

## Research Article

# Modeling Free Surface Flows Using Stabilized Finite Element Method

Deepak Garg , Antonella Longo, and Paolo Papale

*Istituto Nazionale di Geofisica e Vulcanologia, Sezione di Pisa, Via Uguccione della Faggiola 32, I-56126 Pisa, Italy*

Correspondence should be addressed to Deepak Garg; [deepak.garg@ingv.it](mailto:deepak.garg@ingv.it)

Received 3 January 2018; Revised 14 May 2018; Accepted 20 May 2018; Published 11 June 2018

Academic Editor: Luis J. Yebra

Copyright © 2018 Deepak Garg et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This work aims to develop a numerical wave tank for viscous and inviscid flows. The Navier-Stokes equations are solved by time-discontinuous stabilized space-time finite element method. The numerical scheme tracks the free surface location using fluid velocity. A segregated algorithm is proposed to iteratively couple the fluid flow and mesh deformation problems. The numerical scheme and the developed computer code are validated over three free surface problems: solitary wave propagation, the collision between two counter moving waves, and wave damping in a viscous fluid. The benchmark tests demonstrate that the numerical approach is effective and an attractive tool for simulating viscous and inviscid free surface flows.

## 1. Introduction

Numerical simulations of flow with moving boundaries have received a growing interest in last few years. Many engineering applications such as ship hydrodynamics, dam break, sloshing in tanks, shallow water, and mold filling [1–3] are modeled with free surface flows. Moreover, many natural risk problems such as tsunami propagation and flood flows require the numerical methods to tackle free surface flows [4, 5]. The continuous variation of the domain shape with time during the numerical solution of such problems is a challenging issue.

Moving domain problems are modeled with interface-capturing and interface-tracking techniques. In interface-capturing methods such as the level set [6, 7] and the volume of fluid [8, 9] a fixed grid is used. The location of the moving interface is captured by an additional equation for an artificial scalar field. A big disadvantage of these methods is that the level set method captures the free surface accurately but does not conserve the mass whereas the volume of fluid conserves the mass but does not capture the free surface accurately. Contrarily, in interface-tracking methods the location of moving boundary is directly tracked by the computational mesh. Generic-front-tracking [10], volume-tracking [11, 12], immersed boundary method [13], and smoothed particle hydrodynamics (SPH) method [14] are few examples of

interface-tracking methods. Interface-tracking approaches are split in Lagrangian and Arbitrary Lagrangian-Eulerian (ALE). In Lagrangian, all mesh nodes follow the fluid motion, while in ALE method only the boundary nodes move with the fluid velocity and the interior nodes move arbitrarily. Interface-tracking techniques are known to provide greater accuracy over interface-capturing techniques because they define the interface more accurately.

In the framework of finite element methods (FEM), ALE method [15–18] and space-time finite element method (STFEM) [19–21] have been successfully used to solve moving domain problems. In traditional ALE method, the governing equations are written in ALE form by introducing the mesh velocity and solved by a semidiscrete approach; i.e., first the spatial discretization is done by standard Galerkin method leaving the time untouched and then time is discretization by some finite difference scheme. In contrast, in STFEM there is no need to include the mesh velocity explicitly in the Navier-Stokes equations as it is mandatory for ALE. In STFEM both space and time are discretized simultaneously generating space-time slabs. In moving domain problems the computation of mesh velocity is treated as an additional subproblem and can be calculated by any mesh deformation approach; e.g., see [22–24].

Numerical wave tank has become a popular choice for studying the natural disasters involving free surface flows

such as tsunami waves [25–27]. In most cases, the wave tank considers an inviscid flow. Our aim in this work is to develop a robust interface-tracking numerical code for simulating viscous and inviscid flows to be used for the natural hazard problems such as the tsunami wave amplification analysis. We use the stabilized STFEM [28] for solving flow equations. The location of the free surface is tracked by the elastic deformation method [19]. The fluid flow and the mesh motion problem are coupled by an iterative segregated algorithm.

The contents of this paper are organized as follows: In Section 2 we present flow governing equations and the boundary conditions. In Section 3 the partitioned algorithm is introduced followed by the detailed numerical approaches for the solution of fluid flow and mesh updating problems. The validation of the numerical method is done in Section 4 over selected free surface flow problems. In Section 5 the concluding remarks are drawn.

## 2. Flow Governing Equations

Let  $\Omega_t \in \mathbb{R}^d$  ( $d = 1, 2, 3$ ) and  $(0, t_{tot})$  be the spatial and temporal domains, respectively, and let  $\Gamma_t$  denote the boundary of  $\Omega_t$ . The Navier-Stokes equations in conservation variables  $\mathbf{U} = [\rho, \rho v_i, \rho e]^T$  are

$$\mathbf{U}_{,t} + \mathbf{F}_{i,i} = \mathbf{E}_{i,i} + \mathbf{S} \quad (1)$$

where  $\mathbf{F}_i$ ,  $\mathbf{E}_i$ , and  $\mathbf{S}$  are the Euler flux, viscous flux, and source vectors, respectively, and are defined as

$$\begin{aligned} \mathbf{F}_i &= \begin{bmatrix} \rho v_i \\ \rho v_j v_i + p \delta_{ji} \\ \rho v_i e + p v_i \end{bmatrix} \\ \mathbf{E}_i &= \begin{bmatrix} 0 \\ \tau_{ji} \\ \tau_{ij} v_j - q_i \end{bmatrix} \\ \mathbf{S} &= \begin{bmatrix} 0 \\ \rho b_j \\ \rho (b_i u_i + r) \end{bmatrix} \end{aligned} \quad (2)$$

where,  $\rho$ ,  $p$ ,  $v_i$ ,  $\tau_{ij}$ ,  $e$ ,  $q_i$ ,  $b_i$ , and  $r$  are the density, pressure,  $i^{\text{th}}$  component of velocity,  $ij^{\text{th}}$  component of viscous stress tensor, total energy per unit mass,  $i^{\text{th}}$  component of heat flux vector,  $i^{\text{th}}$  component of body force, and heat source, respectively.  $\delta_{ij}$  is the Kronecker delta function. The viscous stress tensor components are defined as

$$\tau_{ij} = \mu (v_{i,j} + v_{j,i}) - \frac{2}{3} \mu (v_{i,i}) \delta_{ij} \quad (3)$$

where  $\mu$  is the viscosity. The total energy per unit mass is defined as  $e = c_v T + |\mathbf{v}|^2/2$ , where  $c_v$  is the specific heat at constant volume,  $T$  is the temperature, and  $\mathbf{v} = \{v_i\}$  is velocity. The heat flux is defined as  $q_i = -\kappa T_{,i}$ , where  $\kappa$  is the thermal conductivity.

By change of variables, (1) can be converted from conservation variables  $\mathbf{U}$  into pressure primitive variables  $\mathbf{Y} = [p, \mathbf{v}, T]^T$ :

$$\mathbf{A}_0 \mathbf{Y}_{,t} + \mathbf{A}_i \mathbf{Y}_{,i} = (\mathbf{K}_{ij} \mathbf{Y}_{,j})_{,i} + \mathbf{S} \quad (4)$$

where  $\mathbf{A}_0 = \mathbf{U}_{,\mathbf{Y}}$ ;  $\mathbf{A}_i = \mathbf{F}_{i,\mathbf{Y}}$  is the  $i^{\text{th}}$  Euler Jacobian matrix;  $\mathbf{K} = [\mathbf{K}_{ij}]$  is the diffusivity matrix with  $\mathbf{K}_{ij} \mathbf{Y}_{,j} = \mathbf{E}_i$ . The above equations are supplemented with an appropriate equation of state relating pressure to other variables. Most compressible flows can be modeled with the ideal gas law,  $p = \rho RT$ , where  $R$  is the specific gas constant. For incompressible flows the density is set to a constant which implies that the isothermal compressibility and isobaric expansiveness are zero. For isothermal flows the energy equation can be excluded from the computations by setting a constant value of temperature. The equations are completed by the following boundary conditions. The boundary  $\Gamma_t$  is decomposed as  $\Gamma_t = \Gamma_n \cup \Gamma_d \cup \Gamma_f$ , where  $\Gamma_n$ ,  $\Gamma_d$ , and  $\Gamma_f$  are slip, no-slip, and free surface parts of the boundary, respectively. For  $(0, t_{tot})$  the boundary conditions read

$$\mathbf{v} \cdot \mathbf{n} = 0 \text{ and } \mathbf{t} \cdot \boldsymbol{\tau} \mathbf{n} = 0 \quad \text{on } \Gamma_n \quad (5)$$

$$\mathbf{v} = \mathbf{v}_d \quad \text{on } \Gamma_d \quad (6)$$

$$-p \mathbf{I} + \boldsymbol{\tau} \mathbf{n} = 0 \quad \text{on } \Gamma_f \quad (7)$$

where  $\mathbf{v}_d$  is the given velocity vector;  $\mathbf{t}$  and  $\mathbf{n}$  are the tangential and outward normal unit vectors on the boundary. At free surface, no restriction is imposed on the velocity and the boundary conditions are similar to those normally imposed at an outlet.

## 3. Numerical Scheme

In free surface problems, the location of the moving boundary is unknown within the fluid mechanics problem and must be determined together with the solution of flow equations. The geometry and flow field evolve in time which makes the problem time-dependent and nonlinear. We decouple the fluid flow and the mesh evolution problems and solve iteratively through a partitioned approach. For each time step, the algorithm is composed of the following steps:

- (1) Solve the Navier-Stokes equations (4) on a given domain  $\Omega(t_n)$  at time  $t_n$ .
- (2) Solve mesh deformation problem by assigning fluid velocity at the free surface.
- (3) Update the interior mesh and go to next time step.

**3.1. Solution of Navier-Stokes Equations.** The Navier-Stokes equations are solved for pressure primitive variables  $\mathbf{Y}$  which are applicable on both compressible and incompressible flows [29]. We use time-discontinuous stabilized space-time finite element method to solve the equations. The numerical technique allows using the same order interpolation functions for all solution variables. In space-time finite element method both space and time are discretized simultaneously by taking

a tensor product of basis functions for the spatial domain and one-dimensional basis function in time direction. We consider piecewise continuous approximation in space and discontinuous approximation in time. Let  $Q = \Omega_t \times (0, t_{tot})$  be the space-time domain with boundary  $P = \Gamma_t \times (0, t_{tot})$ . Let the time interval  $(0, t_{tot})$  be subdivided into an ordered series of time levels  $0 = t_0 < t_1 < \dots < t_n < t_{n+1} < \dots < t_N = t_{tot}$ . Denoting the  $n^{\text{th}}$  time interval as  $I_n = (t_n, t_{n+1})$ , a space-time slab is defined as  $Q_n = \Omega_n \times I_n$  with boundary  $P_n = \Gamma_n \times I_n$ , where  $\Omega_n$  is the spatial domain at time  $t = t_n$ . Each space-time slab is divided into  $(n_{el})_n$  elements  $Q_n^e = \Omega_n^e \times I_n$  having boundary  $P_n^e = \Gamma_n^e \times I_n$  with discontinuous basis functions across the slab boundaries in time direction. The space-time mesh is considered structured in time and the space-time slabs are spatially deformed. The mesh for the upper surface of the slab at  $t_{n+1}$  is obtained by deforming the mesh nodes of the lower surface at  $t_n$ .

The space-time mapping [30] from a physical space-time element  $Q_n^e$  to the master space-time element  $\bar{Q}_n^e = \bar{\Omega}_n^e \times I_n$  is defined as

$$\begin{aligned} \mathbf{x} &= \sum_{i=1}^2 \sum_{a=1}^{n_{el,nodes}} N_a(\boldsymbol{\xi}) T^i(\theta) \mathbf{x}_a^i \\ t &= \sum_{i=1}^2 T^i(\theta) t^i \end{aligned} \quad (8)$$

where subscript  $a$  denotes the node index and superscript  $i$  denotes the time level of a space-time element ( $i=1$  for  $t = t_n$  and  $i=2$  for  $t = t_{n+1}$ );  $\mathbf{x}$  and  $\boldsymbol{\xi}$  are the spatial parts;  $t$  and  $\theta$  are the temporal parts;  $N(\boldsymbol{\xi})$  and  $T(\theta)$  are, respectively, the bilinear spatial and linear temporal shape functions. In case of free surface problems, the position  $\mathbf{x}_a^2$  of the mesh nodes at time  $t_{n+1}$  is not known and is obtained by moving the vertices  $\mathbf{x}_a^1$  to their new position with deformation of the mesh nodes.

$$\mathbf{x}_a^2 = \mathbf{x}_a^1 + \mathbf{v}_a^m \Delta t \quad (9)$$

where  $\mathbf{v}_a^m$  is the mesh velocity of node  $a$  and  $\Delta t = t_{n+1} - t_n$ . The Jacobian matrix for the space-time mapping  $\phi$  over  $Q_n^e$  and its inverse transpose are computed as

$$\mathbf{J}_{ST} = \frac{\partial(\mathbf{x}, t)}{\partial(\boldsymbol{\xi}, \theta)} = \begin{bmatrix} \mathbf{x}_{,\boldsymbol{\xi}} & \mathbf{x}'_{,\theta} \\ t_{,\boldsymbol{\xi}} & t_{,\theta} \end{bmatrix} \quad (10)$$

$$[\mathbf{J}_{ST}^{-1}]' = \begin{bmatrix} [\mathbf{x}_{,\boldsymbol{\xi}}^{-1}]' & \theta'_{,\mathbf{x}} \\ \boldsymbol{\xi}_{,t} & \theta_{,t} \end{bmatrix} = \begin{bmatrix} [\mathbf{x}_{,\boldsymbol{\xi}}^{-1}]' & \mathbf{0}' \\ -\mathbf{v}_m [\mathbf{x}_{,\boldsymbol{\xi}}^{-1}]' & \frac{2}{\Delta t} \end{bmatrix} \quad (11)$$

The derivatives of the shape functions in physical and reference coordinate systems satisfy the following relationship:

$$\begin{bmatrix} (N_a T_i)_{,\mathbf{x}} \\ (N_a T_i)_{,t} \end{bmatrix} = [\mathbf{J}_{ST}^{-1}]' \begin{bmatrix} N_{a,\boldsymbol{\xi}} T_i \\ N_a T_{i,\theta} \end{bmatrix} \quad (12)$$

The fluid mesh velocity  $\mathbf{v}_m$  determines the working frame of reference. If it is zero then we are in Eulerian frame; if it is equal to the fluid particle velocity then we are in Lagrangian

frame; if it is equal to the fluid velocity for the moving boundaries and arbitrary for the internal domain then we recover the ALE frame.

*Remark 1.* The space-time mapping is similar to the ALE mapping except for the temporal derivative terms. In ALE method the time is kept continuous in course of the spatial discretization; therefore the space-time Jacobian coincides with the spatial Jacobian. However, in space-time FEM the time is discretized simultaneously with space which results in a product of spatial Jacobian and a time difference term.

The space-time finite element weighted residual formulation of (4) is written as follows [28]: given a trial function space  $\mathcal{F}_n^h$  and weighting function space  $\mathcal{W}_n^h$ , within each  $Q_n$ ,  $n = 0, 1, \dots, N-1$ , find  $\mathbf{Y}^h \in \mathcal{F}_n^h$  such that  $\forall \mathbf{W}^h \in \mathcal{W}_n^h$  the following relation is satisfied

$$\begin{aligned} & \int_{Q_n} (-\mathbf{W}_{,t}^h \cdot \mathbf{U}(\mathbf{Y}^h) - \mathbf{W}_{,i}^h \cdot \mathbf{F}_i(\mathbf{Y}^h) + \mathbf{W}_{,i}^h \cdot \mathbf{E}_i(\mathbf{Y}^h) \\ & - \mathbf{W}^h \cdot \mathbf{S}) dQ_n + \int_{\Omega(t_{n+1}^-)} \mathbf{W}^h(t_{n+1}^-) \\ & \cdot \mathbf{U}(\mathbf{Y}^h(t_{n+1}^-)) d\Omega - \int_{\Omega(t_n^+)} \mathbf{W}^h(t_n^+) \\ & \cdot \mathbf{U}(\mathbf{Y}^h(t_n^+)) d\Omega + \sum_{e=1}^{(n_{el})_n} \int_{Q_n^e} \mathcal{L}' \mathbf{W}^h \cdot \boldsymbol{\tau}(\mathcal{L} \mathbf{Y}^h \\ & - \mathbf{S}) dQ_n - \int_{P_n} \mathbf{W}^h \cdot \mathbf{U}(\mathbf{Y}^h) \hat{\mathbf{n}} \cdot \mathbf{v}_m dP_n \\ & - \int_{P_n} \mathbf{W}^h \cdot (\mathbf{E}_i(\mathbf{Y}^h) - \mathbf{F}_i(\mathbf{Y}^h)) \hat{\mathbf{n}} dP_n = 0 \end{aligned} \quad (13)$$

In the above formulation, the first and last two integrals are the Galerkin terms obtained from integration by parts. The second and third integrals are the jump terms to enforce continuity of the solution between consecutive slabs. The jump terms facilitate the projection of the solution in case of remeshing. The fourth integral is the least-square stabilization term with differential operator  $\mathcal{L} = \mathbf{A}_0(\partial/\partial t) + \mathbf{A}_i(\partial/\partial x_i)$  and  $\mathcal{L}' = \mathbf{A}'_0(\partial/\partial t) + \mathbf{A}'_i(\partial/\partial x_i)$ .  $\mathcal{L} \mathbf{Y}^h - \mathbf{S}$  is the residual of the equations and  $\boldsymbol{\tau}$  is the stabilization matrix for pressure primitive variables  $\mathbf{Y}^h$  [29, 31].

The space-time discretization can be done as either constant in time or linear in time [28]. In constant in time, the interpolation functions are considered piecewise linear in space and constant in time, while in linear-in-time interpolation functions are linear both in space and in time. For linear-in-time functions within the  $n^{\text{th}}$  space-time slab  $Q_n$ , the finite element trial solution  $\mathbf{Y}^h$  and the weighting function  $\mathbf{W}^h$  are defined as

$$\mathbf{Y}^h = \sum_{a=1}^{(n_{np})_n} N_a T_2 \mathbf{Y}_a^2 + N_a T_1 \mathbf{Y}_a^1 \quad (14)$$

$$\mathbf{W}^h = \sum_{a=1}^{(n_{np})_n} N_a T_2 \mathbf{W}_a^2 + N_a T_1 \mathbf{W}_a^1 \quad (15)$$

where  $\mathbf{Y}_a^2$  and  $\mathbf{Y}_a^1$  are the vectors of unknowns at node  $a$  at times  $t_{n+1}^-$  and  $t_n^+$ , respectively. Analogously,  $\mathbf{W}_a^2$  and  $\mathbf{W}_a^1$  are the vectors of nodal weighting function at times  $t_{n+1}^-$  and  $t_n^+$ , respectively.  $(n_{np})_{(n)}$  is the number of nodal points in  $n$ th space-time slab. The substitution of trial and weighting functions in (13) results in two systems of nonlinear equations which are solved by implicit third-order predictor-corrector method [28].

*Remark 2.* The computation cost of the algorithm can be reduced by implementing the constant-in-time discretization approximation which is first-order time accurate and consists of a linear system having half number of unknowns in comparison to linear-in-time approximation. For constant-in-time the values of  $t_{,\beta}$  should be set to  $\Delta t$  in (10).

**3.2. Mesh Update.** In free surface problems, the location of the free surface is calculated from the fluid velocity  $\mathbf{v}$  at each time step of the computation. Once the displacement of the moving surface is known, the internal nodes can be moved to reduce the distortion of the mesh elements. In order to evaluate the new location of the interior mesh nodes, a mesh deformation technique is used. Mesh deformation can be handled arbitrarily by matching the normal velocity of the mesh with the normal velocity of the moving interface. This requires the computation of a unit normal vector at each node of the moving surface. Since the finite element representation is only  $\mathbb{C}_0$  continuous there is no unique normal at a vertex. Often, a separate normal can be defined for each element that meets at that point and an arithmetic average of the surrounding normals is taken as the normal vector at the mesh node [32]. To avoid this problem as an alternative we match both normal and tangential components of the velocity at the free surface. For mesh deformation, several methods have been proposed in the literature such as Laplacian [22], biharmonic [23], spring [33], transfinite interpolation [34], and the elastic deformation methods [35, 36]. We use the elastic deformation method which has been effectively used in [19, 24]. In elastic deformation method, the computational grid is considered as an elastic body and is deformed subject to applied boundary conditions. For  $n^{\text{th}}$  space-time slab, the partial differential equations governing the displacement of the mesh are

$$\nabla \cdot \boldsymbol{\sigma} = 0 \quad \text{on } \Omega(t_n) \quad (16)$$

where  $\boldsymbol{\sigma}$  is the Cauchy stress tensor which is dependent upon the strain tensor  $\boldsymbol{\epsilon}$  by the Hooke's law for isotropic homogeneous medium.

$$\begin{aligned} \boldsymbol{\sigma} &= \lambda \operatorname{tr}(\boldsymbol{\epsilon}) \mathbf{I} + 2\mu \boldsymbol{\epsilon} \\ \boldsymbol{\epsilon} &= \frac{\nabla \mathbf{u}_m + \nabla \mathbf{u}_m^T}{2} \end{aligned} \quad (17)$$

where  $\mathbf{u}_m$  is the mesh displacement in  $n^{\text{th}}$  space-time slab;  $\lambda$  and  $\mu$  are the Lamé coefficients which can be expressed in terms of modulus of elasticity  $E$  and Poisson ratio  $\nu$ :

$$\lambda = \frac{\nu E}{(1 + \nu)(1 - 2\nu)}$$

$$\mu = \frac{E}{2(1 + \nu)} \quad (18)$$

For  $\nu \rightarrow 0.5$  the material is nearly incompressible and the elastic problem becomes ill-conditioned. For incompressible materials a neo-Hookean elasticity model can be used [37]. The mesh equation (16) is solved by applying the following boundary conditions:

$$\mathbf{u}_m = 0 \quad \text{on } \Gamma_d \quad (19)$$

$$\mathbf{u}_m \cdot \hat{\mathbf{n}} = 0 \quad \text{on } \Gamma_n \quad (20)$$

$$\mathbf{u}_m = \mathbf{v} \Delta t \quad \text{on } \Gamma_f \quad (21)$$

where  $\mathbf{v}$  is the known fluid velocity. In order to avoid excessive distortion and to maintain a better mesh quality near the free surface, the Jacobian based stiffening approach is employed [24, 36].

## 4. Numerical Examples

In order to demonstrate the effectiveness of the model and to validate the computational code, we present three numerical examples: solitary wave propagation, the collision of two waves, and wave damping. The first two examples are for inviscid flow and the third one is for viscous flow. In order to compute the inviscid flow the fluid viscosity is set to zero in (2). In case of no heat exchange this results in a zero diffusion flux vector (i.e.,  $\mathbf{E}_i = 0$ ). For all simulations, the spatial meshes are made of quadrilateral elements and the time slabs are formed simply by moving the spatial meshes in time. The mesh deformation is computed with the elastic deformation method (16) by setting Young's modulus  $E = 1000$  Pa and the Poisson ratio  $\nu = 0.35$ .

**4.1. Solitary Wave Propagation.** This numerical problem is a classical check for the free surface flow models and has been studied in many works [16, 21, 38, 39]. The problem consists in the motion of an inviscid fluid in a tank with three fixed walls (bottom and both vertical walls). The upper surface is free to deform with gravitational force controlling the motion of the wave. The model problem is presented in Figure 1. The initial profile of the wave at the free surface  $\eta$  is given by

$$\eta = d + H \operatorname{sech}^2 \left( \sqrt{\frac{3H}{4d^3}} x \right) \quad (22)$$

The domain parameters are  $d = 10$  m and  $H = 2$  m. The simulation is performed over a structured mesh made of 6400 quadrangle elements. Fluid density is set to 1000 kg/m<sup>3</sup>. The gravitational acceleration is set to 9.81 m/s<sup>2</sup>. The initial conditions for this problem are set according to the



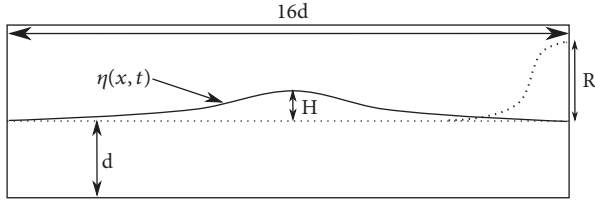
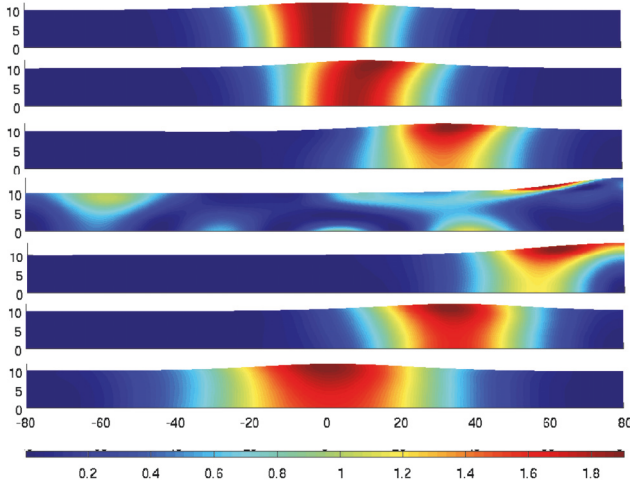


FIGURE 1: Wave propagation: problem description.


 FIGURE 2: Wave propagation: absolute velocity (m/s) from top to bottom at  $t = 0, 1, 3, 7.6, 9, 12,$  and  $15$  s.

approximations given by Laitone [40] for an infinite domain. The initial velocity is set as

$$v_1 = \sqrt{gd} \left( \frac{H}{d} \right) \operatorname{sech}^2 \left( \sqrt{\frac{3H}{4d^3}} x \right) \quad (23)$$

$v_2$

$$= \sqrt{\frac{3g}{d}} \left( \frac{H}{d} \right)^{3/2} y \operatorname{sech}^2 \left( \sqrt{\frac{3H}{4d^3}} x \right) \tanh \left( \sqrt{\frac{3H}{4d^3}} x \right) \quad (24)$$

At the free surface, the boundary condition (7) is imposed. Both top corner points belong to the free surface. The fluid mesh nodes on the vertical walls of the container move in response to the motion of the top corner points. The time step is set to  $10^{-3}$  s.

Figure 2 shows the velocity magnitude at various times. The wave reaches the right vertical wall at time  $t = 7.6$  s. The simulation is carried out until the water wave returns to its initial position at  $x = 0$ . The reflected wave takes approximately the same time to travel back to its initial position. The total run-up height  $d + R_{max}$  against the right wall is 14.28 m, where  $R_{max}$  is the maximum run-up height and can be calculated by the following relation [41]:

$$\frac{R_{max}}{d} = 2 \left( \frac{H}{d} \right) + \frac{1}{2} \left( \frac{H}{d} \right)^2 + O \left( \left( \frac{H}{d} \right)^3 \right) \quad (25)$$

The numerical results are in close agreement with the analytical one proposed in [41]. The total run-up height at the

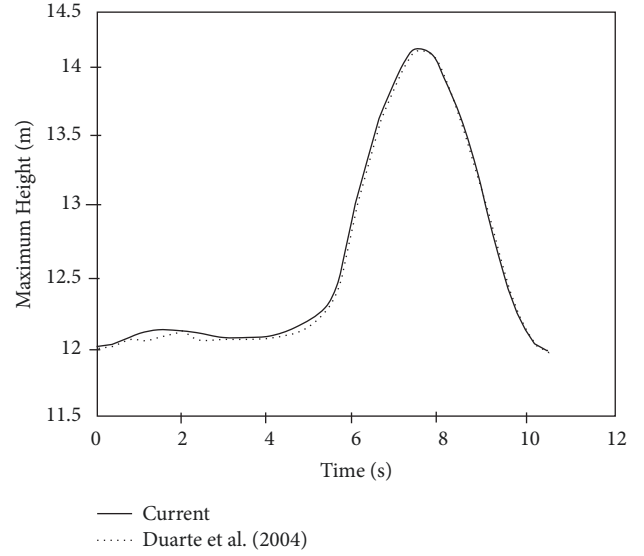


FIGURE 3: Wave propagation: comparison of numerical results with [16].

right wall computed by our model is compared with [16] in Figure 3. The spatial and temporal convergence has been studied by considering two grids and several time step sizes. Table 1 lists the  $L_2$  norm error results.

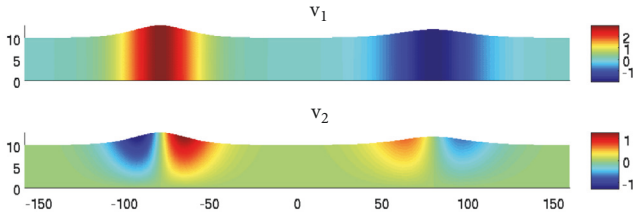
**4.2. Collision between Two Waves.** This problem investigates a head-on collision between two solitary waves and was studied by Su and Mirie [42] for an inviscid, incompressible, homogeneous fluid by a perturbation method up to the third order of approximation. The problem was reformulated in [43] for an overtaking collision between two solitary waves and the results on phase shifts and the time history of interaction were verified with the theoretical solution of the Korteweg-de Vries equation. In [44] the problem was numerically investigated by using a velocity-vorticity based Arbitrary Lagrangian-Eulerian method. We simulate the collision of two solitary waves with two different initial amplitudes: 3 m on the left-hand side and 2 m on the right-hand side. The density and gravity are set to  $1000 \text{ kg/m}^3$  and  $9.81 \text{ m/s}^2$ , respectively. The length of the rectangular domain is  $32d$  with  $d = 10$  m. Initial velocities are set using Laitone's relations (23) and (24). The initial profiles for the velocity are shown in Figure 4. The computational grid is made of 10000 elements. The boundary conditions for the fluid flow and the mesh problem are same as those for the wave damping of viscous fluid problem. The time step is  $10^{-3}$  s.

Figure 5 shows the space-time characteristic curves for free surface elevation. The maximum amplitude (run-up) during the collision can be evaluated as [43]

$$\begin{aligned} \text{Maximum run-up} &= A_L + B_R + \frac{1}{2} A_L B_R \\ &+ \frac{1}{4} A_L B_R (A_L + B_R) \end{aligned} \quad (26)$$

TABLE 1:  $L_2$  norm solution error.

Grid	Time step	$L_2$ error
$240 \times 15$	0.001	1.3583e-3
$240 \times 15$	0.005	9.7483e-2
$240 \times 15$	0.01	7.9872e-1
$320 \times 20$	0.001	5.8368e-4
$320 \times 20$	0.005	2.9428e-2
$320 \times 20$	0.01	3.8183e-1

FIGURE 4: Wave collision: the initial profiles for horizontal ( $v_1$ ) and vertical velocity ( $v_2$ ) (m/s).

where  $A_L$ ,  $B_R$  are the initial amplitudes of the solitary waves on the left and the right-hand sides, respectively. The simulation results give a maximum run-up height of 15.491 m at time  $t = 7.5$  s, which is close to the calculated value of 15.500 m by (26).

**4.3. Wave Damping of a Viscous Fluid.** This problem consists in solving the small-amplitude motion of the free surface of a viscous fluid in a rectangular tank. The initial profile of the free surface is given by

$$h(x) = 1.5 + a_0 \sin(\pi(0.5 - x)) \quad (27)$$

where  $a_0$  is the amplitude of the initial sinusoidal perturbation of the movement. The problem has been investigated in [45, 46]. The simulation domain for the problem is shown in Figure 6. The initial perturbation  $a_0$  is set to 0.01 m. The dynamic viscosity and the density of the fluid are set to 0.01 Pa·s and 1 kg/m<sup>3</sup>, respectively. The gravitational acceleration is set to unity. The computational mesh is composed of  $50 \times 50$  quadrangle elements. The initial conditions for the fluid are zero velocity and a hydrostatic pressure profile. The fluid is allowed to slip at all walls of the container. The time step is set to  $10^{-3}$  s.

Under the effect of gravity, the left part of fluid tends to move downwards and enforces the right part of the fluid to move upwards. The first phase of this movement lasts for the first 1.7 s and the fluid reaches its minimum amplitude of 1.4992 m at the left wall. After that, the motion is reversed and continues periodically. The flow is dominated by the viscous forces which are responsible for the damping of the movement. For small-amplitude waves, the initial-value problem has been solved analytically in [47]. For free surface

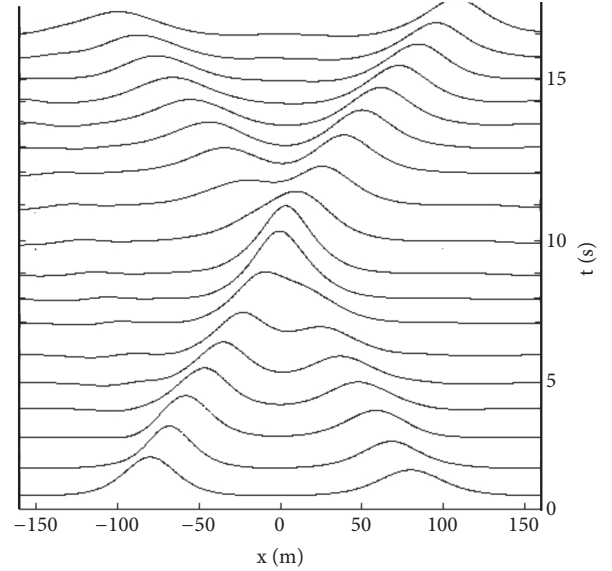


FIGURE 5: Wave collision: space-time characteristic curves for free surface elevation.

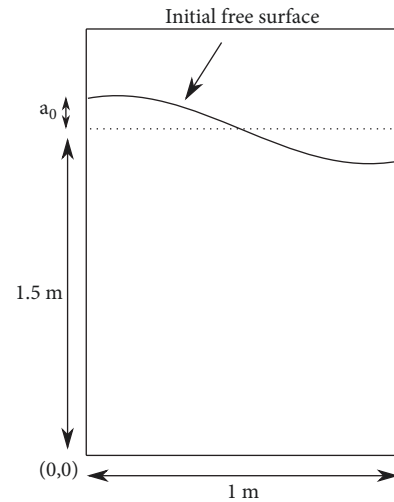


FIGURE 6: Viscous sloshing: schematic description of the problem.

$y = a(t)g(x, y)$  with  $g$  satisfying  $(\partial_x^2 + \partial_y^2 + k^2)g = 0$ , the solution for one fluid is given by

$$a(t) = \frac{4\nu^2 k^4 a_0 f(\sqrt{\nu k^2 t})}{8\nu^2 k^4 + \omega_0^2} + \sum_{i=1}^4 \frac{z_i}{Z_i} \left( \frac{\omega_0^2 a_0}{z_i^2 - \nu k^2} \right) e^{(z_i^2 - \nu k^2)t} f(z_i \sqrt{t}) \quad (28)$$

where  $\nu$  is the kinematic viscosity of the fluid;  $k$  is the wave number;  $\omega_0^2 = gk$  is the inviscid natural frequency; each  $z_i$  is the root of the following algebraic equation:

$$z^4 + 2k^2 \nu z^2 + 4\sqrt{k^6 \nu^3} z + \nu^2 k^4 + \omega_0^2 = 0 \quad (29)$$

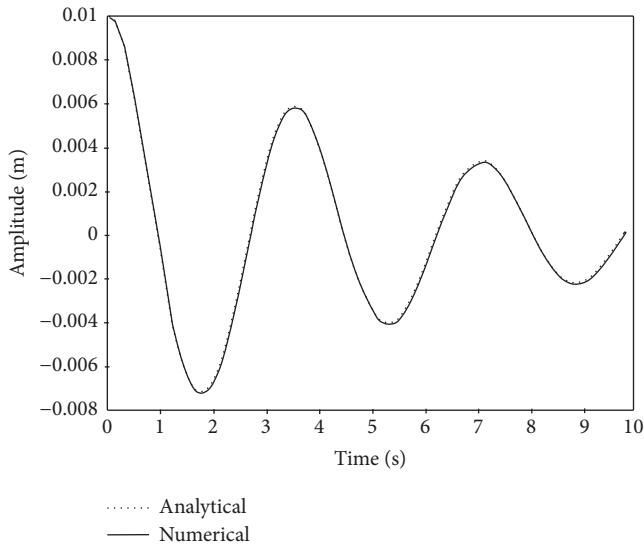


FIGURE 7: Viscous sloshing: time evolution of the amplitude at the top left corner point.

and  $Z_1 = (z_2 - z_1)(z_3 - z_1)(z_4 - z_1)$  with  $Z_2, Z_3, Z_4$  obtained by circular permutation of the indices and  $f(\cdot)$  is the complementary error function defined as

$$f(x) = \frac{2}{\sqrt{\pi}} \int_x^{\infty} e^{-t^2} dt \quad (30)$$

Figure 7 compares the simulation results with the analytical solution [47] for the amplitude of the top left node over time. Numerical results match remarkably well with the analytic solution.

## 5. Conclusions

Our objective in this work was to develop a numerical scheme for simulating viscous and inviscid free surface flows for the numerical study of the wave tank. We used a partitioned algorithm to iteratively solve the fluid flow and the mesh deformation problems. The numerical technique introduced mesh velocity in an implicit manner and avoided the need to write the flow equations in the ALE form a priori. We presented the detailed formulation of the numerical scheme covering space-time mapping and discretization strategies. We successfully validated our numerical code on one viscous and two inviscid flow test cases. For the simulations, our choice of initial meshes preserved a good quality for the entire simulation times and allowed avoiding remeshing for any arbitrary time interval. The accuracy of the numerical results leads to concluding that the numerical approach is highly efficient and stable for a variety of free flow problems.

The numerical scheme is clearly not applicable for free surface flows involving the wave breaking and the fluid separation. In such cases, free surface undergoes a large deformation and a portion of the fluid split from the main body leading to the breakdown of the numerical scheme. Such problems can be modeled with multicomponent/multiphase flows. Future developments will concern the extension of

the developed algorithm to multicomponent flows and fully three-dimensional cases which are relevant and crucial for the natural risk problems.

## Data Availability

The numerical simulations data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

The research leading to these results has received funding from the People Programme (Marie Curie Actions) of the European Union's Seventh Framework Programme (FP7/2007-2013) under the project NEMOH, REA, Grant Agreement no. 289976.

## References

- [1] B. R. Hodges and R. L. Street, "On Simulation of Turbulent Non-linear Free-Surface Flows," *Journal of Computational Physics*, vol. 151, no. 2, pp. 425–457, 1999.
- [2] A. Valiani, V. Caleffi, and A. Zanni, "Case study: Malpasset dam-break simulation using a two-dimensional finite volume method," *Journal of Hydraulic Engineering*, vol. 128, no. 5, pp. 460–472, 2002.
- [3] G. X. Wu, Q. W. Ma, and R. Eatock Taylor, "Numerical simulation of sloshing waves in a 3D tank based on a finite element method," *Applied Ocean Research*, vol. 20, no. 6, pp. 337–355, 1998.
- [4] N. Ba Thuy, K. Tanimoto, N. Tanaka, K. Harada, and K. Iimura, "Effect of open gap in coastal forest on tsunami run-up—investigations by experiment and numerical simulation," *Ocean Engineering*, vol. 36, no. 15-16, pp. 1258–1269, 2009.
- [5] S. Tinti, E. Bortolucci, and A. Armigliato, "Numerical simulation of the landslide-induced tsunamis of 1988 on Vulcano Island, Italy," *Bulletin of Volcanology*, vol. 61, no. 1-2, pp. 121–137, 1999.
- [6] Y. C. Chang, T. Y. Hou, B. Merriman, and S. Osher, "A level set formulation of Eulerian interface capturing methods for incompressible fluid flows," *Journal of Computational Physics*, vol. 124, no. 2, pp. 449–464, 1996.
- [7] J. A. Sethian and P. Smereka, "Level set methods for fluid interfaces," in *Annual review of fluid mechanics*, Vol. 35, vol. 35 of *Annu. Rev. Fluid Mech.*, pp. 341–372, Annual Reviews, Palo Alto, CA, 2003.
- [8] C. W. Hirt and B. D. Nichols, "Volume of fluid (VOF) method for the dynamics of free boundaries," *Journal of Computational Physics*, vol. 39, no. 1, pp. 201–225, 1981.
- [9] R. Scardovelli and S. Zaleski, "Interface reconstruction with least-square fit and split Eulerian-Lagrangian advection," *International Journal for Numerical Methods in Fluids*, vol. 41, no. 3, pp. 251–274, 2003.

- [10] D. J. Torres and J. U. Brackbill, "The Point-Set Method: Front-Tracking without Connectivity," *Journal of Computational Physics*, vol. 165, no. 2, pp. 620–644, 2000.
- [11] F. H. Harlow and J. E. Welch, "Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface," *Physics of Fluids*, vol. 8, no. 12, pp. 2182–2189, 1965.
- [12] T. Nakayama and M. Mori, "An Eulerian finite element method for time-dependent free surface problems in hydrodynamics," *International Journal for Numerical Methods in Fluids*, vol. 22, no. 3, pp. 175–194, 1996.
- [13] M.-C. Lai and Z. Li, "A remark on jump conditions for the three-dimensional Navier-Stokes equations involving an immersed moving membrane," *Applied Mathematics Letters*, vol. 14, no. 2, pp. 149–154, 2001.
- [14] M. B. Liu and G. R. Liu, "Smoothed particle hydrodynamics (SPH): an overview and recent developments," *Archives of Computational Methods in Engineering: State-of-the-Art Reviews*, vol. 17, no. 1, pp. 25–76, 2010.
- [15] Q. W. Ma and S. Yan, "Quasi ALE finite element method for nonlinear water waves," *Journal of Computational Physics*, vol. 212, no. 1, pp. 52–72, 2006.
- [16] F. Duarte, R. Gormaz, and S. Natesan, "Arbitrary Lagrangian-Eulerian method for Navier-Stokes equations with moving boundaries," *Computer Methods Applied Mechanics and Engineering*, vol. 193, no. 45–47, pp. 4819–4836, 2004.
- [17] J. Li, M. Hesse, J. Ziegler, and A. W. Woods, "An arbitrary Lagrangian Eulerian method for moving-boundary problems and its application to jumping over water," *Journal of Computational Physics*, vol. 208, no. 1, pp. 289–314, 2005.
- [18] E. Bänsch, J. Paul, and A. Schmidt, "An ALE finite element method for a coupled Stefan problem and Navier-Stokes equations with free capillary surface, Computers and Structures," *International Journal for Numerical Methods in Fluids*, vol. 71, no. 10, pp. 1282–1296, 2013.
- [19] T. E. Tezduyar, M. Behr, and J. Liou, "A new strategy for finite element computations involving moving boundaries and interfaces—the deforming-spatial-domain/space-time procedure. I. The concept and the preliminary numerical tests," *Computer Methods Applied Mechanics and Engineering*, vol. 94, no. 3, pp. 339–351, 1992.
- [20] T. E. Tezduyar, M. Behr, S. Mittal, and J. Liou, "A new strategy for finite element computations involving moving boundaries and interfaces—the deforming-spatial-domain/space-time procedure. II. Computation of free-surface flows, two-liquid flows, and flows with drifting cylinders," *Computer Methods Applied Mechanics and Engineering*, vol. 94, no. 3, pp. 353–371, 1992.
- [21] A. Masud and T. J. Hughes, "A space-time Galerkin/least-squares finite element formulation of the Navier-Stokes equations for moving domain problems," *Computer Methods Applied Mechanics and Engineering*, vol. 146, no. 1–2, pp. 91–126, 1997.
- [22] R. Löhner and C. Yang, "Improved ale mesh velocities for moving boundaries," *Communications in Numerical Methods in Engineering*, vol. 12, pp. 599–608, 1996.
- [23] B. T. Helenbrook, "Mesh deformation using the biharmonic operator," *International Journal for Numerical Methods in Engineering*, vol. 56, no. 7, pp. 1007–1021, 2003.
- [24] A. A. Johnson and T. E. Tezduyar, "Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and interfaces," *Computer Methods Applied Mechanics and Engineering*, vol. 119, no. 1–2, pp. 73–94, 1994.
- [25] J. Tang, Y. Shen, D. M. Causon, L. Qian, and C. G. Mingham, "Numerical study of periodic long wave run-up on a rigid vegetation sloping beach," *Coastal Engineering Journal*, vol. 121, pp. 158–166, 2017.
- [26] D. D. Prasad, M. R. Ahmed, Y.-H. Lee, and R. N. Sharma, "Validation of a piston type wave-maker using Numerical Wave Tank," *Ocean Engineering*, vol. 131, pp. 57–67, 2017.
- [27] Z. Liu, J. Jin, B. Hyun, and K. Hong, "Review of Application of VOF-Based NWT on Integrated OWC System," *Journal of the Korean Society for Marine Environment & Energy*, vol. 15, no. 2, pp. 111–117, 2012.
- [28] F. Shakib, T. J. Hughes, and Z. e. Johan, "A new finite element formulation for computational fluid dynamics. X. The compressible Euler and Navier-Stokes equations," *Computer Methods Applied Mechanics and Engineering*, vol. 89, no. 1–3, pp. 141–219, 1991.
- [29] G. Hauke and T. J. Hughes, "A comparative study of different sets of variables for solving compressible and incompressible flows," *Computer Methods Applied Mechanics and Engineering*, vol. 153, no. 1–2, pp. 1–44, 1998.
- [30] Y. Bazilevs, K. Takizawa, and T. E. Tezduyar, *Computational Fluid-Structure Interaction*, John Wiley & Sons, Ltd, Chichester, UK, 2013.
- [31] G. Hauke, "Simple stabilizing matrices for the computation of compressible flows in primitive variables," *Computer Methods Applied Mechanics and Engineering*, vol. 190, no. 51–52, pp. 6881–6893, 2001.
- [32] M. A. Walkley, P. H. Gaskell, P. K. Jimack, M. A. Kelmanson, J. L. Summers, and M. C. T. Wilson, "On the calculation of normals in free-surface flow problems," *Communications in Numerical Methods in Engineering*, vol. 20, no. 5, pp. 343–351, 2004.
- [33] J. T. Batina, "Unsteady Euler algorithm with unstructured dynamic mesh for complex-aircraft aerodynamic analysis," *AIAA Journal*, vol. 29, no. 3, pp. 327–333, 1991.
- [34] G. A. Davis and O. O. Bendiksen, "Unsteady transonic two-dimensional euler solutions using finite elements," *AIAA Journal*, vol. 31, no. 6, pp. 1051–1059, 1993.
- [35] D. R. Lynch, "Unified approach to simulation on deforming elements with application to phase change problems," *Journal of Computational Physics*, vol. 47, no. 3, pp. 387–411, 1982.
- [36] T. E. Tezduyar, M. Behr, S. Mittal, and A. A. Johnson, "Computation of unsteady incompressible flows with the stabilized finite element methods-space-time formulations, iterative strategies and massively parallel implementations," *New Methods in Transient Analysis*, vol. 143, pp. 7–24, 1992.
- [37] M. Souli, "Arbitrary lagrangian-eulerian and free surface methods in fluid mechanics," *CMAME*, vol. 191, pp. 451–466, 2001.
- [38] O. C. Zienkiewicz, R. L. Taylor, and P. Nithiarasu, *The finite element method for fluid dynamics*, ELSEVIER, 6th edition, 2006.
- [39] S. M. Zandi, B. Boroomand, and S. Soghrati, "Exponential basis functions in solution of incompressible fluid problems with moving free surfaces," *Journal of Computational Physics*, vol. 231, no. 2, pp. 505–527, 2012.
- [40] E. V. Laitone, "The second approximation to cnoidal and solitary waves," *Journal of Fluid Mechanics*, vol. 9, pp. 430–444, 1960.
- [41] J. G. Byatt-Smith, "An integral equation for unsteady surface waves and a comment on the Boussinesq equation," *Journal of Fluid Mechanics*, vol. 49, pp. 625–633, 1971.
- [42] C. H. Su and R. M. Mirie, "On head-on collisions between solitary waves," *Journal of Fluid Mechanics*, vol. 98, no. 3, pp. 509–525, 1980.



- [43] R. M. Mirie and C. H. Su, "Collisions between two solitary waves. Part 2. A numerical study," *Journal of Fluid Mechanics*, vol. 115, pp. 475–492, 1982.
- [44] D. C. Lo and D. L. Young, "Arbitrary Lagrangian-Eulerian finite element analysis of free surface flow using a velocity-vorticity formulation," *Journal of Computational Physics*, vol. 195, no. 1, pp. 175–201, 2004.
- [45] S. Popinet and S. Zaleski, "A front-tracking algorithm for accurate representation of surface tension," *International Journal for Numerical Methods in Fluids*, vol. 30, no. 6, pp. 775–793, 1999.
- [46] S. Rabier and M. Medale, "Computation of free surface flows with a projection FEM in a moving mesh framework," *Computer Methods Applied Mechanics and Engineering*, vol. 192, no. 41-42, pp. 4703–4721, 2003.
- [47] A. Prosperetti, "Motion of two superposed viscous fluids," *Physics of Fluids*, vol. 24, no. 7, pp. 1217–1223, 1981.



**Hindawi**

Submit your manuscripts at  
[www.hindawi.com](http://www.hindawi.com)

