Heat Transfer Engineering, 35(5):537–547, 2014
Copyright © Taylor and Francis Group, LLC
ISSN: 0145-7632 print / 1521-0537 online
DOI: 10.1080/01457632.2013.833407

Taylor & Francis
Taylor & Francis Group

3D ALE Finite-Element Method for Two-Phase Flows With Phase Change

GUSTAVO ANJOS,[1] NORBERTO MANGIAVACCHI,[2] NAVID BORHANI,[1] and JOHN R. THOME[1]

[1]Heat and Mass Transfer Laboratory (LTCM), Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland
[2]Mechanical Engineering Department, State University of Rio de Janeiro (UERJ), Rio de Janeiro, Brazil

Downloaded by [18.101.24.94] at 04:23 07 February 2014

We seek to study numerically two-phase flow phenomena with phase change through the finite-element method (FEM) and the arbitrary Lagrangian–Eulerian (ALE) framework. This method is based on the so-called "one-fluid" formulation; thus, only one set of equations is used to describe the flow field at the vapor and liquid phases. The equations are discretized on an unstructured tetrahedron mesh and the interface between the phases is defined by a triangular surface, which is a subset of the three-dimensional mesh. The Navier–Stokes equation is used to model the fluid flow with the inclusion of a source term to compute the interfacial forces that arise from two-phase flows. The continuity and energy equations are slightly modified to take into account the heat and mass transport between the different phases. Such a methodology can be employed to study accurately many problems, such as oil extraction and refinement in the petroleum area, design of refrigeration systems, modeling of biological systems, and efficient cooling of electronics for computational purposes, which is the aim of this research. A comparison of the obtained numerical results to the classical literature is performed and presented in this paper, thus proving the capability of the proposed new methodology as a platform for the study of diabatic two-phase flows.

INTRODUCTION

Today, most of the cooling devices found in personal computers and data centers use either single-phase air or water cooling systems. With microprocessor performance increasing exponentially, an efficient and better way to cool and decrease the computer chip temperature is of utmost importance. Furthermore, there are plans for a substantial increase of the number of chips per motherboard to go into multilayer stacks of chips with internal cooling channels, since higher computational resources are continuously required. It is known that the heat exchange of two-phase flow systems is much higher than for those using single-phase flow, mainly due to the nature of the thermal behavior of each phase in the presence of an interface layer separating both fluids. Therefore, a new cooling technique is proposed to maintain simultaneously the temperature of two or more stacked microprocessors, within an optimal working range, by flowing and evaporating two-phase environmentally friendly refrigerants in between. These operating fluids are responsible for removing the excessive heat produced by the processors; however, the cooling channels are limited to the order of 100 μm size.

Despite the available cutting-edge experimental techniques, a deeper insight into the microscale flow field is necessary. However, to access such a small length scale accurately, different techniques are required. In this context, numerical analysis has become an useful tool to simulate the mechanisms of two-phase flows, due to the fast growth of computer resources and the reduction of cost compared to those of experimental facilities. In fact, the modeling of such conditions is not an easy task, due to the complexity of the nonlinear set of equations that govern the flow field. Moreover, the characterization of surface tension forces and the interfacial deformation between the vapor and liquid phases add another level of complexity, all of which require ssignificant efforts to resolve in two-phase flow simulations.

The attempt to solve numerically two-phase flows splits the research in two main categories, namely, "one-fluid" and "two-fluids" formulations. The former uses one set of equations to describe all the phases and it is assisted by a color function, which defines their regions in the domain. The latter describes each phase as a separate fluid with its own set of governing equations, thus requiring an additional mathematical formulation

This research is funded by the Nano-Tera RTD project CMOSAIC (ref. 123618) financed by the Swiss Confederation and scientifically evaluated by SNSF.

Address correspondence to Professor John R. Thome, Heat and Mass Transfer Laboratory (LTCM) at the Swiss Federal Institute of Technology–EPFL, Lausanne, Switzerland. E-mail: john.thome@epfl.ch

Downloaded by [18.101.24.94] at 04:23 07 February 2014

for them. Within the "one-fluid" formulation, the interface description may be divided in two subareas, namely, Eulerian and Lagrangian descriptions. The basic difference between the two methodologies is reflected in the modeling of the interface between the phases. In the Eulerian description, the computational mesh is fixed on the space and an additional hyperbolic equation is required to describe the motion of the interface. Despite its relative ease of handling strong interface distortions, the discretization of such an equation may introduce artificial diffusion, thus leading to leak of accuracy, for instance, in the volume-of-fluid [1] and the level-set [2] methods. On the other hand, the Lagrangian formulation describes the interface between fluids explicitly by computational elements. Such a description allows a sharp representation of the front, but its drawback is the proper treatment of topological changes in the interface, since coalescence and breakups are not inherent to its methodology. The volume-tracking [3] and the front-tracking [4] methods are far the most widely used. Due to the shortcomings of purely Eulerian and purely Lagrangian formulations, the arbitrary Lagrangian–Eulerian (ALE) description allows these two frameworks to be combined in one single formulation so that the best aspect of each separate approach can be used in conjunction; that is, the computational mesh nodes may move with the continuum in normal Lagragian fashion or be held fixed in Eulerian manner. The ALE description has shown to be suitable to describe fluid flow problems (see, e.g., [5]), and this work extends its capability to two-phase flows with phase change.

Two-phase flow problems become even more interesting if phase change occurs, where the mass transfer from one phase to another adds significant complexity to the dynamics of bubbles and droplets. However, the modeling of such phenomena is not an easy task and should be treated with extreme care. Unfortunately, the related literature, in the numerical domain, is not as widely developed for mass transfer in single- and two-phase flows, compared to that for no phase change taking place. Despite the problems related to the modeling of phase change, efforts have been invested to develop tools capable of predicting, to a certain extent, boiling and condensation processes in two-phase flows. The pool boiling process was extensively reviewed in reference [6], in which four basic mechanisms were identified that contribute to the total heat flux, namely, evaporation at the liquid interface, enhanced natural convection, natural convection, and transient conduction at a nucleation site. It was also found that these mechanisms are strongly linked to the temperature of the superheated wall. Reference [7] presented a new model to simulate two-phase flows with phase change in two-dimensional domains. The new formulation was included in the previous front-tracking adiabatic code developed by reference [8], to extend its computation to boiling flows. Since the interface is represented by geometrical objects, special treatment of phase change was considered. The authors of reference [8] also studied the influence of several parameters in the interface temperature. To validate their model, they compared it to the exact solution of a one-dimensional test case, followed by the simulation of film boiling with different fluid properties.

In this work, we present an extension of the previously developed three-dimensional code ([9] and [5]), to simulate two-phase flows with phase change that allows the simulation of complex problems involving the liquid–vapor interface. The equations are written in the arbitrary Lagrangian–Eulerian description and discretized by the finite-element method. An additional equation is required to model the temperature field, and, due to the mass transfer occurring in the interface, the mass conservation equation is slightly modified to take into account boiling and/or condensation. In the following sections of this paper, the mathematical formulation used to model two-phase flows with phase change is detailed. Moreover, the interface representation and the employed remeshing technique are briefly described. Latter, validations for the surface tension term and simulations of the implemented phase change algorithm are presented. Finally, conclusions from this study are discussed.

MATHEMATICAL FORMULATION

Let us consider vapor and liquid occupying a domain where phase change occurs. In the "one-fluid" approach, one set of governing equations, namely, for momentum and for conservation of mass and energy, are written for both phases with the assistance of a step function, which takes into account the jump of properties in the interface separating the phases. Assuming that the fluids in each phase are incompressible and the change of volume, led by the phase change, occurs only at the interface, the momentum equation is written in the arbitrary Lagrangian–Eulerian nondimensional form as:

$$\rho\left[\frac{\partial \mathbf{u}}{\partial t}+(\mathbf{u}-\hat{\mathbf{u}})\cdot\nabla\mathbf{u}\right]=-\nabla p+\frac{1}{Re}\nabla\cdot[\mu(\nabla\mathbf{u}+\nabla\mathbf{u}^{T})]+\frac{1}{Fr^{2}}\rho\mathbf{g}+\frac{1}{We}\kappa\nabla H \tag{1}$$

On the left-hand side of Eq. (1), the term $u-\hat{u}$ represents the relative velocity between the flow field u and the mesh $\hat{u}$ at one time step. The gravity term is represented by g, and the system pressure by p and time by t. Re, We, and Fr are the Reynolds, Weber, and Froude numbers, respectively. The term $\kappa\nabla H$ represents the surface tension force, which is computed based on a new three-dimensional extension of Frenet's formula. Here, κ is the surface's curvature and H is the Heaviside function, which defines the region of each fluid in the domain, so that the inner fluid is 1, the outer fluid is 0, and the surface is 0.5. The fluid properties of density ρ and viscosity μ are kept constant at each phase, but do not necessarily assume equal values. Thus, the Heaviside function is used to defined a generic property Φ distribution along the domain as follows:

$$\Phi=\Phi_{1}H+\Phi_{2}(1-H) \tag{2}$$

Within the ALE context, the energy transport equation should take into account the relative velocity $u-\hat{u}$, thus representing the

heat convection. The energy equation is written in the arbitrary Lagrangian–Eulerian nondimensional form as:

$$\rho c_p \left[\frac{\partial T}{\partial t} + (u - \hat{u}) \cdot \nabla T \right] = \frac{1}{Re\,Pr} [\nabla \cdot k \nabla T + \dot{q} |\nabla H|] \quad (3)$$

In the preceding equation, T is the temperature distribution, c_p is the heat capacity, and k is the heat conductivity, both distributed on the domain in the same manner as the density and viscosity. Pr is the Prandtl number and $\dot{q}$ is the mass flux, defined as:

$$\dot{q} = k \nabla T \quad (4)$$

In mass transfer problems, the interface should be moved according to evaporation or condensation effects; thus, the velocity field is no longer divergence free in the vicinity of the interface. Therefore, the continuity equation is slightly modified to take into account such a phenomenon, so that

$$\nabla \cdot u = \left(1 - \frac{\rho_l}{\rho_v} \right) \dot{q} |\nabla H| \quad (5)$$

Since the gradient of the Heaviside function is only different from zero near the interface, the fluid equation becomes incompressible far from it. The normal velocity of the interface u_I is found to be

$$u_I = \frac{1}{2}(u_l + u_v) + \frac{\dot{q}}{2} \left(1 + \frac{\rho_l}{\rho_v} \right) \quad (6)$$

The new position of the interface nodes x_I may be found by integrating

$$\frac{dx_I}{dt} = u_I n \quad (7)$$

where n is the outward normal vector.

INTERFACE DESCRIPTION AND REMESHING

Unlike the descriptions found in most front-tracking codes, the interface mesh implemented in this work is a subset of the tetrahedron domain mesh; that is, each triangle shares a face of two adjacent tetrahedral elements, and thus the interface is sharply defined. Moreover, due to the finite-element formulation, the fluid properties are kept constant inside the mesh element and no smoothing functions is required to treat the jump conditions at the interface. As can be seen in Figure 1a, the interface between the phases is not part of the computational mesh and it should be defined by a function, which is advected using an additional equation. Moreover, the transition of properties (Φ_1, Φ_2) should be calculated using a smooth function to avoid numerical instabilities. On the other hand, in the Lagrangian approach (Figure 1b), the properties Φ_1 and Φ_2 are sharply defined in each region, thus not requiring any additional function.

In two-phase flows within the moving mesh context, a remeshing process is extremely necessary since the flow field, either due to the motion of a single bubble or due to the imposed velocity condition, tends to move the mesh nodes from one region to another, damaging the uniformity of node distributions. In this work, the interface between the phases is part of the computational mesh and a surface remeshing is also required to keep the element connectivity consistently while the simulation is running. Thus, two sets of data are stored during the simulations, namely, volumetric nodes and surface mesh. The former consists of the mesh nodes distributed on the three-dimensional space, and the latter stands for the interface and boundary triangular meshes. Each set of data is treated separately and exported to an external mesh generator library [10], which is responsible for generating the tetrahedron connectivity array at each time step. The volumetric node distribution is done by solving the Helmholtz equation, in which the boundary conditions consist in the characteristic edge lengths of each surface mesh; thus, a smooth transition between the interface

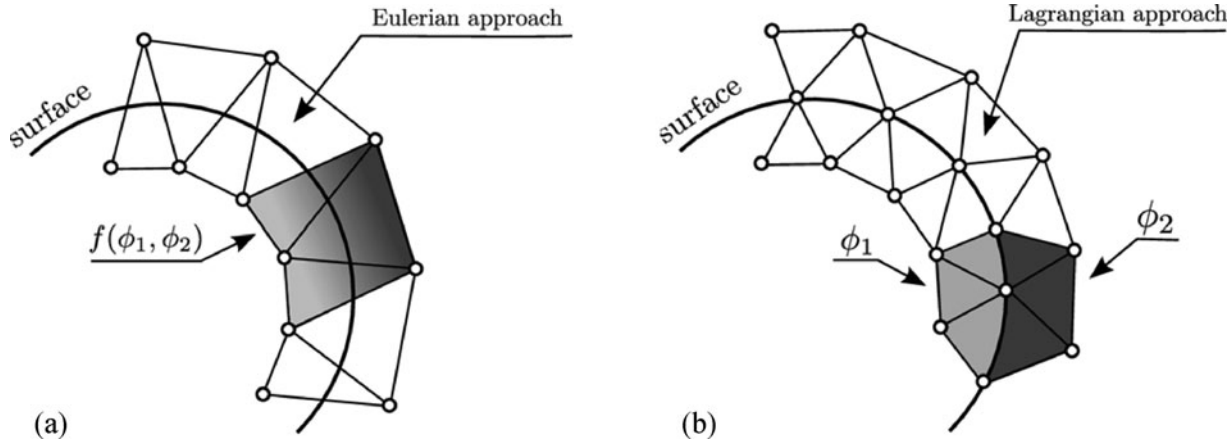

Figure 1 Two-dimensional representation of the interface between the phases. (a) In the Eulerian approach, the interface is implicitly defined as a scalar function, which is advected by an additional equation. (b) A set of geometrical elements is used to represent the interface in the Lagrangian approach, and thus no additional equation is required.

Downloaded by [18.101.24.94] at 04:23 07 February 2014

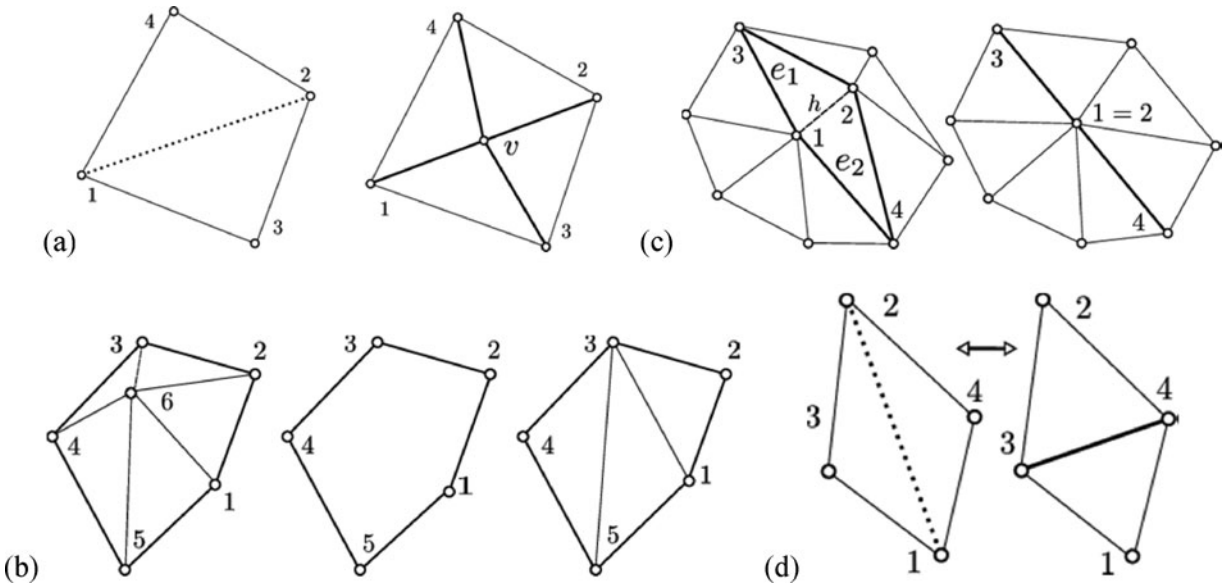

Figure 2 Basic surface mesh operations: (a) point insertion, (b) edge contraction, (c) point deletion, and (d) edge flipping.

Downloaded by [18.101.24.94] at 04:23 07 February 2014

and the boundary meshes is successfully achieved. Note that the treatment of the volumetric nodes does not require the maintenance of the tetrahedron's connectivity. On the other hand, the surface meshes should be modified, at each time step, to keep the triangle's shape bounded to good aspect ratios. This is achieved by the insertion/deletion of points, edge flipping, and edge contractions.

Figure 2a shows the insertion procedure, which aims to occupy "barren" areas and to increase the accuracy in certain regions where a higher precision is required. If an edge $e > e_{max}$, a new point v is inserted by considering the curvature of the adjacent nodes. Figure 2b shows the deletion of a surface node, which is required when the moving mesh causes the clustering of points in the surface mesh. The strategy attempts to find an edge e that is smaller than a predefined length e_{min}, which is defined according to the simulation requirements. Once such an edge is detected, the sum of the edge length of the one-ring neighbors is performed for both extremity vertices. The one that has the lowest value is then eliminated from the surface mesh. Once the vertex deletion is performed, its neighbors form a polyhedron, which must be subdivided to recover the triangular structure. Such a task is done based on the work of [11] and [12] and extended here to triangular surface meshes.

Figure 2c shows the strategy of edge contraction adopted, which is based on the work of [13]. Once an edge $e < e_{min}$ is detected, such a scheme aims to collapse the two extremity vertices into the midpoint of the edge e, and thus the two adjacent triangles are removed from the surface mesh. After being contracted, the edge e is no longer part of the surface mesh, and thus points 1 and 2 occupy the same position while points 3 and 4 remain in the same location. The benefit of such an approach compared to the previous point deletion strategy is its geometrical simplicity, since the surrounding connectivity of the mesh is not affected. Note that the same consideration of the curvature of two adjacent points found in the insertion strategy should be taken into account when collapsing two vertices, thus avoiding displacement errors and undesired losses of mass. The last strategy, but not less important, is shown in Figure 2d. The edge-flipping strategy attempts to restore the mesh quality changing the connectivity of two consecutive elements by comparing the quality of the pair of triangles (1–2–3 and 1–4–2) to the new pair (1–4–3 and 2–3–4). This quality is based on the sum of triangle aspect ratio, the curvature of neighbors points, the sum of area, and the circumcenter of each triangle, and is performed for the entire surface mesh. All these operations are made systematically; thus, the mesh is kept regularized and the simulation can reach the final solution.

Due to the separation of the domain and the surface mesh in the above procedure, the mesh distribution treatments may be combined into a scheme and adjusted by parameters varying from 0 to 1, which is a three-dimensional generalization of the approach presented by reference [14] for two-dimensional simulations. The domain and surface velocities are therefore treated as follows:

$$\hat{\mathbf{u}}(\mathbf{x}) = \begin{cases} \mathbf{u} - \gamma_1(\mathbf{u}\cdot t)t + \gamma_2(\mathbf{u}_e \cdot t)t & \text{if } \mathbf{x} \text{ belongs to the interface} \\ \beta_1\mathbf{u} + \beta_2\mathbf{u}_v + \beta_3\mathbf{u}_e & \text{if } \mathbf{x} \text{ does not belong to the interface} \end{cases} \quad (8)$$

In such a method, due to the description of the interface mesh by computational elements, the surface should move according to the fluid motion. In the preceding equation, if x belongs to the interface, we can define its velocity as u_I. Thus, it is convenient to decompose it into two orthogonal components: u_{In} and u_{It}, which represent the normal and tangential velocities, respectively. To decrease the displacement of nodes in the tangential

Downloaded by [18.101.24.94] at 04:23 07 February 2014

direction, one may remove partially, or even totally, its velocity from the total interface's velocity. This can be achieved by either projecting the interface's velocity u_{In} to the normal vector associated to the node or, in a simpler manner, by removing the tangent component from the total surface mesh velocity $u_{It} = \mathbf{u} - (\mathbf{u} \cdot t)t$. Therewith, the parameter γ_1 controls the magnitude of the tangent velocity in the total interface's velocity. Letting $\gamma_1 = 1$, only the normal interface's velocity is taken into account in the surface mesh motion, and therefore the surface nodes are not allowed to move in the tangent direction. Additionally, the parameter γ_2 includes the smoothing scheme on the surface mesh nodes, thus keeping them all bounded within good aspect ratio. (For more details about the smoothing technique used in this work, see reference [5].) The parameter β_1 controls the Lagrangian motion of the inner and outer volumetric mesh velocity. By setting $\beta_1 = 1$, the flow velocity u is fully included in the moving mesh velocity $\hat{\mathbf{u}}$ and, consequently, the volumetric nodes move according to the flow field. Otherwise, letting $\beta_1 = 0$, the flow velocity $\mathbf{u}$ is not taken into account in the moving mesh velocity. The parameters β_2 and β_3 control the intensity of the velocity smoothing scheme $\mathbf{u}_v$ and the Laplacian smooth scheme $\mathbf{u}_e$ in the moving mesh velocity. Thus, setting both parameters to null, the volumetric mesh smoothing is not performed. Note that the parameters γ and β may vary from 0 to 1 to achieve a desirable node distribution according to the simulation requirements.

NUMERICAL PROCEDURE

The complete set of equations (momentum, continuity and energy) is discretized through the finite-element method. A summary of the underlying principles is given next, and further details can by found in references [9] and [5].

The Galerkin method is applied to the variational form of the governing equations to discretize all terms except the nonlinear convective term. To overcome the obstacle of modeling and implementing a numerical scheme for this term, the semi-Lagrangian technique is employed (for details, see [15] and [16]) based on the idea of representing the acceleration field by a Lagrangian point of view instead of the well-known Eulerian derivative. For each time step the points are moved toward the flow, and once the task is accomplished the coordinate system is reinitialized and the original mesh is recovered. The substantial derivative is evaluated in the strong form along the characteristic trajectory, by estimating the position of a point and solving the equation Dx/Dt backward in time, $t^{n+1} \geq t \geq t^n$, with the initial condition $x(t^{n+1}) = x_i$; then an integration method is used to evaluate the previous point position in the grid. A first-order discretization scheme is adopted, assuming the trajectory as a straight line. The choice of element types for coupled partial differential equation (PDE) problems must take into account the Babuska–Brezzi condition in order to preserve the stability properties intrinsic to the discretization scheme of references [17] and [18]. Such a condition is not mandatory, but must be fulfilled if used in conjunction with the Galerkin method. For these reasons and considering the excellent mass conservation property, the Taylor–Hood $P1^+ - P1$ element (Mini-element) was used to represent pressure and temperature, both calculated at the tetrahedron vertices, and the velocities, by evaluating them at the tetrahedron vertices and its centroid, thus avoiding critical numerical instabilities and pressure oscillations. Once the discretization of the domain is accomplished, the system matrices are assembled and the solution of the time dependent three-dimensional equations is then found by successively solving the linear system in each time step for pressure, velocity, and temperature. Due to the strong coupling between pressure and velocity, the numerical procedure implemented to solve the mentioned linear system uses the Projection method based on the LU decomposition, which was first introduced by reference [19]. The aim of this method is to uncouple pressure and velocity and solve each quantity separately, thus reducing the large linear system size into smaller ones. Additionally, the temperature equation is solved separately and it does not require the same methodology. The solution of the linear system for pressure and velocity is described in the following, followed by the solution of the temperature field.

Let us define the matrix B and the right-hand side vector r^n according to the finite-element method as

$$B = \frac{M_\rho}{\Delta t} + \frac{K}{Re} r^n = \frac{M_\rho}{\Delta t} u_d^n \tag{9}$$

where M_ρ and K are the mass and stiff matrices, respectively. The time step is represented by Δt, Re is the Reynolds number, and u_d^n is the velocity calculated in the previous time step at the departure points from the semi-Lagrangian method. The system is uncoupled and solved in the following way by computing the trial velocity $\tilde{u}$ and solving the linear system with proper velocity boundary conditions bc_1 as:

$$B\tilde{u} = r^n + bc_1 \tag{10}$$

An update of $\tilde{u}$ is performed while considering the gravity g and the surface tension force f:

$$\tilde{u}_{corr} = \tilde{u} + \Delta t M_{\rho L}^{-1} \left(M_\rho g + \frac{1}{We} M f \right) \tag{11}$$

Note that the subscript L refers to the lumped matrix technique, in which a diagonalization is performed to the consistent matrix M_ρ to reduce numerical costs in its inversion. The mass transfer contribution across the interface Z is represented by this expression:

$$Z = \left[1 - \frac{\rho_l}{\rho_v} \right] \dot{q} |G_T H| \tag{12}$$

where ρ_l and ρ_v are the densities for the liquid and vapor phases, respectively, $\dot{q}$ is the mass flux, and $G_T H$ is the gradient of the Heaviside function, defined by the temperature mesh nodes. The pressure P^{n+1} is calculated by solving the linear system

with pressure boundary conditions bc_2:

$$DB_L^{-1}Gp^{n+1} = -D\tilde{u}_{corr} + Z + bc_2 \tag{13}$$

where G is the gradient matrix, D is the divergence matrix, and B_L^{-1} is the inverted lumped matrix B. The velocity solution u^{n+1} is found from

$$u^{n+1} = \tilde{u}_{corr} + B_L^{-1}\, Gp^{n+1} \tag{14}$$

On the other hand, the solution of the linear system of the temperature equation is straightforward. Let define the matrix B_T and r_T^n as

$$B_T = \frac{M_T}{\Delta t} + \frac{K_T}{PrRe} r_T^n = \frac{M_T}{\Delta t} T_d^n + \dot{q}|G_T H| \tag{15}$$

where M_T and K_T are the mass and stiff matrices of the temperature equation, respectively, Pr is the Prandtl number, and Re is the Reynolds number. The temperature calculated in the previous time step at the departure points is represented by u_d^n. Thus, the solution of the temperature is found by solving the following linear system for T with temperature boundary conditions bc_T:

$$B_T T = r_T^n + bc_T \tag{16}$$

The solutions of the velocity and temperature linear systems are obtained iteratively at each time step by the conjugate gradient method and preconditioned by the incomplete Cholesky factorization. The pressure is solved by the generalized minimum residual method and preconditioned by an incomplete LU factorization. Such a methodology has shown to be suitable to solve the linear systems of two-phase flow problems with phase change.

Table 1 Fluid properties

Fluid	ρ (kg/m^3)	μ (μPa-s)	σ (N/m)	k (W/m-K)	c_p (J/kg-K)	δ (—)
R134a, vapor	37.54	12.04	0.0074	0.0173	1065	—
R134a, liquid	1187	185.4	0.0074	0.079	1446	—
Sucrose	1.172	5.650	77.7	—	—	0.0617
Glycerol	1.260	712.0	63.1	—	—	0.1483
Air	1.789	1.225	—	—	—	—

RESULTS

In this section, a numerical test is carried out to validate the code against the terminal velocity and liquid film thickness of elongated bubbles in circular channels. The wall effects are included in the bubble dynamics and the results are compared to the well-known flow pattern map of reference [20] and Brown's theoretical solution for the film thickness [21]. The modeling of such flows increases significantly the obstacles of the remeshing process, since the formation of the thin liquid film between the bubble and wall requires a fine mesh to capture the flow mechanisms. The flexibility of the finite-element method is explored in the development of the computational meshes, in which different mesh element sizes are used in the three-dimensional domain.

Isothermal Rising of Taylor Bubbles

Figure 3 shows the time progression of a Taylor air bubble immersed in a sucrose solution (see Table 1). The same bubble shape and film thickness as in the previous cases are used as an initial shape. In the transient evolution, the bubble's velocity reached its maximum velocity at time $t \approx 1$, and its terminal

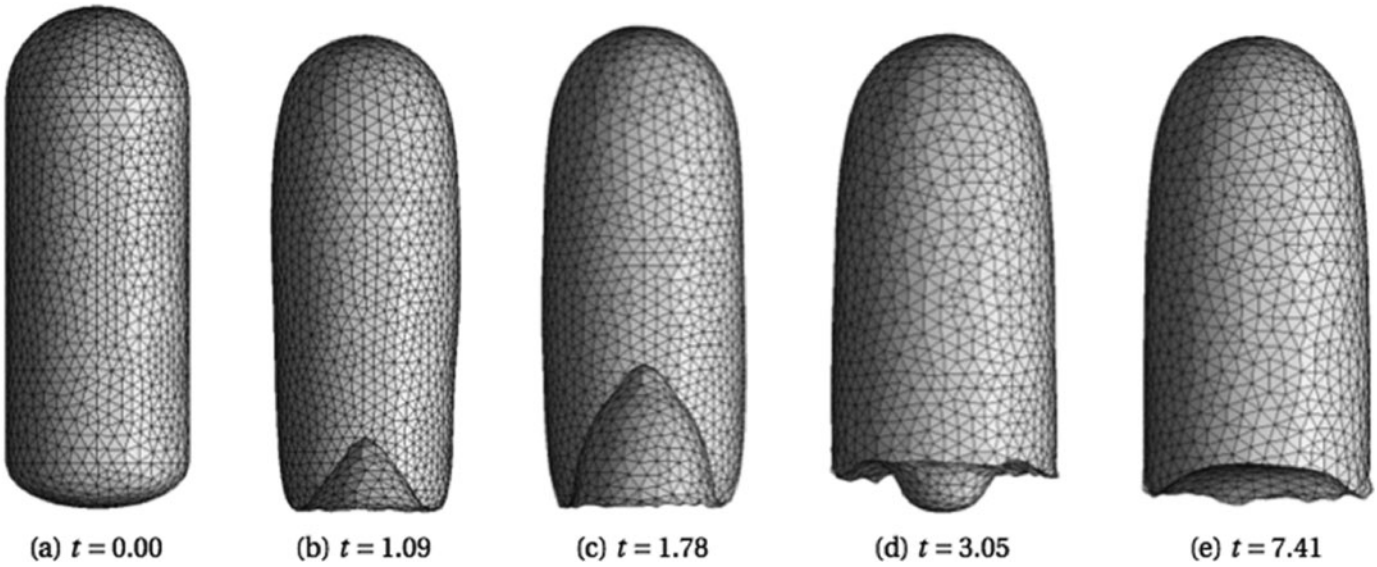

Figure 3 Bubble shape evolution with time for an air bubble in a sucrose solution with dimensionless numbers Mo = 10^{-7}, Fr = 1, and We = 40. The adaptive mesh refinement proposed in this work captures accurately the strong shape distortion produced by the high ascension velocity. (a) Initial bubble shape with $t = 0$. (b–d) Bubble shape change during transient solution. (e) Terminal bubble shape with $t = 7.41$. (Color figure available online.)

Downloaded by [18.101.24.94] at 04:23 07 February 2014

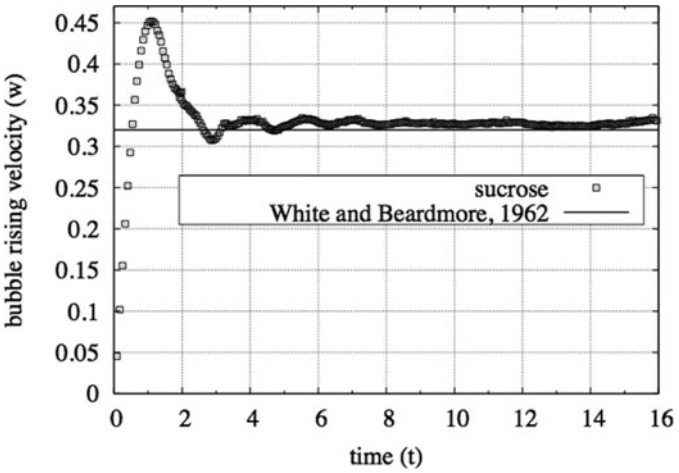

Figure 4 Rising of an air Taylor bubble immersed in a sucrose solution with dimensionless numbers set to Mo = 10^{-7}, Fr = 1, and We = 40. The time evolution of the bubble's center of mass velocity is compared to the terminal bubble's velocity found in White and Beardmore [20]. Velocity and time are nondimensional.

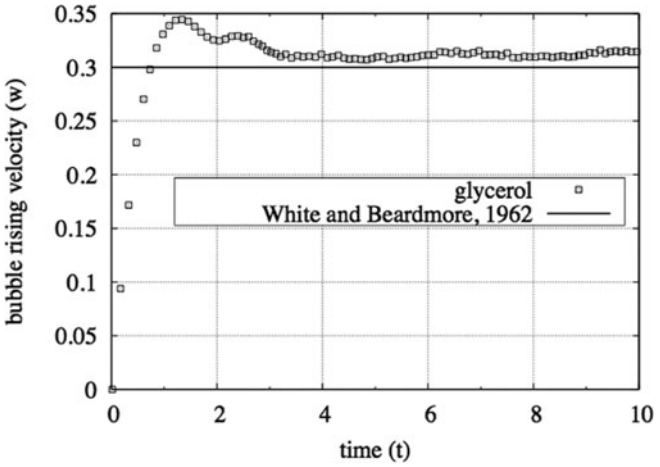

Figure 6 Rising of an air Taylor bubble immersed in a glycerol solution with dimensionless numbers set to Mo = 10^{-2}, Fr = 1, and We = 100. The time evolution of the bubble's center of mass velocity is compared to the terminal bubble's velocity found in White and Beardmore [20]. Velocity and time are nondimensional.

velocity at time $t \approx 3.7$. Also, it was shown that the bottom part of the bubble was pulled in and oscillated until convergence at $t \approx 7.4$. The mesh parameters used in this simulation were $\beta_1 = 0.0$, $\beta_2 = 1.0$, $\gamma_1 = 1.0$, and $\gamma_2 = 0.1$, and the dimensionless numbers were set to Mo = 10^{-7}, Fr = 1, and We = 40. Figure 4 presents the transient solution of the bubble's center of mass velocity. An overshooting of the ascension velocity was observed from time t = 0 to t = 1.1, due to the initial deformation of the bottom part of the bubble, and consequently acceleration of the center of mass. The result agreed with the prediction of the flow pattern map, obtaining an error of 1.1%.

Figure 5 shows the time progression of a Taylor air bubble immersed in a glycerol solution (see Table 1). The mesh parameters used in this simulation were $\beta_1 = 0.0, \beta_2 = 0.5, \gamma_1 = 0.5$, and $\gamma_2 = 0.1$, and the dimensionless numbers was set to Mo = 10^{-2}, Fr = 1, and We = 100. In this case, the mesh parameters were slightly modified to test different mesh conditions in the simulation. The obtained result shows that the bubble shape and terminal velocity agree well with experimental data. Figure 6 shows the evolution in time of the bubble's center of mass velocity, in which the computed terminal velocity approaches the value found in the flow pattern map. The error was found to be 6.5%.

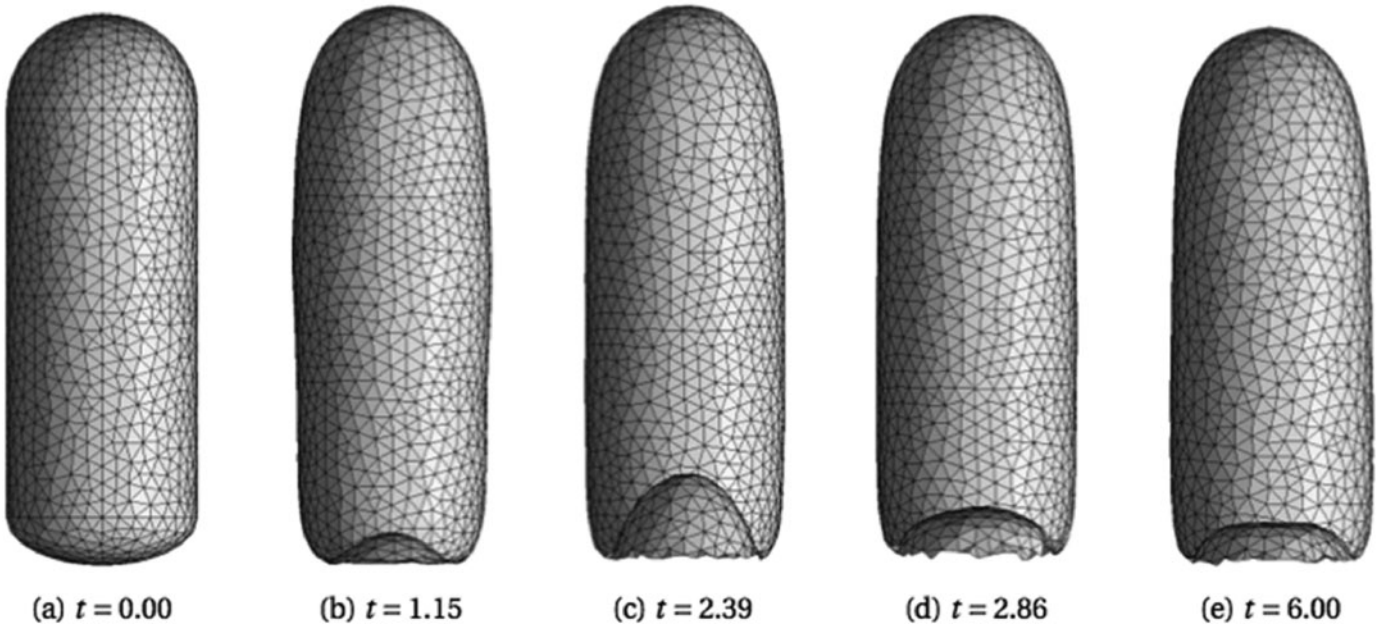

Figure 5 Bubble shape evolution with time for an air bubble in a glycerol solution with dimensionless numbers Mo = 10^{-2}, Fr = 1, and We = 100. (a) Initial bubble shape with $t = 0$. (b–d) Bubble shape transient solution. (e) Terminal bubble shape with $t = 6.00$. (Color figure available online.)

Downloaded by [18.101.24.94] at 04:23 07 February 2014

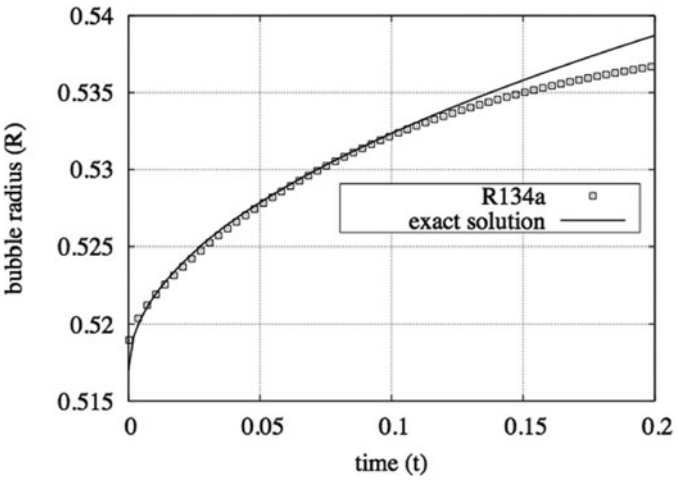

Figure 7 Growth of a vapor bubble due to evaporation.

Spherically Growing Bubble in a Superheated Liquid

To validate the proposed mass transfer model, a spherical vapor bubble is simulated in a superheated liquid with the fluid properties shown in Table 1. The numerical domain was set as a cubic shape and an outflow condition was imposed at all the walls, thus allowing the bubble to expand equally to all directions. Initially, a vapor bubble with its temperature equal to the saturation temperature *Tsat* is placed on the middle of the domain. The surface temperature is set to the saturation temperature, and thus the only difference in temperature is due to the superheated liquid surrounding the vapor bubble. Considering that the temperature is nondimensionalized such that $T^* = (\mathrm{T} - Ts)/(\mathrm{Tw} - Ts)$, the saturation temperature is set to $Ts = 0$ and the temperature of the superheated liquid is set to $T = 0+$. The superheated liquid at the given saturation pressure starts to evaporate; thus, the total liquid volume decreases and the vapor

Downloaded by [18.101.24.94] at 04:23 07 February 2014

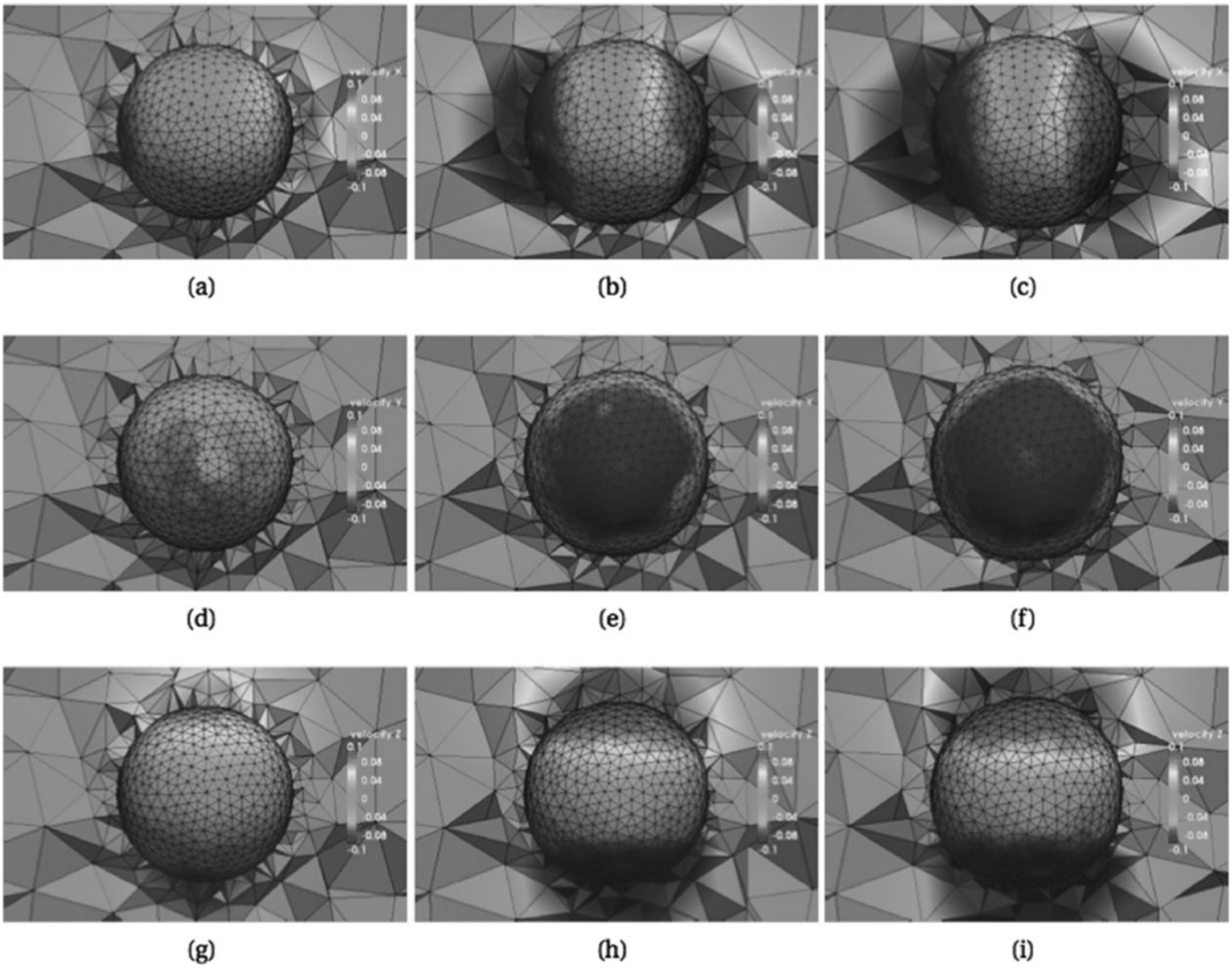

Figure 8 Velocity components of a vapor bubble growing due to evaporation of the superheated liquid: (a–c) x-component, (d–f) y-component, and (g–i) z-component. (Color figure available online.)

volume increases. At this point, an analytical solution can be used to compare the change of vapor volume with time. Such an expression is given by

$$R(t) = 2\beta\sqrt{kt} \tag{17}$$

where R is the bubble's radius, k is the thermal conductivity, and β is a constant calculated in reference [22]. Note that the viscous and surface tension effects are neglected during the simulation; thus, the preceding relationship is valid for the growth of the vapor bubble.

Figure 7 shows the growth of the vapor bubble with time due to the evaporation process. From $t \approx 0.02$ to $t \approx 0.12$, the vapor bubble expands linearly due to heat conduction. From $t \approx 0.13$ to the end of the simulation, the bubble asymptotically reduces its expansion and begins to deviate from the analytical solution, due to different assumptions used to derive the above equation with respect to those in the numerical simulation. Figure 8 depicts the three velocity components, showing the vapor bubble expansion. As can be seen, the velocity increases with time as well as the bubble radius.

CONCLUSIONS

This article presents a new methodology for simulating incompressible two-phase flows within the finite-element method (FEM) context in which the mesh moves in an arbitrary Lagrangian–Eulerian (ALE) fashion. The coupling ALE–FEM methodology provides a sharp representation of the interface between the phases, not only for the geometrical representation itself but also for the definition of the phase properties, thus resulting in a model that accurately describes the actual physical conditions. The proposed treatment of the computational mesh, splitting the surface meshes and the volumetric points, has been shown to be an excellent choice, thus avoiding the obstacles of handling the remeshing process over the tetrahedron mesh, allowing the utilization of a standard Delaunay tetrahedralization library. Moreover, the new adaptive meshing strategy achieves good control of the mesh quality during the simulations by keeping the volumetric and surface elements bounded to a satisfactory shape, thus preserving the accuracy of the calculation. However, excessive linear interpolations on the volumetric and surface mesh may lead to a lack of accuracy in certain domain areas, and this should be used with care. The heat and mass transfer was implemented into the code using the same strategy of the fluid flow solver. However, it requires an extensive reformulation of the isothermal method presented here. The thin thermal boundary layer requires an additional number of nodes and elements to be physically resolved, thus increasing significantly the processing time. The preliminary tests presented for evaporation in square microchannels qualitatively agreed with those observed experimentally. However, a deeper analysis is still required to compare the actual state of the phase change model with different benchmarks available in the literature.

NOMENCLATURE

B	matrix
B_T	matrix associated to energy equation
c_p	specific heat
f	surface tension force
Fr	Froude number
g	gravity
k	thermal conductivity
K	stiff matrix
M	mass matrix
M_T	mass matrix associated to energy
M_ρ	mass matrix
Mo	Morton number
n	outward normal vector
Pr	Prandtl number
q	heat flux
r_T^n	residual vector associated to energy equation
r^n	residual vector from current time step
Re	Reynolds number
t	time
T	temperature
Tw	wall temperature
Ts	saturation temperature
u	Eulerian velocity
u_I	velocity of the interface
u_d^n	velocity vector calculated from the departure points in the previous time step
$\tilde{u}$	trial velocity
$\mathbf{u_I}$	interface's velocity
$\hat{u}$	mesh velocity
We	Weber number
x_I	interface coordinate

Greek Symbols

μ	viscosity
ρ	density
κ	curvature
ψ	generic property
γ	surface mesh coefficient
β	volumetric mesh coefficient
σ	surface tension
δ	liquid film thickness

REFERENCES

[1] Hirt, C., and Nichols, B., Volume of fluid (VOF) method for the dynamics of free boundaries, *Journal of Computational Physics*, vol. 39, pp. 201–225, 1981.

[2] Sussman, M., Smereka, P., and Osher, S., A Level-Set Approach for Computing Solutions to Incompressible Two-Phase Flow, *Journal of Computational Physics*, vol. 114, pp. 146–159, 1994.

Downloaded by [18.101.24.94] at 04:23 07 February 2014

[3] Harlow, F. H., and Welch, J. E., Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid With Free Surface, *Physics of Fluids*, vol. 8, pp. 2182–2189, 1965.
[4] Glimm, J., Grove, J., Lindquist, W. B., McBryan, O., and Tryggvason, G., The Bifurcation of Tracked Scalar Waves, *SIAM Journal of Computations*, vol. 9, pp. 61–79, 1988.
[5] Anjos, G. R., A 3D ALE Finite Element Method for Two-Phase Flows with Phase Change, Lausanne, Ph.D. thesis, 2012.
[6] Dhir, V. K., Nucleate and Transition Boiling Heat Transfer Under Pool and External Flow Conditions, *International Journal of Heat and Fluid Flow*, vol. 21, pp. 290–314, 1991.
[7] Juric, D., and G. Tryggvason, G., Computations of Boiling Flows, *Journal of Computational Physics*, pp. 387–410, 1998.
[8] Unverdi, S. O., and Tryggvason, G., A Front-Tracking Method for Viscous, Incompressible, Multi-Fluid Flows, *Journal of Computational Physics*, vol. 100, pp. 25–37, 1992.
[9] Anjos, G., Hydrodynamics Field Solution of Electrochemical Cells Through Finite Element Method, Metallurgical and Materials Engineering, Federal University of Rio de Janeiro, Rio de Janeiro, Ph.D. Thesis 2007.
[10] Si, H., Adaptive Tetrahedral Mesh Generation by Constrained Delaunay Refinement, *International Jounal for Numerical Methods in Engineering*, vol. 46, pp. 856–880, 2008.
[11] O. Devillers, O., On Deletion in Delaunay Triangulations, *International Journal of Computational Geometry & Applications*, vol. 12, p. 285, 2002.
[12] Xu, X., Pain, C. C., Goddard, A. J. H., and Oliveira, C. R. E., An Automatic Adaptive Meshing Technique For Delaunay Triangulations, *Computer Methods in Applied Mechanics and Engineering*, vol. 161, pp. 297–303, 1998.
[13] Negami, S., Diagonal Flips in Triangulation of Surfaces, *Discrete Mathematics*, vol. 135, pp. 225–232, 1994.
[14] Souza, F. S., and Mangiavacchi, N., A Lagrangian Level-Set Approach for the Simulation of Incompressible Two-Fluid Flows, *International Journal for Numerical Methods in Fluids*, vol. 47, pp. 1393–1401, 2004.
[15] Robert, A., A Stable Numerical Integration Scheme for the Primitive Meteorological Equations, *Atmosphere Oceans*, vol. 19, pp. 35–46, 1981.
[16] Pironneau, O., On the Transport-Diffusion Algorithm and Its Applications to the Navier–Stokes Equation, *Numerische Mathematik*, vol. 38, pp. 309–332, 1982.
[17] Cuvelier, C., Segal, A., and van Steenhoven, A. A., *Finite Element Method and Navier–Stokes Equations*, Reidel Publishing Company, Dordrecht, The Netherlands, 1986.
[18] Zienkiewicz, O. C., and Taylor, R. L., *The Finite Element Method Volume 1: The Basis,* 5th ed., Wiley John and Sons, New York, NY, 2000.
[19] Chorin, A. J., Numerical Solution of the Navier–Stokes Equations, *Mathematics of Computations*, vol. 22, pp. 745–762, 1968.
[20] White, E. T., and Beardmore, R. H., The Velocity of Rise of Single Cylindrical Air Bubbles Through Liquids Contained in Vertical Tubes, *Chemical Engineering Science*, vol. 1998, pp. 351–361, 1962.
[21] Brown, R. A. S., The Mechanisms of Large Bubbles in Tubes I: Bubbles Velocities in Stagnant Liquids, *Canadian Journal of Chemical Engineering*, vol. 43, pp. 217–223, 1965.
[22] Scriven, L. E., On the Dynamics of Phase Growth, *Chemical Engineering Science*, vol. 10, pp. 1–13, 1959.

APPENDIX

The nondimensional numbers Re, Fr, We, Mo, and Pr are the Reynolds, Froude, Weber, Morton, and Prandtl numbers, respectively, which are important to characterize the fluid flow problem. These numbers are defined based on a characteristic length L and velocity U as follows:

$$\mathrm{Re} = \frac{\rho_0 \mathrm{UL}}{\mu_0} \quad \mathrm{Fr} = \frac{\mathrm{U}}{\sqrt{g_0 \mathrm{L}}} \quad \mathrm{We} = \frac{\rho_0 \mathrm{U}^2 \mathrm{L}}{\sigma_0}$$

$$\mathrm{Pr} = \frac{c_p \mu_0}{\kappa_0} \, \mathrm{Mo} = \frac{\mu_0^4 g}{\rho_0 \sigma_0^3} = \frac{\mathrm{We}^3}{\mathrm{Fr}\,\mathrm{Re}^4} \tag{18}$$

Here, g is the gravity, c_p is the fluid specific heat, k is the thermal conductivity, σ is the surface tension coefficient, and the subscript 0 denotes that the properties are taken from the referential fluid.

Gustavo Anjos received a B.Eng. degree in mechanical engineering in 2005 from the State University of Rio de Janeiro (UERJ), an M.Sc. degree in metallurgy and materials engineering in 2007 from the Federal University of Rio de Janeiro (UFRJ), and a Ph.D. degree in mechanical engineering in 2012 from the Swiss Federal Institute of Technology in Lausanne (EPFL). He is currently a postdoctoral associate at the Department of Nuclear Science & Engineering (NSE) at the Massachusetts Institute of Technology (MIT) in the United States. His research focuses on numerical simulations of two-phase flows with phase change for single and multiple bubbles.

Norberto Mangiavacchi is currently an associate professor and head of the Mechanical Engineering Department at the State University of Rio de Janeiro (UERJ). He worked as a research engineer at CSN-Companhia Siderurgica Brasileira from 1986 to 1996. He received his Ph.D. in mechanical engineering and scientific computing at the University of Michigan, Ann Arbor, in 1994. He received the 1995 Robert T. Knapp Award from the ASME Fluids Engineering Division. He worked as postdoctoral researcher at CERFACS, Toulouse, France, in 1995, and at COPPE/UFRJ, Rio de Janeiro, Brazil, in 1997. He also worked as an assistant professor at State University of São Paulo, Brazil, from 1998 to 2002. He joined UERJ in 2002, where he has

Downloaded by [18.101.24.94] at 04:23 07 February 2014

been head of the Graduate Program in Mechanical Engineering since 2006. He directs the GESAR laboratory, an interdisciplinary group that is concerned on numerical and experimental modeling of complex flow phenomena, applied to environmental problems. He hosted the II Workshop on Energy and Environment, Rio de Janeiro, in 2011.

Navid Borhani received a B.Eng. degree in aeronautical engineering in 1992 and a Ph.D. in aerospace engineering in 1996 from the University of Manchester. After a period of postdoctoral research at the University of Colorado's Program in Atmospheric and Oceanic Sciences between 1996 and 1998, he moved to the Swiss Federal Institute of Technology in Lausanne (EPFL) where he was a senior research associate in the Laboratory of Fluid Mechanics (LMF) from 1998 to 2008, and then in the Laboratory of Heat and Mass Transfer (LTCM) to date. His research focuses on experimental investigations of single- and multiphase fluid dynamics and instabilities using quantitative optical flow visualization techniques.

John R. Thome has been a professor of heat and mass transfer at the Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland, since 1998, where he directs a two-phase heat transfer research laboratory with 20+ post-docs and Ph.D. students. Recently, his work has primarily focused on investigation of the fundamental phenomena of microchannel two-phase flows (and proposing of a new smart map), mechanistic two-phase flow pattern-based heat transfer and pressure drop models for microscale evaporating flows, computerized flow control of two-phase microcooling systems, the development of multi-microchannel evaporators for electronics cooling, and numerical modeling of two-phase phenomena. He received his Ph.D. at Oxford University, England, in 1978. He is the author of four books: *Enhanced Boiling Heat Transfer* (1990), *Convective Boiling and Condensation,* 3rd edition (1994), *Wolverine Engineering Databook III* (2004), and *Nucleate Boiling on Micro-Structured Surfaces* (2008). He received the ASME Heat Transfer Division's Best Paper Award in 1998 for a three-part paper on two-phase flow and flow boiling heat transfer published in the *Journal of Heat Transfer* and recently the Very Highly Commended Paper Award from the *International Journal of Refrigeration* for 2011–2012. Prof. Thome received the UK Institute of Refrigeration's J. E. Hall Gold Medal in 2008 for his extensive work in the field of microscale refrigeration heat transfer and the 2010 ASME Heat Transfer Memorial Award for his work on flow pattern-based heat transfer models for macro- and microscale flows. He hosted the 8th ECI International Conference on Boiling and Condensation Heat Transfer in Lausanne in June 2012.

Downloaded by [18.101.24.94] at 04:23 07 February 2014