



Multi-fluid flows with the Particle Finite Element Method

Sergio Idelsohn ^{*,1}, Monica Mier-Torrecilla, Eugenio Oñate

International Center for Numerical Methods in Engineering (CIMNE), Gran Capitan s/n, Campus Nord UPC, Barcelona, Spain

ARTICLE INFO

Article history:

Received 20 January 2009

Received in revised form 27 March 2009

Accepted 3 April 2009

Available online 18 April 2009

Keywords:

Particle method

Finite elements

Heterogeneous fluids

Multi-fluids

Lagrange formulations

Multiphase flows

Incompressible Navier–Stokes equations

Free-surfaces

Interfaces

ABSTRACT

Particle methods are those in which the problem is represented by a discrete number of particles. Each particle moves accordingly with its own mass and the external/internal forces applied on it. In this paper the Particle Finite Element Method based on finite element shape functions is used to solve the continuous fluid mechanics equations in the case of heterogeneous density. To evaluate the external applied forces to each particle, the incompressible Navier–Stokes equations are solved at each time step using a Lagrangian formulation. All the information in the fluid is transmitted via the particles. All kinds of density heterogeneous fluids and multiphase flows with internal interfaces including or not free-surfaces, breaking waves and fluid separations may be easily solved with this methodology.

© 2009 Elsevier B.V. All rights reserved.

1. Introduction

Over the last 20 years, computer simulation of incompressible fluid flow has been based on the Eulerian formulation of the fluid mechanics equations on continuous domains [1]. However, it is still difficult to analyze problems in which the shape of the free-surfaces or internal interfaces changes continuously or in fluid–structure interactions where complicated contact problems are involved.

More recently, particle methods in which each fluid particle is followed in a Lagrangian manner have been used [2–5]. The first ideas on this approach were proposed by Monaghan [2] for the treatment of astrophysical hydrodynamic problems with the so called Smooth Particle Hydrodynamics Method (SPH).

This method was later generalized to fluid mechanic problems [2–5]. Kernel approximations are used in the SPH method to interpolate the unknowns. More particle methods have been developed based on similar ideas and applied to multiphase flows [6–8, and references therein].

SPH belongs to a family of methods called meshless methods. All these methods use the idea of a polynomial interpolant that fits a number of points minimizing the distance between the interpolated function and the value of the unknown point. These ideas were proposed first by Nayroles et al. [9], later used in structural

mechanics by Belytschko et al. [10] and in fluid mechanics problems by Oñate et al. [11–13]. Lately, the meshless ideas were generalized to take into account the finite element type approximations in order to obtain the same computing time in mesh generation as in the evaluation of the meshless connectivities [14]. This method was called the Meshless Finite Element Method (MFEM) and uses the Extended Delaunay Tessellation [15] to build the mesh in a computing time which is linear with the number of nodal points.

It must be noted that particle methods may be used with both: mesh or meshless shape functions. The only practical limitation is that the connectivity in meshless methods or the mesh generation in methods with mesh need to be evaluated at each time step. For these reason the evaluation of the connectivity must not consume much computing time.

The Particle Finite Element Method (PFEM) [16,17] combines the particle precept with the finite element shape functions using an auxiliary finite element mesh that is quickly built at each time step. PFEM has been successfully used to solve the Navier–Stokes equations [16–20] and fluid–structure interactions problems [21–23]. In this paper the advantages of PFEM for the tracking of internal interfaces will be explored and used to solve fluid mechanics problems including immiscible heterogeneous flows. Although in [24] we have solved already some multi-fluids problems using PFEM, there the density jump was small. In this work we have developed a special stabilization technique and interface definition that allow arbitrary jumps in density.

* Corresponding author. Tel.: +34 93 4011829; fax: +34 93 401 6517.

E-mail address: sergio@cimne.upc.edu (S. Idelsohn).

¹ ICREA Research Professor.

Heterogeneous fluid flows occur frequently in nature and engineering practice. Nevertheless, they still pose a major research challenge from both, theoretical and computational points of view. Contrary to heterogeneous solid materials where the main cause of variation in the physical properties are the Young modulus or the anisotropy of the different materials involved, in fluids the main reasons for the heterogeneity are the variations of the density and the viscosity.

Density variations for incompressible flows may be due to a change in the temperature field or due to the presence of several immiscible flows with different densities. For the first case, most works in the field have used the Boussinesq approximation [25] in which momentum effects due to density variation are ignored and the temperature dependence on the density is taken into account via a linearized force term as a buoyancy force. When the variation of the density with temperature is important, Boussinesq approximation may not be applied. The second case, that is the presence of two or more immiscible flows, represents a bigger challenge from the numerical point of view because now there is a real jump in the physical properties. Furthermore the most important difference between solids and fluids is perhaps, the large deformation and mobility in a fluid compared with a solid. In presence of immiscible flows, this large mobility produces a strong mixing of every fluid with the others leading to multiple interfaces. Typically the different fluids start totally unmixed and they become totally mixed after a few time steps (Fig. 1.1). Until now the only way to treat this kind of mixing fluids is via homogenization procedures [26]. Nevertheless, a homogenization technique is possible when there is a relative homogeneous distribution of all the fluids involved (Fig. 1.1b), but it cannot be used when the fluid starts to separate (Fig. 1.1a).

In the method described in this work the Boussinesq approximation is introduced in all terms of the momentum equations: the sources and the inertial ones. Furthermore, the method allows solving the problem of mixing different fluids without any homogenization approach.

Viscous variations of the fluid can be again due to variation of the temperature. Jumps in the viscosity value between two or more fluids may be also involved but they are not so important in general. In this paper only heterogeneity due to changes or jumps in the density will be considered.

Computing the interface between two immiscible fluids or the free-surfaces is difficult because neither the shape nor the position of the domains between the fluids are known a priori. There are basically two approaches for computing interfaces in this kind of flows: interface-tracking and interface-capturing methods. The former computes the motion of the flow particles via a Lagrangian approach, where the numerical domain adapts itself to the shape and position of the interfaces. Standard interface-capturing methods consider both fluids as a single effective fluid with variable properties [27–29]. The interfaces are considered as a region of sudden change in the fluid properties. This approach requires an accurate modelling of the jump in the properties of the two fluids taking into account that the interfaces can move, bend and reconnect in

arbitrary ways. Furthermore, prescribing exact boundary conditions in the interface is usually approximated.

The PFEM belongs to the category of interface-tracking methods. At each time step a new mesh is generated that tracks exactly the interface. Contrary with other tracking techniques where a special mesh is built around each interface [30,31], PFEM rebuilds the complete mesh with a fast mesh generator. The method becomes competitive when the interfaces between the different fluids are dominant (see Fig. 1.1).

The Lagrangian fluid flow equations for the Navier–Stokes approximation will be revised in the next sections including an implicit fractional-step method for the time integration. Then, the basis of PFEM will be summarized. Finally, the PFEM will be used to solve some heterogeneous flows problems including interfaces, free-surfaces and breaking waves.

2. Governing equations and fractional-step method in Lagrangian formulations

In the final x_i position, the mass and momentum conservation equations can be written as:

$$\text{Mass conservation : } \frac{D\rho}{Dt} + \rho \frac{\partial u_i}{\partial x_i} = 0, \tag{2.1}$$

$$\text{Momentum conservation : } \rho \frac{Du_i}{Dt} = -\frac{\partial}{\partial x_i} p + \frac{\partial}{\partial x_j} \tau_{ij} + \rho f_i, \tag{2.2}$$

where $\rho = \rho(x)$ is the density, u_i are the Cartesian components of the velocity field, p the pressure, τ_{ij} the deviator stress tensor, f_i the source term (normally the gravity) and $\frac{D\phi}{Dt}$ represents the total or material time derivative of a function ϕ .

For heterogeneous materials ρ is a function of the position $\rho = \rho(x)$. For incompressible flows the material time derivative $\frac{D\rho}{Dt} = 0$. Nevertheless the spatial time derivative is not ($\frac{\partial \rho}{\partial t} \neq 0$). This is the reason why heterogeneous materials are more easily solved with Lagrangian formulations.

From Eq. (2.1), for incompressible flows the mass conservation in Lagrangian formulation may be simply written as: $\frac{\partial u_i}{\partial x_i} = 0$, while in Eulerian formulation mass conservation must be written as: $\frac{\partial \rho}{\partial t} + u_i \frac{\partial \rho}{\partial x_i} = 0$.

For Newtonian fluids the stress tensor τ_{ij} may be expressed as a function of the velocity field through the viscosity μ by

$$\tau_{ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \frac{\partial u_l}{\partial x_l} \delta_{ij} \right). \tag{2.3}$$

For incompressible or near incompressible flows ($\frac{\partial u_l}{\partial x_l} = 0$ or $\frac{\partial u_l}{\partial x_l} \ll \frac{\partial u_i}{\partial x_i}$) the term:

$$\frac{2\mu}{3} \frac{\partial u_l}{\partial x_l} = 0 \text{ or } \approx 0 \tag{2.4}$$

and it may be neglected in Eq. (2.3). Then:

$$\tau_{ij} \approx \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right). \tag{2.5}$$

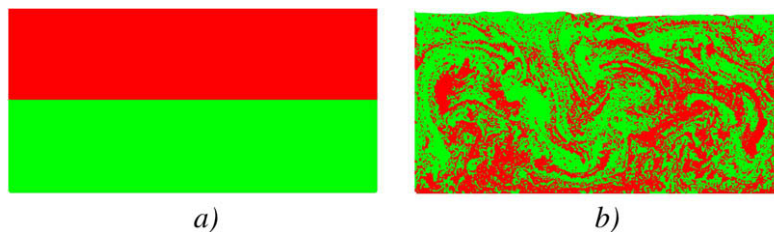


Fig. 1.1. Two phase-flows with a fluid of lower density at the bottom part. (a) Initial position and (b) position after 1 s.

The particular case of an incompressible or near incompressible flow with constant viscosity can be further simplified given the well known Laplace form of the Navier–Stokes equations. This approximation will be used in this work because then the equations to be solved separate in each velocity component. It must be noted however that this formulation may be used only for fluids with constant viscosity in the entire domain. This approximation does not hold for fluids with variable viscosity or element by element constant viscosity [32]. In the Laplace form the term $\frac{\partial}{\partial x_j} \tau_{ij}$ in the momentum equations is simplified as:

$$\begin{aligned} \frac{\partial}{\partial x_j} \tau_{ij} &= \frac{\partial}{\partial x_j} \left(\mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right) = \mu \frac{\partial}{\partial x_j} \left(\frac{\partial u_i}{\partial x_j} \right) + \mu \frac{\partial}{\partial x_j} \left(\frac{\partial u_j}{\partial x_i} \right) \\ &= \mu \frac{\partial}{\partial x_j} \left(\frac{\partial u_i}{\partial x_j} \right) + \mu \frac{\partial}{\partial x_i} \left(\frac{\partial u_j}{\partial x_j} \right) \approx \mu \frac{\partial}{\partial x_j} \left(\frac{\partial u_i}{\partial x_j} \right). \end{aligned} \tag{2.6}$$

Then, the momentum equations can be finally written as:

$$\rho \frac{Du_i}{Dt} = -\frac{\partial}{\partial x_i} p + \frac{\partial}{\partial x_j} \tau_{ij} + \rho f_i \approx -\frac{\partial}{\partial x_i} p + \mu \frac{\partial}{\partial x_j} \left(\frac{\partial u_i}{\partial x_j} \right) + \rho f_i. \tag{2.7}$$

As stated before, Eq. (2.7) show that only the u_i velocity component is present in each “ith” equation. When using pressure segregation methods as the fractional-step method, this implies that each velocity component may be solved independently from the others, which leads to considerable savings in computing time. For variable viscosity however the fully coupled Navier–Stokes equations (Eq. (2.2)) must be used in all cases.

2.1. Boundary and interface conditions

The standard boundary conditions for the Navier–Stokes equations are:

$$\tau_{ij} v_j - p v_i = \bar{\sigma}_{ni} \text{ on } \Gamma_\sigma; \quad u_i v_i = \bar{u}_n \text{ on } \Gamma_n; \quad u_i \zeta_i = \bar{u}_t \text{ on } \Gamma_t,$$

where v_i and ζ_i are the components of the normal and tangent vectors to the boundary.

On the internal interfaces the conditions are:

$$(\tau_{ij}^+ v_j - p^+ v_i) = \sigma_{ni}^+ = -(\tau_{ij}^- v_j - p^- v_i) = -\sigma_{ni}^-; \quad u_i^+ = u_i^- \text{ on } \Gamma_i,$$

where super indices (+) and (−) represent the variable value at each side of the interface.

2.2. Implicit time integration

Eq. (2.7) will be integrated implicitly in time as:

$$\begin{aligned} \rho \frac{Du_i}{Dt} &\approx \rho \frac{u_i(x_i, t^{n+1}) - u_i(x_i, t^n)}{\Delta t} = \rho \frac{u_i^{n+1} - u_i^n}{\Delta t} \\ &= \left[-\frac{\partial}{\partial x_i} p + \mu \frac{\partial}{\partial x_j} \left(\frac{\partial u_i}{\partial x_j} \right) + \rho f_i \right]^{n+\theta}, \end{aligned} \tag{2.8}$$

where $[\phi(x, t)]^{n+\theta}$ means $\theta \phi(x, t^{n+1}) + (1 - \theta) \phi(x, t^n) = \theta \phi^{n+1} + (1 - \theta) \phi^n$ and $\phi^n = \phi(x, t^n)$ represents the value of the function at time t^n but at the final position x .

Only the case of $\theta = 1$ (full implicit) will be considered next. Other values as for instance $\theta = 1/2$ can be considered without major changes.

Then, the time integrated equations become:

$$\rho \frac{u_i^{n+1} - u_i^n}{\Delta t} = \left[-\frac{\partial}{\partial x_i} p \right]^{n+1} + \left[\mu \frac{\partial}{\partial x_j} \left(\frac{\partial u_i}{\partial x_j} \right) + \rho f_i \right]^{n+1}. \tag{2.9}$$

The mass conservation is also integrated implicitly by:

$$\frac{D\rho}{Dt} \approx \frac{\rho^{n+1} - \rho^n}{\Delta t} = -\rho^{n+1} \frac{\partial(u_i^{n+1})}{\partial x_i}. \tag{2.10}$$

2.2.1. The time splitting

The time integration of Eq. (2.9) presents some difficulties as it is a full coupled equation involving four degrees of freedom for each node. When the fluid is incompressible or nearly incompressible advantages can be taken from the fact that in Eq. (2.9) the three components of the velocity are only coupled via the pressure. A fractional-step method has been proposed [16] which consists in splitting each time step in two pseudo-time steps. In the first step the implicit part of the pressure is avoided leading to a decoupled equation in each of the velocity components. The implicit part of the pressure is added in a second step. The fractional-step algorithm is:

$$\begin{aligned} \frac{Du_i}{Dt} &\approx \frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{u_i^{n+1} - u_i^* + u_i^* - u_i^n}{\Delta t} \\ &= -\frac{1}{\rho} \frac{\partial}{\partial x_i} p^{n+1} + \frac{1}{\rho} \frac{\partial \tau_{ij}^{n+1}}{\partial x_j} + f_i, \end{aligned} \tag{2.11}$$

where u_i^* are intermediate variables defined by the split:

$$(A) \quad u_i^* = u_i^n + f_i \Delta t - \frac{\Delta t}{\rho} \frac{\partial}{\partial x_i} \gamma p^n + \frac{\Delta t}{\rho} \frac{\partial}{\partial x_j} \tau_{ij}^{n+1}, \tag{2.12}$$

$$(C) \quad u_i^{n+1} = u_i^* - \frac{\Delta t}{\rho} \frac{\partial}{\partial x_i} (p^{n+1} - \gamma p^n), \tag{2.13}$$

γ is a parameter between 0 and 1 defining the amount of pressure splitting.

Taking into account (2.6), the last term in (2.12) may be approximated as:

$$\frac{\partial}{\partial x_j} \tau_{ij}^{n+1} = \mu \frac{\partial}{\partial x_j} \left(\frac{\partial u_i^{n+1}}{\partial x_j} \right) \approx \mu \frac{\partial}{\partial x_j} \left(\frac{\partial u_i^*}{\partial x_j} \right),$$

which allows to write Eq. (2.12) as:

$$(A) \quad u_i^* = u_i^n + f_i \Delta t - \frac{\Delta t}{\rho} \frac{\partial}{\partial x_i} \gamma p^n + \frac{\Delta t}{\rho} \mu \theta \frac{\partial}{\partial x_j} \left(\frac{\partial u_i^*}{\partial x_j} \right).$$

The mass conservation equations may be also split as:

$$\begin{aligned} \frac{D\rho}{Dt} &\approx \frac{\rho^{n+1} - \rho^n}{\Delta t} = \frac{\rho^{n+1} - \rho^* + \rho^* - \rho^n}{\Delta t} \\ &= -\rho \frac{\partial(u_i^{n+1} - u_i^* + u_i^*)}{\partial x_i}, \end{aligned} \tag{2.14}$$

where ρ^* is again an intermediate variable defined by the split:

$$\frac{\rho^* - \rho^n}{\Delta t} = -\rho \frac{\partial u_i^*}{\partial x_i}, \tag{2.15}$$

$$\frac{\rho^{n+1} - \rho^*}{\Delta t} = -\rho \frac{\partial(u_i^{n+1} - u_i^*)}{\partial x_i}. \tag{2.16}$$

From Eqs. (2.13) and (2.16) the coupled mass–momentum equation becomes:

$$(B) \quad \frac{\rho^{n+1} - \rho^*}{\Delta t^2} = \frac{\partial^2}{\partial x_i^2} (p^{n+1} - \gamma p^n). \tag{2.17}$$

Taking into account Eq. (2.15) the above expression can be written as:

$$(B) \quad \frac{\rho^{n+1} - \rho^n}{\Delta t^2} + \frac{\rho}{\Delta t} \frac{\partial u_i^*}{\partial x_i} = \frac{\partial^2}{\partial x_i^2} (p^{n+1} - \gamma p^n). \tag{2.18}$$

In Eq. (2.18) the incompressibility condition has not be introduced yet. The simplest way to introduce the incompressibility condition in a Lagrangian formulation is to write:

$$\rho_{(x)}^{n+1} = \rho_{(x)}^n = \rho_{(x)}^0 = \rho_{(x)}. \tag{2.19}$$

Then, the first term of Eq. (2.18) disappears, remaining

$$(B) \quad \frac{\rho_{(x)} \partial u_i^*}{\Delta t \partial x_i} = \frac{\partial^2}{\partial x_i^2} (p^{n+1} - \gamma p^n).$$

The three-step fractional method for a Lagrangian formulation may be summarized as:

$$\begin{aligned} (A) \quad & u_i^* - \frac{\Delta t}{\rho_{(x)}} \mu \frac{\partial}{\partial x_j} \left(\frac{\partial u_i^*}{\partial x_j} \right) = u_i^n + f_i \Delta t - \frac{\Delta t}{\rho_{(x)}} \frac{\partial}{\partial x_i} \gamma p^n \Rightarrow u_i^*, \\ (B) \quad & \frac{\rho_{(x)} \partial u_i^*}{\Delta t \partial x_i} = \frac{\partial^2}{\partial x_i^2} (p^{n+1} - \gamma p^n) \Rightarrow p^{n+1}, \\ (C) \quad & u_i^{n+1} = u_i^* - \frac{\Delta t}{\rho_{(x)}} \frac{\partial}{\partial x_i} (p^{n+1} - \gamma p^n) \Rightarrow u_i^{n+1}. \end{aligned} \quad (2.20)$$

It must be noted that although Eqs. (2.20) seem linear they are not because they are all evaluated at the final position. The algorithm must be necessarily iterative. Once u_i^{n+1} is evaluated during each iteration, then the coordinates are updated to the new position as described in a next section and the three steps A, B and C are repeated until convergence.

3. The Particle Finite Element Method

Particle methods aim to represent the behavior of a physical problem by a collection of particles. Each particle moves accordingly to its own mass and the internal/external forces applied on it. External forces are evaluated by the interaction with the neighbor particles by simple rules [16,17].

Another characteristic of particle methods is that all the physical and mathematical properties are attached to the particle itself and not to the elements as in FE methods. For instance physical properties like viscosity or density, physical variables like velocity,

temperature or pressure and also mathematical variables like gradients or volumetric deformations are assigned to each particle and they represent an average of the property around the particle position.

Particle methods are advantageous to treat discrete problems such as granular materials but also to treat continuous problems for which possibilities exist of internal interfaces, frictional-contact problems in fluid–solid interactions or free-surfaces with breaking waves.

The most relevant characteristic of a particle method is that there is not a specified solution domain. The problem domain is defined by the particle positions and hence, there is not a boundary contour. This is the reason why, when a differential equation is to be solved in order to evaluate the forces, the boundary needs to be identified in order to impose the boundary conditions. In addition, the particles can be used to generate a discrete domain within which the integral forms of the governing differential equations are solved (see Fig. 3.1).

In this paper, the Particle Finite Element Method (PFEM) [16,17] is used to deal with the incompressible Navier–Stokes equations for heterogeneous fluid flows. Different fluids acting together will be modelled by an arbitrary number of particles. On each particle the forces will be the gravity force (internal force of the particle) and the interacting forces with the neighboring particles (external force to the particle). The external forces will be evaluated by solving the Navier–Stokes equations. For this purpose an analysis domain needs to be defined at each time step with a known contour where the boundary conditions will be imposed.

Also at each time step a new mesh is generated in order to define the new interfaces between the different fluids and the corresponding shape functions to solve the differential equations. This

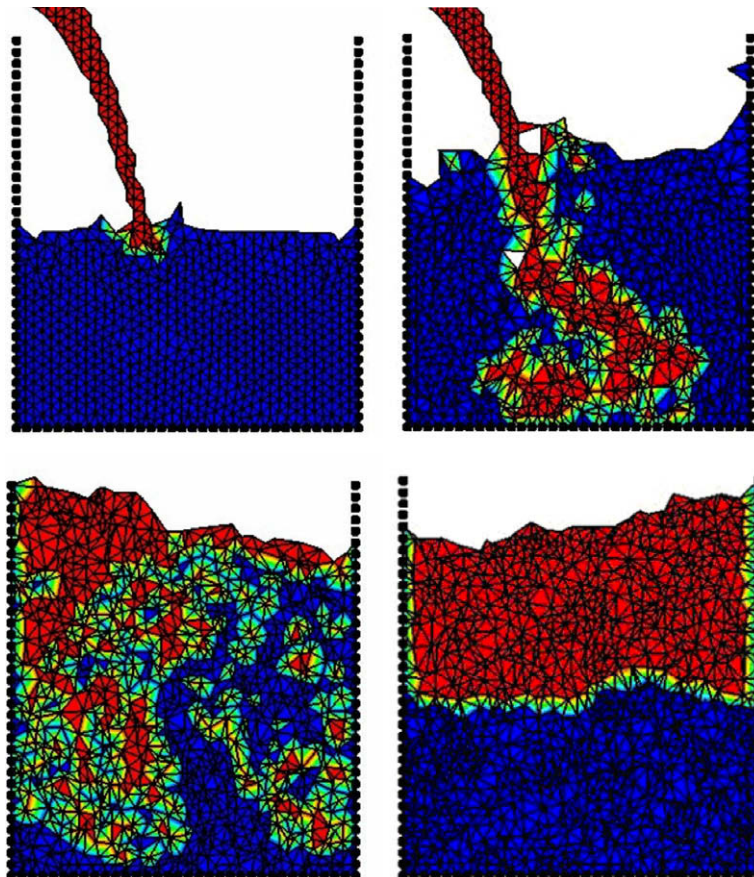


Fig. 3.1. Mixing two different fluids. Recognition of the internal interfaces and free-surfaces of the analysis domain and mesh update for successive point distributions.

mesh is only useful for the definition of the interacting forces and vanishes once the forces are evaluated (see Fig. 3.1).

The interpolation functions are a particular case of the finite element shape functions [14,15]. The boundary surface is defined using the alpha-shape technique [16,17].

3.1. Particle position update

The particle positions are updated by solving the Lagrangian form of the Navier–Stokes equations.

Let X_i be the initial position of a particle at time t^n .

Let x_i be the final position of a particle at time t^{n+1} and the time increment $\Delta t = t^{n+1} - t^n$.

Being $u_i(x, t^{n+1}) = u_i^{n+1}$ the velocity of the particle at time t^{n+1} , the final position can be evaluated by:

$$x_i = X_i + \int_{t^n}^{t^{n+1}} u_i dt \approx X_i + u_i^{n+1} \theta \Delta t + u_i^n (1 - \theta) \Delta t. \quad (3.1)$$

In the same way the displacement of the particle is computed as:

$$d_i(x, t^{n+1}) = d_i^{n+1} \approx u_i^{n+1} \theta \Delta t + u_i^n (1 - \theta) \Delta t \quad (3.2)$$

with θ between 0 and 1.

3.2. Generation of a new mesh

One of the key points for success of the Lagrangian formulation described here is the fast regeneration of a mesh at every time step on the basis of the position of the nodes in the space domain. In this work the mesh is generated using the so called Extended Delaunay Tessellation (EDT) presented in [15]. The EDT allows generating meshes of elements with arbitrary polyhydic shapes (combining triangles, quadrilaterals and other polygons in 2D and tetrahedra, hexahedra and arbitrary polyhedra in 3D) in a computing time of order n , being n the total number of nodes in the mesh. Details of the mesh generation procedure and the shape functions for arbitrary polyhedra can be found in [14,15]. Once the new mesh has been generated at each time step the numerical solution is found using the standard finite element method (FEM).

3.3. Spatial discretization

Using FEM, the unknown functions are approximated using an equal order interpolation for all variables in the final configuration:

$$\begin{aligned} u_i &= \sum_l N_l(X, t) u_{il}, \\ p &= \sum_l N_l(X, t) p_l. \end{aligned} \quad (3.3)$$

In compact form:

$$\begin{aligned} u_i &= \mathbf{N}_i^T(X, t) \mathbf{u}_i, \\ p &= \mathbf{N}_p^T(X, t) \mathbf{P} \end{aligned} \quad (3.4)$$

or in matrix form:

$$u_i = \mathbf{N}^T \mathbf{u} = \begin{bmatrix} \mathbf{N}_i^T & & \\ & \mathbf{N}_i^T & \\ & & \mathbf{N}_i^T \end{bmatrix} \mathbf{u}, \quad (3.5)$$

where \mathbf{N}_i and \mathbf{N}_p are the FEM shape functions and \mathbf{u} , \mathbf{P} the nodal values of the three components of the unknown velocity and the pressure, respectively.

It must be noted that the shape functions $\mathbf{N}(X, t)$ are functions of the particle coordinates. Then, the shape functions may change in time following the particles position. During the time step a mesh update may introduce change in the shape functions which must be taken into account. During the time integration there two times are involved: t^n and t^{n+1} . The following notation will be used to distinguish between $\mathbf{N}(X, t^n)$ and $\mathbf{N}(X, t^{n+1})$:

$$\mathbf{N}(X, t^n) = \mathbf{N}^n \quad \text{and} \quad \mathbf{N}(X, t^{n+1}) = \mathbf{N}.$$

The following hypothesis will be introduced: the mesh is not updated during the iterations within each time step. This means that if a mesh update is introduced at the beginning of a time step, the same mesh (but deformed) will continue until the end of the time step. Mathematically expressed, $\mathbf{N}(X, t^n) = \mathbf{N}(X, t^{n+1})$. Unfortunately, this hypothesis is not always possible to satisfy for all meshes, and this introduces small errors in the computations which are neglected in this paper.

3.4. Discrete equations

Using the Galerkin weighted residual method to solve the split equations, the following integral forms are obtained:

$$\begin{aligned} \text{(A)} \quad & \frac{1}{\Delta t} \int_V \mathbf{N}_i \rho \mathbf{u}_i^* dV - \frac{1}{\Delta t} \int_V \mathbf{N}_i \rho \mathbf{u}_i^n dV - \int_V \mathbf{N}_i f_i \rho dV + \int_V \mathbf{N}_i \frac{\partial}{\partial x_i} \gamma p^n dV \\ & - \int_V \mathbf{N}_i \mu \frac{\partial}{\partial x_j} \left(\frac{\partial u_i^*}{\partial x_j} \right) dV - \int_{\Gamma_\sigma} \mathbf{N}_i (\bar{\sigma}_{ni} - (\tau_{ij}^* v_j - \gamma p^n v_i)) d\Gamma = \mathbf{0}, \end{aligned} \quad (3.6)$$

$$\begin{aligned} \text{(B)} \quad & \int_V \mathbf{N}_p \left\{ \frac{\rho}{\Delta t} \left(\frac{\partial u_i^*}{\partial x_i} \right) - \frac{\partial^2}{\partial x_i^2} (p^{n+1} - \gamma p^n) \right\} dV \\ & + \frac{1}{\Delta t} \int_{\Gamma_u} \mathbf{N}_p \rho (\bar{u}_i^{n+1} v_i - u_i^{n+1} v_i) d\Gamma = \mathbf{0}, \end{aligned} \quad (3.7)$$

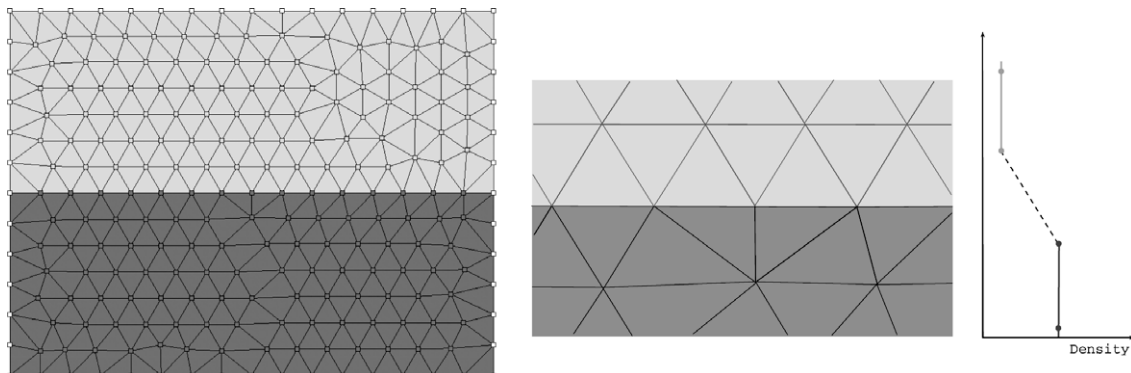


Fig. 5.1. Interface across elements.

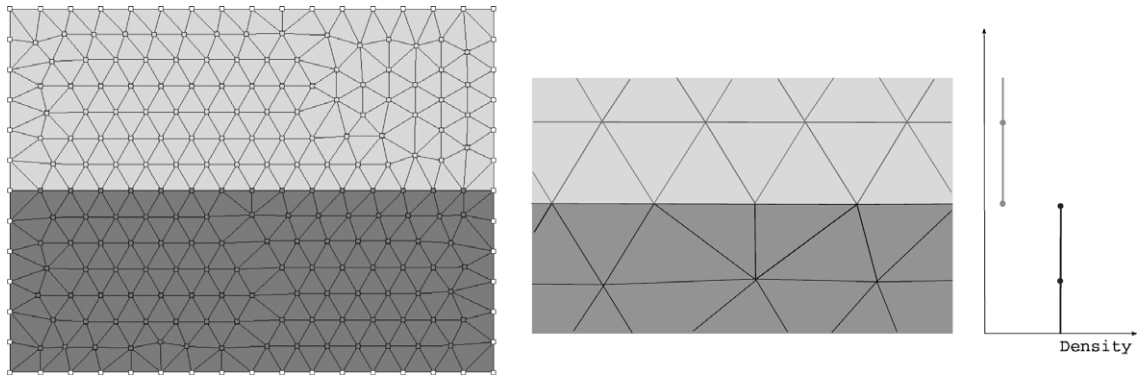


Fig. 5.2. Nodal interface.

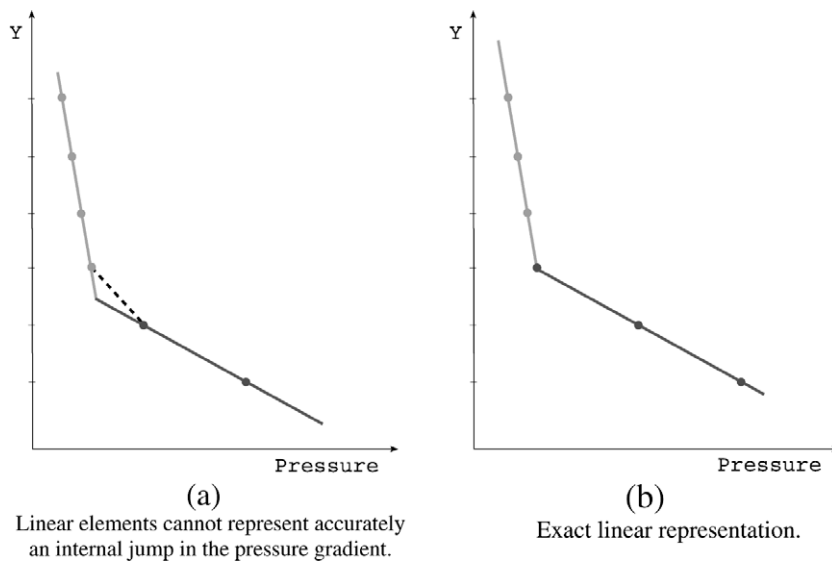


Fig. 5.3. Pressure approximation of the different interface definitions.

$$(C) \int_V \mathbf{N}_i \left\{ \frac{\rho}{\Delta t} (u_i^{n+1} - u_i^*) + \frac{\partial}{\partial x_i} (p^{n+1} - \gamma p^n) \right\} dV - \int_{\Gamma_\sigma} \mathbf{N}_i (p^{n+1} - \gamma p^n) v_i d\Gamma = \mathbf{0}, \quad (3.8)$$

where the boundary conditions have been also split and V is the volume at time t^{n+1} .

Integrating by parts some of the terms, the above equations become:

$$(A) \int_V \mathbf{N}_i \left(\frac{\rho}{\Delta t} u_i^* - \rho f_i \right) dV - \int_V \mathbf{N}_i \frac{\rho}{\Delta t} u_i^n dV + \int_V \mathbf{N}_i \gamma \frac{\partial p^n}{\partial x_i} dV + \mu \int_V \frac{\partial \mathbf{N}_i}{\partial x_j} \frac{\partial u_i^{n+\theta}}{\partial x_j} dV - \int_{\Gamma_\sigma} \mathbf{N}_i (\bar{\sigma}_{ni} + \gamma p^n v_i) d\Gamma = \mathbf{0}, \quad (3.9)$$

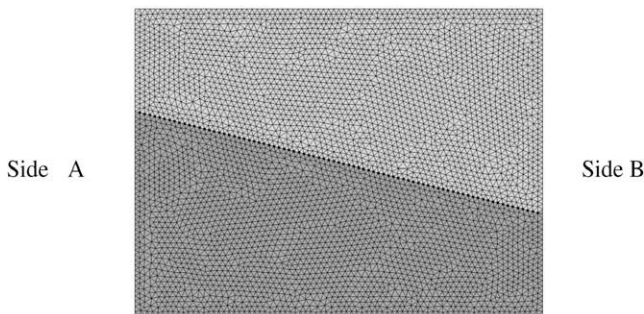


Fig. 6.1. Initial mesh (4305 nodes).

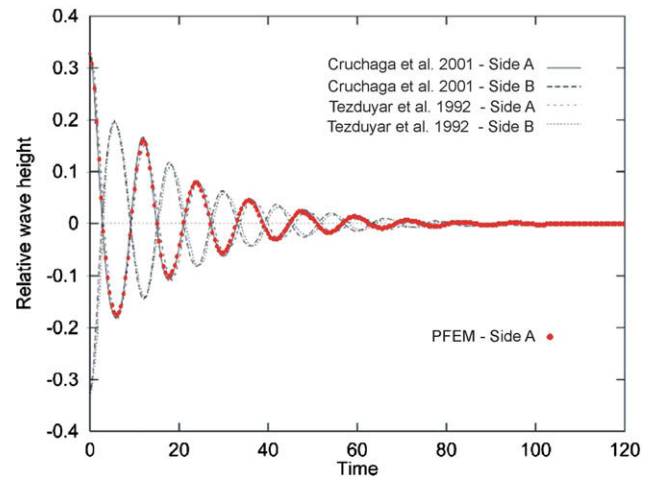
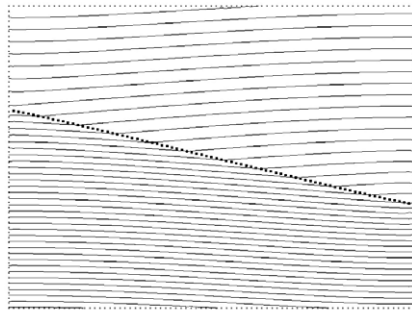


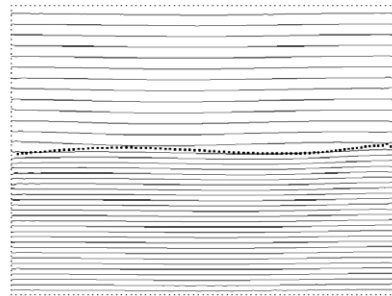
Fig. 6.2. Relative wave height (black dots) compared to references.

$$(B) \quad -\frac{1}{\Delta t} \int_V \frac{\partial \mathbf{N}_p}{\partial x_i} \rho u_i^* dV + \int_V \frac{\partial \mathbf{N}_p}{\partial x_i} \frac{\partial (p^{n+1} - \gamma p^n)}{\partial x_i} dV + \frac{1}{\Delta t} \int_{\Gamma_u} \mathbf{N}_p \rho \bar{u}_n^{n+1} d\Gamma = \mathbf{0}, \quad (3.10)$$

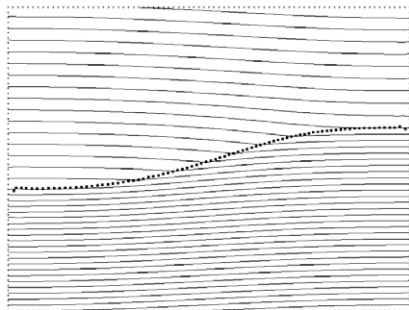
$$(C) \quad \int_V \mathbf{N}_i \left\{ \frac{\rho}{\Delta t} (u_i^{n+1} - u_i^*) + \frac{\partial}{\partial x_i} (p^{n+1} - \gamma p^n) \right\} dV - \int_{\Gamma_\sigma} \mathbf{N}_i (p^{n+1} - \gamma p^n) d\Gamma = \mathbf{0}. \quad (3.11)$$



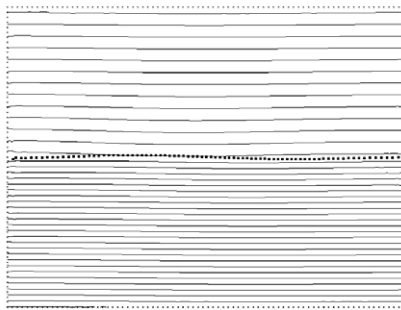
a) t = 0.5 s



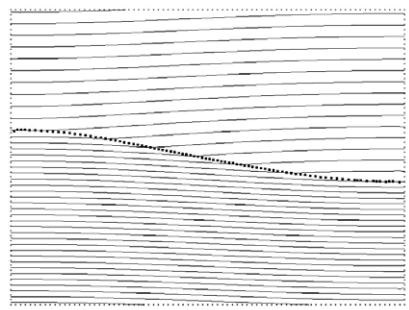
b) t = 3 s



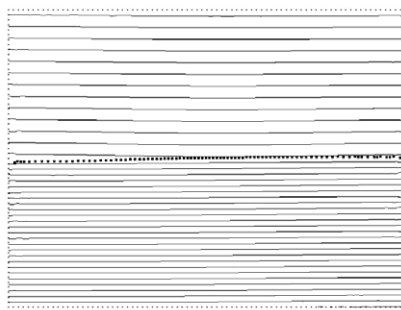
c) t = 6 s



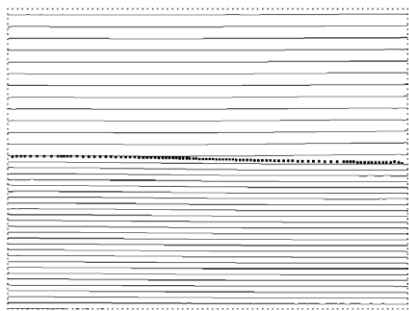
d) t = 9 s



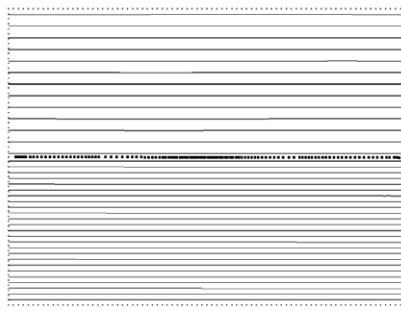
e) t = 12 s



f) t = 15 s



g) t = 21 s



h) t = 120 s

Fig. 6.3. Pressure contour lines for two-fluid sloshing 2:1.

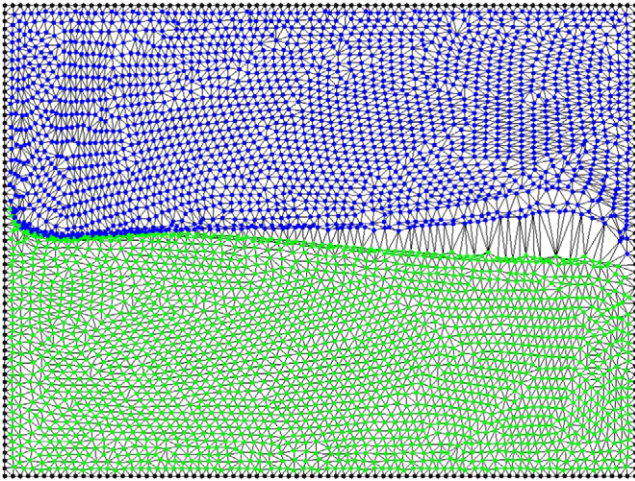


Fig. 6.4. Wrong interface equilibrium with the standard stabilization method (Eq. (3.22)) which does not take properly into account the density jump at the interface.

The essential and natural boundary conditions of Eq. (3.10) are:

$$p = 0 \quad \text{on } \Gamma_\sigma, \quad (3.12)$$

$$\bar{\mathbf{u}}^{n+1} \cdot \mathbf{v} = 0 \quad \text{on } \Gamma_u. \quad (3.13)$$

3.5. Compact form

Using the approximations (3.4) the discrete equations become in compact form:

$$(A) \quad (\mathbf{M}_\rho + \Delta t \mu \mathbf{K}) \mathbf{u}^* = \mathbf{M}_\rho \mathbf{u}^n + \Delta t \mathbf{F} - \gamma \Delta t \mathbf{B}^T \mathbf{P}^n, \quad (3.14)$$

$$(B) \quad \mathbf{L} \mathbf{P}^{n+1} = \frac{1}{\Delta t} (\mathbf{B}_\rho \mathbf{u}^* - \bar{\mathbf{u}}) + \gamma \mathbf{L} \mathbf{P}^n, \quad (3.15)$$

$$(C) \quad \mathbf{M}_\rho \mathbf{u}^{n+1} = \mathbf{M}_\rho \mathbf{u}^* - \Delta t \mathbf{B}^T (\mathbf{P}^{n+1} - \gamma \mathbf{P}^n), \quad (3.16)$$

where

$$\mathbf{L} = \int_V \frac{\partial \mathbf{N}_i}{\partial x_i} \frac{\partial \mathbf{N}_i^T}{\partial x_i} dV; \quad \mathbf{M}_\rho = \begin{bmatrix} \mathbf{M}_{11} & 0 & 0 \\ 0 & \mathbf{M}_{22} & 0 \\ 0 & 0 & \mathbf{M}_{33} \end{bmatrix};$$

$$\mathbf{M}_{ii} = \int_V \mathbf{N}_i \rho_{(x)} \mathbf{N}_i^T dV; \quad \mathbf{K} = \begin{bmatrix} \mathbf{L} & 0 & 0 \\ 0 & \mathbf{L} & 0 \\ 0 & 0 & \mathbf{L} \end{bmatrix}; \quad (3.17)$$

$$\mathbf{B} = \left[\int_V \left(\frac{\partial \mathbf{N}_p}{\partial x} \mathbf{N}_1^T \right) dV; \int_V \left(\frac{\partial \mathbf{N}_p}{\partial y} \mathbf{N}_2^T \right) dV; \int_V \left(\frac{\partial \mathbf{N}_p}{\partial z} \mathbf{N}_3^T \right) dV \right];$$

$$\bar{\mathbf{u}}_i = \int_{\Gamma_u} \mathbf{N}_i \bar{u}_i^{n+1} d\Gamma, \quad (3.18)$$

$$\mathbf{B}_\rho = \left[\int_V \left(\frac{\partial \mathbf{N}_p}{\partial x} \rho_{(x)} \mathbf{N}_1^T \right) dV; \int_V \left(\frac{\partial \mathbf{N}_p}{\partial y} \rho_{(x)} \mathbf{N}_2^T \right) dV; \int_V \left(\frac{\partial \mathbf{N}_p}{\partial z} \rho_{(x)} \mathbf{N}_3^T \right) dV \right], \quad (3.19)$$

$$\mathbf{F}_i = \int_V \mathbf{N}_i f_i \rho_{(x)} dV + \int_{\Gamma_\sigma} \mathbf{N}_i \bar{\sigma}_m d\Gamma. \quad (3.20)$$

The Lagrangian split scheme described has two important advantages:

- (1) Steps A and C are linear. The Lagrangian formulation eliminates the standard convection terms present in Eulerian formulations. The convection terms are responsible for non-

linearity, non symmetry and non self-adjoint operators which require stabilization terms to avoid numerical oscillations. All these problems are not present in this formulation. Only the nonlinearity due to the unknown final positions of the particles remains.

- (2) Steps A and/or C may be solved explicitly with relative large time steps lumping the mass matrix \mathbf{M}_ρ and approximating the viscous term $\Delta t \mu \mathbf{K} \mathbf{u}^*$ by $\Delta t \mu \mathbf{K} \mathbf{u}^n$.
- (3) In all steps the above systems of equations are solved for evaluation of the velocity components (steps A and C) and the pressure (step B). These systems are scalar (only one degree of freedom by node), symmetric and positive definite. Then, it is very easy to solve them using a symmetric iterative scheme (such as the conjugate gradient method).

3.6. Stabilization of the incompressibility condition

In the Eulerian form of the momentum equations, the discrete form must be stabilized in order to avoid numerical wiggles in the velocity and pressure results. This is not the case in the Lagrangian formulation where no stabilization parameter must be added in steps A and C. Nevertheless, the incompressibility condition must be stabilized for equal-order approximations to avoid possible pressure oscillations in some particular cases.

For instance, for small pressure split ($\gamma \neq 0$) or for small time step increments (Courant number much less than one) it is well known that the fractional step does not stabilize the pressure waves. In those particular cases, stabilization term must be introduced in step B in order to eliminate pressure oscillations [33]. On the other hand, examples where the γ parameter is always fixed to zero and the time increments correspond to Courant number ≈ 1 avoid all the stabilization problems.

Pressure stabilization means to add a term into step B that in compact form represents a matrix \mathbf{S} in such way that:

$$(\mathbf{L} + \mathbf{S}) \mathbf{P}^{n+1} = \frac{1}{\Delta t} (\mathbf{B}_\rho \mathbf{u}^* - \bar{\mathbf{u}}) + \gamma \mathbf{L} \mathbf{P}^n. \quad (3.21)$$

Residual-based stabilization methods like Finite Calculus (FIC) [34,35] propose a stabilization term of the form:

$$- \int_V \mathbf{N}_p \frac{\partial}{\partial x_i} \left[\tau \left(\frac{\partial p}{\partial x_i} - \pi_i \right) \right] dV, \quad (3.22)$$

where the term in brackets is interpreted as an approximation of the residual of the momentum equations and π_i is a continuous function obtained from the projection of the pressure gradient on the velocity field and τ is a stabilization parameter.

In order to take into account for jumps on the interfaces for the density, the π_i functions will be defined as projections of the pressure gradients divided by $\rho_{(x)}$. The new stabilization term becomes:

$$- \int_V \mathbf{N}_p \frac{\partial}{\partial x_i} \left[\tau \left(\frac{\partial p}{\partial x_i} - \rho_{(x)} \pi_i \right) \right] dV. \quad (3.23)$$

The projection vector π_i is obtained from its definition using the following weak expression:

$$\int_V \mathbf{N}_i \left(\frac{\partial p}{\partial x_i} - \rho_{(x)} \pi_i \right) dV = 0$$

with $\pi_i = \mathbf{N}_i^T \Pi_i$ being Π_i the vector with the nodal values of π_i .

Then,

$$\int_V \mathbf{N}_i \frac{\partial \mathbf{N}_p^T}{\partial x_i} dV \mathbf{P} - \int_V \mathbf{N}_i \rho_{(x)} \mathbf{N}_i^T dV \Pi_i = 0 \Rightarrow \mathbf{B}^T \mathbf{P} = \mathbf{M}_\rho \Pi \Rightarrow \Pi = \mathbf{M}_\rho^{-1} \mathbf{B}^T \mathbf{P}. \quad (3.24)$$

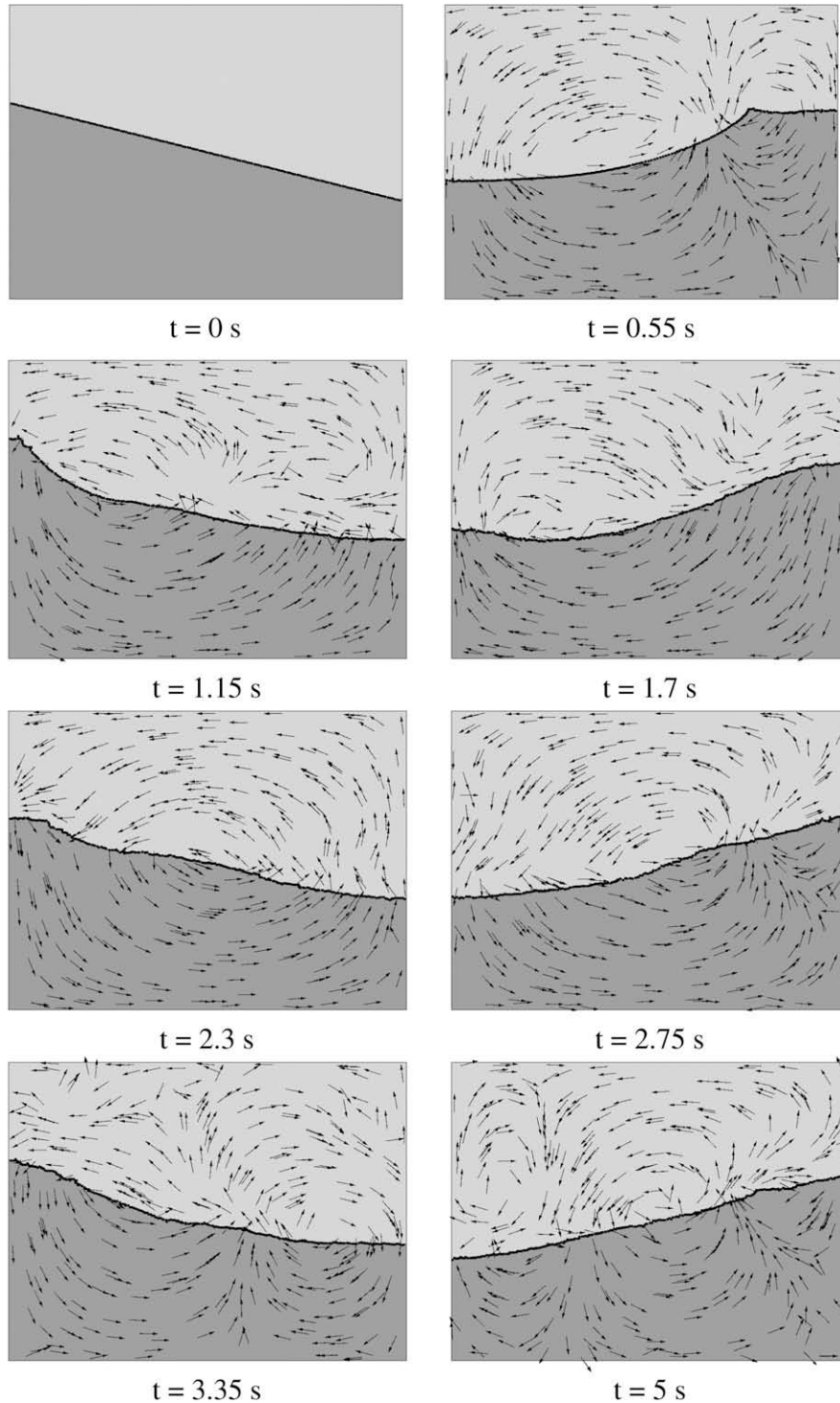


Fig. 6.5. Interface position and velocity vectors (fixed length) for two-fluid sloshing 1000:1 during the first three periods. Bottom fluid: water. Top fluid: air.

The stabilization term (Eq. (3.22)) or the new version (Eq. (3.23)) is integrated by parts and the boundary term is neglected here. This boundary term however may be important for preserving the total volume in free-surface flow problems [34]:

$$\begin{aligned}
 - \int_V \mathbf{N}_p \frac{\partial}{\partial x_i} \left[\tau \left(\frac{\partial p}{\partial x_i} - \rho_{(x)} \pi_i \right) \right] dV &\approx \int_V \frac{\partial \mathbf{N}_p}{\partial x_i} \tau \left(\frac{\partial p}{\partial x_i} - \rho_{(x)} \pi_i \right) dV \\
 &= \int_V \frac{\partial \mathbf{N}_p}{\partial x_i} \tau \frac{\partial \mathbf{N}_p^T}{\partial x_i} dV \mathbf{P} - \int_V \frac{\partial \mathbf{N}_p}{\partial x_i} \tau \rho_{(x)} \mathbf{N}_i^T dV \Pi_i.
 \end{aligned} \tag{3.25}$$

The stabilization term may be written in compact form as:

$$\mathbf{S}\mathbf{P} = \left(\mathbf{L}_\tau - \mathbf{B}_{\tau\rho} \mathbf{M}_\rho^{-1} \mathbf{B}^T \right) \mathbf{P}, \tag{3.26}$$

where \mathbf{L}_τ and $\mathbf{B}_{\tau\rho}$ are the same matrices defined previously but multiplied element by element by τ and $\tau\rho_{(x)}$, respectively, and then assembled.

Eq. (3.21) with the stabilization terms becomes:

$$\left(\mathbf{L} + \mathbf{L}_\tau - \mathbf{B}_{\tau\rho} \mathbf{M}_\rho^{-1} \mathbf{B}^T \right) \mathbf{P}^{n+1} = \frac{1}{\Delta t} \left(\mathbf{B}_\rho \mathbf{u}^* - \bar{\mathbf{u}} \right) + \gamma \mathbf{L} \mathbf{P}^n. \tag{3.27}$$

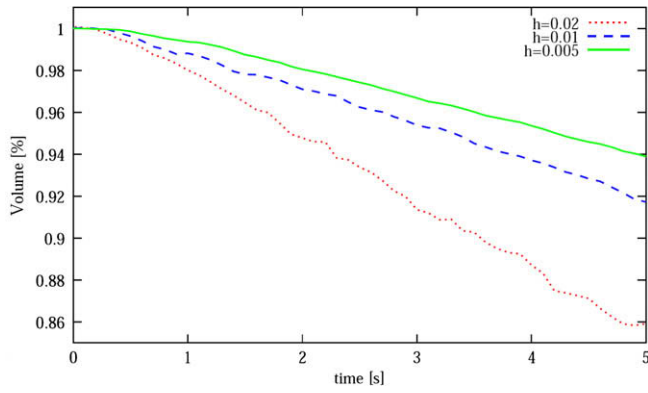


Fig. 6.6. Water mass variation due to remeshing for a large time period (5 s) for different mesh sizes (same time step).

In order to avoid the evaluation of the term $\mathbf{B}_{\tau\rho}\mathbf{M}_{\rho}^{-1}\mathbf{B}^T$, this term is solved explicitly using \mathbf{P}^n for the first iteration or the last pressure \mathbf{P}^{n+1} obtained during the last iteration. Let us call it for simplicity \mathbf{P} . Then

$$(\mathbf{L} + \mathbf{L}_{\tau})\mathbf{P}^{n+1} = \frac{1}{\Delta t}(\mathbf{B}_{\rho}\mathbf{u}^* - \bar{\mathbf{u}}) + \gamma\mathbf{L}\mathbf{P}^n + \mathbf{B}_{\tau\rho}\mathbf{M}_{\rho}^{-1}\mathbf{B}^T\mathbf{P}. \quad (3.28)$$

In two sub-steps:

$$\mathbf{M}_{\rho}\mathbf{\Pi} = \mathbf{B}^T\mathbf{P}, \quad (3.29)$$

$$(\mathbf{L} + \mathbf{L}_{\tau})\mathbf{P}^{n+1} = \frac{1}{\Delta t}(\mathbf{B}_{\rho}\mathbf{u}^* - \bar{\mathbf{u}}) + \gamma\mathbf{L}\mathbf{P}^n + \mathbf{B}_{\tau\rho}\mathbf{\Pi}. \quad (3.30)$$

The value used for τ is $\tau = \left(\frac{1}{\Delta t} + \frac{\mu}{h^2\rho}\right)^{-1}$ which is evaluated element by element [34]. The final algorithm reads:

(A) $(\mathbf{M}_{\rho} + \Delta t\mu\mathbf{K})\mathbf{u}^* = \mathbf{M}_{\rho}\mathbf{u}^n + \Delta t\mathbf{F} - \gamma\Delta t\mathbf{B}^T\mathbf{P}^n \Rightarrow \mathbf{u}^*$,
(B') $\mathbf{M}_{\rho}\mathbf{\Pi} = \mathbf{B}^T\mathbf{P} \Rightarrow \mathbf{\Pi}$,
(B) $(\mathbf{L} + \mathbf{L}_{\tau})\mathbf{P}^{n+1} = \frac{1}{\Delta t}(\mathbf{B}_{\rho}\mathbf{u}^* - \bar{\mathbf{u}}) + \gamma\mathbf{L}\mathbf{P}^n + \mathbf{B}_{\tau\rho}\mathbf{\Pi} \Rightarrow \mathbf{P}^{n+1}$,
(C) $\mathbf{M}_{\rho}\mathbf{u}^{n+1} = \mathbf{M}_{\rho}\mathbf{u}^* - \Delta t\mathbf{B}^T(\mathbf{P}^{n+1} - \gamma\mathbf{P}^n) \Rightarrow \mathbf{u}^{n+1}$.

(3.31)

Steps B' and C may be solved explicitly with a lumped \mathbf{M}_{ρ} matrix. Step A may be solved explicitly for small time steps and a small value of the viscosity parameter. For incompressible flows however step B must be solved implicitly with the full Laplace matrix $(\mathbf{L} + \mathbf{L}_{\tau})$.

4. Heat conduction equation in Lagrangian formulation

In a Lagrangian description, the conduction–convection heat equation reads:

$$\rho C \frac{DT}{Dt} = \frac{\partial}{\partial x_i} \left(\kappa \frac{\partial T}{\partial x_i} \right) + Q, \quad (4.1)$$

where T is the temperature, κ the thermal conductivity, C the heat capacity and Q a possible source terms contributions. The problem definition is completed with standard boundary conditions including heat convection and radiation fluxes, and imposed temperature boundaries.

The term $\frac{DT}{Dt}$ means the Lagrangian or material derivative and ρ the density corresponding to the material points. For this reason, the convective terms are not present in Eq. (4.1).

After discretization of the temperature field $T = \mathbf{N}_T^T \mathbf{T}$, Eq. (4.1) reads:

$$(\mathbf{M}_{\rho C} + \Delta t\mathbf{L}_{\kappa})\mathbf{T}^{n+1} = \mathbf{M}_{\rho C}\mathbf{T}^n + \Delta t\mathbf{Q} \quad (4.2)$$

with:

$$\mathbf{M}_{\rho C} = \int_V \mathbf{N}_T \rho_{(x)} C_{(x)} \mathbf{N}_T^T dV; \quad \mathbf{L}_{\kappa} = \int_V \frac{\partial \mathbf{N}_T}{\partial x_i} \kappa_{(x)} \frac{\partial \mathbf{N}_T^T}{\partial x_i} dV;$$

$$\mathbf{Q} = \int_V \mathbf{N}_T Q_{(x)} dV + \int_{\Gamma_q} \mathbf{N}_T \bar{q}_n d\Gamma.$$

Boussinesq approximation for small variations in the density due to thermal dilatation of the materials stands:

$$\rho = \rho_r [1 - \alpha(T - T_r)], \quad (4.3)$$

where T_r is a reference temperature, ρ_r is the density of the material with temperature T_r and α is a dilatation parameter (in general α is also a function of the temperature).

The Boussinesq approximation is usually modelled by introducing a buoyancy force in the right hand side of the momentum equations. Effectively, when the main forces of source terms in Eq. (2.1) are the gravity forces, (e.g. $f_i = -g_i$, with g_i the gravity acceleration), the momentum equations with the standard Boussinesq approximation read:

$$\rho_r \frac{Du_i}{Dt} = -\frac{\partial}{\partial x_i} p + \frac{\partial}{\partial x_j} \tau_{ij} - g_i \rho_r [1 - \alpha(T - T_r)], \quad (4.4)$$

while the mass conservation equation remains unchanged:

$$\frac{D\rho_r}{Dt} + \rho_r \frac{\partial u_i}{\partial x_i} = 0.$$

For incompressible flows this implies $\frac{\partial u_i}{\partial x_i} = 0$.

In this work, the Boussinesq approximation will be introduced in all the terms of the momentum equations, not only in the source terms but also in the inertial terms. This is easily performed with the Lagrangian formulations where each particle will change their own density using the Boussinesq approximation. This means:

$$\rho_r [1 - \alpha(T - T_r)] \frac{Du_i}{Dt} = -\frac{\partial}{\partial x_i} p + \frac{\partial}{\partial x_j} \tau_{ij} - g_i \rho_r [1 - \alpha(T - T_r)]. \quad (4.5)$$

In problems for which the dilatation effect is large, the Boussinesq approach may be also introduced in the mass conservation equation as

$$\frac{D\rho}{Dt} + \rho \frac{\partial u_i}{\partial x_i} = \frac{D}{Dt} \{ \rho_r [1 - \alpha(T - T_r)] \} + \rho \frac{\partial u_i}{\partial x_i}$$

$$= -\rho_r \alpha \frac{DT}{Dt} + \rho \frac{\partial u_i}{\partial x_i} = 0, \quad (4.6)$$

which implies:

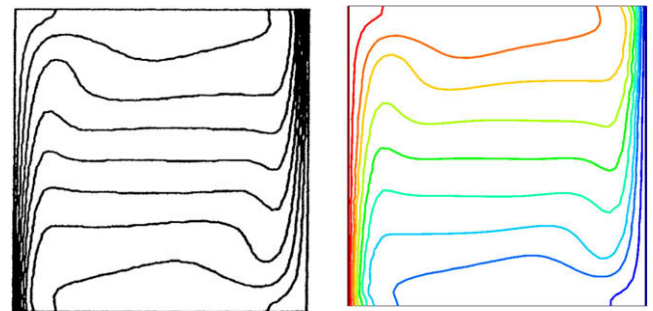


Fig. 6.7. Contour lines of temperature at steady state.

$$\frac{\partial u_i}{\partial x_i} = \frac{\rho_r}{\rho} \alpha \frac{DT}{Dt} = \varepsilon_T. \tag{4.7}$$

In this case, the fluid remains incompressible due to the pressure forces but becomes compressible for temperature dilatation. The temperature volumetric variation ε_T is explicitly introduced as an initial volumetric variation at the beginning of each time step.

The density variation due to the temperature dilatation introduces a change in step B of the fractional-step method described previously. In fact, Eq. (2.19) does not hold anymore as $\rho_{(x)}^{n+1} \neq \rho_{(x)}^n$. It may be replaced by:

$$\frac{D\rho}{Dt} \approx \frac{\rho^{n+1} - \rho^n}{\Delta t} = -\rho^{n+\theta} \frac{\partial u_i}{\partial x_i} = -\rho^{n+\theta} \varepsilon_T. \tag{4.8}$$

Eq. (2.18) for $\theta = 1$ remains:

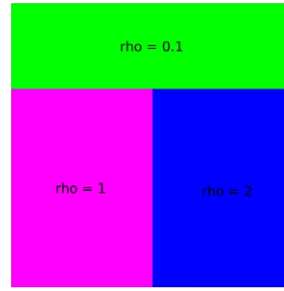
$$\frac{\rho_{(x)}^{n+1}}{\Delta t} \left(\frac{\partial u_i^*}{\partial x_i} - \varepsilon_T \right) = \frac{\partial^2}{\partial x_i^2} (p^{n+1} - \gamma p^n). \tag{4.9}$$

The volumetric temperature strain ε_T may be evaluated explicitly from the previous time step or from the previous iteration using (4.7):

$$\varepsilon_T = \frac{\rho_r}{\rho} \alpha \frac{DT}{Dt} \approx \frac{\alpha(T^{n+1} - T^n)}{[1 - \alpha(T^{n+1} - T_r)]\Delta t}, \tag{4.10}$$

where T^{n+1} is the last temperature evaluated.

It must be noted that the case of an incompressible flow inside a closed recipient may not be computed with this theory.



Fluid A: green $\rho = 0.1 \text{ kg/m}^3$
 Fluid B: purple $\rho = 1 \text{ kg/m}^3$
 Fluid C: blue $\rho = 2 \text{ kg/m}^3$

Fig. 6.9. Fluid densities.

After discretization, this term introduces in equations (B) a new vector \mathbf{E}^{n+1} :

$$\mathbf{E}^{n+1} = \int_V \mathbf{N}_p \left\{ \frac{\rho}{\Delta t} (\varepsilon_T) \right\} dV \tag{4.11}$$

and Eq. (B) remains:

$$(\mathbf{L} + \mathbf{L}_\tau) \mathbf{P}^{n+1} = \frac{1}{\Delta t} (\mathbf{B}_\rho \mathbf{u}^* - \bar{\mathbf{u}}) + \gamma \mathbf{L} \mathbf{P}^n + \mathbf{B}_{\tau\rho} \Pi + \mathbf{E}^{n+1}. \tag{4.12}$$

5. Nodal or element interfaces?

One of the main features of particle methods is that all the physical properties are attached to the nodes instead to the elements. The mesh is permanently updated and hence, it is difficult to keep physical properties at element level. In heterogeneous

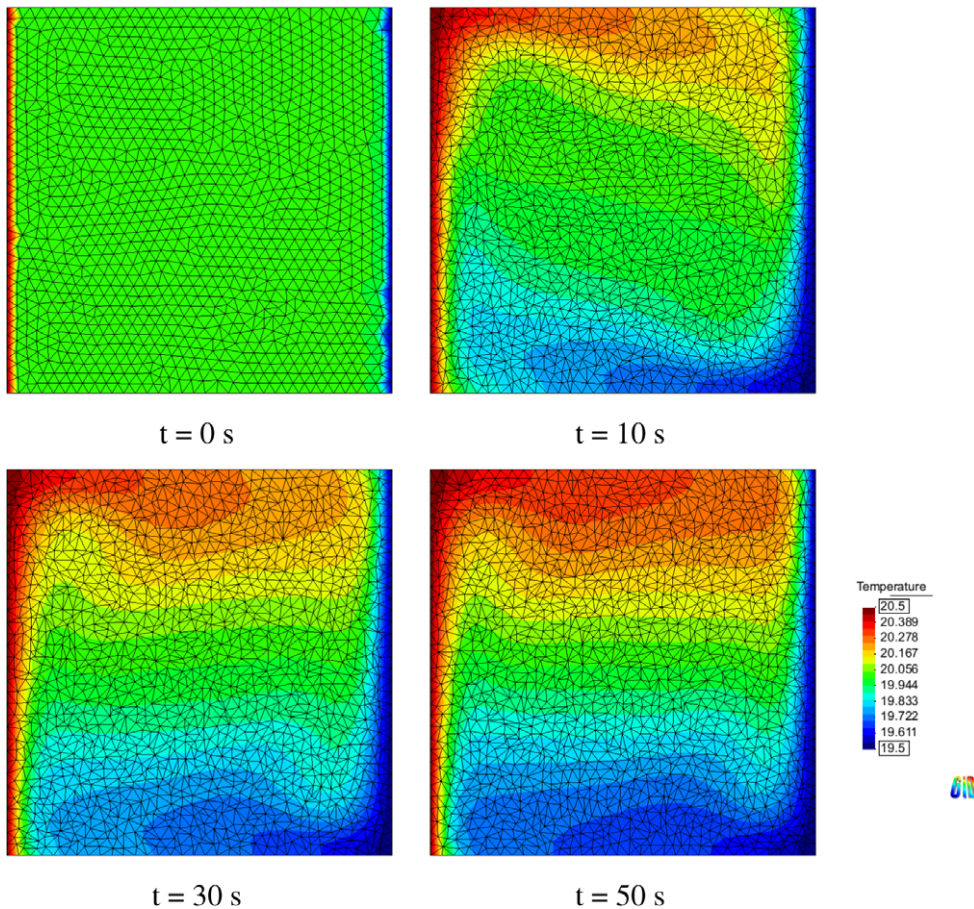


Fig. 6.8. Contour plots of temperature field and mesh at different times.

materials we can have a jump in the material density of several orders of magnitude. We must decide where the internal interface between two different materials occurs. The most *natural* solution for a particle method is to have the interface inside the elements sharing particles with different densities, so that in the element integration points k density takes the mean value $\rho_k = \frac{1}{nv} \sum_{a=1}^{nv} \rho_a$ (where nv is the number of nodes of the element). We will call this possibility as *interface across elements* (Fig. 5.1)

Other possibility is to impose that the interfaces between different materials are between two elements, which will be called *nodal interface* (Fig. 5.2). To achieve this we must accept that elements sharing particles with two different densities have one or the other particular density value. Now the density in the element integration points k takes the value

$$\rho_k = \begin{cases} \rho_1 & \text{if } k \in \text{Fluid1,} \\ \rho_2 & \text{if } k \in \text{Fluid2.} \end{cases}$$

Both possibilities have advantages and disadvantages. Element interfaces are more stable as they do not change much when remeshing is performed. On the other hand, nodal interfaces are more accurate because they allow representing exactly the gradient pressure gap that normally occurs when there is a jump in the density value (see Fig. 5.3).

Two different kinds of problems will be considered:

- (1) problems in which the density varies through the domain in a smooth way. This is the case for instance of thermal variations of the density due to dilatations. In these cases element interface will be used (as in the numerical examples 6.2 and 6.5).
- (2) problems in presence of immiscible different materials where there are jumps in the density properties. In these cases nodal interface will be used (as in numerical examples 6.1, 6.3 and 6.4).

The second case is more difficult to solve numerically because the solution has a discontinuity in the pressure gradients at the

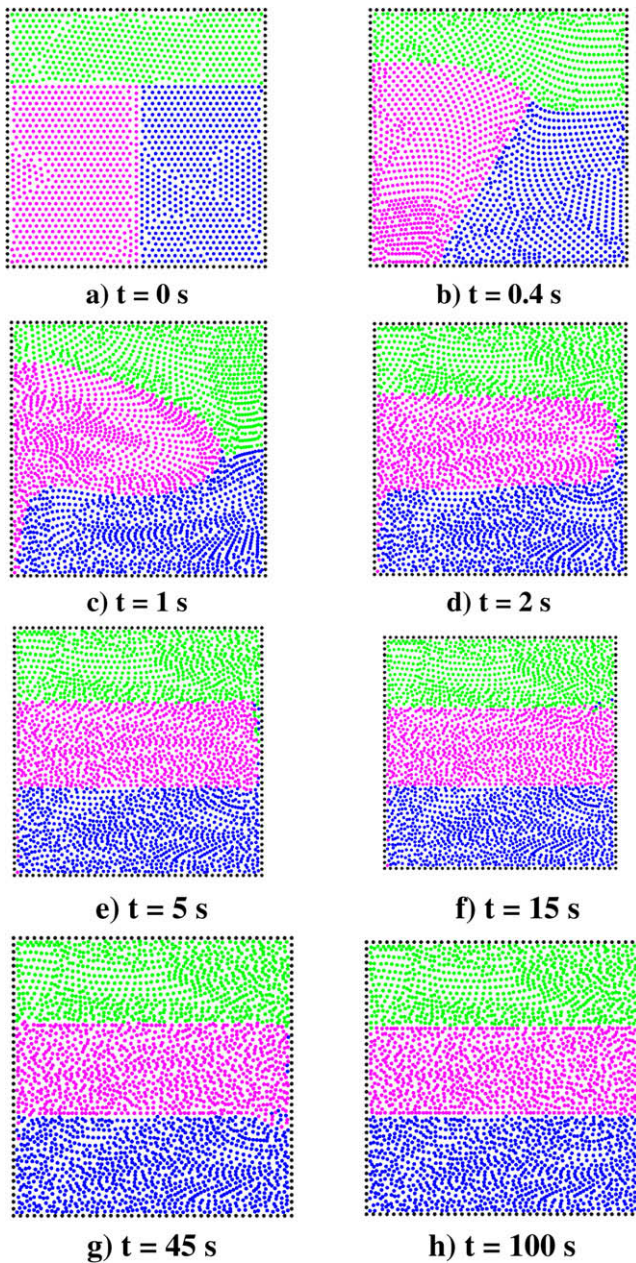


Fig. 6.10. Three different density fluids.

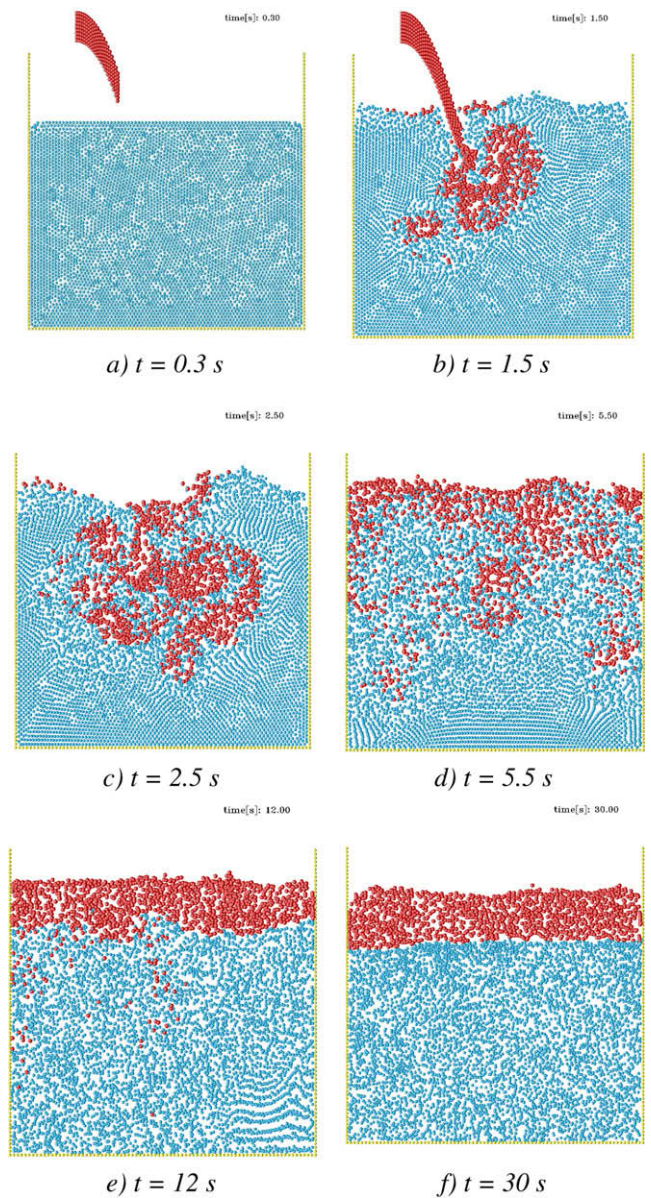


Fig. 6.11. Impinging jet: particle positions.

interface. For this reason immiscible heterogeneous materials will be solved using the nodal interface algorithm in order to be able to represent exactly the pressure gradient jumps. As a consequence of remeshing, the interface position may change suddenly from one time step to the next one introducing small instabilities in the flow. Nevertheless these changes in the interface position between two time steps are of order h . This means that they decrease linearly with the mesh size and are of the same order than the alpha-shape technique used for the determination of the free-surface. In spite of this drawback, nodal interface method improves substantially the results and becomes essential in immiscible fluids.

6. Numerical results

The capabilities of the formulation described above will be shown in several examples with two types of heterogeneities: density jump and continuous variable density. A first example is a sloshing test for which other numerical tests are available. The example is also used to test the stabilization proposed and the ability of the method to deal with large density jumps. In the second example the classical Boussinesq cavity is tested and compared with existing results. Boussinesq cavity is an interesting test for lagrangian formulations because the problem never reaches a sta-

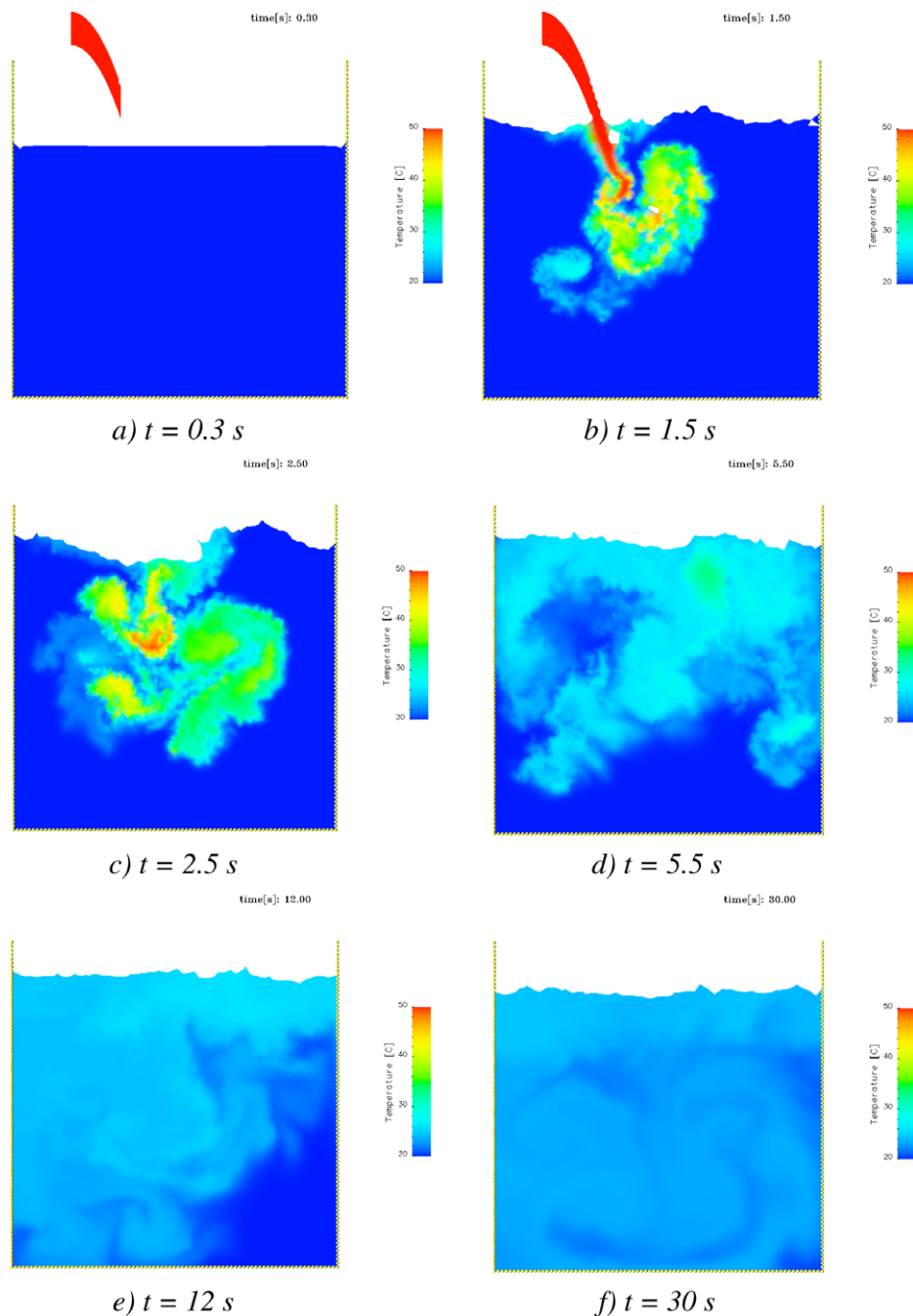


Fig. 6.12. Impinging jet: temperature field.

tionary result for the particle position while the temperature remains at a stationary value. In order to test the ability of the method to carry out with several heterogeneous materials a three density fluid is solved without showing any particular drawback. The fourth example consists in a strong mixing of two immiscible materials. At the beginning, both fluids mix strongly due to the initial kinetic energy. After some time, both fluids separate due to the gravity forces. Finally, the case of two different fluids including thermal dilatation representing a magma chamber is presented.

6.1. Two-fluid sloshing

This numerical test for two-fluid interface flows was first proposed by Tezduyar et al. in [30] and deeply investigated in [31]. It consists in a closed rectangular cavity with two immiscible fluids and small different density. The interface between both fluids is tilted in the initial configuration (Fig. 6.1).

The bottom fluid has a density $\rho = 2 \text{ kg/m}^3$, the top fluid has $\rho = 1 \text{ kg/m}^3$, and the dynamic viscosity is $\mu = 10^{-3} \text{ Pa s}$ for both fluids. Gravity force is set to -0.294 in the vertical direction. Vertical walls are considered free-slip while horizontal walls are no-slip.

This is a typical example where there is a jump in the pressure gradient on the interface between the two materials. This means that it is better to use nodal interfaces to allow the pressure gradients reproduce exactly this possibility.

The free-slip condition is not trivial in Lagrangian methods. We have modelled it assigning $\mu = 0$ in the elements adjacent to the free-slip walls.

Fig. 6.2 shows the interface relative height at the vertical walls (side A and B) for previous works [30,31] compared with PFEM. PFEM results reproduce well both the interface amplitude and oscillation frequency.

When the flow reaches the steady state, the interface height is 0.2998 instead of the correct value of 0.3. Notice that no mass correction algorithm was implemented. Neither was necessary to use enhanced element integration nor shape function enrichment as in interface-capturing methods. The pressure field and the interface discretization are shown in Fig. 6.3.

This example is interesting to test the correct use of the stabilization term of the incompressibility condition for equal order formulation as the presented here. Effectively, as stated before, standard stabilizations based on residual projections like FIC [34] propose a stabilization term based on Eq. (3.22). This stabilization is based on the difference between the pressure gradient and the continuous pressure gradients. Nevertheless, in many cases such as density heterogeneous materials, the pressure gradients are physically discontinuous along the interface. The introduction of a continuous vector π produces an error in the incompressibility

condition along the interface that can be observed in Fig. 6.4. The results shown previously have been obtained using the new stabilization term (Eq. (3.23)) where the density jump for heterogeneous materials has been taken into account in the definition of π .

A very interesting advantage of the present formulation is to allow arbitrary large jumps in density between the materials used. For example, in a water–air sloshing problem the density jump is 1–1000. This kind of problems is normally treated as a single flow, neglecting the air flow. Nevertheless, in many situations the influence of air must be taken into account, especially when bubble production is important and the air incompressibility conditions must be considered. The present method allows carrying out large density jumps without any drawback.

For instance the interface position and fluid velocity for a water–air sloshing problem are shown in Fig. 6.5. The density jump is 1:1000 and the gravity was set to $g = 10 \text{ m/s}^2$. Not even from the jump of pressure gradient nor from the stability algorithm the method has presented any difficulty.

However, this problem defined with the same geometry and initial condition than the previous test, has now a very strong movement of all the internal particles and the interface includes the possibility of breaking waves. Now remeshing is necessary and connectivity changes, in particular near the interface. During remeshing the algorithm to define the interface between the two materials (see Section 5) introduces an error which is of order h (the average distance between two neighbor particles) in the volume definition of each material.

In order to check the volume variation due to the remeshing process, the water mass conservation was measured. Fig. 6.6 shows the water mass for different mesh sizes and fixed time step during a large time period (5 s). We can observe that mesh sizes of order $h = 0.005$ are necessary in order to preserve the mass during all this period at a reasonable level (less than 5% of mass loss).

6.2. Boussinesq cavity

This example deals with one fluid only, but now density changes as a function of temperature: $\rho = \rho_r[1 - \alpha(T - T_r)]$. We have applied the Boussinesq approximation as described in Section 4.

The problem domain is a square cavity with vertical walls at $T = 20.5$ and $19.5 \text{ }^\circ\text{C}$, and adiabatic horizontal walls. Fluid properties are the following: $\rho = 1 \text{ kg/m}^3$, $\mu = 10^{-3} \text{ Pa s}$, $g = 10 \text{ m/s}^2$, $\alpha = 0.1 \text{ }^\circ\text{C}^{-1}$, $k = 10^{-3} \text{ m}^2/\text{s}$, $\Delta T = 1 \text{ }^\circ\text{C}$, $L = 1 \text{ m}$. It gives a Rayleigh number of $Ra = \frac{\rho g \alpha \Delta T L^3}{\mu k} = 10^6$. All walls are considered no-slip.

This is a standard problem used for verification of thermal codes. However we must note that for the Lagrangian formulation it is a difficult test because although the temperature distribution

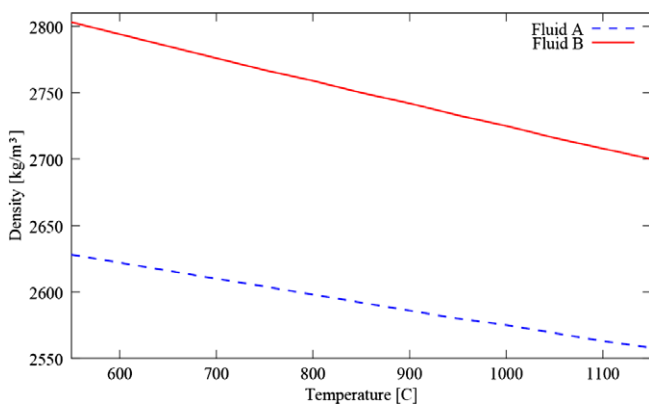


Fig. 6.13. Density variation with temperature.

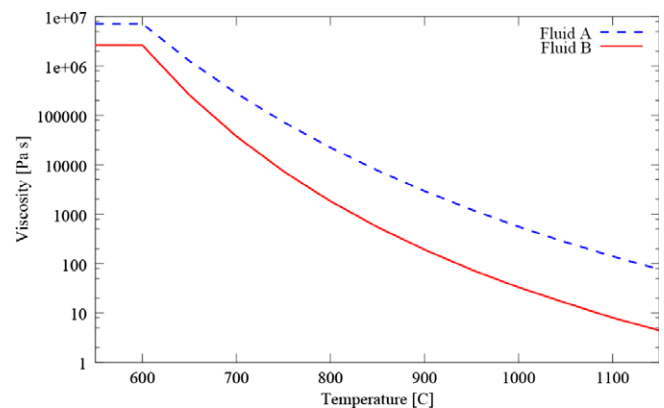


Fig. 6.14. Viscosity variation with temperature.

achieves very quickly a stationary state, the position of the particles is never stationary.

A benchmark solution can be found in [36]. Fig. 6.7 shows a good agreement between ref. [36] and PFEM's solutions. Fig. 6.8 shows the temperature field evolution to the steady state during the first 50 s.

6.3. Heterogeneous flow with several materials

The formulation proposed can be applied with no modification to any arbitrary number of fluids with different densities. In this example we check the behavior with three different fluids with different densities and same viscosity ($\mu = 0.1$ Pa s). The computa-

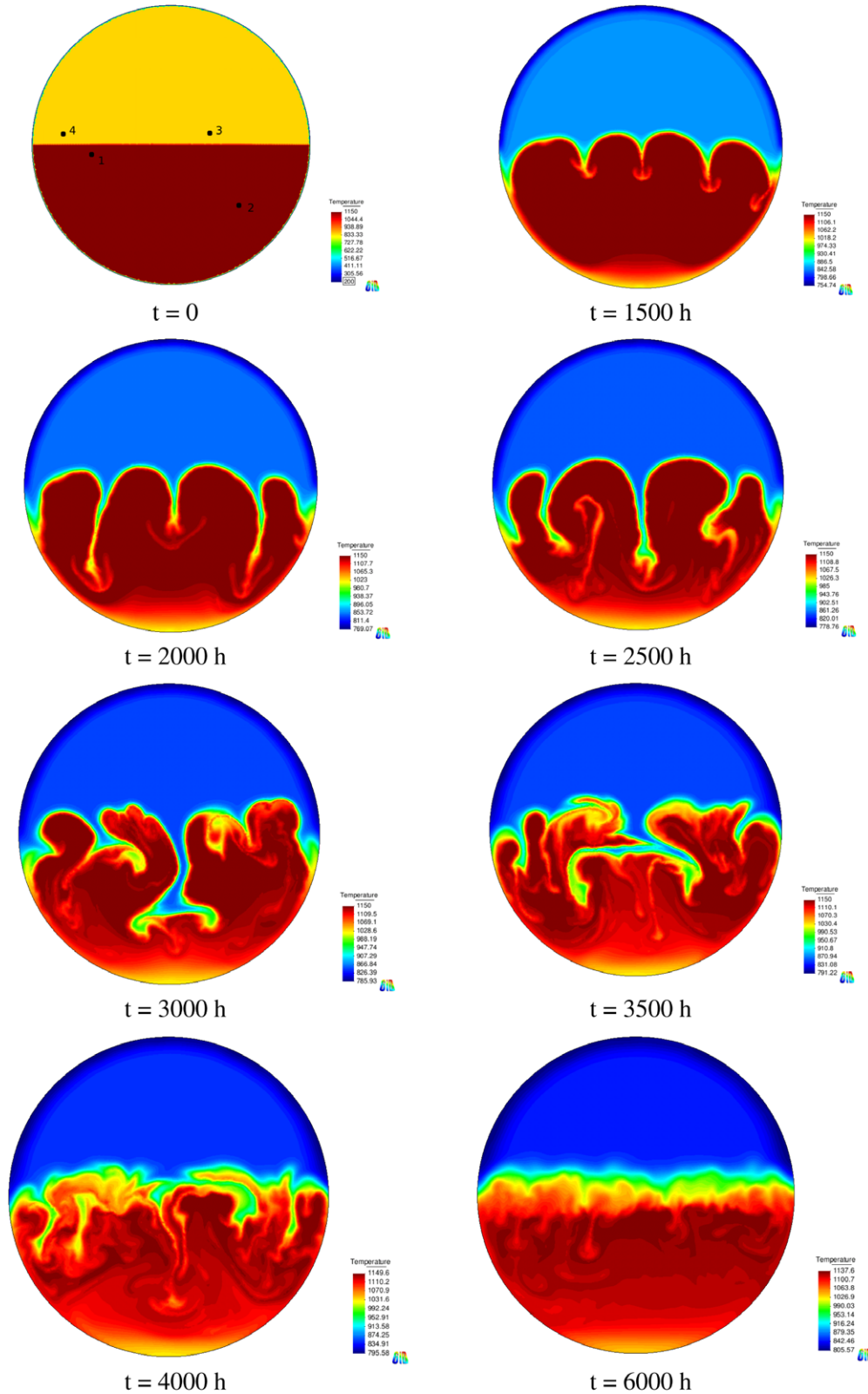


Fig. 6.15. Temperature field at different time steps.

tional domain is a 1×1 m cavity, with free-slip walls and gravity force $g = 10 \text{ m/s}^2$.

In the initial configuration (see Fig. 6.9), fluid A is in a stable position, but fluids B and C will move to find the stable horizontal stratification. Fig. 6.10 shows the particle positions at different time steps. During the first time steps fluid C moves from right to left by the bottom part of the domain while fluid B moves from left to right by the upper part. The movement is so sudden that at time $t = 15$ s some particles of fluid C get trapped inside fluid A on the right hand side. Due to their higher density, they move downwards until they reach fluid C.

There is not any comparison result to test the validity of this example. However the results show clearly the ability of the method to deal with any kind of heterogeneous material.

6.4. Impinging jet

The objective that motivated this work from the beginning is to be able to track the movement of one fluid inside another. We believe that particle methods are especially well suited for this kind of problems, where the fluids fragment and the multiple interfaces become too complex for interface-capturing methods.

In this example a fluid of density $\rho = 800 \text{ kg/m}^3$ and temperature $T = 50 \text{ }^\circ\text{C}$, is poured into a 2×2 m container filled 3/4 with water ($\rho = 1000 \text{ kg/m}^3$) at $T = 20 \text{ }^\circ\text{C}$. Both fluids are immiscible and have the same viscosity, $\mu = 10^{-3} \text{ Pa s}$. Density is considered here as constant and independent of temperature. Thermal diffusivity $k = \kappa/\rho C$ has been set to $k = 10^{-2} \text{ m}^2/\text{s}$ and inflow jet velocity to $v = 1 \text{ m/s}$.

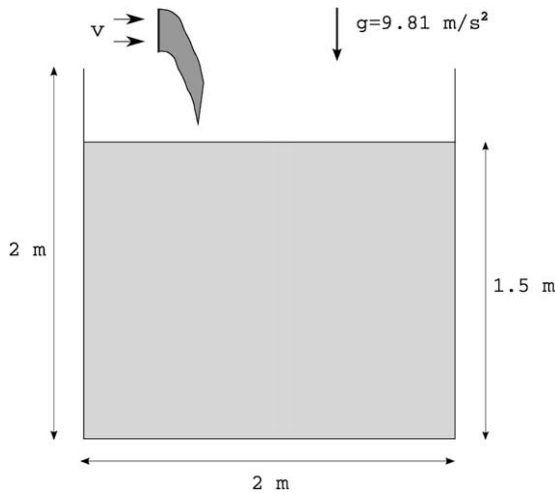


Fig. 6.11 shows how one of the fluids with smaller density penetrates in the other (namely water) with higher density. The jet generates some vorticity and mingling, and at the end fluids separate due to their different densities. The temperature field is also represented in Fig. 6.12.

6.5. Heterogeneous fluids with bouyant thermal effects

The last example represents an application of the method to the study of a magma chamber in which two different fluids with different properties and temperature mix and evolve to a more stable position.

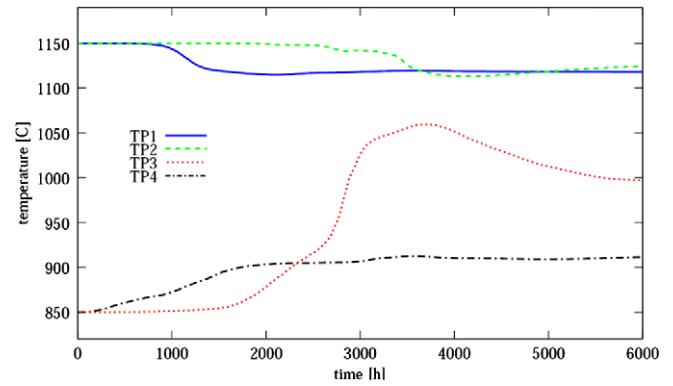


Fig. 6.16. Variation of the temperature in time for four different particles.

Magmas are molten rock below the earth’s surface. Their properties depend mainly on their composition (SiO_2 , Al_2O_3 , FeO , H_2O , etc.) and temperature. Typically at the beginning the chamber is filled with one fluid only (fluid A) that cools due to the lower temperature of the surrounding crustal rock. The decrease in temperature produces that some chemical components start to crystallize and deposit at the bottom. The crystallization process changes the density and the composition of the remaining melt. At some instant new magma (fluid B) enters the chamber. It is hotter, denser and less viscous than fluid A. The difference in temperature and properties between both fluids triggers the convective transport, and A and B start to mingle. Molecular diffusion is of order $10^{-10} \text{ m}^2/\text{s}$ and can be neglected. Fluids are considered as immiscible.

We assume in our simulation that the chamber (of radius 680 m) contains 50% of each fluid and they are at rest at $t = 0$. Initial temperature of fluid A is $850 \text{ }^\circ\text{C}$ and of fluid B, $1150 \text{ }^\circ\text{C}$. Walls are considered no-slip, adiabatic and initially at $200 \text{ }^\circ\text{C}$. Gravity takes the usual value 9.81 m/s^2 .

Magma density can be linearized as $\rho = \rho_r[1 - \alpha(T - T_r)]$ with [37]

Fluid A	Fluid B
$T_r = 1000$	$T_r = 1000$
$\rho_0 = 2575$	$\rho_0 = 2725$
$\alpha = 4.52\text{e-}5$	$\alpha = 6.34\text{e-}5$

Viscosity is modelled as $\log \mu = A + \frac{B}{T-C}$ [38] with

Fluid A	Fluid B
$A = -4.55$	$A = -4.55$
$B = 8106.06$	$B = 5432$
$C = 162.34$	$C = 377.94$

At temperature lower than $600 \text{ }^\circ\text{C}$ viscosity is so high that magma would behave as solidified. However, in this temperature range we model magma as a fluid with constant viscosity (see Figs. 6.13 and 6.14).

Thermal diffusivity is set to $k = 10^{-4} \text{ m}^2/\text{s}$ constant (two orders of magnitude higher than realistic values).

Although our formulation is not complete for variable viscosity terms (see discussion on Section 2, Eqs. (2.6) and (2.7)), numerical results are quite reasonable: fluid B cools rapidly at the material interface and chamber boundary (see Fig. 6.15). Its density increases and moves downwards due to gravity and low viscosity. This movement drags fluid A downwards too, but fluid A is less

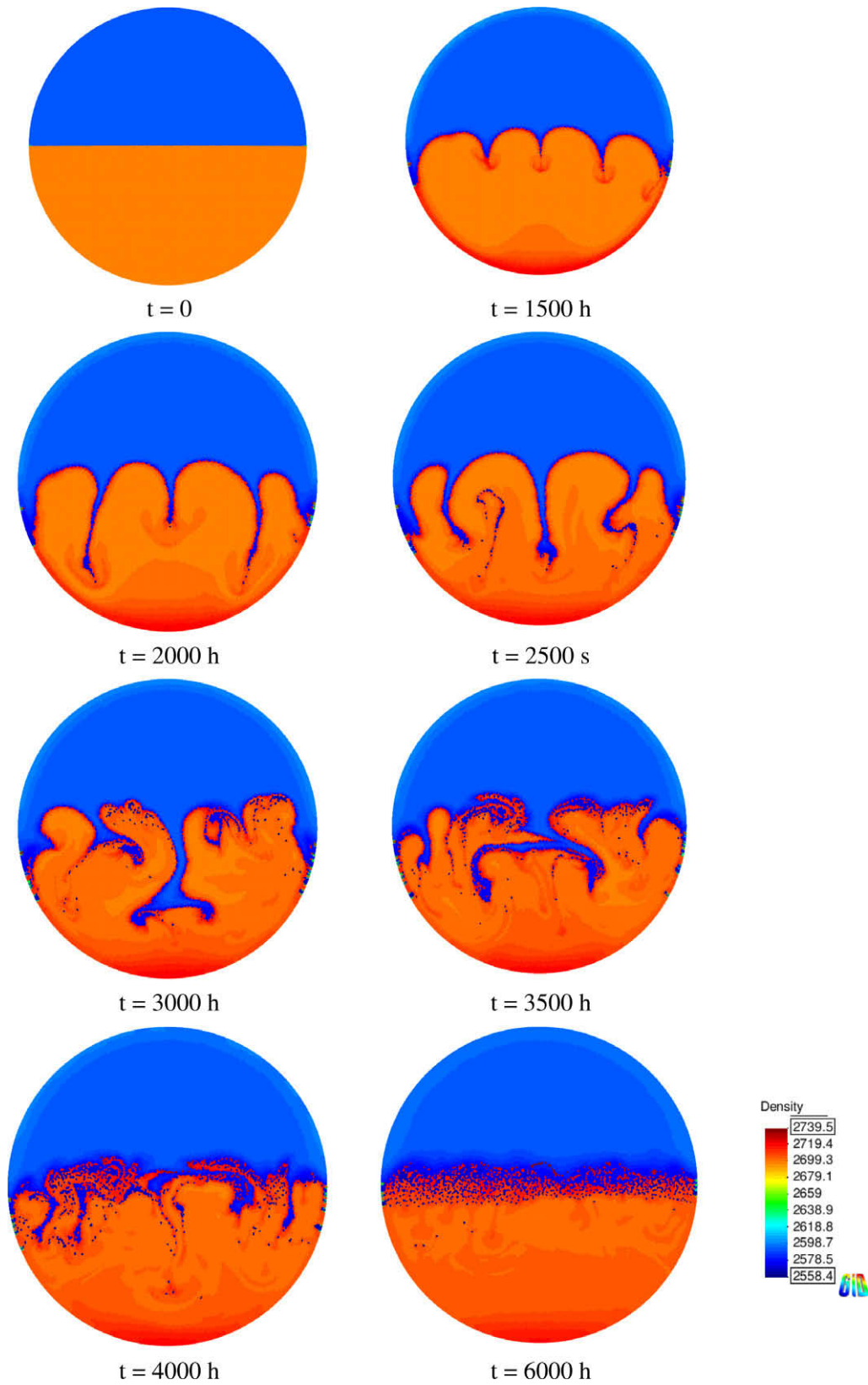


Fig. 6.17. Density field at different time steps.

dense than B and pulls upwards. Convection produces fragmentation and fluid mixing. The evolution of the mixing of the two mate-

rials may be seen in Fig. 6.17 representing the density field at different time steps.

We have tracked also the temperature of four particles along the simulation. The initial positions of the four particles can be seen in Fig. 6.15 at $t = 0$. Their temperature history with time is shown in Fig. 6.16.

The capability of the Lagrangian formulation to track any physical parameter and the ability of particle methods to handle this kind of problems is demonstrated in this example.

7. Conclusions

The Particle Finite Element Method (PFEM) has been used to solve the incompressible Navier–Stokes equations for heterogeneous fluid flows. The results show the ability of the method to deal with problems from the simple case of fluids with a single interface to the case of strong mixed fluids with multiple interfaces. Problems with a big difference between the two materials were also performed without showing any instability.

Many of the previous works for incompressible Navier–Stokes equations have been thought for homogeneous materials. For heterogeneous fluids there are two main differences: one is the possibility to have a discontinuity on the pressure field. The other drawback is to have discontinuous pressure gradients. The first case appears when there is a jump in the viscosity parameter (viscous heterogeneous flows). The second case is typical when there is a density jump (density heterogeneous flows). In this paper we do not treat the first one (we solve problems with viscous variation but without a jump in the viscosity). For the case of density jumps however, we conclude that standard stabilized methods such as pressure projection methods are not accurate because they all work with a continuous pressure gradient projection. To include a discontinuous pressure gradient projection is fundamental to achieve acceptable results.

Finally we conclude that in such cases where there is a jump in the density properties it is essential to force the interface to match the nodes (nodal interfaces). Otherwise very poor approximations are obtained and cases with large density jumps cannot be solved. The disadvantage to use nodal interfaces is the lack of mass preservation during the remeshing process. Nevertheless this mass variation is order h and may be reduced decreasing the mesh size.

Acknowledgements

M. Mier-Torrecilla thanks the Catalan Agency for Administration of University and Research Grants (AGAUR), the European Social Fund and CIMNE for their support. She also thanks F. Costa, A. Geyer and J. Martí from the Volcanology Group at the Spanish National Research Council (CSIC) for their useful suggestions and comments.

References

- [1] J. Donea, A. Huerta, *Finite Element Method for Flow Problems*, John Wiley, 2003.
- [2] J.J. Monaghan, An introduction to SPH, *Comput. Phys. Commun.* 48 (1988) 89–96.
- [3] R.A. Gingold, J.J. Monaghan, Kernel estimates as a basis for general particle methods in hydrodynamics, *J. Comput. Phys.* 46 (1981) 429–453.
- [4] R.A. Gingold, J.J. Monaghan, Smoothed particle hydrodynamics, theory and application to non-spherical stars, *Mon. Not Roy. Astron. Soc.* 181 (1997) 375–389.
- [5] W.K. Liu, S. Jun, Y.F. Zhang, Reproducing Kernel particle methods, *Int. J. Numer. Methods Fluids* 20 (1995) 1081–1106.
- [6] S. Koshizuka, H. Ikeda, Y. Oka, Numerical analysis of fragmentation mechanisms in vapor explosions, *Nucl. Engrg. Design* 189 (1999) 423–433.
- [7] N. Shirakawa, H. Horie, Y. Yamamoto, Y. Okano, A. Yamaguchi, Analysis of jet flows with the two-fluid particle interaction method, *J. Nucl. Sci. Technol.* 38 (2001) 729–738.
- [8] X.Y. Hu, N.A. Adams, A multi-phase SPH method for macroscopic and mesoscopic flows, *J. Comput. Phys.* 213 (2006) 844–861.
- [9] B. Nayroles, G. Touzot, P. Villon, Generalizing the FEM: diffuse approximation and diffuse elements, *Comput. Mech.* 10 (1992) 307–318.
- [10] T. Belytschko, Y. Lu, L. Gu, Element free Galerkin methods, *Int. J. Numer. Methods Engrg.* 37 (1994) 229–256.
- [11] E. Oñate, S.R. Idelsohn, O.C. Zienkiewicz, A finite point method in computational mechanics. Applications to convective transport and fluid flow, *Int. J. Numer. Methods Engrg.* 39 (1996) 3839–3866.
- [12] E. Oñate, S.R. Idelsohn, O.C. Zienkiewicz, R.L. Taylor, C. Sacco, A stabilized finite point method for analysis of fluid mechanics problems, *Comput. Methods Appl. Mech. Engrg.* 139 (1–4) (1996) 315–346.
- [13] E. Oñate, C. Sacco, S.R. Idelsohn, A finite point method for incompressible flow problems, *Comput. Visual. Sci.* 2 (2000) 67–75.
- [14] S.R. Idelsohn, E. Oñate, N. Calvo, F. Del Pin, The meshless finite element method, *Int. J. Numer. Methods Engrg.* 58 (6) (2003) 893–912.
- [15] S.R. Idelsohn, N. Calvo, E. Oñate, Polyhedrization of an arbitrary point set, *Comput. Methods Appl. Mech. Engrg.* 192 (22–24) (2003) 2649–2668.
- [16] S.R. Idelsohn, E. Oñate, F. Del Pin, The particle finite element method: a powerful tool to solve incompressible flows with free-surfaces and breaking waves, *Int. J. Numer. Methods Engrg.* 61 (7) (2004) 964–984.
- [17] E. Oñate, S.R. Idelsohn, F. Del Pin, R. Aubry, The particle finite element method. An overview, *Int. J. Comput. Methods* 2 (1) (2004) 267–307.
- [18] E. Oñate, J. Garcia, S. Idelsohn, Ship hydrodynamics, in: E. Stein, R. De Borst, T.J.R. Hughes (Eds.), *Encyclopedia of Computational Mechanics*, J. Wiley, 2004.
- [19] S.R. Idelsohn, E. Oñate, F. Del Pin, N. Calvo, Lagrangian formulation: the only way to solve some free-surface fluid mechanics problems, in: H.A. Mang, F.G. Rammerstorfer, J. Eberhardsteiner (Eds.), *Fifth World Congress on Computational Mechanics*, Vienna, Austria, 2002 July 7–12.
- [20] A. Larese, R. Rossi, E. Oñate, S.R. Idelsohn, Validation of the particle finite element method (PFEM) for simulation of the free-surface flows, *Engrg. Comput.* 25 (4) (2008) 385–425.
- [21] S.R. Idelsohn, E. Oñate, F. Del Pin, N. Calvo, Fluid–structure interaction using the particle finite element method, *Comput. Methods Appl. Mech. Engrg.* 195 (2006) 2100–2113.
- [22] E. Oñate, S.R. Idelsohn, M.A. Celigueta, R. Rossi, Advances in the particle finite element method for the analysis of fluid–multibody interaction and bed erosion in free-surface flows, *Comput. Methods Appl. Mech. Engrg.* 197 (2008) 1777–1800.
- [23] S.R. Idelsohn, J. Martí, A. Limache, E. Oñate, Unified Lagrangian formulation for elastic solids and incompressible fluids. Application to fluid–structure interaction problems via the PFEM, *Comput. Method Appl. Mech. Engrg.* 197 (2008) 1762–1776.
- [24] E. Oñate, J. Garcia, S. Idelsohn, F. Del Pin, Finite calculus formulations for finite element analysis of incompressible flows: Eulerian, ALE and Lagrangian approaches, *Comput. Method Appl. Mech. Engrg.* 195 (2006) 3001–3037.
- [25] D. Gray, A. Giorgini, The validity of the Boussinesq approximation for liquids and gases, *Int. J. Heat Mass Transfer* 19 (1976) 545–551.
- [26] A. Trykozko, W. Zijl, Complementary finite element methods applied to the numerical homogenization of 3D absolute permeability, *Commun. Numer. Methods Engrg.* 18 (n°1) (2002) 31–41.
- [27] J.A. Sethian, Evolution, implementation, and application of level set and fast marching methods for advancing fronts, *J. Comput. Phys.* 169 (2001) 503–555.
- [28] S.J. Osher, R.P. Fedkiw, Level set methods: an overview and some recent results, *J. Comput. Phys.* 169 (2001) 463–502.
- [29] J.-L. Guermond, L. Quartapelle, A projection FEM for variable density incompressible flows, *J. Comput. Phys.* 165 (2000) 167–188.
- [30] T.E. Tezduyar, M. Behr, S. Mittal, 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, *Comput. Methods Appl. Mech. Engrg.* 94 (1992) 353–371.
- [31] M. Cruchaga, D. Celentano, T. Tezduyar, A moving Lagrangian interface technique for flow computations over fixed meshes, *Comput. Methods Appl. Mech. Engrg.* 191 (2001) 525–543.
- [32] A. Limache, S.R. Idelsohn, R. Rossi, E. Oñate, The violation of objectivity in Laplace formulations of the Navier–Stokes equations, *Int. J. Numer. Methods Fluids* 54 (6–8) (2007) 639–664.
- [33] R. Codina, Pressure stability in fractional step finite element methods for incompressible flows, *J. Comput. Phys.* 170 (2001) 112–140.
- [34] E. Oñate, A stabilized finite element method for incompressible viscous flows using a finite increment calculus formulation, *Comput. Methods Appl. Mech. Engrg.* 182 (1–2) (2000) 355–370.
- [35] E. Oñate, Derivation of stabilized equations for numerical solution of advective–diffusive transport and fluid flow problems, *Comput. Methods Appl. Mech. Engrg.* 151 (1998) 233–265.
- [36] G. De Vahl Davis, Natural convection of air in a square cavity. A benchmark numerical solution, *Int. J. Numer. Methods Fluids* 3 (1983) 249–264.
- [37] R.A. Lange, I.S.E. Carmichael, Thermodynamic properties of silicate liquids with an emphasis on density, thermal expansion and compressibility, *Rev. Mineral. Geochem.* 24 (1990) 25–64.
- [38] D. Giordano, J.K. Russell, D.B. Dingwell, Viscosity of magmatic liquids: a model, *Earth Planet. Sci. Lett.* (2008).