematical model that describes liquid-solid impact phenomena. The mathematical model is based on the potential theory under the assumptions that the fluid be incompressible and the flow irrotational.

The theoretical formulation is referred to the computational liquid domain $\Omega(t)$, which is a simply connected and time dependent region of the two-dimensional space, see Figure 1. The domain boundary is at least piecewise Lipschitz continuous smooth and is defined by

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

by 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

$$\mathbf{S}_0(t,\theta) = \mathbf{x}^s(t,0) = \mathbf{x}^w(t,1),$$

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

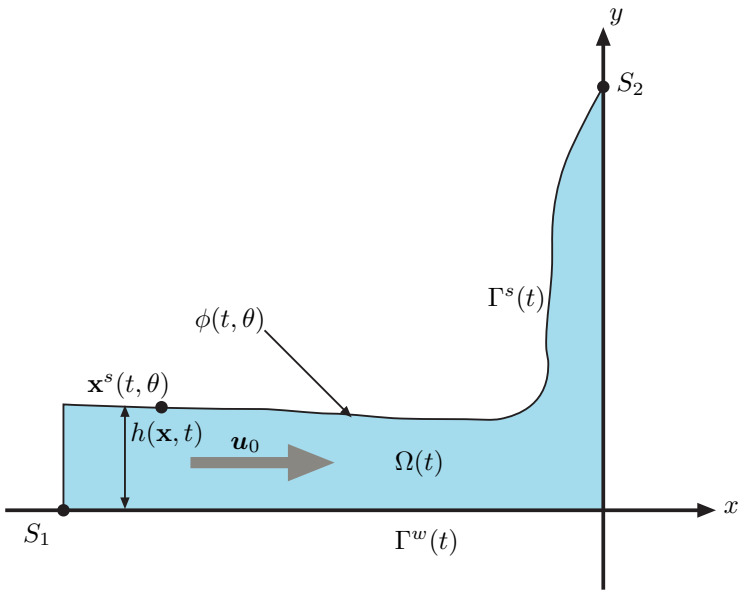


FIG. 1. Bore impact on a rigid wall: sketch of the problem and notation.

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 $\Gamma^w(t)$

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

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

The fluid energy conservation expressed by the Bernoulli's law for the velocity potential $\Phi = \Phi(t, \mathbf{x})$ requires that

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

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.

Substituting the relation $\Phi(t, \mathbf{x}^s(t, \theta)) = \phi(t, \theta)$ into (2), applying the chain rule of derivation, and taking $p = 0$ at $\Gamma^s(t)$ yields the final relation implemented in our numerical model, which is

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

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 unit vector 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 yield the required model equation. 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 of the form

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

By applying the chain rule, we have

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

Finally, by using

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

and by introducing the scalar field

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

we get the following model equation

$$\mathbf{v}^s(t,\theta) = \nabla\Phi(t,\mathbf{x}^s(t,\theta)) + s(t,\theta)\frac{\partial \mathbf{x}^s}{\partial \theta}(t,\theta). \tag{3}$$

The scalar field $s(t,\theta)$ in equation (3) 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

$$s(t,\theta) = \varepsilon \left|\frac{\partial^2 \mathbf{x}^s}{\partial s^2}\right| \left|\frac{\partial \mathbf{x}^s}{\partial \theta}\right|^{-1} \tag{4}$$

in the evolution equation (3) where $s(\theta) = \displaystyle\int_0^\theta |d\mathbf{x}^s/d\theta|\, d\theta$ is the arc length, and ε a coefficient which is linearly decreasing with the mesh size (see the beginning of the next section for a more precise definition). 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 $t = 0$.

3. THE NUMERICAL MODEL

In this section, we formulate the numerical model suitable to describe the liquid-solid impact of a bore against a a rigid vertical wall. This numerical model relies on the potential based model of the previous section. The three next subsections focus on distinct aspects of the numerical algorithm. In the first subsection, we describe how our numerical method evolves in time the shape of the free surface separating air and water. In the second subsection, we briefly describe the mixed finite element solver that computes the approximate flow field within $\Omega(t)$. Finally, in the third subsection, we report some details on the Least Squares-based reconstruction algorithm that is implemented in the present solver to improve the approximation of the shape of the free surface.

3.1. The free surface moving 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 section the node index 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-dependent coefficient in the definition of $s(t,\theta_i)$ given in (4) is $\varepsilon = \dfrac{1}{2N}$. The node positions and potentials on the free boundary line satisfy the time-dependent set of discrete equations:

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

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 [9] by evaluating node positions at the free boundary line and the boundary potentials at time t^{n+1}

at the node labeled by i:

$$\mathbf{x}_i^{n+1} = \mathbf{x}_i^n + \frac{\Delta t}{2}\left[\nabla\Phi(t^{n+1},\mathbf{x}_i^{n+1}) + \nabla\Phi(t^n,\mathbf{x}_i^n)\right] + \frac{\Delta t}{2}\left[s_i^{n+1}\frac{\mathbf{x}_{i+1}^{n+1}-\mathbf{x}_{i-1}^{n+1}}{\theta_{i+1}-\theta_{i-1}} + s_i^n\frac{\mathbf{x}_{i+1}^n-\mathbf{x}_{i-1}^n}{\theta_{i+1}-\theta_{i-1}}\right], \tag{5a}$$

$$\phi_i^{n+1} = \phi_i^n + \frac{\Delta t}{4}\left[\left|\nabla\Phi(t^{n+1},\mathbf{x}_i^{n+1})\right|^2 + |\nabla\Phi(t^n,\mathbf{x}_i^n)|^2\right] + \frac{\Delta t}{2}\left[s_i^{n+1}\frac{\mathbf{x}_{i+1}^{n+1}-\mathbf{x}_{i-1}^{n+1}}{\theta_{i+1}-\theta_{i-1}}\cdot\nabla\Phi(t^{n+1},\mathbf{x}_i^{n+1}) + s_i^n\frac{\mathbf{x}_{i+1}^n-\mathbf{x}_{i-1}^n}{\theta_{i+1}-\theta_{i-1}}\cdot\nabla\Phi(t^n,\mathbf{x}_i^n)\right] - g\frac{\Delta t}{2}\left[\mathbf{x}_i^{n+1}+\mathbf{x}_i^n\right]\cdot\hat{\mathbf{z}}. \tag{5b}$$

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 derivation of this iterative procedure follows. First, we 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

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

that allows equations (5a-5b) to be re-written in the more compact form:

$$\begin{pmatrix}\mathbf{x}\\ \phi\end{pmatrix}_i^{n+1} = \begin{pmatrix}\mathbf{x}\\ \phi\end{pmatrix}_i^{n} + \mathcal{F}\left[\begin{pmatrix}\mathbf{x}\\ \phi\end{pmatrix}_i^{n+1}\right]. \tag{6}$$

Let us assume that the solution of (6) is known at the iterative step $n+\ell/L$. The fixed-point algorithm 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 given by introducing the relaxation parameter α and using the relation

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

The fixed-point algorithm takes the final form

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

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

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 iterations L that are needed to achieve convergence in the fixed-point iterative algorithm (7a-7b) generally depends on the time-step iterative level n. At any time step t^n, the complete re-meshing of $\Omega_h(t^n)$ is performed, 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. Convergence is achieved for values of L that are typically less than 10.

3.2. 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 [7, 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 [5] 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 [5]:

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

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 *MA57* of the *HSL-2000* library [16].

3.3. 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 by assuming that the linear dependence on $\mathbf{x}$

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

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

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

4. NUMERICAL EXPERIMENTS

A clear water bore hitting upon a rigid, vertical, plane wall is considered. The surge is generated by a dam break and propagates over a dry horizontal bed. This impact phenomenon has been extensively investigated and experimental results have been collected in References [28, 35]. The interested reader can also found a classification for bore impact in Reference [34]. Basically, the aim of this section 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 the wall pressure behavior, whose knowledge is crucial in the rational design of transversal structures. Such a liquid-solid impact problem can be typically formulated by assuming that at the initial time $t = 0$ the liquid mass meets the wall. At this precise moment, the wall location, the liquid domain and the flow field are assumed to be known. Instead, for $t > 0$ the flow field together with liquid actions on the wall have to be determined.

The model presented in section 2 to describe mathematically the collision process is true under some simplifying hypotheses that we discuss below. The first one concerns the compressibility of

pure water that does not play a significant role in this kind of impact [27]: 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 [36]. 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 laboratory experimental impact chosen to test the numerical model is the clear water bore, detailed in References [28] and [35]. The initial condition for the whole flow field is not known from the physical experiments. Instead, only the toe velocity, which is apparently a rough estimate of the bulk velocity, was measured, providing a value $U = 2.77$ m/s. It should be pointed out that the measured velocity is the picture of an instant of time and can hardly represent a highly unsteady phenomenon. It is also 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 stream-wise fluctuations.

The complete simulation of the dam-break surge, starting from the removal of the gate and proceeding towards the impact wall, clearly provides a reasonable flow field to start the calculation of the impact. Such a flow field is calculated by using the freely available Shallow Water flow solver CLAWPACK, which is run until the bore arrives at the wall. More details on this software package are available at the URL of Reference [19]. The height field $h(\mathbf{x}, t)$ (see Figure 1), which is obtained by solving the initial Shallow Water problem, also provides a good approximation of the initial computational domain. As it was observed in the video pictures of the experiments, the shape of this computational domain has a strong resemblance with the mildly elongated physical toe when the collision starts to take place.

In order to validate the prediction capability of the physical and numerical model presented in this paper, we simulated four different experiments that are detailed in References [28] and [35]. The four experiments differs for the height h_f of the surge that is initially at rest before the removal of the gate; see also Figure 1. For $h_f = 10$, 20, 30 and 40, two sets of normalized experimental measures of the wall force are available and can be used for comparison with the force that is numerically predicted by the model. By trial and error a convenient temporal step is found to be 10^{-4} seconds. The typical number of generated triangles is 1000. The typical CPU time on a 500 *Mhz* processor computing machine is half an hour for a complete simulation involving about $3.5 \cdot 10^3$ time steps.

Figures 2 and 3 shows the free surface profile evolution every 5 ms and starting from $t = 0$. The computer simulations is arrested when the free surface that is detaching from the wall collides against itself. The last profile is displayed at respectively $T = 0.3$ s, 0.45 s, 05 s and 0.65 s for the four simulations for $h_f = 10$, $h_f = 20$, $h_f = 30$ and $h_f = 40$. A comparison between the profiles of the simulated free surface with the real one generally shows a clear discrepancy. The discrepancy is likely due to the huge air entrainment, phenomenon which has not been accounted for in this model.

Instead, the time-dependent impulsive increasing of the wall force, which is the most relevant quantity for engineering design, is predicted quite well. This fact is clearly evident in Figures 4 and 5, that illustrate the comparison between the normalized experimental time series and the numerical predictions. for the same four simulations. In this pictures, the force predicted by the model has been compared against the experimental force measured in two distinct experiments and obtained by integrating the pressure diagrams, [35]. It is important to mention that the model predicts the moving free surface while verifying closely the mass conservation, the maximum error being well less than 1%. It has also 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 can be observed in all these numerical simulations 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 [8] 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. Finally, Figures 6-10 shows the detail near the wall of the time dependent evolution of the computational mesh and the pressure field for the complete simulation run when $h_f = 40$.

5. CONCLUSIONS

A new numerical procedure has been developed and presented for the analysis of the motion of free-surface flow configurations in liquid-solid impacts. The method approximates the solution of a mathematical model that is formulated on the basis of the 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 node position at the free surface in accord with the Bernoulli's law. A full re-meshing of the computational domain is required at each time-step to track the motion of free-surface flows. Despite the appearance, the re-meshing 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 impacts are presented. In particular, the discussion is focused about to what extent the impact process of interest is simplified by applying this model which is based on the potential theory. 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.

REFERENCES

1. S. Aliabadi and T. Tezduyar, *Stabilized-finite-element/interface-capturing technique for parallel computation of unsteady flows with interface*, Computs Methods Appl. Mech. Engrg., 190 (2000), pp. 243–261.
2. 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.
3. E. Bertolazzi and G. Manzini, *A mixed finite element solver for liquid-liquid impacts,*, Communications in Numerical Methods in Engineering, 20 (2004), pp. 595–606. 2003-36-PV.
4. H. Braess and P. Wriggers, *Arbitrary Lagrangian Eulerian finite element analysis of free surface flow*, Computs Methods Appl. Mech. Engrg., 190 (2000), pp. 95–109.
5. F. Brezzi and M. Fortin, *Mixed and hybrid finite element methods*, in Springer Series in Computational Mathematics, 15, Springer Verlag, 1991.
6. 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.
7. P. Ciarlet, *The Finite Element Method for Elliptic Problems*, North-Holland, Amsterdam, 1978.
8. M. J. Cooker and D. H. Peregrine, *Pressure impulse theory for liquid impact problems*, Journal of Fluid Mechanics, 297 (1995), pp. 193–214.
9. J. Crank and P. Nicholson, *A practical method for numerical evaluation solutions of partial differential equations of the heat conduction type*, 1947.
10. M. Cruchaga, D. Celentano, and T. Tezduyar, *A moving Lagrangian interface for flow computations over fixed meshes*, Computs Methods Appl. Mech. Engrg., 191 (2001), pp. 525–543.
11. 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.
12. ———, *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.
13. F. Harlow and J. Welch, *Numerical calculation of time-dependent viscous incompressible flow*, Phys. Fluids, 8 (1965), p. 2182.
14. C. Hirt, A. Amsden, and J. Cook, *An arbitrary Lagrangian-Eulerian computing method for all flow speeds*, J. Comp. Phys., 14 (1974), p. 227.
15. 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.
16. HSL, *(Harwell Subroutine Library), a collection of Fortran codes for large scale scientific computation*, CCLRC, Oxfordshire, UK, 2004. URL http://www.cse.crlc.ac.uk/nag/hsl/.
17. S. Idelsohn, M. Storti, and E. Onate, *Lagrangian formulations to solve free surface incompressible inviscid fluid flows*, Computs Methods Appl. Mech. Engrg., 191 (2001), pp. 583–593.
18. A. A. Korobkin and V. V. Pukhnachov, *Initial stage of water impact*, Annual review of fluid mechanics, 1988, ch. 20, pp. 159–185.
19. R. J. LeVeque, *Clawpack: Conservation law package*. URL http://www.amath.washington.edu/~ claw/index.html, 2004.
20. 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.
21. 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.
22. 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.
23. ———, *Numerical simulation of BWR vent-clearing hydrodynamics*, Nucl. Sci. Eng., (1980).
24. H. Oguz and A. Prosperetti, *Bubble entrainment by the impact of drops on liquid surfaces*, J. Fluid Mech., 219 (1990), pp. 143–179.
25. H. Oguz, A. Prosperetti, and A. Kolaini, *Air entrapment by a falling water mass*, J. Fluid Mech., 294 (1995), pp. 181–207.
26. S. Osher and J. Sethian, *Fronts propagating with curvature dependent speed: Algorithms based on a Hamilton-Jacobi formulation*, J. Comput. Phys., 79 (1988).
27. 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.

28. P. Scotton, *Dynamic impact of debris flow : experimental study*, tech. rep., IDR2, Department of Civil and Environmental Engineering, University of Trento, Italy, 1996.

29. J. Shewchuk, *Triangle homepage. URL:˜* `http://almond.srv.cs.cmu.% edu/afs/cs/project/quake/public/-www/triangle.html`.

30. ———, *Delaunay Refinement Mesh Generation*, PhD thesis, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania, 1997. Technical Report CMU-CS-97-137.

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

32. 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.

33. T. Tezduyar, S. Aliabadi, and M. Behr, *Enhanced-Discretization Interface-Capturing Technique (EDICT) for computation of unsteady flows with interfaces*, Computs Methods Appl. Mech. Engrg., 155 (1998), pp. 235–248.

34. 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.

35. 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.

36. 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.

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

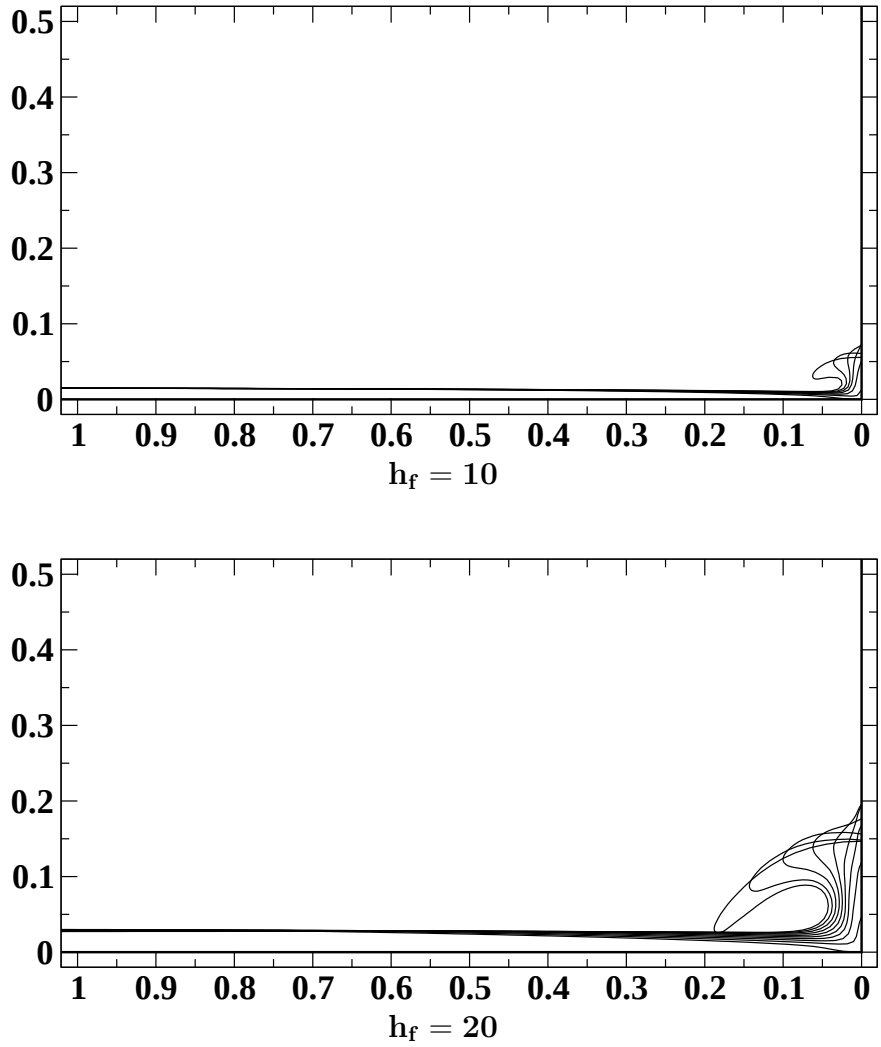


FIG. 2. Free surface profiles for $h_f = 10$ and $h_f = 20$ at different time-steps. Profiles are shown every $5\,10^{-2}$ s from $t = 0$ up to $t = 0.33$ s for $h_f = 10$ and $t = 0.45$ ms for $h_f = 20$.

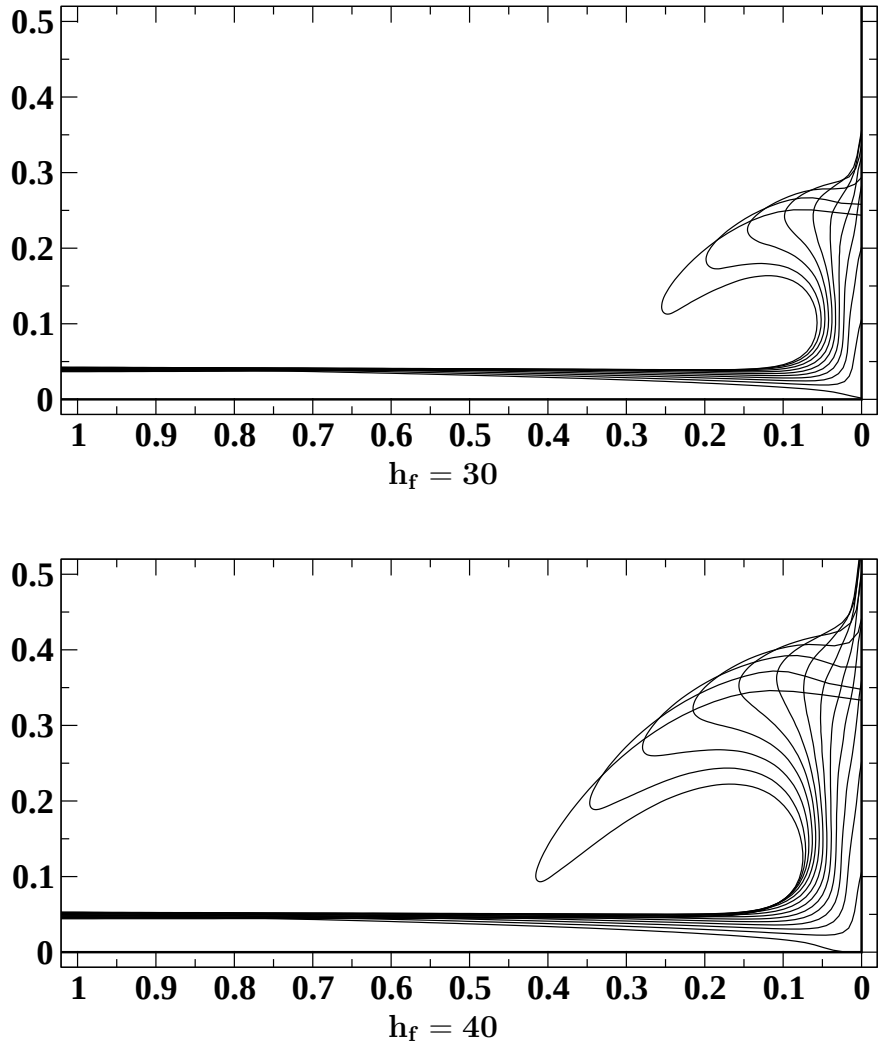
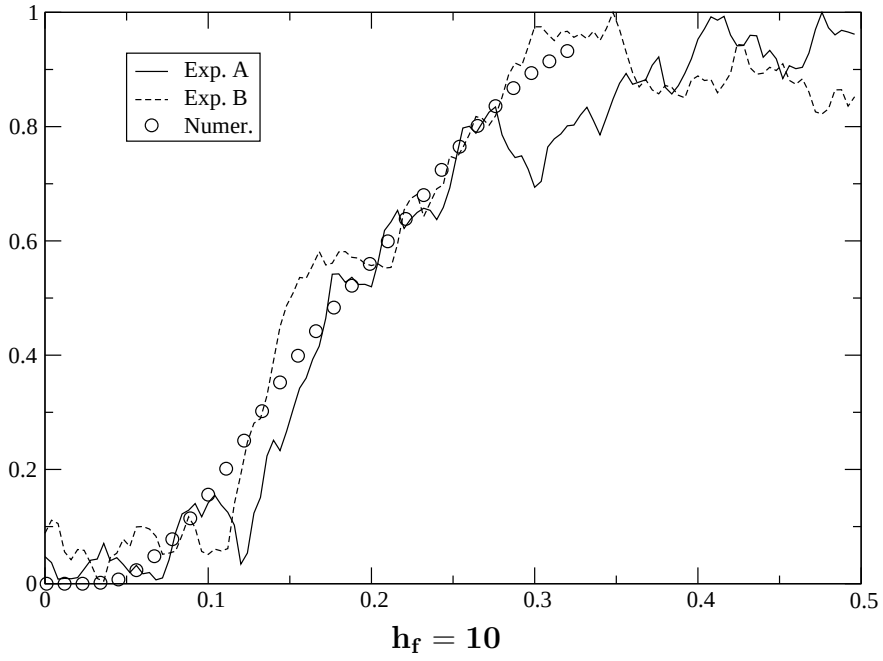


FIG. 3. Free surface profiles for $h_f = 30$ and $h_f = 40$ at different time-steps. Profiles are shown every $5\,10^{-2}$ s from $t = 0$ up to $t = 0.54$ ms for $h_f = 30$ and $t = 0.67$ ms for $h_f = 40$.



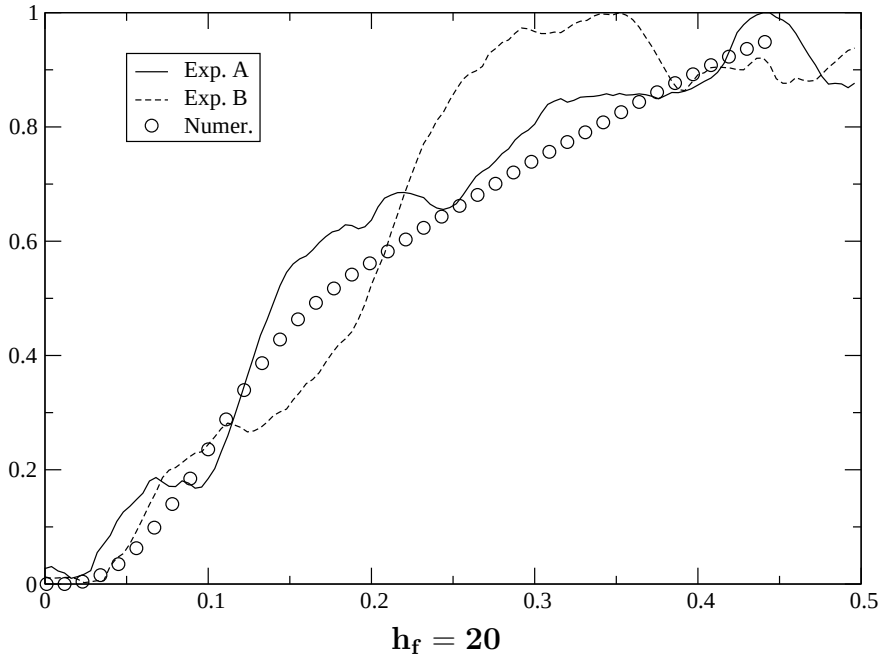
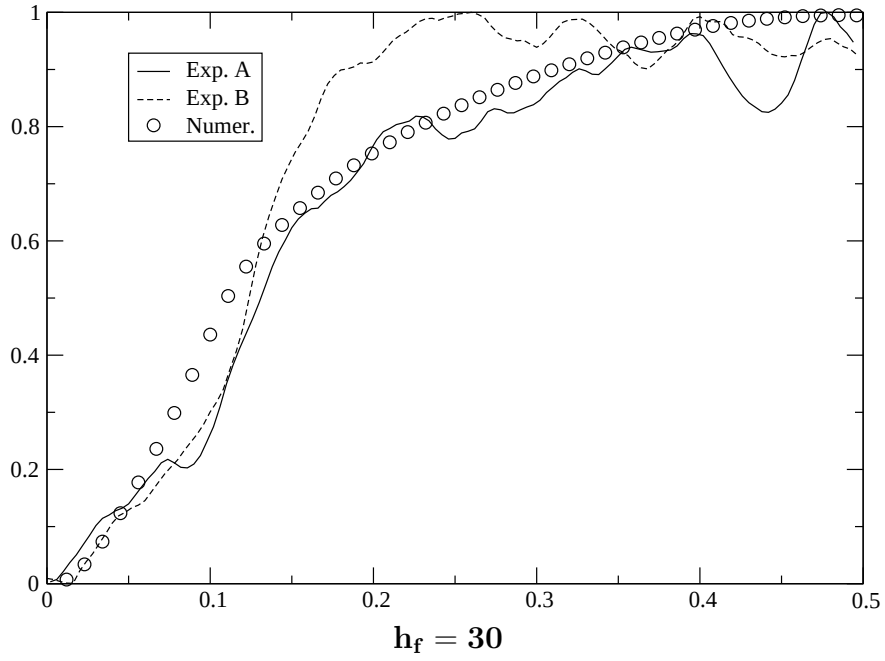


FIG. 4. Normalized time series of the force integrated along the wall for $h_f = 10$ and $h_f = 20$. The numerical result (circles) is super-imposed to two experimental measures labelled by A and B. The horizontal units are given in s, the vertical units are adimensional.



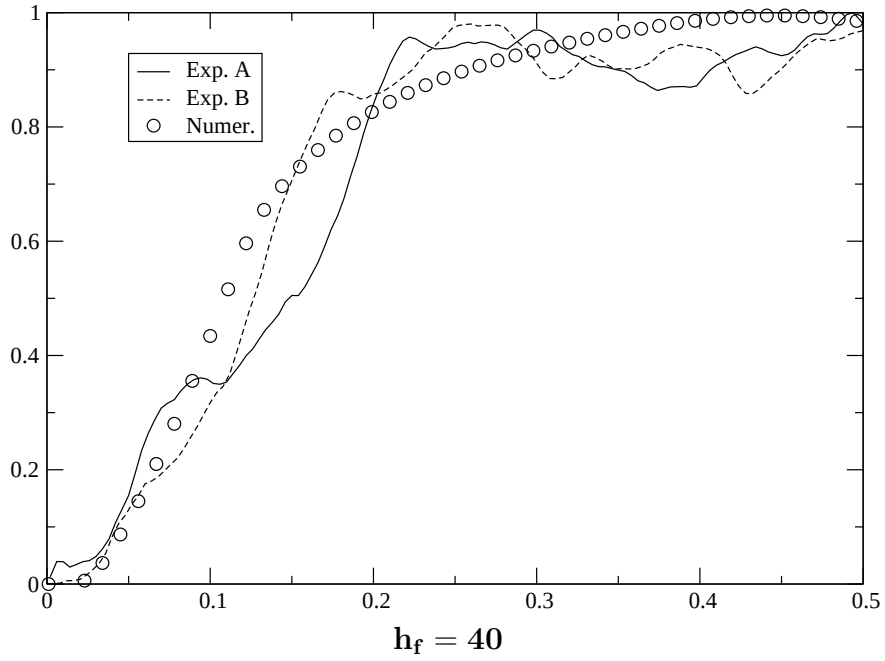


FIG. 5. Normalized time series of the force integrated along the wall for $h_f = 30$ and $h_f = 40$. The numerical result (circles) is super-imposed to two experimental measures labelled by A and B. The horizontal units are given in s, the vertical units are adimensional.

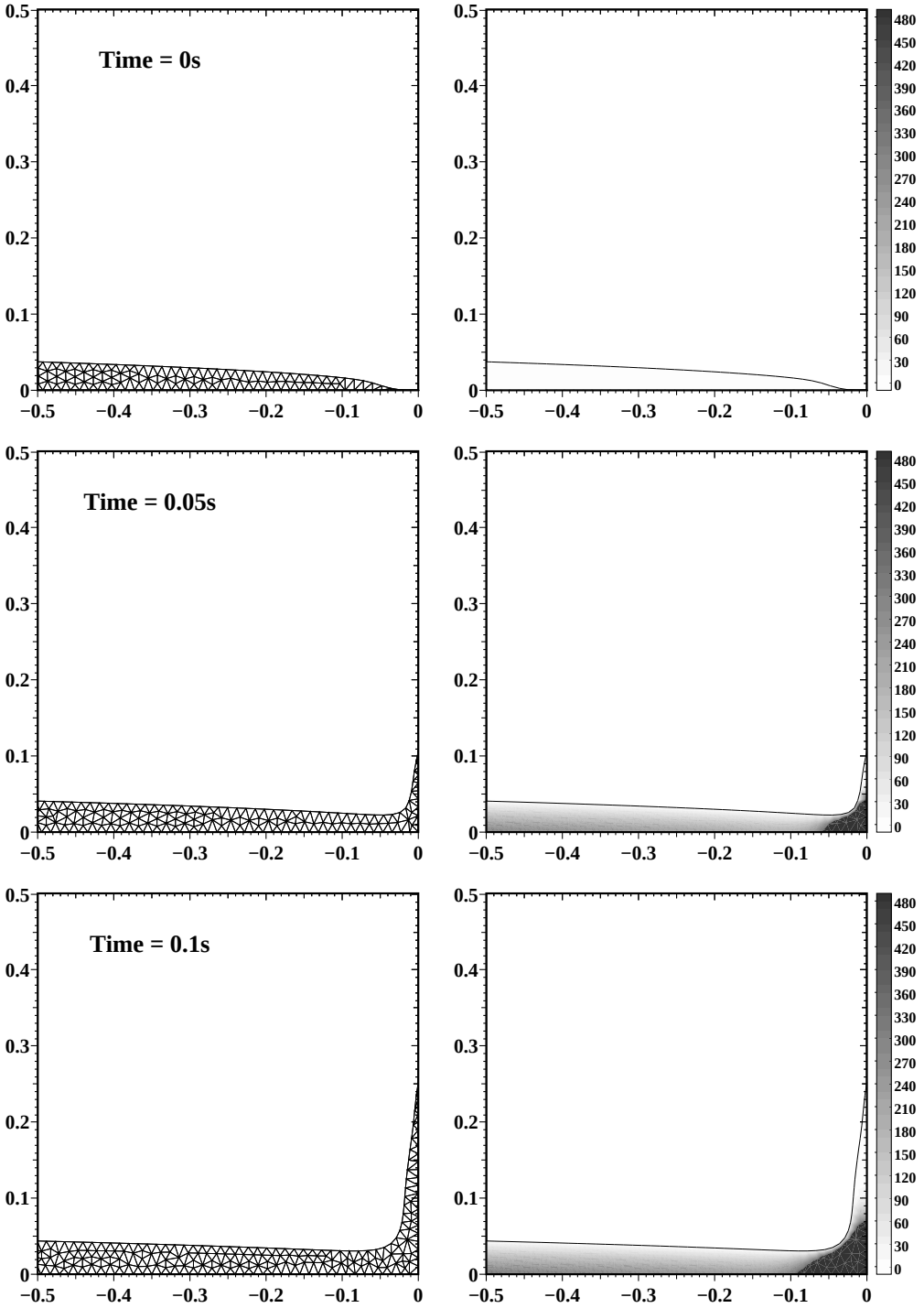


FIG. 6. Mesh and pressure field detail for $h_f = 40$

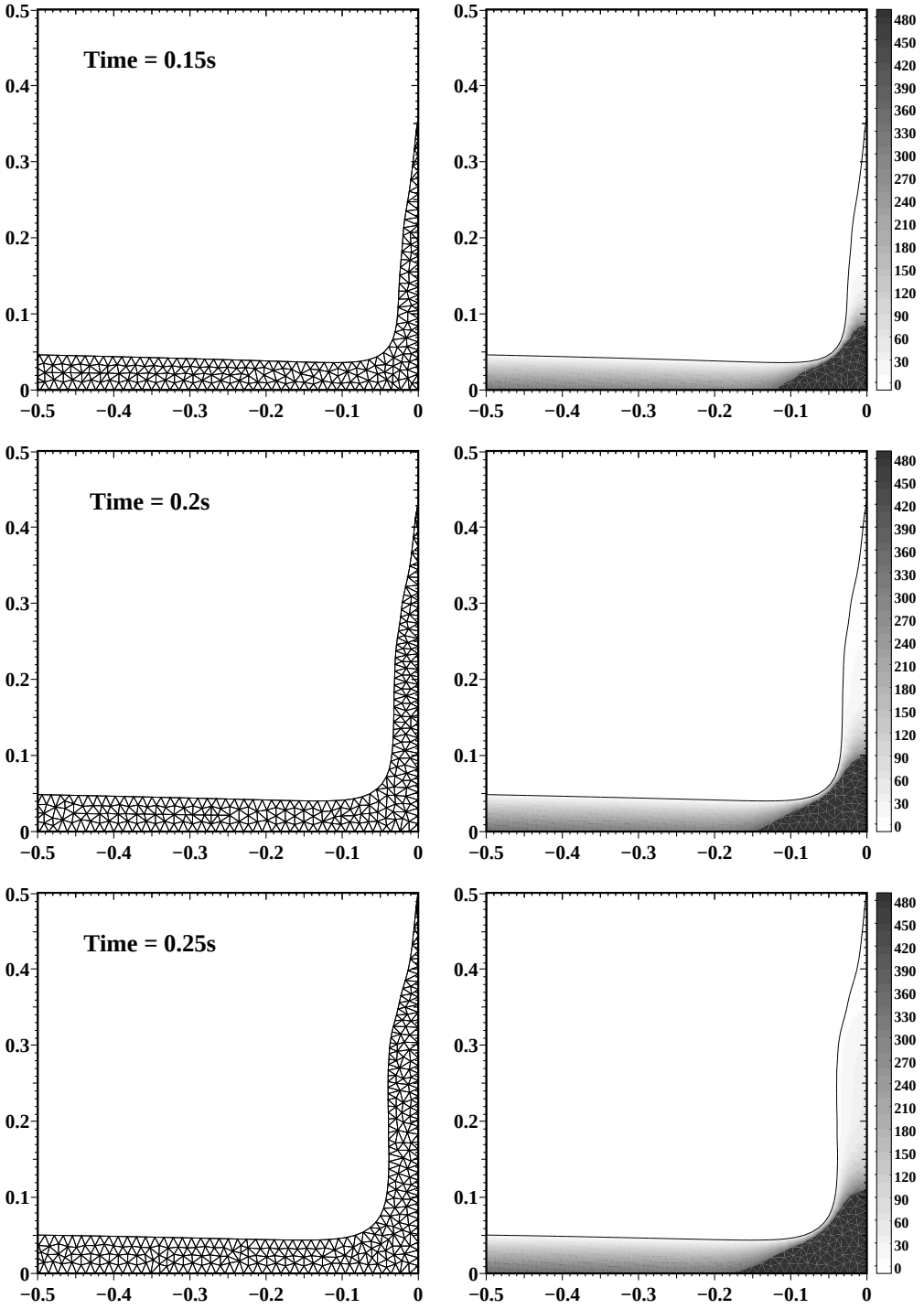


FIG. 7. Mesh and pressure field detail for $h_f = 40$

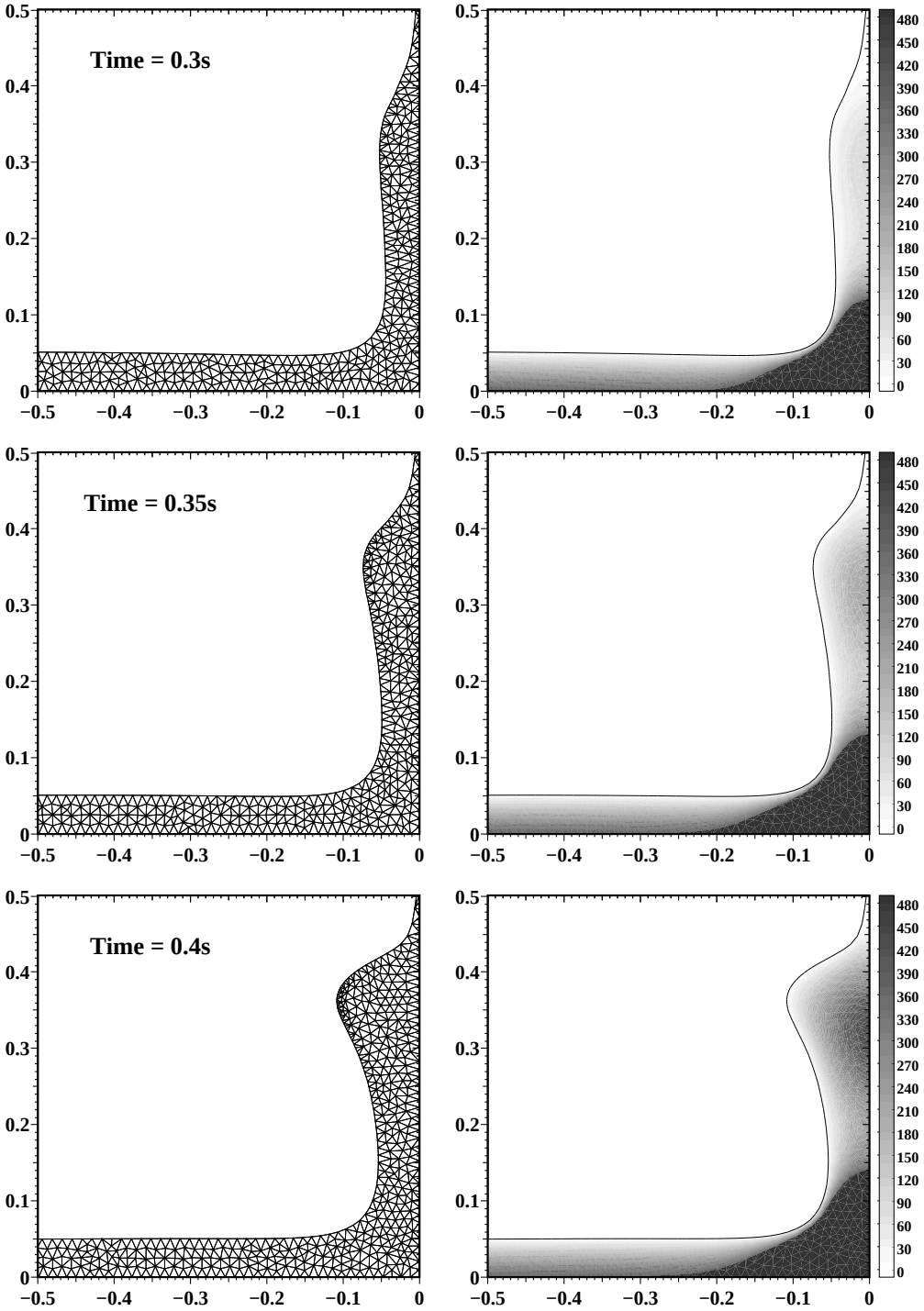


FIG. 8. Mesh and pressure field detail for $h_f = 40$

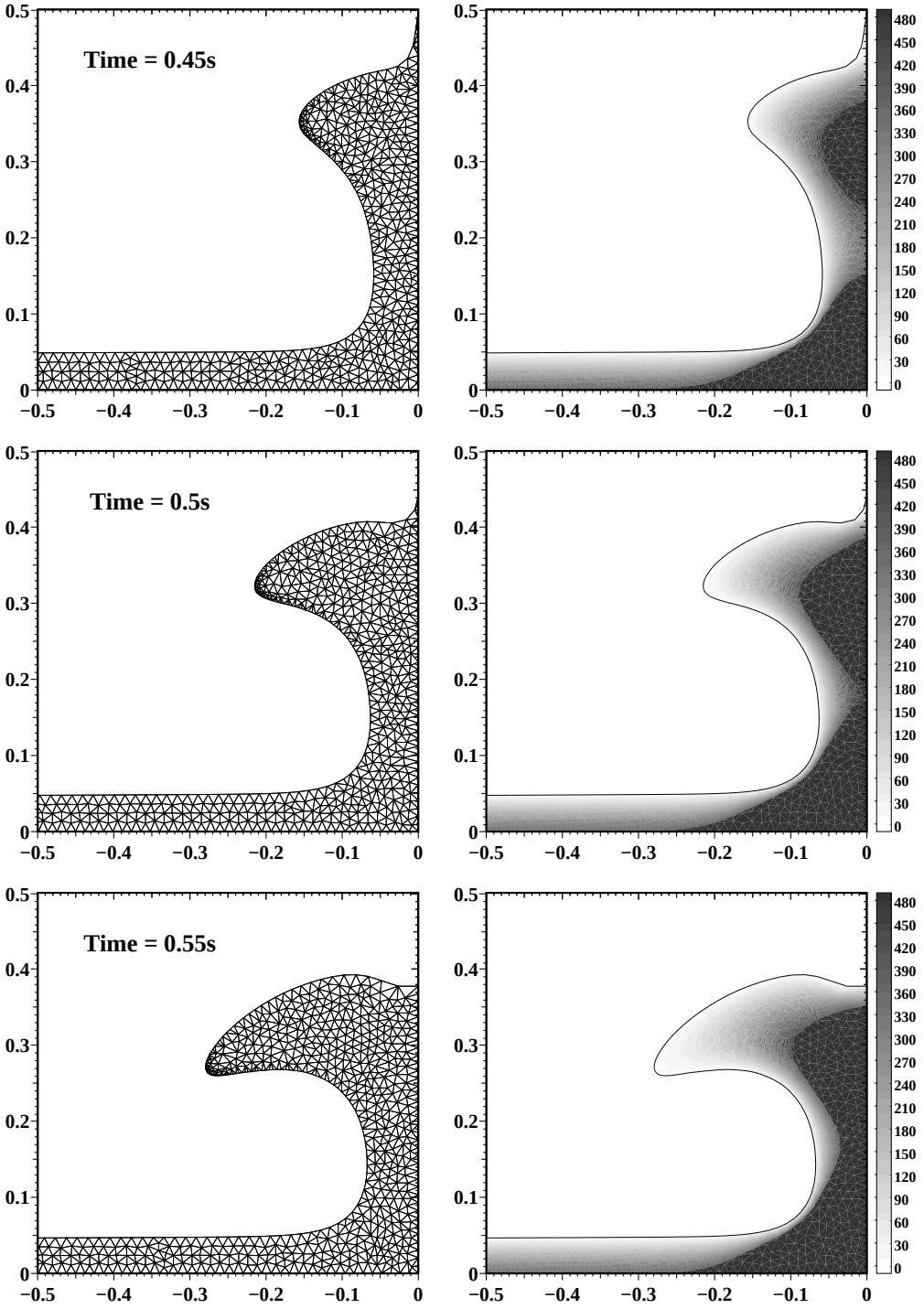


FIG. 9. Mesh and pressure field detail for $h_f = 40$

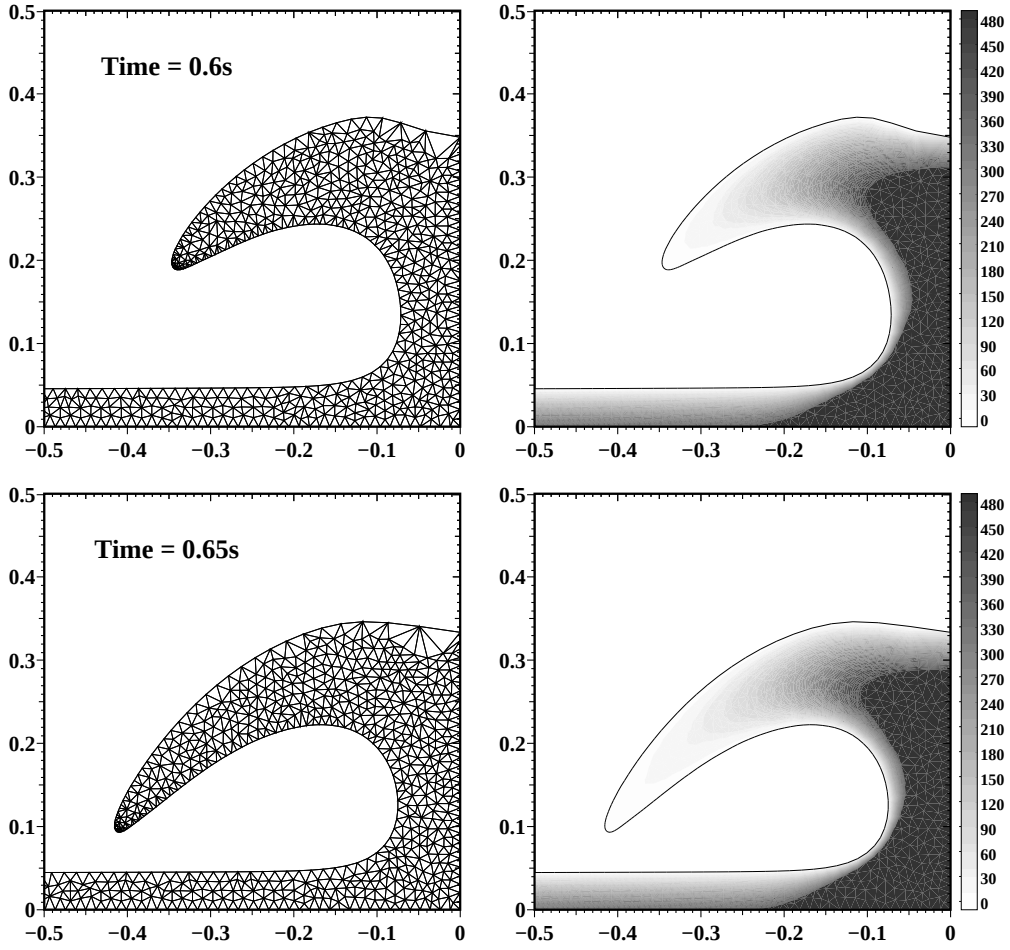


FIG. 10. Mesh and pressure field detail for $h_f = 40$