INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids **28**: 719–736 (1998)

LOCAL MESH ADAPTATION TECHNIQUE FOR FRONT TRACKING PROBLEMS

N. LOCK*, M. JAEGER, M. MEDALE AND R. OCCELLI

Institut Universitaire des Systèmes Thermiques Industriels, UMR C.N.R.S. 139, Technopole de Château-Gombert, 5 rue Enrico Fermi, 13453 Marseille Cedex 13, France

SUMMARY

A numerical model is developed for the simulation of moving interfaces in viscous incompressible flows. The model is based on the finite element method with a pseudo-concentration technique to track the front. Since a Eulerian approach is chosen, the interface is advected by the flow through a fixed mesh. Therefore, material discontinuity across the interface cannot be described accurately. To remedy this problem, the model has been supplemented with a local mesh adaptation technique. This latter consists in updating the mesh at each time step to the interface position, such that element boundaries lie along the front. It has been implemented for unstructured triangular finite element meshes. The outcome of this technique is that it allows an accurate treatment of material discontinuity across the interface and, if necessary, a modelling of interface phenomena such as surface tension by using specific boundary elements. For illustration, two examples are computed and presented in this paper: the broken dam problem and the Rayleigh–Taylor instability. Good agreement has been obtained in the comparison of the numerical results with theory or available experimental data. © 1998 John Wiley & Sons, Ltd.

KEY WORDS: front tracking; mesh adaptation; finite element method

1. INTRODUCTION

The study of the problem of moving interfaces is one of the most important areas in technological and engineering fields. Its application domain comprises solid mechanics (material shaping processes) and fluid mechanics (break-up of a jet, coalescence of liquid drops, wave breaking, etc.). However, from a numerical modelling viewpoint, different approaches have been developed to treat these two classes of problems. For material shaping problems, the finite element method combined with a Lagrangian description of the movement is generally used. On the other hand, the most suitable kinematic description for the study of fluid–structure interactions is a mixed one, Lagrangian in solids and Eulerian in fluids, with a coupling at the interface. Finally, the mixed description or the Eulerian one is usually employed for the treatment of flows with fluid–fluid interfaces. This last approach appears to be the one that can handle complex interface problems. State of the art numerical methods can be found in Hyman [1], Laskey *et al.* [2] and Floryan and Rasmussen [3].

In the present study, we are concerned with the last category of problems, in which particularly complex shaped interfaces can be found. The difficulties in the treatment of such

* Correspondence to: Institut Universitaire des Systèmes Thermiques Industriels, UMR C.N.R.S. 139, Technopole de Château-Gombert, 5 rue Enrico Fermi, 13453 Marseille Cedex 13, France.

CCC 0271–2091/98/140719–18$17.50

Received 22 July 1996
Accepted 29 March 1997

flows can be attributed to (i) the interface location (ii) the interface topology and (iii) the interfacial mechanisms. Most of the developed models use the Eulerian description of the flow and various techniques have been employed to track the interfaces through the fixed mesh. The two well-known basic methods are surface tracking and volume tracking methods.

In surface tracking methods, markers are initially located on the interface and are subsequently followed within the flow. However, interfacial mechanisms such as coalescence cannot be easily treated with these techniques. Moreover, Osher and Sethian [4] quoted another drawback of the marker-based method, when the curvature effect has to be considered. For large and complex motion, particles come together in regions of high curvature causing numerical instability. A regridding mechanism must then be employed, but it usually contains a diffusion-like error, which dominates the real effects of curvature. To overcome these difficulties Osher and Sethian developed a class of algorithms (PSC schemes [4]) based on numerically solving Hamilton–Jacobi equations.

In the volume tracking methods, the interface is implicitly tracked. Markers are used to identify the fluids. These markers can be, e.g. one of the fluid properties, or another function. For each cell of the mesh, the fluid which is present is found. The interface is located somewhere inside cells that contain more than one fluid and it is built cell by cell. The two commonly used methods are the MAC [5,6] (marker-and-cell) and VOF [7] (volume of fluid) methods. The MAC technique uses massless particles as markers. The main problem with this method is that the particles may accumulate in some zones of the mesh, leading to an inaccuracy in locating the interface. The VOF technique defines a marker function F which represents the fraction of a cell volume occupied by one of the fluids. For a given cell, if F is zero or unity, the cell is considered to be either empty, or filled with the fluid respectively. If F is between zero and unity, the cell is an interface cell. The VOF technique was developed by Hirt and Nichols for finite difference structured meshes. The determination of the F value for a cell uses the donor–acceptor flux approximation of Ramshaw and Trapp [8] which requires rectangular cells. The advantage of these volume tracking methods is that they can handle interfaces subject to large deformations and treat multiple fronts.

The extension of the volume tracking approach based on a marker function to finite element unstructured meshes was initially introduced by Thompson [9]. For this purpose, he introduced a marker function, named pseudo-concentration, designed to be advected by a standard finite element computation. The main feature of this function is that it is continuous on the whole domain and therefore can be accurately represented by finite element interpolation. Furthermore, this function must remain single-valued during the whole computation, and a specified value is assigned to the interface. The latter is then identified by finding the contour lines corresponding to this specified value. Initially, Thompson [9] applied his method to creeping flows. Later Dhatt *et al.* [10] and then Usmani *et al.* [11] and Lewis *et al.* [12] extended the method to Navier–Stokes flows with special interest in mould casting. In the last 2 years extensions of the VOF method to unstructured meshes have also been proposed. Such extensions can be found in Wang and Wang [13] and in FIDAP [14] (since version 7.5). In the former the fluid flow is computed with the control volume finite element method of Baliga and Patankar [15], whereas in the second case, the standard finite element method is used. Nevertheless, both track the front with a VOF based method.

The main drawback of a Eulerian description for moving interface flows is that the material discontinuity across the interface is poorly described because the front crosses elements. To remedy this problem, methods based on a local and temporary mesh adaptation to the interface position have been developed by authors such as Glimm *et al.*[16], Sato and Richardson [17] and Mashayek and Ashgriz [18]. In the work of Glimm *et al.* [16], the interface is modelled by nodes which are connected by linear pieces called bonds and is tracked in a Lagrangian manner. Mashayek and Ashgriz [18] used the VOF method combined with an element generation technique along the interface. The front tracking strategy employed by Sato and Richardson [17] is a combination of both, the front being advected in a Lagrangian way and smoothed according to the volume fraction criteria. However, all the proposed mesh adaptation strategies are based on structured meshes. Mesh cells at the beginning of the computation have a rectangular shape. The wet cells (neither full nor empty) are adapted so that one of their boundaries coincides with the interface after adaptation. Triangular elements are usually used for this purpose, although Mashayek and Ashgriz [18] employed degenerated quadrilateral elements.

The finite element model developed in this study uses the idea of pseudo-concentration function to track the interface. It has been presented in previous works and applied to the simulation of mould casting problem [19,20]. It has also been used to study the corium discharge from a molten core of a nuclear reactor [21]. In the present work, it is supplemented by a mesh adaptation algorithm for general triangular finite element meshes. Therefore, the new numerical model is able to accurately describe material discontinuity across the interface and can also take into account interfacial phenomena such as surface tension, radiation, etc. Also, to deal with the pressure discontinuity at the interface, a triangular finite element based on discontinuous pressure approximation has been chosen to compute the flow field. As in our previous works [19–21], this element is employed here with a penalty formulation of the governing flow equations using primitive variables. In Section 2, the moving interface model is presented. The governing equations for a two-fluid system are shown in Section 3 and the finite element model is described in Section 4. The efficiency of the present approach is illustrated by two numerical examples in Section 5. Concluding remarks are made in Section 6.

2. MOVING INTERFACE MODEL

2.1. Volume tracking model

As previously mentioned in the introduction, the model developed in this study is a volume tracking type in the context of finite element approach. The notion of pseudo-concentration function is used to describe the motion of the interface. This function F is advected by the velocity field V through the fixed finite element mesh according to the following hyperbolic equation:

$$\frac{\partial F}{\partial t} + \mathbf{V} \cdot \nabla F = 0. \qquad (1)$$

This relation is a pure advection equation of first order. Boundary values for F may then be specified only along the part of the boundary where the fluid enters the domain. An initial linear F-field must also be defined according to the initial interface position. This field is built in such a way that:

$$F = Fc \quad \text{at the interface;} \qquad (2)$$

$$F < Fc \quad \text{in zones filled with fluid 1;} \qquad (3)$$

$$F > Fc \quad \text{in zones filled with fluid 2;} \qquad (4)$$

where Fc is the F-value associated with the interface.

Instead of defining a typical piece-wise marker function, e.g. density, the pseudo-concentration function F is chosen here as being continuous and monotonic. This choice comes from the fact that the finite element method, which is employed to solve Equation (1), cannot handle the advection of discontinuous function. Moreover the numerical scheme used to advect the function F must keep the property of monotonicity. Otherwise numerical artificial interfaces may appear during the computation. An analysis of this problem in the case of a Euler time marching scheme can be found in Medale and Jaeger [20].

At each time step, once the velocity field has been computed, the solution of Equation (1) yields the new F-field. The new interface position is then identified at the points where F takes the prescribed value Fc. From this update front position, the elements of the mesh can then be divided into three groups as follows:

- elements filled with fluid 1 ($F < Fc$);
- elements filled with fluid 2 ($F > Fc$);
- elements crossed by the interface, called transition elements.

2.2. Local mesh adaptation technique

In the finite element method, fluid properties such as density, viscosity, etc. are specified on each node and interpolated throughout the mesh or assigned to each element. Material discontinuity can be only be accurately taken into account in the last case and only if the front lies along element boundaries. In the case of a Eulerian approach, this front moves through a fixed mesh. Physical properties assigned to transition elements are then fixed to an average value calculated from those of fluids present in the element. Therefore, the discontinuity is not really taken into account because it is smeared over a finite region of the order of elements size. The coarser the mesh is, the more the errors are significant.

An additional difficulty arises with a two-fluid system of high density ratio. This is shown in Figure 1, which corresponds to the case of a stable stratification of the two-fluid system when the lightest fluid lies over the heaviest one. In Figure 1(a), the horizontal interface crosses two neighbouring triangular elements. Since these two elements have inverted orientation (the base is down on the left, whereas on the right it is up), they have two different filling rates and thus different averaged density. Under gravitational action this leads to the generation of artificial surface waves which must be dumped down. This was obtained in our previous work [20] using an augmented viscosity for the crossed elements. However, this further raises the degree of empiricism of the model. The same analysis can be driven in the situation depicted in Figure 1(b), where the horizontal interface crosses an inclined structured mesh made up with rectangular elements.

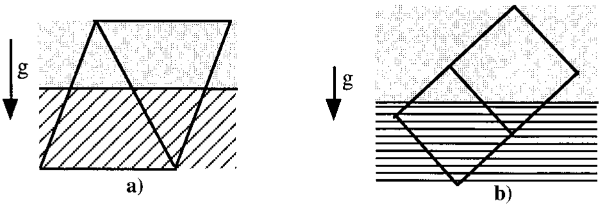


Figure 1. Particular configurations where the average properties based on the filling rate yield numerical instability.

The second drawback is that interfacial phenomena such as surface tension action or heat transfer by radiation can only be modelled by body forces distributed in transition elements (see Unverdi and Tryggvason [22], Brackbill *et al.* [23]). Again, the error is proportional to the element size. Nevertheless, the results obtained by this approach are satisfying, providing the treatment of the interface does not have to be very accurate. Several examples can be found in References [10–12,19,20].

When dealing with problems where an accurate interface description is required, two solutions exist. The first one consists of using very fine meshes so that error does not exceed an acceptable value. This approach has been employed by other authors for a direct simulation of very complex interfacial mechanisms, such as surface breaking [24]. However such a strategy needs a high performance computer.

The other alternative, which we have adopted, is an update of the mesh at each time step. For this purpose, a local mesh adaptation technique has been developed. In our Eulerian model, the interface tracking is always performed on the same mesh, called reference mesh. This is composed of linear three-node triangular elements and can be completely unstructured. The local mesh adaptation technique consists of dividing the transition elements so that the generated element boundaries lie along the front. By doing so, the reference mesh is locally adapted to the front position and a temporary mesh is built from the reference one. This temporary mesh is used for solving flow equations or any other governing equation of a given physical problem (e.g. energy equation, to study heat transfer mechanism). It allows us to take into account the fluid physical properties discontinuity across the interface since transition elements do not exist any more. Another advantage of the developed technique is that interfacial phenomena can be treated accurately by using specific boundary elements at the interface.

Since the finite element used for solving the interface tracking problem is a linear one, the cutting out of the transition elements is obvious. In fact, since the F function is piece-wise linear on the reference mesh, the Fc-contour line that identifies the interface is a line segment inside each transition element. Therefore, these elements are divided into two parts which have triangular and trapezoid shapes. By adding a new node at each of the intersection points of the front line with the edges of the element, this latter can be divided into three triangular elements as shown in Figure 2(a).

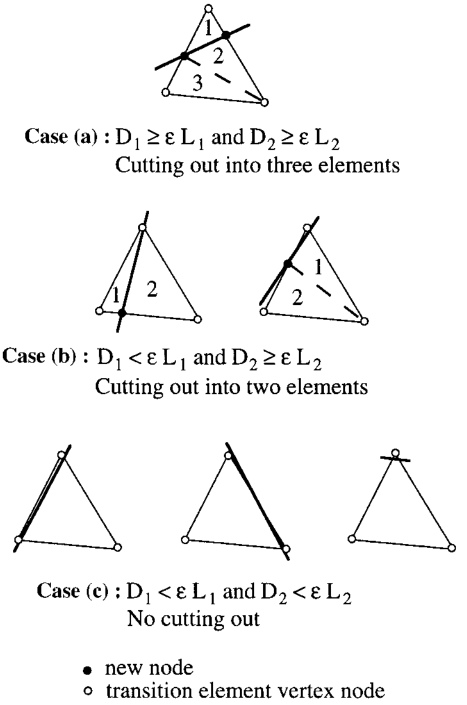
In this cutting out procedure, caution must be taken not to generate elements which are too small. This situation is encountered each time the front line intersects an element edge very close to one of its vertex nodes. To overcome this difficulty, a cutting out criterion has been introduced. For each element edge crossed by the interface, the distance D between the intersection point and the nearest vertex node is compared with the length L of the edge. A new node is generated at the intersection point only if the following condition is satisfied:

$$D \geq \varepsilon L \tag{5}$$

where ε is a real number: $0 < \varepsilon < 1$.

In Figure 2(b), Equation (5) is checked for only one of the intersected edges. Therefore, only one new node is generated and the corresponding transition element is transformed into two elements. As for Figure 2(c), Equation (5) is violated for both intersected edges. Then, no new node is created and the transition element is not divided. In both cases, the interface position is slightly altered regarding the temporary mesh. However, the introduced error is not significant since the F-field itself is not modified at this step of the computation.

In order to clarify the connection between the various steps of the computation, an illustration of the complete solution procedure is given in Figure 3. The initial position of the



Case (a) : $D_1 \geq \varepsilon L_1$ and $D_2 \geq \varepsilon L_2$
Cutting out into three elements

Case (b) : $D_1 < \varepsilon L_1$ and $D_2 \geq \varepsilon L_2$
Cutting out into two elements

Case (c) : $D_1 < \varepsilon L_1$ and $D_2 < \varepsilon L_2$
No cutting out

- ● new node
- ○ transition element vertex node

Figure 2. Classification and cutting out of transition elements.

interface in the reference mesh is shown in Figure 3(a). From this information, the initial F-field is reconstructed by assigning a value of F to every node of the reference mesh (Figure 3(b)) according to Equations (2), (3) and (4). Then, by implementing the local mesh adaptation technique, transition elements are divided and a temporary mesh is generated as shown in Figure 3(c). After the flow equations have been solved by finite element method on the temporary mesh, the computed nodal velocities (Figure 3(d)) are projected onto the reference mesh as seen in Figure 3(e). In the next step, using the obtained velocity values, the new F-field is determined by solving F advection equation on the reference mesh. This F-field is then employed to identify the new interface position by plotting the contour line Fc as depicted in Figure 3(f). This will complete the sequences used in the first time step and to carry on the computation, the steps of Figure 3(c–f) will be repeated for each time step.

3. GOVERNING EQUATIONS FOR THE TWO-FLUID SYSTEM

3.1. Flow equations

The flow considered here is that of a system composed of two viscous incompressible and non-miscible fluids as illustrated in Figure 4. The domain of interest is Ω, with Ω_1 and Ω_2 the sub-domains associated with fluid 1 and 2 respectively. Γ is the boundary of Ω with Γ_1 and Γ_2, the parts of Γ induced by Ω_1 and Ω_2. Finally, Γ_{12} represents the interface separating fluid 1 from fluid 2 in Ω.

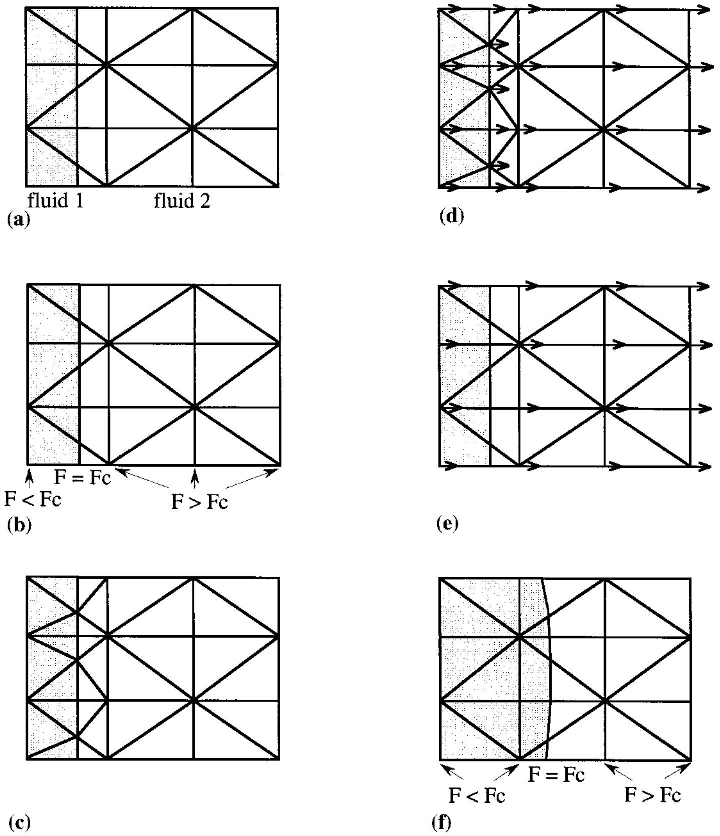


Figure 3. Illustration of the local mesh adaptation algorithm. (a) Interface initial position on the reference mesh. (b) Description of the F-field on the reference mesh. (c) Loal mesh adaptation—physical properties and boundary conditions update. (d) Computation of the velocity field on the temporary mesh. (e) Projection of the computed velocity field on the reference mesh. (f) Computation of the new F-field—update of the interface locaton on the reference mesh.

The equations governing the motion of this two-fluid system are the momentum and continuity equations with discontinuous physical property fields. If ρ_α and μ_α denote the density and the dynamic viscosity of fluid $\alpha(\alpha = 1, 2)$, $\mathbf{V}$ is the velocity and $\mathbf{b}$ the body forces,

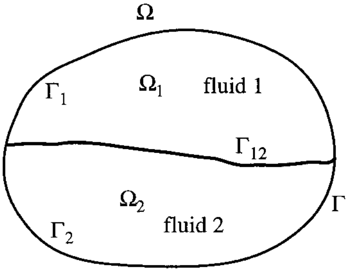


Figure 4. Two-fluid system with a moving interface.

these equations are written as:

$$\rho_\alpha\left(\frac{\partial \mathbf{V}}{\partial t}+(\mathbf{V}\cdot\nabla)\mathbf{V}\right)=\nabla\cdot\boldsymbol{\sigma}+\mathbf{b} \tag{6}$$

$$\nabla\cdot\mathbf{V}=\mathbf{0} \tag{7}$$

In the momentum equation (6), $\boldsymbol{\sigma}$ is the stress tensor, the expression of which, for both fluids, is given by the following Newtonian behaviour law:

$$\boldsymbol{\sigma}=-p\mathbf{I}+\mu_\alpha(\nabla\mathbf{V}+\nabla\mathbf{V}^T) \tag{8}$$

where p is the pressure and $\mathbf{I}$ the identity tensor.

3.2. Boundary conditions

Different types of boundary conditions may be used on Γ. For more generalities, we will express them in the curvilinear co-ordinate system $(\boldsymbol{\tau}, \mathbf{n})$ where $\boldsymbol{\tau}$ and $\mathbf{n}$ are the unit tangent and the unit outward normal vectors respectively.

(i) The Dirichlet boundary conditions may be used for every one of the tangential and normal velocity components:

$$V_\tau=V_\tau^\Gamma \tag{9}$$

$$V_n=V_n^\Gamma \tag{10}$$

Equation (9) may be employed to specify a no-slip condition at solid walls, with a zero value for V_τ^Γ if the walls are fixed. It may also be used as inlet or outlet boundary conditions.

Equation (10) may be applied to impose a normal velocity profile at inlet or an impermeability condition at solid walls. In the last case, V_n^Γ is set to zero.

(ii) The Neumann boundary conditions may be applied as specified normal and/or tangential forces:

$$T_\tau=\mathbf{T}\cdot\boldsymbol{\tau}=(\sigma\cdot\mathbf{n})\cdot\boldsymbol{\tau}=T_\tau^\Gamma \tag{11}$$

$$T_n=\mathbf{T}\cdot\boldsymbol{n}=(\boldsymbol{\sigma}\cdot\mathbf{n})\cdot\boldsymbol{n}=T_n^\Gamma \tag{12}$$

where T_τ and T_n are the tangential and normal components of the stress vector $\mathbf{T}$, respectively.

Equation (11) is exclusively used to impose a wall shear stress condition or a slip condition if T_τ^Γ is set to zero. This is also the correct condition to specify symmetry lines. In our numerical model, in order to allow the movement of the interface along solid walls, equation (11) was employed, instead of using a no-slip condition. In that case, either $T_\tau^\Gamma=0$ or $T_\tau^\Gamma=-C_\mathrm{f}V_\tau$ is imposed, where C_f is a specified friction coefficient.

Equation (12) is generally applied with a zero value of T_n^Γ in order to impose a traction-free boundary condition at outlet. It is also a useful condition to specify a pressure gradient.

3.3. Interfacial conditions

At the interface Γ_{12}, both kinematical and mechanical conditions must be satisfied. The first one, which is velocity continuity condition, is automatically satisfied since the discretized sub-domains Ω_1 and Ω_2 share the same nodes at this interface. In the absence of surface tension gradient along the interface, the mechanical condition indicates that the jump in traction forces balance in the surface tension action due to curvature:

$$\mathbf{T}_1+\mathbf{T}_2+\frac{\gamma}{R}\mathbf{n}=\mathbf{0} \tag{13}$$

where $\mathbf{T}_1$ and $\mathbf{T}_2$ are the stress vectors of fluid 1 and fluid 2 at the interface, the surface tension coefficient and R the radius of curvature of the interface. $\mathbf{n}$ is the unit normal vector to Γ_{12} pointing towards the center of curvature.

4. FINITE ELEMENT MODEL

4.1. Front tracking problem

The finite element formulation for the front tracking problem has been presented in previous studies [19–21]. The integral form of the variational statement is written as:

$$W_F = \int_\Omega \delta F\left(\frac{\partial F}{\partial t} + \mathbf{V}\cdot\nabla F\right)\mathrm{d}\Omega \tag{14}$$

where δF represents the weighting function for F.

This integral is spatially discretized using an iso-parametric Lagrangian three-node triangular element, which ensures a piece-wise linear C^0 approximation for the pseudo-concentration function F.

4.2. Flow problem

For the flow problem, some modifications have been made to the numerical model presented in a previous paper [20], in order to take into account the interface condition (13).

In References [19–21], a finite element that ensured a C^0 approximation for the pressure was used. However, this element is not suitable to handle pressure jump across the interface. So in the present numerical model, a seven-node triangular finite element that ensures a quadratic C^0 approximation for the velocity and a piece-wise linear C^1 approximation for the pressure was used. Thus, pressure jump and traction forces jump can be modelled. This choice of a discontinuous pressure approximation is made in combination with a penalized formulation of the incompressibility condition. Therefore, the pressure variable can be eliminated at element level by static condensation in order to reduce the number of variables to compute.

The integral form of the variational statement is then written as:

$$W_V = \int_\Omega \left\{ \boldsymbol{\delta}\mathbf{V}\left[\rho_\alpha\left(\frac{\partial \mathbf{V}}{\partial t} + (\mathbf{V}\cdot\nabla)\mathbf{V}\right) - \mathbf{b}\right] + \delta p\left[\nabla\cdot\mathbf{V} - \frac{p}{\lambda}\right] + \nabla\boldsymbol{\delta}\mathbf{V}{:}\boldsymbol{\sigma}\right\}\mathrm{d}\Omega - \int_\Gamma \boldsymbol{\delta}\mathbf{V}\cdot\mathbf{T}\,\mathrm{d}\Gamma$$
$$- \int_{\Gamma_{12}} \boldsymbol{\delta}\mathbf{V}\cdot(\mathbf{T}_1 + \mathbf{T}_2)\,\mathrm{d}\Gamma_{12}, \tag{15}$$

where $\boldsymbol{\delta}\mathbf{V}$ and δp are the weighting functions associated with the velocity and the pressure, respectively. λ is the penalty coefficient of the order of 10^9. In Equation (15), the boundary integrals result from the integration by parts of the stress terms. For convenience, they may be expressed in terms of tangential and normal components in order to incorporate Neumann boundary conditions (11) and (12) and interface condition (13). The integral form (15) then becomes:

$$W_V = \int_\Omega \left\{ \boldsymbol{\delta}\mathbf{V}\left[\rho_\alpha\left(\frac{\partial \mathbf{V}}{\partial t} + (\mathbf{V}\cdot\nabla)\mathbf{V}\right) - \mathbf{b}\right] + \delta p\left[\nabla\cdot\mathbf{V} - \frac{p}{\lambda}\right] + \nabla\boldsymbol{\delta}\mathbf{V}{:}\boldsymbol{\sigma}\right\}\mathrm{d}\Omega$$
$$- \int_\Gamma (\delta V_\tau T_\tau + \delta V_n T_n)\,\mathrm{d}\Gamma + \int_{\Gamma_{12}} \delta V_n \frac{\gamma}{R}\,\mathrm{d}\Gamma_{12}. \tag{16}$$

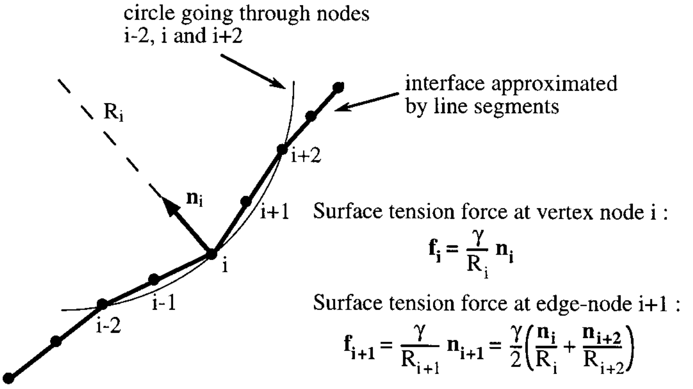

circle going through nodes i-2, i and i+2
interface approximated by line segments
R_i
$\mathbf{n}_i$
i+2
i+1
i
i-1
i-2
Surface tension force at vertex node i :
$\mathbf{f}_i = \frac{\gamma}{R_i}\mathbf{n}_i$
Surface tension force at edge-node i+1 :
$\mathbf{f}_{i+1} = \frac{\gamma}{R_{i+1}}\mathbf{n}_{i+1} = \frac{\gamma}{2}\left(\frac{\mathbf{n}_i}{R_i} + \frac{\mathbf{n}_{i+2}}{R_{i+2}}\right)$

Figure 5. Computation of surface tension force.

The first boundary integral is equal to zero on the parts of Γ where Dirichlet boundary conditions (9) and (10) prevail. It is spatially discretized by employing three-node quadratic boundary finite elements when Neumann boundary conditions must be specified.

4.3. Surface tension force modelling

In Equation (16), the surface tension action on Γ_{12} is modelled by using quadratic three-node boundary elements. The calculation of the unit normal vector **n** and the radius of curvature R at the interface may then be obtained from the element curvature. However, in the present study, in order to simplify the mesh adaptation algorithm, the interface is approximated by a set of line segments. Each one of these line segments corresponds to a three-node boundary element. Thus, the parameters **n** and R must be determined differently. They are calculated only for each one of the boundary element vertex nodes. Let i be one of these nodes and $i-2$ and $i+2$ the two others neighbouring vertex nodes as illustrated in Figure 5. The parameters **n** and R, and then the surface tension force are computed at node i, by considering the circle going through nodes $i-2$, i and $i+2$. Then, the surface tension force at mid-node $i+1$, for example, is obtained from linear interpolation of nearby vertex nodes contributions.

4.4. Solution procedure

The solution strategy of the governing equations of this moving interface flow has been presented in References [19,20]. The implicit Euler scheme is used to advance the solution in time. The solution is staggered in the sense that the flow problem and the front tracking one are solved in a sequential manner. Indeed, at each time step, the last calculated velocity field is employed to compute the interface location. This in turn is used, to update fluid properties before solving flow equations, and so on. This way of coupling is of explicit nature which therefore induces a time step limitation. A reliable guide to select the most suitable time step is that the interface does not skip any domain element.

5. NUMERICAL EXAMPLES

In order to demonstrate the efficiency of the proposed model, we present two numerical examples.

The first one is the broken dam problem studied experimentally by Martin and Moyce [25] and numerically by other authors (Hirt and Nichols [7], Huerta and Liu [26] and Wang and Wang [13]). This problem has retained our attention because it belongs to the domain of application of both models (average properties for transition elements or accurate discretization of the material discontinuity), in the limit of comparable meshes. Thus the numerical results obtained with our model (including mesh adaptation), and its previous version [20] (using average properties for transition elements), have been compared with the data of Martin and Moyce [25].

The second example is the well-known Rayleigh–Taylor instability that occurs when a heavy fluid is superimposed over a light one in a gravitational field acting downward. The aim of that study is to check if our numerical model (supplemented with local mesh adaptation) can now accurately handle such class of flows (dominated by interfacial phenomena). This problem has been analyzed theoretically by Chandrasekhar [27] in the early stages of the instability development. Authors such as Daly [28,29], Dervieux and Thomasset [30], Baker *et al.* [31], Tryggvason and Unverdi [32,33] and Andrews [34] have studied it numerically. The results obtained in the present study have been compared with the analytical predictions of Chandrasekhar [27] for different fluid density ratios. Surface tension effect on the instability growth for a given ratio has also been examined.

5.1. Broken dam problem

A column of water, in hydrostatic equilibrium, is initially confined between two vertical walls. At initial time ($t = 0$), the right wall is suddenly removed and the water column flows out, under gravity, along a dry horizontal floor. The water column is chosen to be $a = 5.715$ cm wide and $b = 11.430$ cm high in order to compare this numerical solution with the experimental data obtained by Martin and Moyce [25]. The computational domain as shown in Figure 6 is extended by approximately six times the width of the column in the horizontal direction and twice the height vertically.

The density and the dynamic viscosity are 1000 kg m^{-3} and 0.001 kg m^{-1} s^{-1} respectively for the water, and 1 kg m^{-3} and 0.0001 kg m^{-1} s^{-1} for the surrounding air. No surface tension force is applied since its effect is negligible compared with gravity force.

The initial values for velocity field are zero in the whole domain and the initial F-field is defined as follows:

$$F(x, y) = y - b \quad \text{for } y > 2\text{x} \quad \text{and} \quad F(x, y) = x - a \quad \text{for } y \leq 2x.$$

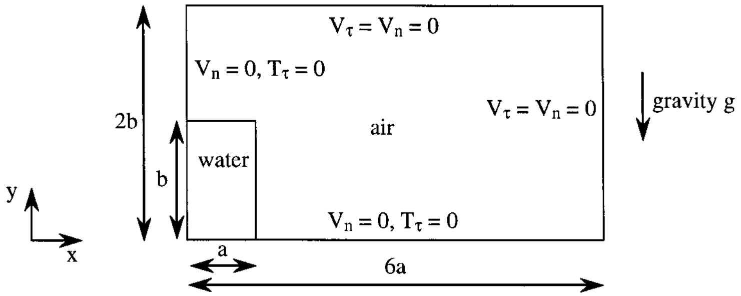


Figure 6. Computational domain and boundary conditions for broken dam problem.

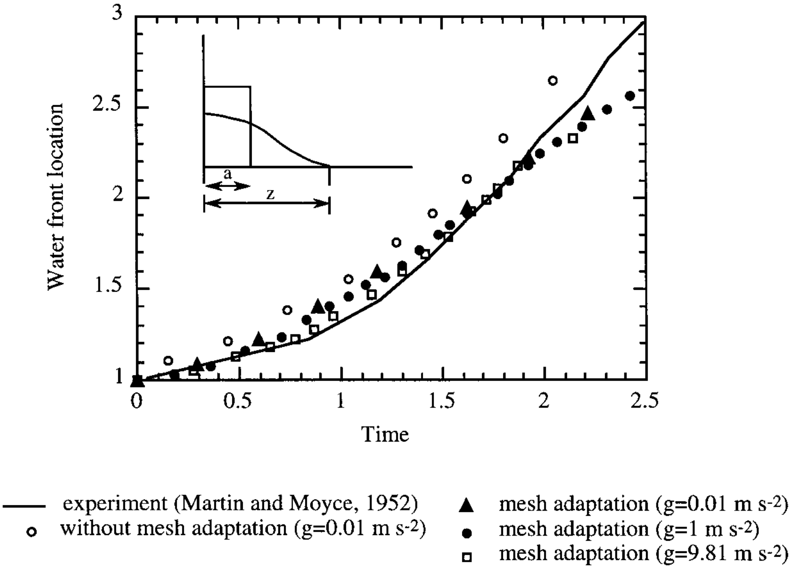


Figure 7. Water front location z/a versus time $t\sqrt{2g/a}$ along the floor.

The specified value Fc used to identify the interface position at each time step is then equal to zero.

The boundary conditions used in the calculation are indicated in Figure 6. They correspond to those of a flow in a closed cavity, although Martin and Moyce [25] used an open channel. However, the influence of these boundary conditions can be clearly identified in the results, thus defining a non-influenced region where numerical and available experimental results can be compared.

For the computation, the unstructured reference mesh of the rectangular domain comprises 1488 elements. The problem has been run for three values of gravity ($g = 0.01$, 1 and 9.81 m s^{-2}). During the collapse of the column, the water accelerates and smaller time steps have to be used. They vary from 0.5 s to 0.001 s.

The predicted water front location along the floor have been reported for the three gravity values in Figure 7 and compared with the Martin and Moyce's experimental data [25]. The results are presented in a dimensionless form by defining a as the reference length and $\sqrt{a/2g}$ as the reference time. The numerical solutions obtained by the model, including mesh adaptation are in good agreement with the measurements, until $t\sqrt{2g/a} \approx 2$, which defines the limit of the non-influenced region. One can observe that the front speed is slightly over-estimated by the numerical model. This is an outcome of the slip boundary condition used on the horizontal floor. However, the influence of this condition decreases with increasing gravity magnitude.

The main features of this flow depends on gravity and viscosity, whereas the details that happen at the interface are of second order. Thus, as depicted in Figure 7 for the case $g = 0.01$ m s^{-2}, the local mesh adaptation model improves only slightly global behaviour, such as the prediction of the water front location on the floor. However for the same mesh, no successful calculation has been obtained by the model without mesh adaptation, for gravity values greater than 0.01 m s^{-2}. This failure may be related to numerical instabilities produced when using average properties for transition elements, as previously mentioned in Section 2.

Therefore, in the presence of high property ratio, local mesh adaptation can be seen as a more accurate way than artificial dumping to overcome such numerical instabilities.

Two other computations were performed, without mesh adaptation, for $g = 0.01$ m s^{-2}. Instead of using average fluid physical properties in transition elements, the physical properties of the air for the first calculation and those of the water for the second were used. Figure 8 shows the variation of the velocity y-component along the x-direction for a given interface. Because of the unstructured mesh used, the curves were plotted at a height where the node number is greatest. Then, it can be seen that the results given by the mesh adaptation model are situated between the two limit curves obtained without this adaptation technique. So if a very fine mesh is used, the model not including mesh adaptation will give results close to those of the improved model.

5.2. Rayleigh–Taylor instability

When a heavy fluid (density ρ_2) is superimposed over a light one (density ρ_1) in a gravitational field g acting downward, the fluid interface is unstable. Any perturbation of this interface tends to grow with time producing the phenomenon known as Rayleigh–Taylor instability.

Various factors such as fluid density ratio, viscosity, surface tension, etc., influence the development of this instability. We have chosen here to study, firstly in the absence of surface tension, the effect of the density ratio ρ_2/ρ_1 on the growth of the instability and to compare the results, in the early stages of the instability development, with the analytical predictions given by Chandrasekhar [27]. Then, an additional calculation including surface tension was performed in order to examine its effect on this instability. Chandrasekhar [27] predicted, by means of the linear theory, that there exists a most unstable wavelength disturbance λ_m for which the growth rate n_m of Rayleigh–Taylor instability is greatest. Both values are given by:

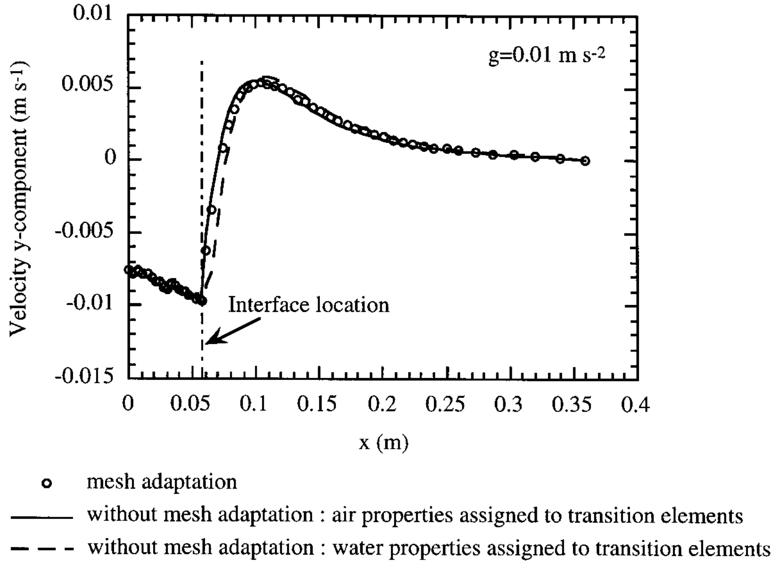


Figure 8. Velocity y-component variation along the x-direction.

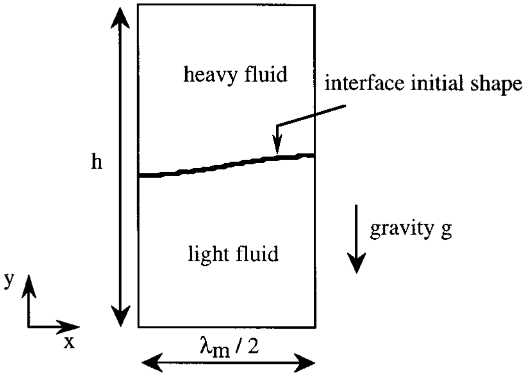


Figure 9. Computational domain and interface initial shape for Rayleigh–Taylor instability.

$$n_m = \left(\frac{2\pi g}{\lambda_m}\frac{\rho_2-\rho_1}{\rho_2+\rho_1}\right)^{1/2} \qquad \lambda_m = 4\pi\left(\frac{4\mu^2}{\rho_2^2-\rho_1^2}\frac{1}{g}\right)^{1/3},$$

where μ is the dynamic viscosity, supposedly the same for both fluids ($\mu = 0.001$ kg m^{-1} s^{-1} in the present study). The density of the light fluid ρ_1 is fixed at 1 kg m^{-3} and remains unchanged, while that of the heavy fluid ρ_2 varies.

The height of the computational domain, as shown in Figure 9, has been fixed at $h = 18$ cm, whereas the width is adjusted in order to simulate a half wavelength of the most unstable perturbation ($\lambda_m/2$). The two horizontal boundaries of the domain are considered as slip walls and the vertical ones correspond to symmetry lines. In both cases, the boundary conditions are:

$$V_n = 0 \quad \text{and} \quad T_\tau = 0.$$

At initial time ($t = 0$), a zero velocity field is imposed in the whole computational domain and the interface perturbation is introduced by specifying the following initial F-field:

$$F(x, y) = y - \left(\frac{h}{2} + A_0\cos(k_0 x)\right),$$

where the amplitude and the wave number of the perturbation are $A_0 = \lambda_m/100$ and $k_0 = 2\pi/\lambda_m$, respectively. With this definition of the F-field, the index value Fc is equal to zero and the interface has the initial shape shown in Figure 9.

The calculations are performed on an unstructured mesh of 1200 elements. The gravity value is 9.81 m s^{-2} and the time steps vary from 0.02 s to 0.005 s depending on the density ratios.

Figure 10 shows the time variation of the calculated spike's amplitude (large dots) for four problems in which the fluid density ratios are 1.1, 2, 3 and 4. These results are compared with linear theory, which is only applicable during the early phases of the instability development and indicates that the perturbation amplitude grows with time as:

$$A = A_0\cosh(n_m t).$$

We notice that the numerical solutions are in good agreement with the linear theory at early stages of the computations. Then non-linear effects become dominant and the results diverge from this theory.

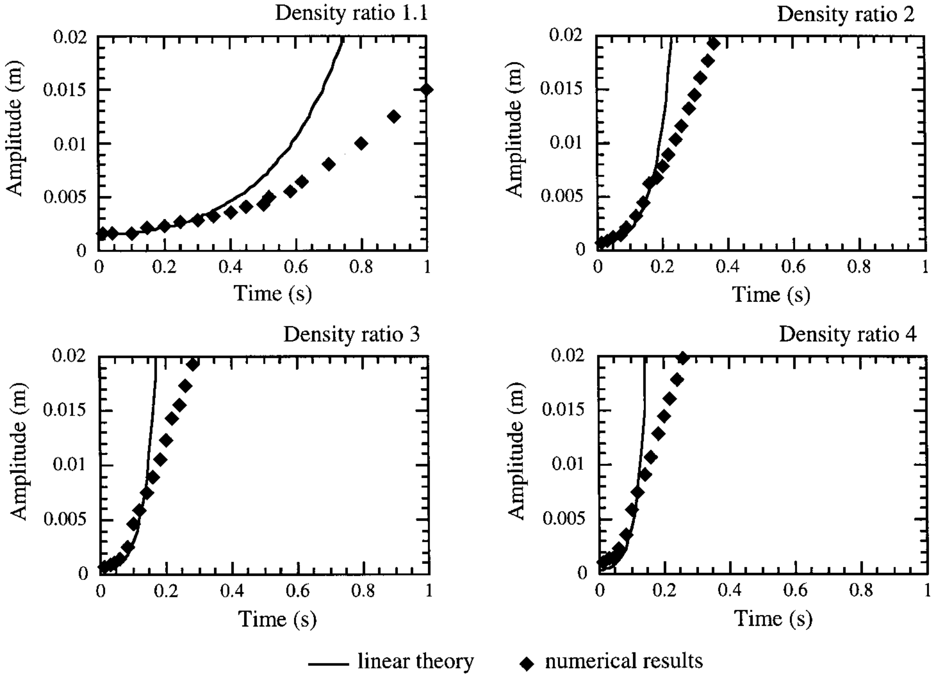


Figure 10. Comparison of the calculated spike amplitude with the linear theory.

The growth rate of the perturbation in the early stages is plotted versus Atwood number $A = (\rho_2 - \rho_1)/(\rho_2 + \rho_1)$ in Figure 11 and is compared with the prediction of Chandrasekhar [27]. Excellent agreement with this prediction is obtained. As found by Daly [28], the growth rate increases with the Atwood number.

In order to examine the effect of surface tension on the instability growth, an additional computation incorporating surface tension has been performed for a fluid density ratio of 1.1.

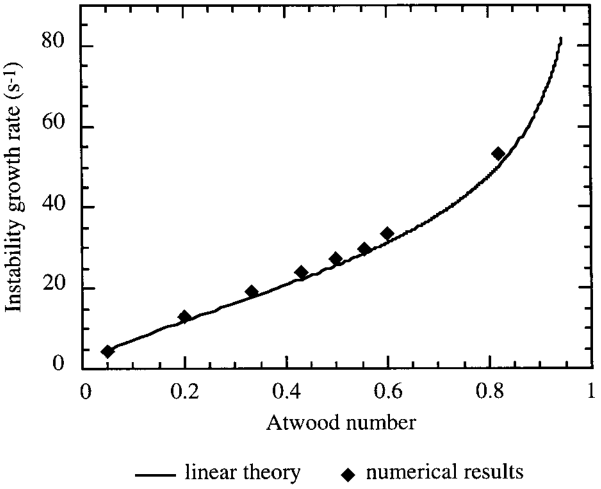


Figure 11. Comparison of the numerical growth rate with the linear theory.

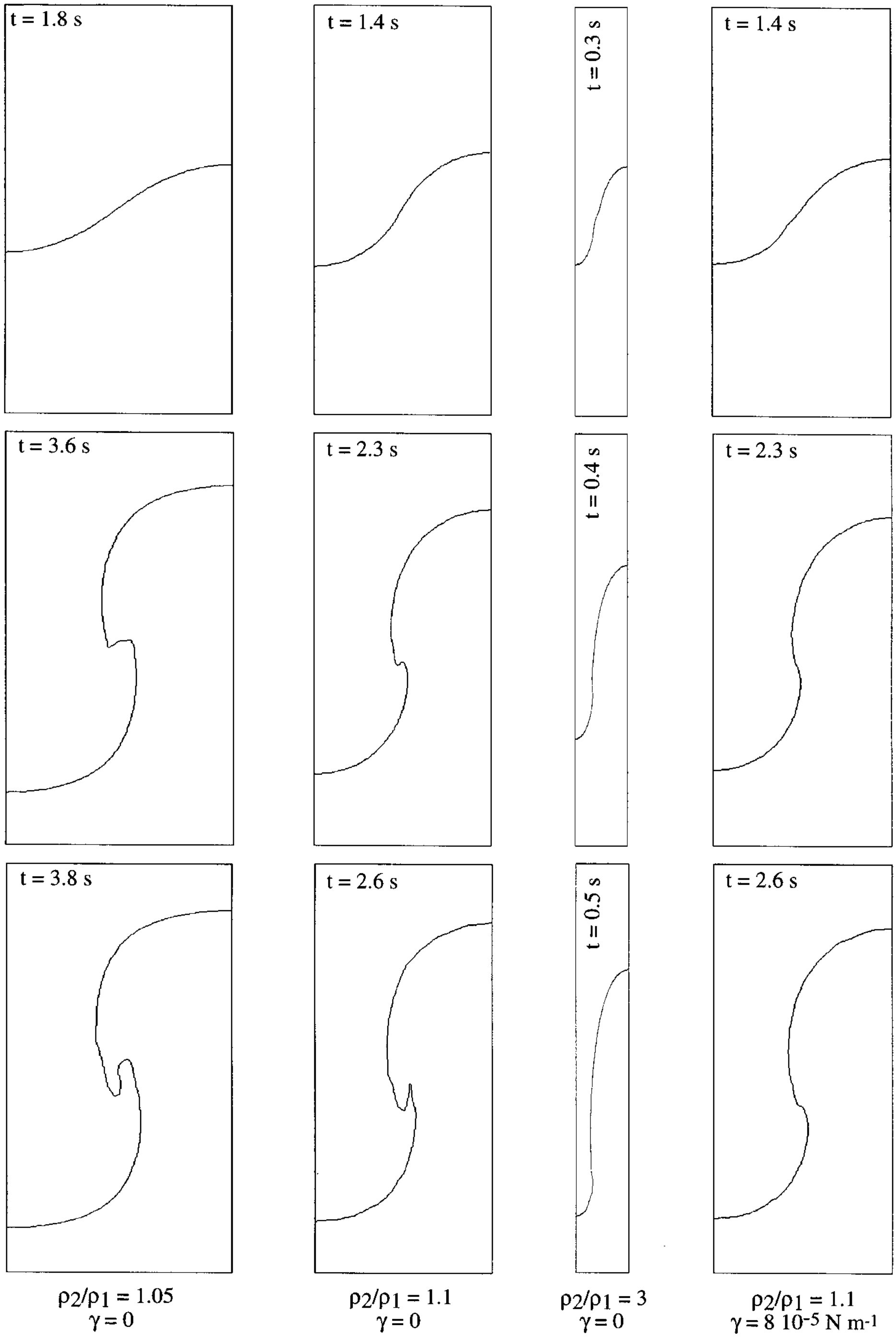


Figure 12. Development of Rayleigh–Taylor instability at different density ratios and surface tension coefficients.

The surface tension coefficient is chosen to be smaller than a critical value for which the flow is stabilized. According to Chandrasekhar [27], for an initial perturbation whose wavelength is λ_m, this surface tension critical value is given by:

$$\gamma_c = \frac{\lambda_m^2(\rho_2 - \rho_1)g}{4\pi^2}.$$

For a density ratio of 1.1, that critical value is equal to 1.53 10^{-4} N m^{-1}.

Figure 12 shows the development with time of Rayleigh–Taylor instability for the following four cases:

$a - \rho_2/\rho_1 = 1.05$ and $\gamma = 0$

$b - \rho_2/\rho_1 = 1.1$ and $\gamma = 0$

$c - \rho_2/\rho_1 = 3$ and $\gamma = 0$

$d - \rho_2/\rho_1 = 1.1$ and $\gamma = 8 \cdot 10^{-5}\,\mathrm{N\,m^{-1}}$.

The first three columns show that the late time shape of the interface and the instability growth are strongly influenced by the density ratio. At density ratios of 1.05 and 1.1, the interface rolls up along the side of the spike giving this mushroom form. This phenomenon, known as Kelvin–Helmholtz instability, is due to the development of short wavelength perturbations along the fluid interface and parallel to the main flow. At a density ratio of three, however, there is no roll-up. The heavy fluid simply falls through the light fluid in the form of a narrow spike, the amplitude of which rapidly increases with time.

Finally, the last column plot series compared with the second column ones reveal the surface tension retarding effect on the short wavelength mode development. Indeed, Kelvin–Helmoltz instability appears later in the calculations that include surface tension, rather than in the zero surface tension problem.

6. CONCLUSIONS

A finite element model for the simulation of time-dependent incompressible viscous flows with moving interface has been presented. The 2D Navier–Stokes equations are solved with a fixed mesh approach, and the interface is tracked by a pseudo-concentration function based method. This latter allows us to deal with problems with complex interface topology. In order to keep the physical fluid property fields sharp, a local mesh adaptation technique has been developed for general unstructured triangular meshes. Owing to this technique, the description of material discontinuity across the interface is accurate. Interface phenomena such as the surface tension effect can also be incorporated in a natural way by using specific boundary elements at the interface. The local mesh adaptation then allows us to treat the interface as a real discontinuity which has better accuracy in comparison with the techniques that smear it over some finite region. Finally, this technique is more economical with computational time than the remeshing-based methods which entirely redefine the mesh. Here, only crossed interface elements are affected by the modification. The numerical examples presented in this paper have demonstrated the efficiency of our finite element model.

REFERENCES

1. J.M. Hyman, 'Numerical methods for tracking interface', *Physica D*, **120**, 396 (1984).
2. K.J. Laskey, E.S. Oran and J.P. Boris, 'Approaches to resolving and tracking interfaces and discontinuities', *Naval Research Laboratory Report 5999*, Arlington, VA, 1987.
3. J.M. Floryan and H. Rasmussen, 'Numerical methods for viscous flows with moving boundaries', *Appl. Mech. Rev.*, **42**, 323 (1989).
4. S. Osher and J.A. Sethian, 'Fronts propagating with curvature dependent speed: algorithms based on Hamilton–Jacobi formulations', *J. Comput. Physics*, **79**, 12 (1988).
5. J.E. Welch, F.H. Harlow, J.P. Shannon and B.J. Daly, 'The MAC method: a computing technique for solving viscous, incompressible, transient fluid-flow problems involving free surfaces', *Los Alamos Scientific Laboratory Report LA-3425*, 1966.
6. F.H. Harlow and J.F. Welch, 'Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface', *Phys. Fluids*, **8**, 2182 (1965).
7. C.W. Hirt and B.D. Nichols, 'Volume of fluid (VOF) method for the dynamics of free boundaries', *J. Comput. Phys.*, **39**, 201 (1981).
8. J.D. Ramshaw and J.A. Trapp, 'A numerical technique for low-speed homogeneous two-phase flow with sharp interfaces', *J. Comput. Phys.*, **21**, 438 (1976).
9. E. Thompson, 'Use of pseudo-concentration to follow creeping viscous flow during transient analysis', *Int. j. numer. methods fluids*, **6**, 749 (1986).
10. G. Dhatt, D.M. Gao and A. Ben Cheikh, 'A finite element simulation of metal flow in moulds', *Int. j. numer. methods eng.*, **30**, 821 (1990).
11. A.S. Usmani, J.T. Cross and R.W. Lewis, 'A finite element model for the simulations of mould filling in metal casting and the associated heat transfer', *Int. j. numer. methods eng.*, **35**, 787 (1992).
12. R.W. Lewis, A.S. Usmani and J.T. Cross, 'Efficient mould filling simulation in castings by an explicit finite element method', *Int. j. numer. methods fluids*, **20**, 493 (1995).
13. S.P. Wang and K.K. Wang, 'A net inflow method for incompressible viscous flow with moving free surface', *Int. j. numer. methods fluids*, **18**, 669 (1994).
14. *FIDAP 7.5 Update Manual*, Fluid Dynamics International, Evanston, IL, 1995.
15. B.R. Baliga and S.V. Patankar, 'A control volume finite-element method for two-dimensional fluid flow and heat transfer', *Numer. Heat Transfer*, **6**, 245 (1980).
16. J. Glimm, J. Grove, B. Lindquist, O.A. McBryan and G. Tryggvason, 'The bifurcation of tracked scalar waves', *SIAM J. Sci. Stat. Comput.*, **9**, 61 (1988).
17. T. Sato and S.M. Richardson, 'Numerical simulation method for viscoelastic flows with free surfaces-fringe element generation method', *Int. j. numer. methods fluids*, **19**, 555 (1994).
18. F. Mashayek and N. Ashgriz, 'A hybrid finite-element-volume-of-fluid method for simulating free surface flows and interfaces', *Int. j. numer. methods fluids*, **20**, 1363 (1995).
19. M. Medale, 'Modélisation numérique de l'étape de remplissage des moules de fonderie par la méthode des éléments finis', *Thèse de Doctorat*, Université de Technologie de Compiègne, France, 1994.7
20. M. Medale and M. Jaeger, 'Numerical simulation of incompressible flows with moving interfaces', *Int. j. numer. methods fluids*, **24**, 615 (1997).
21. M. Jaeger, M. Medale, N. Lock and R. Martin, 'A finite element model for thermal incompressible flows with moving free boundaries', *Proc. 10th Int. Heat Transfer Conf.*, Brighton, UK, **2**, 372 (1994).
22. S.O. Unverdi and G. Tryggvason, 'A front-tracking method for viscous, incompressible, multi-fluid flows', *J. Comput. Phys.*, **100**, 25 (1992).
23. J.U. Brackbill, D.B. Kothe and C. Zemach, 'A continuum method for modeling surface tension', *J. Comput. Phys.*, **100**, 335 (1992).
24. S. Zaleski, 'Méthodes de simulation d'interfaces fibres entre fluides', *Proc. Ecole de Printemps Mécanique des Fluides Numérique*, Carcans-Maubuisson, France, 1995.
25. J.C. Martin and W.J. Moyce, 'An experimental study of the collapse of liquid columns on a rigid horizontal plane', *Philos. Trans. Ser. A*, *Math. Phys. Sci.*, **244**, 312 (1952).
26. A. Huerta and W.K. Liu, 'Viscous flow with large free surface motion', *Comput. Methods Appl. Mech. Eng.*, **69**, 277 (1988).
27. S. Chandrasekhar, *Hydrodynamic and Hydromagnetic Stability*, Oxford University Press, Oxford, 1961.
28. B.J. Daly, 'Numerical study of two fluid Rayleigh–Taylor instability', *Phys. Fluids*, **10**, 297 (1967).
29. B.J. Daly, 'Numerical study of the effect of surface tension on interface instability', *Phys. Fluids*, **12**, 1340 (1969).
30. A. Dervieux and F. Thomasset, 'A finite element method for the simulation of a Rayleigh–Taylor instability', *Lect. Notes in Math.*, **771**, 145 (1980).
31. G.R. Baker, D.I. Meiron and S.A. Orszag, 'Vortex simulations of the Rayleigh–Taylor instability', *Phys. Fluids*, **23**, 1485 (1980).
32. G. Tryggvason and S.O. Unverdi, 'Computations of three-dimensional Rayleigh–Taylor instability', *Phys. Fluids*, **2**, 656 (1990).
33. S.O. Unverdi and G. Tryggvason, 'Computations of multi-fluid flows', *Physica D*, **60**, 70 (1992).
34. M.J. Andrews, 'Accurate computation of convective transport in transient two-phase flow', *Int. j. numer. methods fluids*, **21**, 205 (1995).