INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2007; **53**:735–751
Published online 23 August 2006 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1303

A two-phase flow interface capturing finite element method

C. Devals[1,‡], M. Heniche[1,§], F. Bertrand[1,*,†], P. A. Tanguy[1,¶] and R. E. Hayes[2,‖]

[1]*Department of Chemical Engineering, Ecole Polytechnique, Montreal, Que., Canada H3C 3A7*
[2]*Department of Chemical and Materials Engineering, University of Alberta, Edmonton, Alta., Canada T6G 2G6*

SUMMARY

A new interface capturing algorithm is proposed for the finite element simulation of two-phase flows. It relies on the solution of an advection equation for the interface between the two phases by a streamline upwind Petrov–Galerkin (SUPG) scheme combined with an adaptive mesh refinement procedure and a filtering technique. This method is illustrated in the case of a Rayleigh–Taylor two-phase flow problem governed by the Stokes equations. Copyright © 2006 John Wiley & Sons, Ltd.

Received 27 March 2006; Revised 12 June 2006; Accepted 12 June 2006

KEY WORDS: interface capturing; finite element method; Stokes equations; adaptive meshing; filtering technique; Rayleigh–Taylor flow

1. INTRODUCTION

Two-phase flows involving the combination of a gas phase and a liquid phase are commonly encountered in daily life, especially in chemical engineering processes and the oil industry [1, 2]. These complex flows are difficult to handle from an engineering standpoint, and their accurate modelling is still an open issue. One approach that can be used to develop a better insight is to resort to numerical simulation.

A correct simulation of two-phase flow problems requires a reliable representation of the interface between the two phases. Several approaches have been proposed for this purpose, which can be

*Correspondence to: F. Bertrand, Department of Chemical Engineering, Ecole Polytechnique, C.P. 6079 Station CV, Montreal, Que., Canada H3C 3A7.
†E-mail: francois.bertrand@polymtl.ca
‡E-mail: christophe.devals@polymtl.ca
§E-mail: mourad.heniche@polymtl.ca
¶E-mail: philippe.tanguy@polymtl.ca
‖E-mail: Bob.Hayes@Ualberta.ca

Contract/grant sponsor: Natural Sciences and Engineering Research Council of Canada (NSERC)

divided into Lagrangian, mixed Eulerian–Lagrangian and Eulerian methods. With Lagrangian methods, the mesh moves and is distorted as its interface moves. In practice, this movement leads to large amplitude deformation, and breaking or merging patterns cannot be well captured unless a repetitive remeshing strategy is used, which makes the computation costly. In the case of the mixed Eulerian–Lagrangian methods, the mesh is updated in the vicinity of the moving interface, the mesh being generally kept fixed elsewhere. Note that, in recent Arbitrary–Lagrangian–Eulerian (ALE) methods, the mesh can be improved not only near the interface, but also anywhere in the domain where the element aspect ratio has deteriorated [3]. Thorough reviews for these two approaches can be found in References [4, 5].

The Eulerian approach, which makes use of a fixed grid, is probably the most popular method. The first Eulerian models, introduced 50 years ago at Los Alamos [6], are the PIC (particle-in-cell) method and the MAC (marker-and-cell) method, both based on marker particles to follow the interface. Later, Rider and Kothe [7] extended this idea with their so-called *marker particle* method. Other methods appeared in the 1980s, such as the volume of fluid (VOF) method of Hirt and Nichols [8]. In this case, the position of the interface is calculated through the transport of a function representing the fractional volume of the fluid in a cell. Interface reconstruction is needed and in the 'original' VOF method, the interface was approximated by vertical or horizontal straight lines. One property of the VOF method is that mass conservation is ensured, although the interface reconstruction is time consuming and its extension to 3D is complex. This method has since been improved by many authors [9–12] and a review of the VOF methods can be found in Reference [13].

Another popular way to capture an interface is the level-set method [14], where the zero level of a smooth distance function indicates the position of the interface. Contrary to the VOF method, this method does not conserve mass intrinsically so that high-order schemes or mesh refinement are required for an accurate solution of the underlying advection equation. A reinitialization step is also needed to ensure that the level-set function remains a distance function. A mass-preserving level-set method can be found in Reference [15].

Finally, the pseudo-concentration method is based on the value of a colour function F to determine the location of the two fluids in the domain. The value of F ranges between 0 and 1, and the position of the interface corresponds to the position of the contour line $F = 0.5$. As in the VOF or the level-set methods, F is transported by an advection equation. One of its advantage is that it does not need interface reconstruction as with the VOF method but adaptive or mesh refinement strategies are however required to obtain accurate solutions [16, 17].

In this work, we have chosen the pseudo-concentration method because its implementation is relatively easy in 2D and can be extended without much difficulty to 3D. The flow situation considered is the classical viscous Rayleigh–Taylor (RT) flow problem, which is a good benchmark problem to assess the efficiency of an interface capturing method. In this problem, a heavy fluid of density ρ_1 is placed initially above a light fluid of density ρ_2. A small initial perturbation in the interface combined with the effect of the gravitational force results in a collapsing of fluid 1 into fluid 2. This problem is challenging owing to the evolution of the interface shape, and it is particularly suitable to investigate the coupling of the advection of the colour function and the momentum equations, and how the cumulative error can be controlled when the mesh is adapted around the moving interface.

The RT flow problem is often used in literature to compare different interface tracking and capturing methods, and a great variety of results can be found in References [18–20] for instance. If some studies are in 3D, as in Reference [21], most of them are 2D.

DOI: 10.1002/fld

Fraigneau *et al*. [18] developed a Eulerian method for simulating variable density incompressible viscous flows. They compared the finite element method and the finite volume method for the RT flow problem in the viscous regime for two different Reynolds numbers. They showed that, even at moderate Reynolds numbers, this problem is very sensitive to the numerical method used and, in particular, to the mesh refinement algorithm.

Zhao *et al*. [19] solved the incompressible Navier–Stokes equations for two superimposed viscous fluids on unstructured grids with the finite volume method. The free surface was computed with the VOF method and the surface tension was taken into account. The motion of the characteristic RT mushroom shape of the interface was studied at $Re = 283$.

Rudman [20] used an algorithm for volume tracking based on the concept of flux-corrected transport (FCT) to solve the RT flow problem. He compared this method with three other VOF techniques: the simple line interface calculation (SLIC) method, the VOF method of Hirt and Nichols [8] and Youngs'method [22].

Popinet and Zaleski [23], as Smolianski [4] with a level-set method, revisited the work of Puckett *et al*. [24] and solved the RT flow problem with a front-tracking algorithm. Marker particles were advected in a Lagrangian manner to follow the interface. A source term to take into account surface tension was added to the momentum equation. In Reference [24], a second-order VOF method was used and surface tension was not taken into account.

Rider *et al*. [25] compared four different methods to compute the RT flow problem: an interface capturing method, two PLIC (for piecewise linear interface calculation) methods and a coloured particle method. In this study, the surface tension was neglected at the fluid interface. The density ratio was 7.25 and the viscosity ratio was 1.06. They all led to similar results, though the results obtained with the two PLIC methods seem to be slightly more accurate.

The interested reader may refer to the article of Tryggvason [26] for an overview of the Rayleigh–Taylor flow problem.

The objective of this paper is to extend to the case of two-phase Stokes equations a finite element-based interface capturing algorithm that was recently developed [27]. The solution scheme can be divided into four steps: (1) solution of the advection equation for the pseudo-concentration F by a streamline upwind Petrov–Galerkin (SUPG) method [28]; (2) filtering of F as proposed in Reference [29]; (3) solution of the Stokes equations for the velocity and pressure; and (4) automatic remeshing following the work of Bertrand *et al*. [30] and Devals *et al*. [31]. In Reference [27], the overall interface capturing algorithm was used to solve moving interface with a known velocity field. In the present work, the velocity field is unknown and the method is coupled to a mixed finite element method for the solution of the Stokes equations.

First, the two-phase flow problem to be solved will be presented. Next, the interface capturing algorithm will be recalled and the overall method will be described. Finally, the accuracy and the robustness of the method will be investigated in the case of the classical Rayleigh–Taylor flow problem.

2. TWO-PHASE FLOW PROBLEM

Let us consider the flow of two immiscible and incompressible fluids in computational domain Ω, obtained by solving the Navier–Stokes and continuity equations:

$$\rho(F)\left(\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v}\cdot\mathbf{grad}\,\mathbf{v}\right) = \mu(F)\Delta\mathbf{v} - \mathbf{grad}\,p + \mathbf{f} \quad \text{in } \Omega \tag{1}$$

$$\operatorname{div}\mathbf{v} = 0 \quad \text{in } \Omega \tag{2}$$

DOI: 10.1002/fld

and the advection equation of a pseudo-concentration for F

$$\frac{\partial F}{\partial t} + \mathbf{v} \cdot \mathbf{grad}\, F = 0 \quad \text{in } \Omega \tag{3}$$

where ρ, μ, $\mathbf{v}$, p, $\mathbf{f}$ represent the density, the viscosity, the velocity, the pressure and a body force like gravity, for instance. The value of F that lies between 0 and 1 depends on the distribution of the two fluids in the domain. F is set to 1 for fluid 1 and 0 for fluid 2, and $F = 0.5$ corresponds to an approximation of the interface position. F is also used to set the value of μ and ρ in the Navier–Stokes equations, as will be described later. Initial and boundary conditions for F and $\mathbf{v}$ must also be applied for mathematical well-posedness.

In the forthcoming, it will be assumed without loss of generality that the flow is inertialess. The proposed strategy is by no means restricted to creeping flows and can be extended in a straightforward manner to the full Navier–Stokes equations.

3. FINITE ELEMENT-BASED INTERFACE CAPTURING ALGORITHM

The overall numerical strategy for the simulation of the flow of two immiscible fluids as governed by Equations (1)–(3) is based on the interface capturing method described in Reference [27], and consists of four main steps:

1. solution of the advection equation (3);
2. filtering of the colour function F;
3. solution of the flow equations (1) and (2); and
4. automatic remeshing of the computational domain.

Each of these steps will now be described in detail in turn.

3.1. Advection equation

As mentioned in the previous section, the location of the two fluids is computed by a pseudo-concentration or colour function F, which is the solution of the advection equation (3). In the present work, this transient advection equation is solved by the SUPG method [28] and three-node linear triangles. The second-order single step implicit midpoint rule (IMR) is used to integrate this equation with respect to time, as advocated by Malidi *et al*. [32] for its nice mass conservation properties.

3.2. Filtering technique

As described in detail in References [27, 29], the filtering technique used in this work relies on a change of variable that smoothes out the oscillations of the colour function F. It also forces the values of F to remain within the range $[0, 1]$, as shown in Figure 1. Let us denote by $\tilde{F}$ the oscillatory solution of Equation (3), obtained with the SUPG finite element method. Then the oscillation-free solution, $\overline{F}$, is obtained after application of the filter:

$$\overline{F} = \frac{1}{1 + \exp(-2\alpha(2\tilde{F} - 1))} \tag{4}$$

DOI: 10.1002/fld

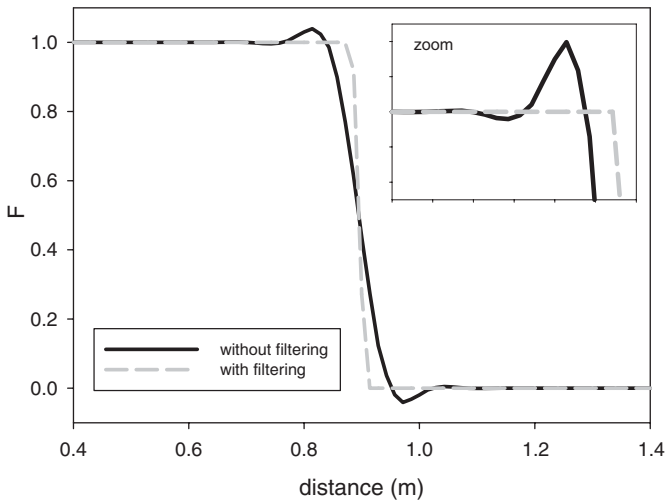

Figure 1. Illustration of the efficiency of the filtering technique.

where α is a scaling factor that is set to 5 in this work (refer to Reference [27] for more details on the value of α). It is important to underline that, by construction, the function $\overline{F}$ also satisfies Equation (3). The direct consequence of this property is that one can take as initial and boundary conditions for $\tilde{F}$ those for $\overline{F}$. $\overline{F}$ can be recovered *a posteriori* by applying transformation (4).

3.3. Flow equations

The momentum equation (1) and the continuity equation (2) are solved directly in a decoupled manner by means of the Uzawa algorithm [33]. The equations are discretized by means of the discontinuous-pressure quadratic $P_2^+ - P_1$ Crouzeix–Raviart element [34]. The first-order implicit Euler scheme is used to integrate the momentum equation with respect to time. The values of the dynamic viscosity μ and density ρ are determined as a weighted average with respect to the value of $\overline{F}$:

$$\varphi = \varphi_1 \overline{F} + (1 - \overline{F})\varphi_2 \tag{5}$$

where φ_1 and φ_2 denote the properties (dynamic viscosity or density) of phases 1 and 2, respectively.

3.4. Automatic mesh refinement technique

An automatic mesh refinement technique is required as part of the interface capturing strategy to model accurately the shock-like behaviour of this interface and to keep it as sharp and as smooth as possible. The mesh refinement technique used in this work and described in Reference [31], relies on one single coarse reference mesh that is refined locally around the interface, using the values of the colour function $\tilde{F}$ as a remeshing indicator.

Figure 2 illustrates the four steps of this technique in the case of the Rayleigh–Taylor flow problem. First, the contour line $\tilde{F} = 0.5$ is used to determine the elements to be refined on the reference mesh of the computational domain (Figures 2(a) and (b)). In fact, they correspond to

DOI: 10.1002/fld

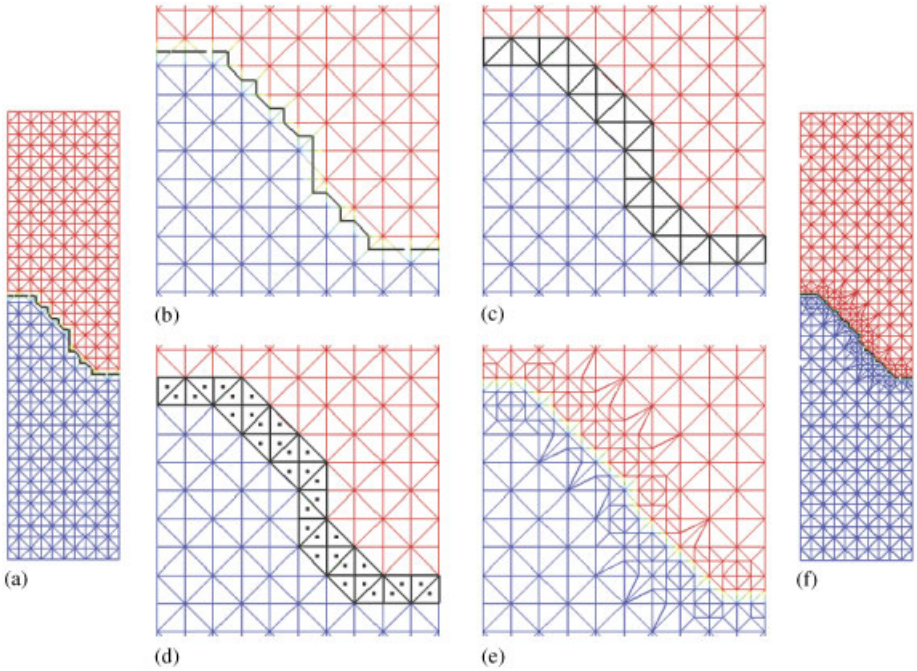

Figure 2. Automatic mesh refinement technique: (a) reference mesh with contour line $F = 0.5$; (b) zoom of (a); (c) selection of elements to be refined; (d) creation of a control point for each element to be refined; (e) generation of new locally refined mesh; and (f) zoom out of (e) with contour line $F = 0.5$.

those elements for which $\tilde{F}$ lies between preset values $\tilde{F} = 0.5 \pm \varepsilon$, where ε is a parameter that controls the width of the region to refine (Figure 2(c)). The targeted elements are next flagged with control points arbitrarily located at their centre (Figure 2(d)). The mesh is then adapted for each element lying within a preset distance from these control points. The element is split into four elements in such a way that the regularity of the child elements is preserved, and a further refinement step in neighbouring elements may be required, owing to the presence of hanging nodes to satisfy finite element continuity (Figures 2(e) and (f)). Note that this procedure can be repeated to generate meshes that are finer and finer in the vicinity of the fluid interface.

In our current implementation, the number of mesh refinements or iterations is set by the user, but a criterion based on the variation of $\tilde{F}$ between two successive iterations could be used for more flexibility. In this study, the maximum number of iterations was arbitrarily fixed to 2 to have the same minimum mesh size Δh at the interface for the fine uniform grid and for the locally refined grid (see Section 4.1). Moreover, the mesh size away from the interface for the latter grid case corresponds to the mesh size of the coarse grid.

The algorithm of the overall method is summarized in Figure 3 for the steady-state and the unsteady-state cases. This algorithm will now be briefly discussed for the unsteady-case. First, $\tilde{F}$ must be initialized. Then, at each time step, the following set of actions must be taken: $\tilde{F}$ is filtered according to (4) to generate $\overline{F}$, from which physical parameters $\mu = \mu(\overline{F})$ and $\rho = \rho(\overline{F})$ can be obtained; the flow equations and the advection equation are solved for $\mathbf{v}$, p and $\tilde{F}$; the interface is

DOI: 10.1002/fld

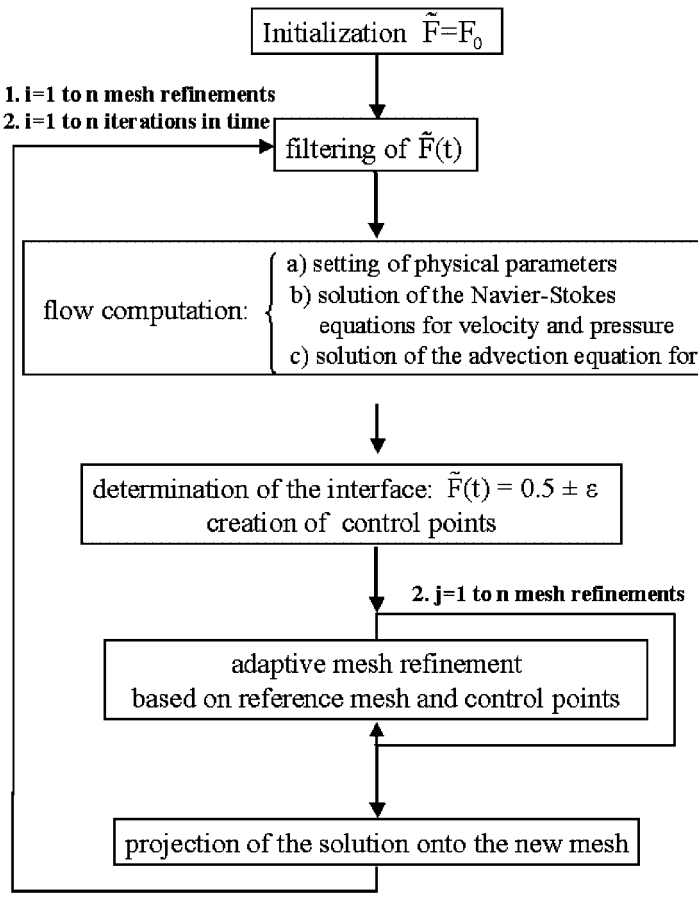

Figure 3. Schematic of the overall algorithm: 1 refers to the steady-state case and 2 to the unsteady-state case.

updated from $\tilde{F}$, and control points are generated according to $\tilde{F} = 0.5 \pm \varepsilon$; the reference mesh is then adapted by performing one or more iterations of the automatic refinement procedure described in this section; finally, the current solution is projected onto the new mesh using standard finite element interpolation before going to the next time step. For the steady-state case, many refinement passes can be achieved, each of which consists of recomputing the current solution after fine-tuning the position of the interface and adapting the mesh accordingly.

4. RESULTS AND DISCUSSION

In this study, the classical viscous Rayleigh–Taylor (RT) flow problem is solved to assess the efficiency of the interface capturing method.

DOI: 10.1002/fld

4.1. Problem definition

The domain is a rectangular box of dimension $L \times 4L$, with $L = 0.25$ m, as shown in Figure 4. Symmetric boundaries are used on the left and the right sides of the domain. At the top and at the bottom of the domain, slip conditions are imposed. The initial pressure is the hydrostatic pressure, and a slight perturbation is applied initially at the interface, defined by $y = 4L/2 + 0.1\cos(\pi x/L)$, where (x, y) represents the co-ordinates of the domain.

The two fluids are incompressible with a density ratio ρ_1/ρ_2 and a dynamic viscosity ratio μ_1/μ_2 both equal to 2, which gives the same kinematic viscosity $v = \mu/\rho$ for both fluids. More precisely, ρ_1 is equal to 998 kg m^{-3} and μ_1 to 20 Pa s.

Three meshing strategies were considered for the simulations of this work. The first one is based on a fixed and structured mesh of $2 \times 10 \times 40$ regular triangular elements (Figure 5(a)), which corresponds to a mesh size Δh of 0.025 m and yields 440 equations for the advection problem and 3198 velocity equations for the flow problem. The second strategy is also based on a fixed mesh but the latter is globally refined with $2 \times 40 \times 160$ regular triangular elements and $\Delta h = 0.00625$ m, which corresponds to 6601 pseudo-concentration unknowns and 51 198 velocity unknowns (Figure 5(b)). Finally, the third strategy is based on the $2 \times 10 \times 40$ finite element mesh, which is adapted at the interface at each time step by performing two passes of the automatic refinement procedure of Section 3.4. This scenario led to meshes with Δh comprised between 0.00625 and 0.025 m, a number of pseudo-concentration unknowns between 745 and 2062, and a number of velocity unknowns between 5599 and 16 127. Figure 5(c) displays the mesh obtained with this strategy at $t = 0$ s. A summary of the characteristic of the meshes can be

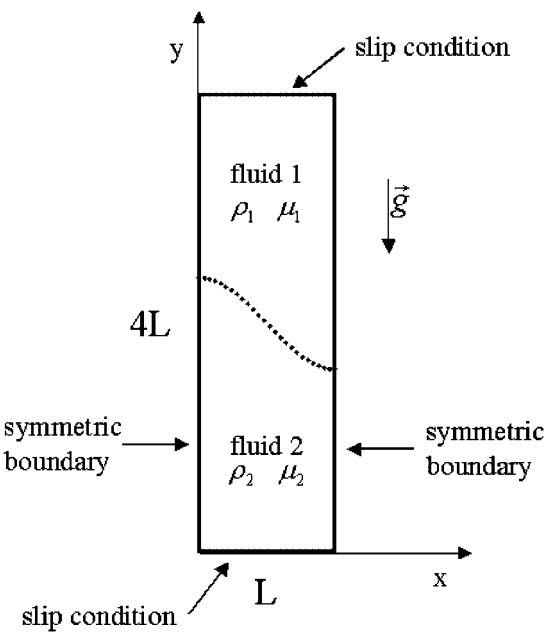

Figure 4. Geometry for the Rayleigh–Taylor flow problem.

DOI: 10.1002/fld

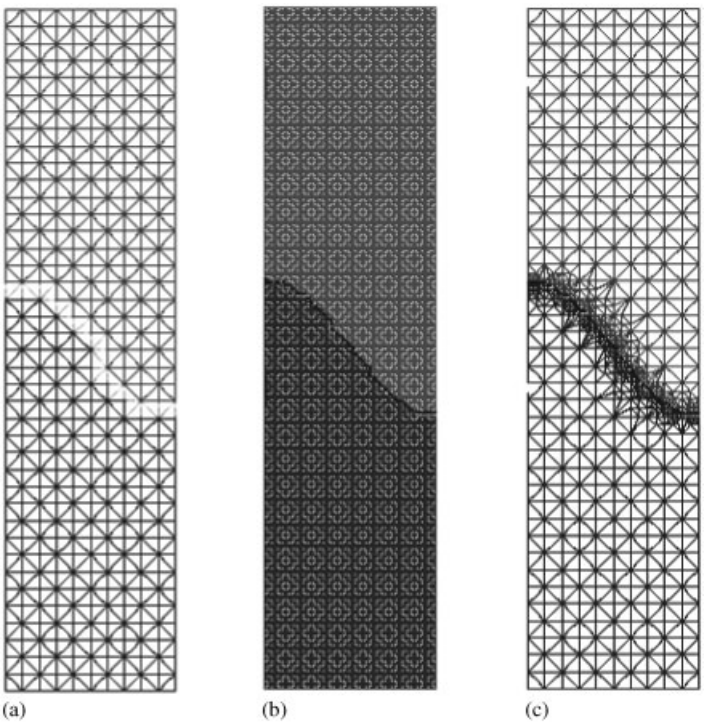

Figure 5. Meshes for the Rayleigh–Taylor flow problem at $t = 0$ s: (a) structured mesh with $2 \times 10 \times 40$ regular triangular elements; (b) structured mesh with $2 \times 40 \times 160$ regular triangular elements; and (c) $2 \times 10 \times 40$ locally refined mesh.

Table I. Characteristics of the meshes for the different strategies.

Strategy	Mesh	Number of unknowns	
		Pseudo-concentration	Velocity
1	$2 \times 10 \times 40$ structured mesh	440	3198
2	$2 \times 40 \times 160$ structured mesh	6601	51 198
3	Locally refined reference mesh at $t = 0$ s	745	5599
3	Locally refined reference mesh at $t = 1.1$ s	2062	16 127
	$2 \times 80 \times 320$ structured mesh	26 001	20 4798

found in Table I. In all the cases discussed below, a transient simulation was carried out using a time step $\Delta t = 5 \times 10^{-3}$ s, which was determined using a CFL rule with the finest mesh.

4.2. Results

First, to assess the solution accuracy with the $2 \times 40 \times 160$ finite element mesh, the solution obtained with this mesh at $t = 1.1$ s was compared to that obtained with an even finer mesh

DOI: 10.1002/fld

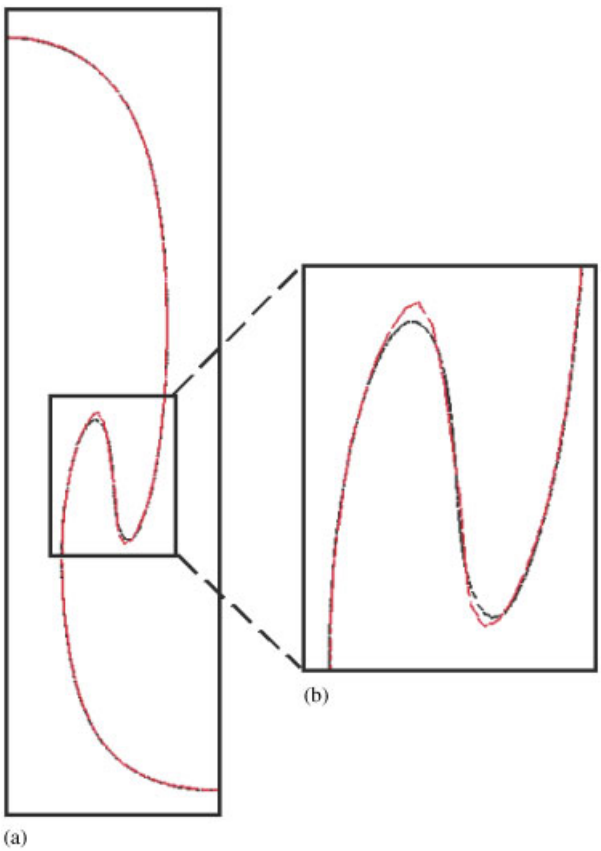

Figure 6. (a) Comparison of the solution at $t = 1.1$ s with a $2 \times 40 \times 160$ finite element mesh (in red) and a $2 \times 80 \times 320$ finite element mesh (in black); and (b) zoom of (a).

comprising $2 \times 80 \times 320$ regular mesh (26 001 and 204 798 equations for the advection and the Stokes equations, respectively). As can be seen in Figure 6, both solutions are similar except near the tip of the 'mushroom'. The difference between the two velocity fields (not shown here) is also small, with a maximum relative error on the velocity less than 0.7%. For these reasons and as it is less time consuming, the $2 \times 40 \times 160$ finite element mesh was chosen as the reference solution mesh.

Figure 7 presents at different times (from 0.3 to 0.9 s) the velocity field and the contour line $F = 0.5$ obtained with the three meshing strategies summarized in Table I. The patterns seem to be globally the same, but some differences are highlighted in Figure 8 that displays the contour line $F = 0.5$ for the three meshing strategies at $t = 0.9$ s. In particular, it shows that the interface predicted with two passes of the automatic mesh refinement algorithm is comparable to that obtained with the $2 \times 40 \times 160$ finite element mesh. However, the interface is poorly described with the $2 \times 10 \times 40$ finite element mesh. These observations are confirmed in Figure 9, which compares the maximum velocity predicted with the three meshing strategies. It shows that the $2 \times 40 \times 160$ finite element mesh and the locally refined mesh yield similar maximum velocity

DOI: 10.1002/fld

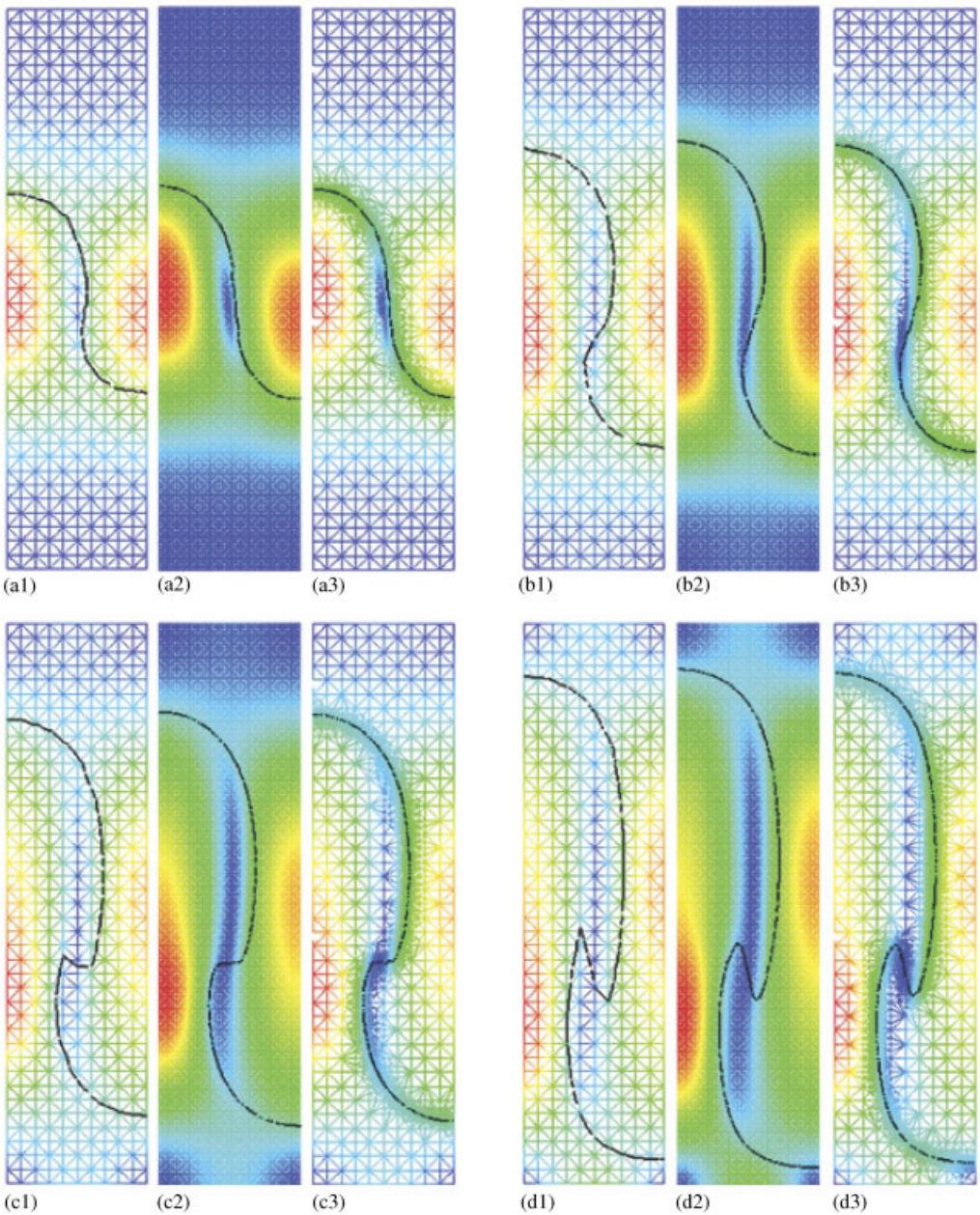

Figure 7. Velocity fields and contour lines $F = 0.5$ for the: (1) $2 \times 10 \times 40$ finite element mesh (meshing strategy #1); (2) $2 \times 40 \times 160$ finite element mesh (meshing strategy #2); and (3) locally refined mesh (meshing strategy #3), at various times: (a) $t = 0.3$ s; (b) $t = 0.5$ s; (c) $t = 0.7$ s; and (d) $t = 0.9$ s.

DOI: 10.1002/fld

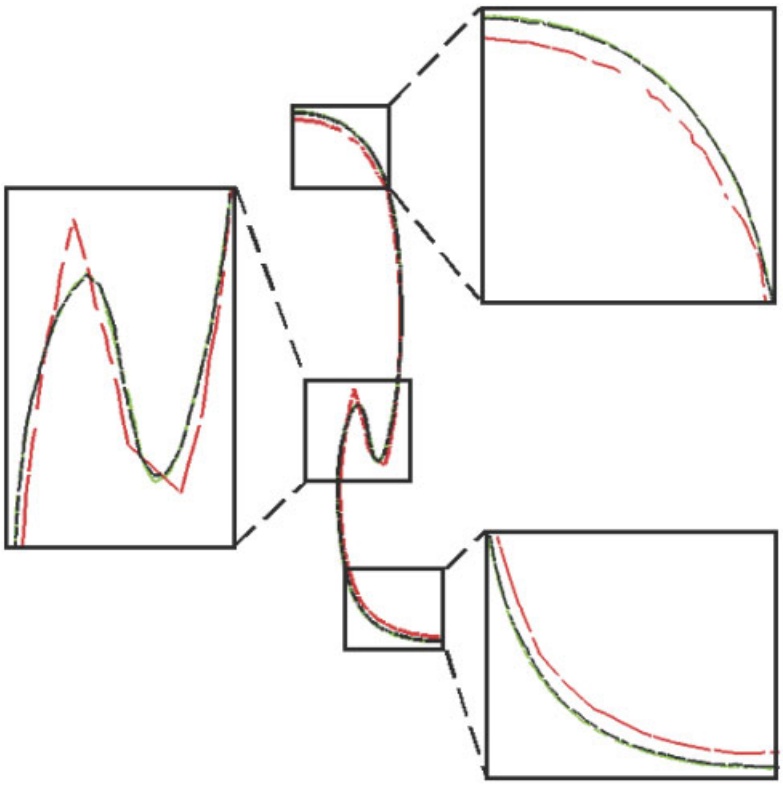

Figure 8. Comparison of the contour line $F = 0.5$ for the $2 \times 10 \times 40$ finite element mesh (in red), the locally refined mesh (in black) and the $2 \times 40 \times 160$ finite element mesh (in green) at $t = 0.9$ s.

variations over time. In fact, the relative differences are smaller than 3.5%. This difference goes up to 8.6% between the $2 \times 10 \times 40$ finite element mesh and the locally refined mesh.

Figure 10 illustrates the importance of the filtering technique. The pseudo-concentration field is shown without and with the filtering technique in Figures 10(a) and (b), respectively. It is pretty clear that with a coarse grid, the diffusion is important around the interface (Figure 10(a1)). As the mesh size is reduced (Figure 10(a2) and (a3)), the diffusion is less important around the interface. The combination of a fine mesh and the filtering technique gives accurate results as shown in Figures 10(b2) and (b3) for the strategy 2 and 3, respectively. In the first case (Figure 10(b1)), the mesh is too coarse to give good results.

As it is an important point of this work, the numerical conservation of F was studied. Let us denote by F_h and $\mathbf{u}_h$ the computed colour function and the velocity field, respectively. The conservation equation can then be integrated over the computational domain Ω to yield the conservation error E:

$$E = \int_{\Omega} \left(\frac{\partial F_h}{\partial t} + \operatorname{div}(F_h \mathbf{u}_h) \right) \mathrm{d}\Omega \tag{6}$$

DOI: 10.1002/fld

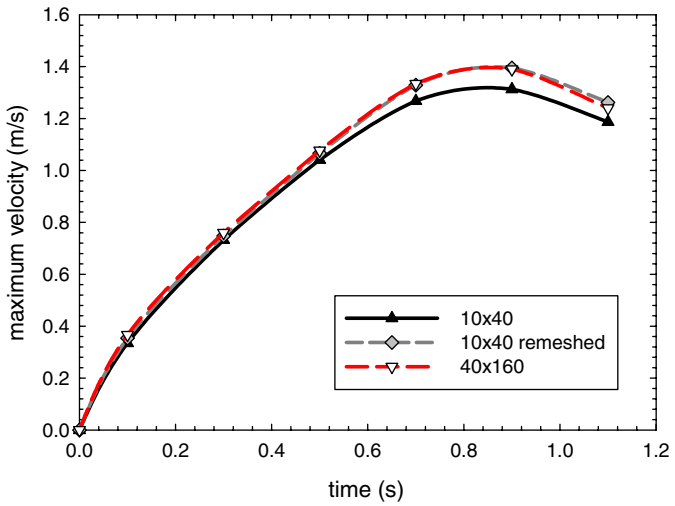

Figure 9. Maximum velocity ($\mathrm{m\,s^{-1}}$) as function of time (s).

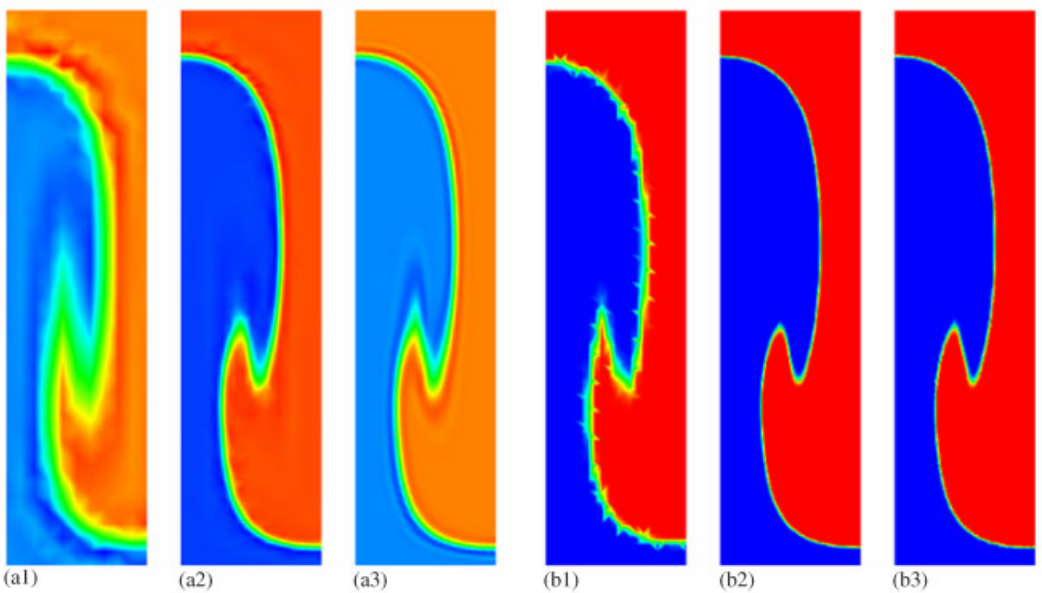

Figure 10. (a) Non-filtered solution; and (b) filtered solution, at $t = 0.9\,\mathrm{s}$ for the different strategies: (1) $2 \times 10 \times 40$ finite element mesh; (2) $2 \times 40 \times 160$ finite element mesh; and (3) automatic mesh refinement strategy.

As can be seen in Figure 11 that displays the variation of E with respect to time for the coarse $2 \times 10 \times 40$ finite element mesh, the global conservation error E increases from 0.2×10^{-4} to 0.2×10^{-3} after 1.1 s. For the fine $2 \times 40 \times 160$ finite element mesh, the variation is much less

DOI: 10.1002/fld

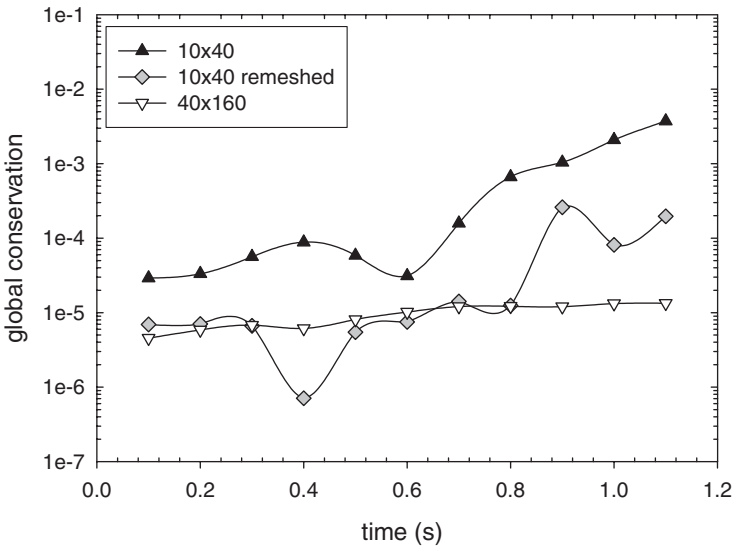

Figure 11. Variation of error on mass conservation with respect to time.

steep and smaller than 10^{-5}. For the automatic mesh refinement strategy, the error oscillates but is significantly smaller than that with the coarse mesh. Despite this oscillatory behaviour, the error is comparable or smaller than that of the fine mesh before $t = 0.8$ s, after which it increases more rapidly. Note that time $t = 0.8$ s corresponds to the onset of the formation of the tips of the mushroom, which makes the shape of the interface more complex.

Figure 12 shows the local conservation of fluid mass over the domain at time $t = 0.7$ s, for the three meshing strategies. In all three cases, the interface is clearly the place where the local error is higher. One may also notice that local mass conservation is best satisfied with the fine $2 \times 40 \times 160$ finite element mesh, closely followed by the local refined meshing strategy. Mass conservation with the coarse $2 \times 10 \times 40$ finite element mesh is not as well satisfied.

5. CONCLUSION

In this work, an efficient finite element-based strategy for the two-phase Navier–Stokes equations was presented. The overall simulation strategy includes an interface capturing method that is based on a filtering technique for the colour function F and an automatic mesh refinement procedure to be applied in the vicinity of the interface. The strategy was described in detail and then applied to the solution of the classical Rayleigh–Taylor flow problem.

It was shown in particular that the proposed strategy is efficient and yields solutions that are of comparable accuracy as those that can be obtained using a fine mesh with many more equations to solve.

In future work, the proposed strategy will be applied to other multiphase flow problems of industrial relevance. The influence of the integration scheme will also be investigated since recent

DOI: 10.1002/fld

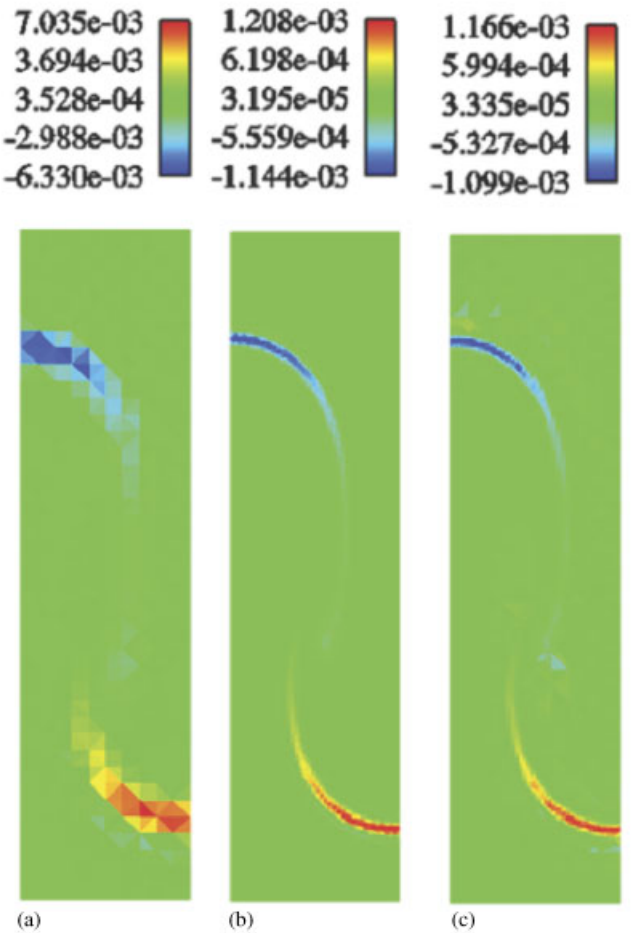

Figure 12. Fringes showing local mass conservation over the domain for the: (a) $2 \times 10 \times 40$ finite element mesh; (b) $2 \times 40 \times 160$ finite element mesh; and (c) automatic mesh refinement strategy, at time $t = 0.7$ s.

work by Malidi *et al.* [32] showed that the time scheme used may have a significant impact on the accuracy of the solution.

ACKNOWLEDGEMENTS

The financial contribution of the Natural Sciences and Engineering Research Council of Canada (NSERC) is gratefully acknowledged.

REFERENCES

1. Ghorai S, Nigam KDP. CFD modeling of flow profiles and interfacial phenomena in two-phase flow in pipes. *Chemical Engineering and Processing* 2006; **45**:55–65.
2. Chang C, Nguyen QD, Rønningsen HP. Isothermal start-up of pipeline transporting waxy crude oil. *Journal of Non-Newtonian Fluid Mechanics* 1999; **87**:127–154.

DOI: 10.1002/fld

3. Vachal P, Garimella RV, Shashkov MJ. Untangling of 2D meshes in ALE simulations. *Journal of Computational Physics*, submitted, on the web at http://cnls.lanl.gov/~shashkov/
4. Smolianski A. Finite-element/level-set/operator-splitting (FELSOS) approach for computing two-fluid unsteady flows with free moving interfaces. *International Journal for Numerical Methods in Fluids* 2005; **48**:231–269.
5. Prosperetti A, Tryggvason G. Appendix 3: Report of study group on computational physics. *International Journal of Multiphase Flow* 2003; **29**:1089–1099.
6. Johnson NL. The legacy and future of CFD at Los Alamos. *CFD 2004*, *12th Annual Conference of the CFD Society of Canada*, Ottawa, Canada, 9–11 May 2004.
7. Rider WJ, Kothe DB. A marker particle method for interface tracking. *Presented at the 6th International Symposium on Computational Fluid Dynamics*, 4–8 September 1995, on the web at http://www-xdiv.lanl.gov/XHM/personnel/wjr/Web_papers/pubs.html
8. Hirt CW, Nichols BD. Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics* 1981; **39**:201–225.
9. Kothe DB, Rider WJ. Comment on modeling interfacial flows with volume of fluid methods. On the web at http://www-xdiv.lanl.gov/XHM/personnel/wjr/Web_papers/pubs.html
10. Ashgriz N, Poo JY. FLAIR: flux line-segment model for advection and interface reconstruction. *Journal of Computational Physics* 1991; **93**:449–468.
11. Scardovelli R, Zaleski S. Interface reconstruction with least-square fit and split Eulerian–Lagrangian advection. *International Journal for Numerical Methods in Fluids* 2003; **41**:251–274.
12. Dyadechko V, Shashkov M. Moment-of-fluid interface reconstruction. *LA-UR-05-7571*, Los Alamos National Laboratory, 2006, on the web at http://math.lanl.gov/~vdyadechko/doc/2005-mof.pdf
13. Rider WJ, Kothe DB. Reconstructing volume tracking. *Journal of Computational Physics* 1998; **141**:112–152.
14. Sussman M, Smereka P, Osher S. A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational Physics* 1994; **114**:146–159.
15. van der Pijl SP, Segal A, Vuik C, Wesseling P. A mass-conserving level-set for modelling of multi-phase flows. *International Journal for Numerical Methods in Fluids* 2005; **47**:339–361.
16. Dufour S, Pelletier D. Computations of multiphase flows with surface tension using an adaptive finite element method. *Numerical Heat Transfer*, *Part A* 2001; **40**:335–362.
17. Lock N, Jaegger M, Medale M, Occelli R. Local mesh adaptation technique for front tracking problems. *International Journal for Numerical Methods in Fluids* 1998; **28**:719–736.
18. Fraigneau Y, Guermond JL, Quartapelle L. Approximation of variable density incompressible flows by means of finite elements and finite volumes. *Communications in Numerical Methods in Engineering* 2001; **17**:893–902.
19. Zhao Y, Tan HH, Zhang B. A high-resolution characteristics-based implicit dual time-stepping VOF method for free surface flow simulation on unstructured grids. *Journal of Computational Physics* 2002; **183**:233–273.
20. Rudman M. Volume-tracking methods for interfacial flow calculations. *International Journal for Numerical Methods in Fluids* 1997; **24**:671–691.
21. Biausser B, Guignard S, Marcer R, Fraunié P. 3D two-phase flows numerical simulations by SL-VOF method. *International Journal for Numerical Methods in Fluids* 2004; **45**:581–604.
22. Youngs DL. Time-dependent multi-material flow with large fluid distortion. In *Numerical Methods for Fluid Dynamics*, Morton KW, Baines MJ (eds). Academic Press: New York, 1982; 273–285.
23. Popinet S, Zaleski S. A front-tracking algorithm for accurate representation of surface tension. *International Journal for Numerical Methods in Fluids* 1999; **30**:775–793.
24. Puckett EG, Almgren AS, Bell JB, Marcus DL, Rider J. A high-order projection method for tracking fluid interfaces in variable density incompressible flows. *Journal of Computational Physics* 1997; **130**:269–282.
25. Rider WJ, Douglas B, Kothe DB, Mosso J, Cerutti JH. Accurate solution algorithms for incompressible multiphase flows. *Presented at the 33rd Aerospace Sciences Meeting and Exhibit*, Reno NV, 9–12 January 1995, on the web at http://www-xdiv.lanl.gov/XHM/personnel/wjr/Web_papers/pubs.html
26. Tryggvason G. Numerical simulations of the Rayleigh–Taylor instability. *Journal of Computational Physics* 1988; **75**:253–282.
27. Devals C, Heniche M, Bertrand F, Hayes RE, Tanguy PA. A finite element strategy for the solution of interface tracking problems. *International Journal for Numerical Methods in Fluids* 2005; **49**:1305–1327.
28. Brooks AN, Hughes TJR. Streamline upwind Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 1982; **32**:199–259.

DOI: 10.1002/fld

29. Devals C, Heniche M, Bertrand F, Tanguy PA, Hayes RE. A filtering technique for solving the advection equation in two-phase flow problems. *Proceedings of CFD 2004*, *12th Annual Conference of the CFD Society of Canada*, Ottawa, Canada, 9–11 May 2004.
30. Bertrand F, Thibault F, Delamare L, Tanguy PA. Adaptive finite element simulations of fluid flow in twin-screw extruders. *Computers and Chemical Engineering* 2003; **27**:491–500.
31. Devals C, Heniche M, Bertrand F, Tanguy PA, Hayes RE. A finite element solver for the advection equation applied to interface tracking. *Proceedings of Numiform 2004*, *Materials Processing and Design*: *Modeling*, *Simulation and Applications*. *AIP Conference*, *Proceedings 712*, Ghosh S, Castro JM, Lee JK (eds). Columbus, Ohio, U.S.A., 13–17 June 2004.
32. Malidi A, Dufour S, N'dri D. A study of time integration schemes for the numerical modelling of free surface flows. *International Journal for Numerical Methods in Fluids* 2005; **48**:1123–1147.
33. Fortin M, Glowinski R. *Augmented Lagrangian Methods*. North-Holland: Amsterdam, 1983.
34. Crouzeix M, Raviard PA. Conforming and non-conforming finite element methods for solving the stationary Stokes equation. *RAIRO*, *Série Analyse Numérique* 1973; **7**:33–76.

DOI: 10.1002/fld