Proceedings of the ASME 2015 International Technical Conference and Exhibition on Packagin and Integration of Electronic and Photonic Microsystems and ASME 2015 12th International Conference on Nanochannels, Microchannels, and Minichannels
InterPACK/ICNMM 2015
July 6-9, 2015, San Francisco, California, USA

InterPACKICNMM2015-48441

ALE-FEM FOR TWO-PHASE FLOWS WITH HEAT AND MASS TRANSFER IN MICROCHANNELS

Gustavo R. Anjos*
gustavo.anjos@uerj.br
Norberto Mangiavacchi, Jose Pontes
norberto@uerj.br,jose.pontes@uerj.br

Group for Studies and Environmental
Simulations in Reservoirs
Department of Mechanical Engineering
State University of Rio de Janeiro
20940-903, Rio de Janeiro, RJ – Brazil
Tel: +55 21 2332-4733, Fax: +55 21 2332-4733
WWW: http://www.gesar.uerj.br

John Thome
Heat and Mass Transfer Laboratory
Swiss Federal Institute of Technology
Lausanne, Vaud, 1015, Switzerland
john.thome@epfl.ch

ABSTRACT

A numerical method is described to study two-phase flows for single and multiple bubbles with phase change. The fluid flow equations are based on the Arbitrary Lagrangian-Eulerian formulation (ALE) and the Finite Element Method (FEM), creating a new two-phase method with an improved model for the liquid-gas interface in microchannels. A successful adaptive mesh update procedure is also described for effective management of the mesh at the two-phase interface to remove, add and repair surface elements, since the computational mesh nodes move according to the flow. The Lagrangian description explicitly defines the two-phase interface position by a set of interconnected nodes which ensures a sharp representation of the boundary, including the role of the surface tension. The methodology proposed for computing the curvature leads to accurate results with moderate programming effort and computational cost and it can also be applied to different configurations with an explicit description of the interface. 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, modelling of biological systems and efficient cooling of electronics for computational purposes, being the latter the aim of this research. The obtained numerical results will be described, therefore proving the capability of the proposed new methodology.

NOMENCLATURE

D	Bubble's diameter
$\mathbf{u}$	Velocity vector
$\mathbf{x}$	Node coordinate
t	Time
$\hat{\mathbf{u}}$	Mesh velocity vector
$\mathbf{u}^T$	Transpose velocity vector
∇	Differential operator
p	Pressure
ρ	Density
ρ_v	Vapor phase density
ρ_l	Liquid phase density
μ	Viscosity

[*] Address all correspondence to this author.

Copyright © 2015 by ASME

μ_v	Vapor viscosity
μ_l	Liquid viscosity
c_p	Heat capacity
h_{fg}	Latent Heat of vaporization
H	Heaviside function
Re	Reynolds number
Fr	Froude number
We	Weber number
Pr	Prandtl number
T	Temperature
k	Thermal conductivity
$\dot{q}$	Heat flux
$\mathbf{u}_I$	Velocity vector at the interface nodes
$\mathbf{u}_{I_n}$	Normal velocity vector at the interface node
$\mathbf{u}_{I_t}$	Tangential velocity vector at the interface node
$\mathbf{u}_e$	Velocity vector based on Laplacian smooth operator
$\mathbf{u}_{vel}$	Velocity vector based on velocity smoothing scheme
$\mathbf{u}_l$	Velocity vector in liquid phase
$\mathbf{u}_v$	Velocity vector in vapor phase
U_∞	Free-stream velocity
$\mathbf{n}$	Normal direction vector
$\mathbf{t}$	Tangential direction vector
γ_1	Parameter for the tangent velocity at the interface node
γ_2	Parameter for Laplacian Smoothing at the interface node
β_1	Parameter for Lagrangian motion at volumetric mesh node
β_2	Parameter for Laplacian Smoothing at volumetric mesh node
β_3	Parameter for Velocity scheme at volumetric mesh node
L_b	Bubble's length
B	Mass and diffusion matrix
B_T	Mass and diffusion temperature matrix
M_ρ	Mass matrix
M_T	Mass temperature matrix
K	Diffusion matrix
K_T	Diffusion temperature matrix
D	Divergent matrix
G	Gradient matrix
G_T	Gradient temperature matrix
r^n	Right hand side vector
r_T^n	Right hand side temperature vector
u_d^n	Right hand side vector in departure
$\tilde{u}$	Computed velocity
bc_1	Boundary condition vector for velocities
bc_2	Boundary condition vector for pressure
bc_T	Boundary condition vector for temperature
g	Gravity vector
f	Surface tension force vector
$\tilde{u}\mathrm{corr}$	Corrected computed velocity
M_ρ^{-1}	Lumped mass matrix
B_L^{-1}	Lumped B matrix
n	Current time step
$n+1$	Next time step

INTRODUCTION

A thermal collapse concerning the cooling of computer microprocessors may be expected if an efficient and better way to cool and decrease the computer chip temperature is not achieved. Today, most of the cooling devices found in personal computers and datacenters use either single-phase air or water-cooling systems. Furthermore, a substantial increase of the number of chips per motherboard plans to go to multi-layer 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 are much higher than those using single-phase flow, mainly due to the nature of the thermal behavior of each phase in 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 microns 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 non-linear 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 adds another level of complexity, all of which require significant 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 for their. Within the one-fluid formulation, the interface description may be divided in two sub-areas, namely Eulerian and Lagrangian descriptions. The basic difference between both methodologies is reflected on 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 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 break-ups are not inherent to its methodology. The Volume-tracking Harlow and Welch, 1965 and the Front-tracking [3] methods are far the most widely used. Due to the shortcomings of purely Eulerian and purely Lagrangian formulations, the Arbitrary Lagrangian-Eulerian description allows these two frameworks to be combined in one single formulation so that the best aspect of each separately approach can be used in conjunction, that is, the computational mesh nodes may move with the continuum in normal Lagrangian fashion or to be held fixed in Eulerian manner. The ALE description has shown to be suitable to describe fluid flow problems (see, for instance [4]) 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 significantly 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 so widely developed for mass transfer in single and two-phase flows, compared to that for no phase change taking place. Despite the problems related on the modeling of phase change, efforts have been invested to develop tools capable to predict, to a certain extent, boiling and condensation processes in two-phase flows. The pool boiling process was extensively reviewed by [5], in which 4 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. In [6], they 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 [7] to extend its computation to boiling flows. Since the interface is represented by geometrical objects, special treatment of phase change was considered. They also studied the influence of several parameters in the interface temperature. To validate their model, they compared it to the exact solution of a 1-dimensional test case, followed by the simulation of film boiling with different fluid properties.

In this work, we present an extension of the previous developed 3-dimensional code [8, 9], to simulate two-phase flows with heat and mass transfer, which allows the simulation of complex problems involving 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 will be detailed. Moreover, the interface representation and the employed re-meshing technique will be shortly described. Latter, several numerical simulations will be presented and focused to validate the implemented phase change algorithm. Finally, conclusions from this study will be discussed.

NUMERICAL SIMULATION

Field Equations

Let us consider vapor and liquid occupying a domain where phase change occurs. In the 'one-fluid' approach, one set of governing equations, namely momentum, 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 non-dimensional form as:

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

On the left hand side of Eq. (1), the term $\mathbf{u}-\hat{\mathbf{u}}$ represents the relative velocity between the flow field $\mathbf{u}$ and the mesh $\hat{\mathbf{u}}$ at one time step. The gravity term is represented by g, while the system pressure by p and time by t. These equations are non-dimensionalized by defining the following non-dimensional parameters:

$$\rho=\rho_\infty\rho^* \qquad x=Lx^* \qquad \mathbf{g}=g_\infty\mathbf{g}^* \qquad \mathbf{c}=U\mathbf{c}^*$$

$$\mu=\mu_\infty\mu^* \qquad \mathbf{v}=U\mathbf{v}^* \qquad t=\frac{L}{U}t^* \qquad \kappa=\frac{1}{L}\kappa^*$$

$$p=\rho_\infty U^2 p^* \qquad \frac{\partial}{\partial t}=\frac{U}{L}\frac{\partial}{\partial t^*} \qquad \nabla=\frac{1}{L}\nabla^* \qquad \sigma=\sigma_0\sigma^*$$

$$c_p=c_{p_\infty}c_p^* \qquad k=k_\infty k^* \qquad T=(T_w-T_s)T^*+T_s$$

Re, We and Fr are the Reynolds, Weber and Froude numbers respectively, defined if referential velocity is known, as follow:

Copyright © 2015 by ASME

$$Re = \frac{\rho_l U_\infty D}{\mu_l} \qquad We = \frac{\rho_l U_\infty^2 D}{\sigma} \qquad Fr = \frac{U_\infty}{\sqrt{gD}} \qquad Pr = \frac{c_p \mu_l}{k_l}$$

If velocity is not known “a priori” and cannot be used as referential value, the non-dimensional form is taken using the known diameter D and gravity g as the reference, and the non-dimensional number are written as follow:

$$Re = \frac{\rho_l \sqrt{gD^3}}{\mu_l} \qquad We = \frac{\rho_l g D^2}{\sigma} \qquad Fr = 1 \qquad Pr = \frac{c_p \mu_l}{k_l}$$

The term $(1/We)\kappa\nabla H$ represents the surface tension force, where 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. κ is the surfaces curvature computed based on a new 3-dimensional extension of the Frenets formula [9]. The fluid properties density ρ and viscosity μ are kept constant at each phase, but not necessarily assume equal values. Thus, the Heaviside function is used to defined a generic property ϕ distribution along the domain as follow:

$$\phi = \phi_1 H + \phi_2(1-H) \tag{2}$$

Within the ALE context, the energy transport equation should take into account the relative velocity $\mathbf{u}-\hat{\mathbf{u}}$, thus representing the heat convection. The energy equation is written in the Arbitrary Lagrangian-Eulerian non-dimensional form as:

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

In the above 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 heat flux defined as:

$$\dot{q} = -k\nabla T \tag{4}$$

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 phenomena, so that:

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

Since the gradient of the Heaviside function is only different of zero near the interface, the fluid equation becomes incompressible far from it. The normal velocity of the interface $\mathbf{u}_I$ is found to be:

$$\mathbf{u}_I = \frac{1}{2}(\mathbf{u}_l + \mathbf{u}_v) + \frac{\dot{q}}{2}\left(1+\frac{\rho_l}{\rho_v}\right) \tag{6}$$

The new position of the interface nodes $\mathbf{x}_I$ may be found by integrating:

$$\frac{dx_I}{dt} = \mathbf{u}_I\cdot\mathbf{n} \tag{7}$$

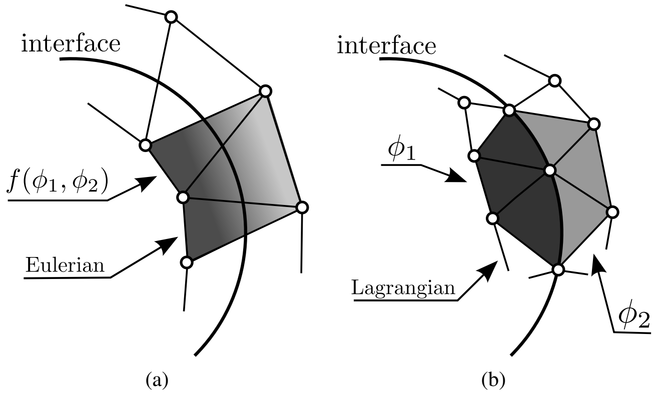
where $\mathbf{n}$ is the outward normal vector.

Interface and Re-meshing Procedure

Unlike the descriptions found in most front-tracking codes, the interface mesh implemented in this work is a sub-set of the tetrahedron volumetric mesh, i.e. each triangle shares a face of two adjacent tetrahedral elements, 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 Fig. (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 (Fig. (1b)), the properties Φ_1, Φ_2 are sharply defined in each region, thus not requiring any additional function.

In two-phase flows within the moving mesh context, a re-meshing 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 re-meshing is also required to keep the element connectivity consistently while the simulation is running. Thus, two sets of data are stored during the simulations,

Copyright © 2015 by ASME



(a)

(b)

FIGURE 1: Interface representation in two-phase flows. (a) In the *Eulerian* approach, the interface between the fluids is located somewhere in between the computational elements. On the other hand (b) the *Lagrangian* description represents the interface by computational objects, such as nodes, segments and elements, thus achieving a sharp interface.

namely volumetric nodes and surface mesh. The former consists in the mesh nodes distributed on the 3-dimensional space and the later 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 to generate the tetrahedron connectivity array at each time step. The volumetric node distribution is done by solving Helmholtz equation, in which the boundary conditions consist in the characteristic edge lengths of each surface mesh, thus a smooth transition between the interface and the boundary meshes is successfully achieved. Note that the treatment of the volumetric nodes does not require the maintenance of the tetrahedrons connectivity. On the other hand, the surface meshes should be modified, at each time step, to keep the triangles shape bounded to good aspect ratios. This is achieved by the common mesh operations such as insertion and deletion of points, flipping and contraction of and edges. Moreover, due to the implementation within the code where the volumetric and surface meshes are treated separately, the mesh distribution treatments may be combined into a scheme and adjusted by parameters varying from 0 to 1. Therefore, the volumetric and surface velocities are therefore treated as follows:

$$\hat{\mathbf{u}}(\mathbf{x}) = \begin{cases} \beta_1 \mathbf{u} + \beta_2 \mathbf{u}_{vel} + \beta_3 \mathbf{u}_e & \text{if } \mathbf{x} \notin \mathbf{x}_I \\ \mathbf{u} - \gamma_1 (\mathbf{u} \cdot \mathbf{t})t + \ \gamma_2 (\mathbf{u}_e \cdot \mathbf{t})t & \text{if } \mathbf{x} \in \mathbf{x}_I \end{cases} \tag{8}$$

Due to the description of the interface mesh by computational elements, the surface should move according to the fluid motion. In the above equation, if $\mathbf{x}$ belongs to the interface, we can define its velocity as $\mathbf{u}_I$. Thus, it is convenient to decompose it into two orthogonal components: $\mathbf{u}_{I_n}$ and $\mathbf{u}_{I_t}$ which represent the normal and tangential velocities, respectively. To decrease the displacement of nodes in the tangential 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 $\mathbf{u}_{I_n}$ to the normal vector associated to the node or, in a simpler manner, by removing the tangent component from the total surface mesh velocity $\mathbf{u}_{I_t} = \mathbf{u} - (\mathbf{u} \cdot \mathbf{t})t$. Such a procedure may be included into a scheme so that the intensity of the tangential velocity can be easy modified. 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 a Laplacian smoothing scheme [11] on the surface mesh nodes, thus keeping them all bounded within good aspect ratio. The parameter β_1 controls the Lagrangian motion of the inner and outer volumetric mesh velocity. By setting $\beta_1 = 1$, the flow velocity $\mathbf{u}$ 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 on the moving mesh velocity. The parameters β_2 and β_3 control the intensity of the velocity smoothing scheme $\mathbf{u}_{vel}$ and the Laplacian smooth scheme $\mathbf{u}_e$ into 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 the desirable node distributions according to the simulation requirements.

Finite Element Method

The Navier-Stokes and energy equations are discretized through the Finite Element method. A summary of the underlying principles is given below and further details can by found at [4].

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 modelling and implementing a numerical scheme for this term, the semi-Lagrangian technique is employed (for details, see [12] and [13]) 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 towards 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 backwards in time with the initial condition $x(t_{n+1}) = xi$ than 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.

Copyright © 2015 by ASME

The choice of element types for coupled PDEs problems must take into account the Babuska-Brezzi condition in order to preserve the stability properties intrinsic to the discretization scheme ([14] and [15]). 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 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 3-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 [16]. 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 below 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:

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

where and M_ρ and K are the mass and stiffness 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 a 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:

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

Note that the subscript L refers to the Lumped matrix technique, in which a digitalization 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 the expression below:

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

where ρ_l and ρ_v are the densities for the liquid and vapour phases respectively. $\dot{q}$ is the discrete heat 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:

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

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:

$$u^{n+1} = u_{\text{corr}} + B_L^{-1}Gp^{n+1} \tag{14}$$

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:

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

where M_T and K_T are the mass and stiffness 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 as:

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

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.

Copyright © 2015 by ASME

0.1 Time step restriction

In the governing equations, the explicitly treated terms are limited by a time step. The time step constraint for Semi-Lagrangian term is given by the expression below:

$$\Delta t_{sl} < \frac{h_{min}}{\mathbf{u}_{max}} \tag{17}$$

where h_{min} is the smallest tetrahedral mesh edge length and $\mathbf{u}_{max}$ is the maximum value of velocity given by $\max\{|u|,|v|,|w|\}$. On the other hand, the motion of the mesh nodes is due to $\hat{\mathbf{u}}$ and the time step constraint should prevent the nodes from moving more than one mesh cell, otherwise the mesh is corrupted due to element distortions and collision of nodes. The strategy of the moving mesh time constrain calculation is based on the velocity differences and the local mesh edge size, thus limiting one node to overlap the other or cross the face of an element. Let us consider v_1 and v_2 as the vertices of a tetrahedral edge h_e and $\Delta\hat{\mathbf{u}}_e = \hat{\mathbf{u}}_1 - \hat{\mathbf{u}}_2$ as the mesh velocity difference, thus the proposed Lagrangian time step should be bounded by:

$$\Delta t_l < \min\left(\frac{h_e}{|\Delta\hat{\mathbf{u}}_e|}\right) \tag{18}$$

Additionally, gravity and surface tension adds a constraint to the final time step which is related to the wave velocity propagating into the computational mesh. According to Brackbill and Kothe [17] and Fortuna [18], such a criteria for both surface tension and gravity may be written as:

$$\Delta t_s < \left[\frac{\rho h^3 \mathrm{Eo}}{2\pi}\right]^{1/2} \tag{19}$$

and

$$\Delta t_g < \left[\frac{1}{h_{min}}\right]^{1/2} \tag{20}$$

In the above equations, ρ is an average fluid density between the inner and outer fluids, h is the mesh characteristic length, h_{min} is the smallest mesh length and Eo is the *Eötvös*. Thus, the final simulation constraint may be written as:

$$\Delta t \leq \min\left\{\Delta t_l, \Delta t_s, \Delta t_g, \Delta t_{sl}\right\} \tag{21}$$

RESULTS

Rising Bubble in Thermally Stratified Water Layer

The free rising of an air bubble immersed in a thermally stratified water layer is investigated in this numerical simulation. The physical properties of the fluid stands for those of the binary air and water, i.e. density $\rho_l = 1000\,kg/m^3$ and $\rho_v = 1.225\,kg/m^3$, viscosity $\mu_l = 0.001Ns/m^2$ and $\mu_v = 1.7894e-5Ns/m^2$ and surface tension $\sigma = 0.072N/m$, resulting in $Re = 1455.66$, $Fr = 1$ and $We = 4.90$. As can be seen in Fig. (2), the numerical domain used in this simulation was a parallelepiped domain with dimensions $13.3Dx10Dx27D$ and the initial temperature distribution follows the rule:

$$T_{ini} = \begin{cases} 0.78 & y \leq 7.83D \\ z*3.428E-3+7.531E-1 & 7.83D < y < 25.33D \\ 0.84 & y \geq 25.33D \end{cases} \tag{22}$$

Note that despite heat transfer occurs, no mass transfer is accounted in for this simulation of a free rising air bubble. For this simulation, the volumetric mesh parameters $\beta_1 = 0.0$ and $\beta_2 = 0.0$ were set to zero, while $\beta_3 = 0.7$ to guarantee the volumetric nodes distant to the interface. The surface mesh parameters were set to $\gamma_1 = 1.0$ and $\gamma_2 = 0.5$. The domain's discretization used approximately 40000 triangles and 6100 nodes. The surface mesh (boundary and bubbles) had approximately 9500 triangles in which 5700 were part of the interface meshes.

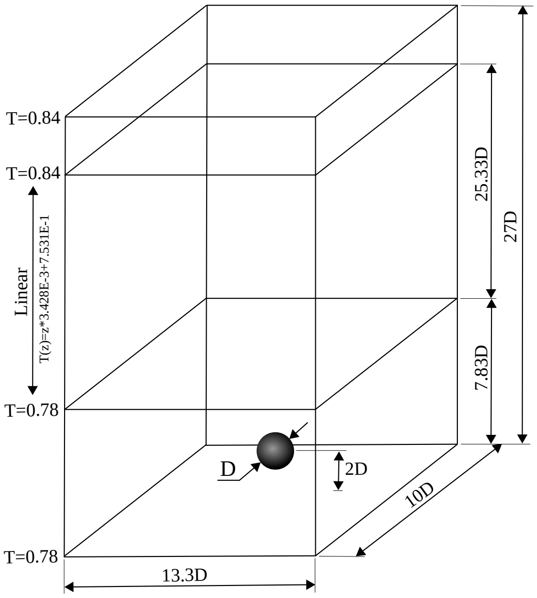
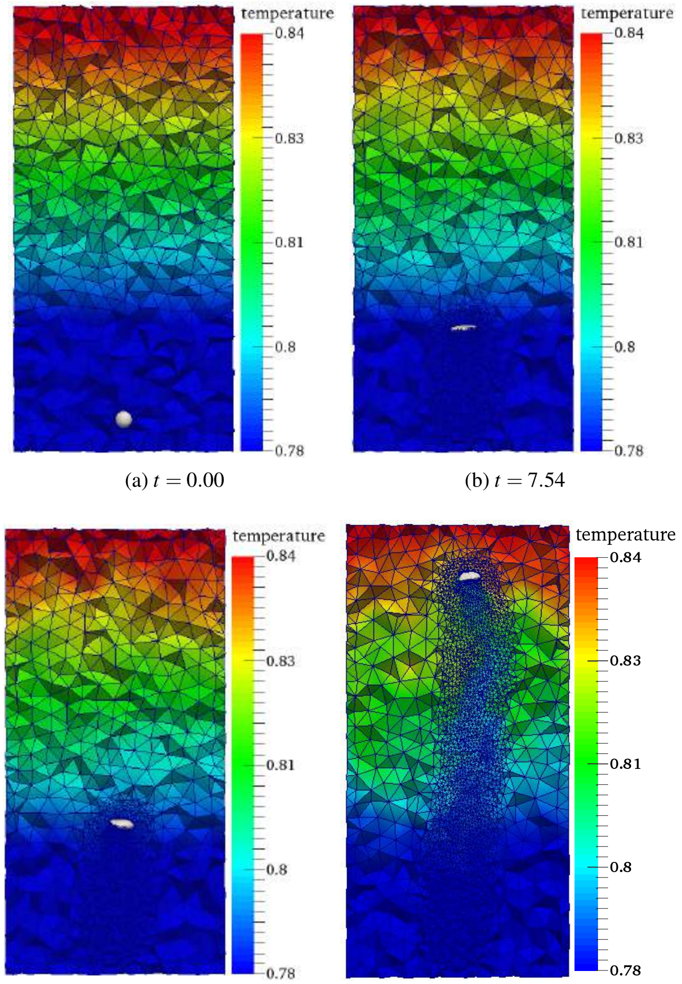


FIGURE 2: Numerical domain dimensions and thermal profile along y direction.

Copyright © 2015 by ASME

In Fig. (3), the free rising of an air bubble immersed in thermally stratified water layer was carried out, showing the evolution of bubble's shape crossing different temperature layers along the y axis. In Fig. (3a), the mesh used in this simulation is shown as well as its initial conditions. It was found that the bubble trajectory is not linear due to the generation of vortex while in ascension. Moreover, the passage of the bubble in temperature gradient zones increase significantly the mixture of the thermal layers.



(a) $t = 0.00$

(b) $t = 7.54$

(c) $t = 9.66$

(d) $t = 25.48$

FIGURE 3: Free rising of an air bubble immersed in thermally stratified water layer. (a) Initial condition $t = 0.00$ and finite element mesh. (b) Air bubble crossing temperature gradient zone for $T = 0.78$ and $t = 0.00$. (c) Air bubble in temperature gradient zone $T = 0.81$ and $t = 0.00$. (d) Air bubble at $T = 0.84$ and $t = 0.00$. Temperature and time are non-dimensional.

TABLE 1: FLUID PROPERTIES OF REFRIGERANT R236FA

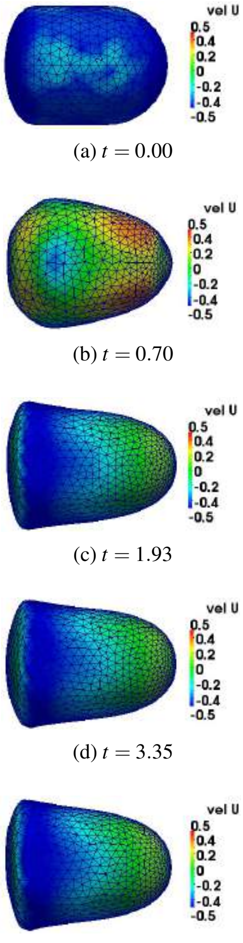
Fluid: R236fa		
vapor phase:	ρ=18.294kg/m^3	$\mu = 10.846\,\mu Pa\cdot s$
	k=12.760mW/mK	$c_p = 0.8844\,kJ/kgK$
liquid phase:	ρ=1360.3kg/m^3	$\mu = 286.03\,\mu Pa\cdot s$
	k=72.870mW/mK	$c_p = 1.2641\,kJ/kgK$
interface:	σ=10.086mN/m	

Isothermal Microchannel Two-Phase Flow

The isothermal simulation of an isolated bubble is performed to evaluate the conditions found in microchannel flows using refrigerant R236fa as the working fluid. A micro channel with a square cross-section of $100x100\,\mu m$ and $700\,\mu m$ length long, was represented in non-dimensional form as $1Dx1Dx7D$. The bubbles length was set to $L_b = 1.2D$ where D is the cross-section width. The simulation was performed using a moving referential frame, where the bubble remains fixed in space. The initial conditions were set to fully developed velocity profile previously simulated for single-phase flow and later interpolated in the two-phase mesh. The boundary conditions were set as fully developed velocity profile in the inlet (left wall) and outlet (right wall) in which 1 node was set for prescribed pressure (Dirichlet condition). All the remaining walls were set to zero velocity. For the moving frame context, the bubble's centroid velocity is subtracted each time step of the current wall boundary condition, therefore the bubble nodes were fixed in space, and all the remaining mesh nodes (including boundaries) moves backwards relatively to the bubble's centroid velocity. Figure (4) shows the evolution of the bubble's shape with time and colored by the velocity component in the x-direction. After a transient stage, the bubble reaches its stable solution for the bubble shape. However, the tail of the bubble presents a small continuous oscillation, typically observed experimentally. The dimensionless liquid film thickness remained stable and its minimum value was measured, and it was found to be $\delta_{min} = 0.057$, which represents 57μm.

For this simulation, the volumetric mesh parameters $\beta_1 = 0.0$ and $\beta_2 = 0.0$ were set to zero, while $\beta_3 = 1.0$ to guarantee the volumetric nodes distant to the interface. The surface mesh parameters were set to $\gamma_1 = 1.0$ and $\gamma_2 = 1.0$. The domain's discretization used approximately 69000 tetrahedrons and 8200 nodes. The surface mesh (boundary and bubbles) had approximately 16000 triangles in which 9000 were part of the interface meshes.

Copyright © 2015 by ASME



(a) $t = 0.00$

(b) $t = 0.70$

(c) $t = 1.93$

(d) $t = 3.35$

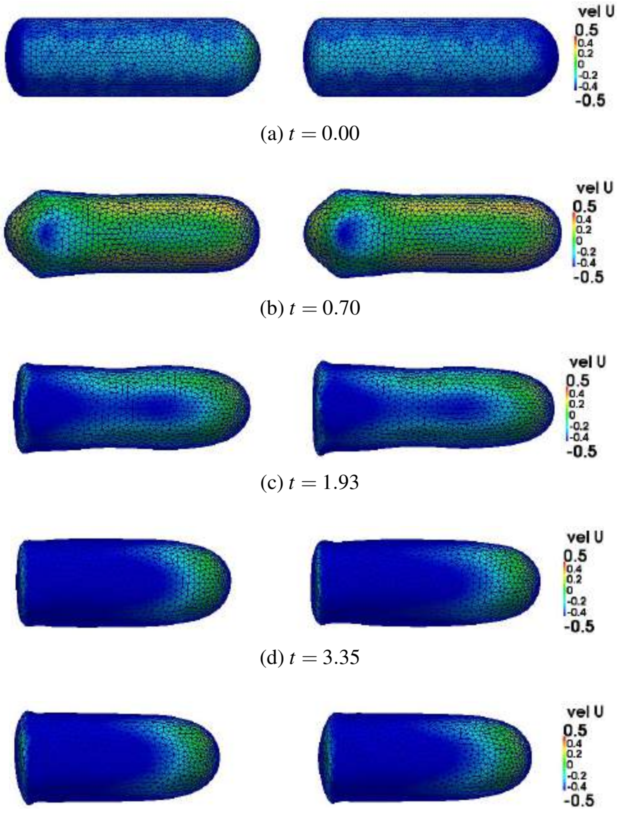
(e) $t = 8.76$

FIGURE 4: Bubble's shape evolution with time for a single vapor bubble. The working fluid is R236fa, whose details can be found in the Table (1) The fluid is entering at the left of the domain and exiting at the right. A transient stage occurs in (a-b), followed by a nearly stable bubble shape in (c-e).

Isothermal Microchannel Two-Phase Flow of Two Bubbles

A more interesting test case was simulated using the refrigerant R1234ze as working fluid with two bubble placed $0.5D$ behind the other one. Therefore, the second bubble is affected by the wake formed by the motion of the first one, characterizing a slug flow. Each bubble length was set to be $L_b = 2D$ and the domain dimension was set as the previous test case $1Dx1Dx7D$. The simulation was performed using a moving referential frame, where the bubble remains fixed in space with same boundary conditions for the previous test case.Figure (5) depicts the bubble's shape evolution of two vapor bubbles of the refrigerant R1234fa. Different patterns were found in the bubble's shapes as well as the evolution of the film thickness with time, as can be seen in Fig. (6). The liquid film thicknesses were found to be $\delta = 20\mu m$ and $\delta = 37\mu m$ for the left and right bubble respectively. From time $t \approx 8$ till the end of the simulation, the two bubbles remained equally distanced.

The volumetric mesh parameters $\beta_1 = 0.0$ and $\beta_2 = 0.0$ were set to zero, while $\beta_3 = 1.0$ to guarantee the volumetric nodes distant to the interface. The surface mesh parameters were set to $\gamma_1 = 1.0$ and $\gamma_2 = 1.0$. The domain's discretization used approximately 69000 tetrahedrons and 8200 nodes. The surface mesh (boundary and bubbles) had approximately 16000 triangles in which 9000 were part of the interface meshes.



(a) $t = 0.00$

(b) $t = 0.70$

(c) $t = 1.93$

(d) $t = 3.35$

(e) $t = 8.76$

FIGURE 5: Bubble's shape evolution with time for two vapor bubbles. The working fluid is refrigerant R1234ze, whose details can be found in Table (2). The fluid is entering at the left of the domain and exiting at the right. (a-c) Transient stage. (d) initial formation of the surface waves in the tail of the vapor bubble. (d) Stable bubble shape.

TABLE 2: FLUID PROPERTIES OF REFRIGERANT R1234ZE

Fluid: R1234ze		
vapor phase:	ρ=25.898kg/m^3	$\mu = 12.464\,\mu Pa \cdot s$
	k=15.664W/mK	$c_p = 1.0421\,kJ/kgK$
liquid phase:	ρ=1157.4kg/m^3	$\mu = 202.71\,\mu Pa \cdot s$
	k=62.060mW/mK	$c_p = 1.3882\,kJ/kgK$
interface:	σ=8.5156mN/m	$h_{fg} = 164.7\,kJ/kg$

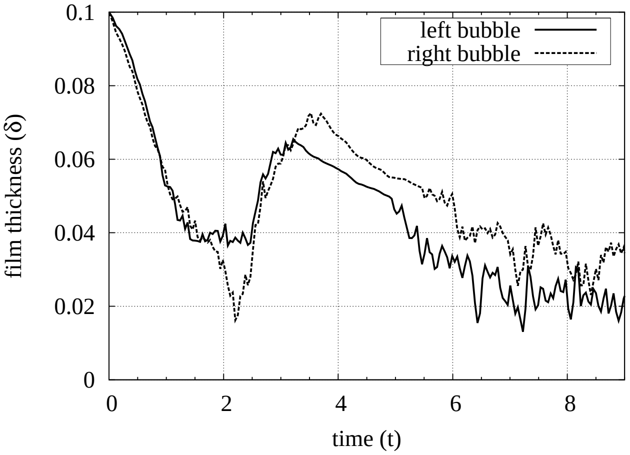
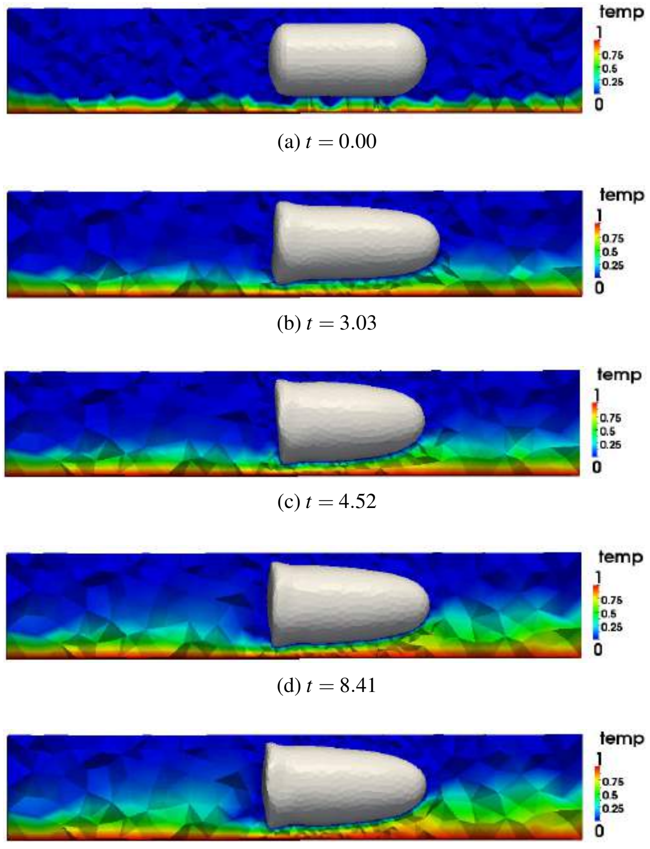


FIGURE 6: Transient solution of the liquid film thickness computed for the two vapor bubbles of the refrigerant R1234ze.

Two-phase Flow Boiling of Single Bubble

The dynamics of a single vapor bubble of the refrigerant R1234ze was simulated taking into account the evaporation process produced by the constant and uniform heat flux $\dot{q}$ applied to the bottom part (wall) of the numerical domain (see 2 for fluid properties). Thus, the simulation of the microchannel placed on the top of a working chip can be achieved. The bubble's length was set to $L_b = 1.3D$ and the initial film thickness to $\delta_i = 0.35$. The domain dimension was set to $1Dx1Dx7D$ for a square cross-sectional microchannel. The simulation was performed using a moving referential frame, where the bubble remains fixed in space. The initial and boundary conditions set to this problems was the same of previous test cases and for the energy equation was set for constant saturated temperature at bottom wall and Neumann condition ($\mathbf{n} \cdot \nabla T$) for the remaining walls. Figure (7) shows the transient solution of the temperature profile of two-phase evaporation of R1234ze. Due to the evaporation process, it was verified that the bubble's volume changed 12% compared to its initial volume. Moreover, the transient bubble's shape presented a different pattern compared to the isothermal simulation performed at [4] for the same working fluid, indicating that the evaporation of the liquid film affected the bubble's shape. The surface waves were not identified in the tail of the bubble. The measured film thickness found was $\delta = 0.073$. A further investigation is still required to validate the implemented mass transfer model.

The volumetric mesh parameters $\beta_1 = 0.0$ and $\beta_2 = 0.0$, while $\beta_3 = 1.0$. The surface mesh parameters were set to $\gamma_1 = 1.0$ and $\gamma_2 = 1.0$. The domain's discretization used approximately 69000 tetrahedrons and 8200 nodes. The surface mesh (boundary and bubbles) had approximately 16000 triangles in which 9000 were part of the interface meshes.



(a) $t = 0.00$

(b) $t = 3.03$

(c) $t = 4.52$

(d) $t = 8.41$

(e) $t = 13.03$

FIGURE 7: Transient solution of two-phase flow boiling of refrigerant R1234ze. A constant heat flux $\dot{q}$ is applied in the bottom part of the domain. The fluid is entering at the left of the domain and exiting at the right. Time and temperature are non-dimensional.

CONCLUSIONS

This article presents a new methodology for simulating incompressible two-phase flows with heat and mass transfer within the Finite Element Method context in which the mesh moves in an Arbitrary Lagrangian-Eulerian 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, which accurately describes the actual physical conditions. The proposed treatment of the computational mesh, splitting the surface meshes and the volumetric points, has 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.

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 rising bubble in thermally stratified water layer and boiling two-phase flow of single bubble in square microchannels qualitatively agreed to those observed experimentally. The two-phase flow boiling simulation requires finer mesh to capture the heat and mass transfer accurately. The typical computational time for each simulation presented in the results sections is approximately three days long for hundred thousand finite elements. A time step and mesh convergence analysis is still required, as well as validations on the proposed model, to compare the actual state of the phase change implementation with different benchmarks available in the literature.

ACKNOWLEDGMENT

This research was funded by the Brazilian Program CAPES – Coordination for the Improvement of Higher Education Personnel, granted by Science Without Borders/Young Talent Attraction program.

REFERENCES

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

[2] Sussman, M., Smereka, P., and Osher, S., 1994. "A level-set approach for computing solutions to incompressible two-phase flow". *Journal of Computational Physics,* ***114***, pp. 146–159.

[3] Glimm, J., Grove, J., Lindquist, W., McBryan, O., and Tryggvadson, G., 1988. "The bifurcation of tracked scalar waves". *SIAM Journal of Computations,* ***9***(1), pp. 61–79.

[4] Anjos, G., 2012. "A 3d ale finite element method for two-phase flows with phase change". PhD thesis, École Polytechnique Fédérale de Lausanne, July.

[5] Dhir, V., 1991. "Nucleate and transition boiling heat transfer under pool and external flow conditions". *International Journal of Heat and Fluid Flow,* ***21***, pp. 290–314.

[6] Juric, D., and Tryggvason, G., 1998. "Computations of boiling flows". *Journal of Computational Physics,* ***24***(3), pp. 387–410.

[7] Unverdi, S., and Tryggvason, G., 1992. "A front-tracking method for viscous, incompressible, multi-fluid flows". *Journal of Computational Physics,* ***100***, pp. 25–37.

[8] Anjos, G., Mangiavacchi, N., and Pontes, J., 2014. "Three-dimensional finite element method for rotating disk flows". *Journal of the Brazilian Society of Mechanical Sciences and Engineering,* ***36***(4), pp. 709–724.

[9] Anjos, G., Borhani, N., Mangiavacchi, N., and Thome, J., 2014. "3d moving mesh finite element method for two-phase flows". *Journal of Computational Physics,* ***270***, pp. 366–377.

[10] Si, H., 2008. "Adaptive tetrahedral mesh generation by constrained delaunay refinement". *International Journal for Numerical Methods in Engineering,* ***46***(7), pp. 856–880.

[11] Desbrun, M., Meyer, M., Schröder, P., and Barr, A., 1999. "Implicit fairing of irregular meshes using diffusion and curvature flow". In Proceedings of Siggraph, pp. 317–324.

[12] Robert, A., 1981. "A stable numerical integration scheme for the primitive meteorological equations". *Atmosphere Oceans,* ***19***, pp. 35–46.

[13] Pironneau, O., 1982. "On the transport-diffusion algorithm and its applications to the navier-stokes equation". *Numerische Mathematik,* ***38***, pp. 309–332.

[14] Cuvelier, C., Segal, A., and van Steenhoven, A. A., 1986. *Finite Element Method and Navier-Stokes Equations*. Dordrecht, Holland.

[15] Zienkiewicz, O., and Taylor, R., 2000. *The Finite Element Method: Its Basis and Fundamentals*, 5th. ed. Butterworth-Heinemann, Oxford.

[16] Chorin, A. J., 1968. "Numerical solution of the navier-stokes equations". *Mathematics of Computation,* ***22***, pp. 745–762.

[17] Brackbill, J., and Kothe, D., 1992. "A continuum method for modeling surface tension". *Journal of Computational Physics,* ***100***, pp. 335–354.

[18] Fortuna, A., 2000. *Técnicas Computacionais para Dinâmica dos Fluidos*, 1a. ed. edUSP.

Copyright © 2015 by ASME