INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2004; **45**:1–19 (DOI: 10.1002/fld.591)

A projection scheme for incompressible multiphase flow using adaptive Eulerian grid

T. Chen[1,†], P. D. Minev[2,*,‡] and K. Nandakumar[1,§]

[1]*Department of Chemical and Materials Engineering, University of Alta., Edmonton, Alta., Canada T6G 2G6*
[2]*Department of Mathematical and Statistical Sciences, University of Alta., Edmonton, Alta., Canada T6G 2G1*

SUMMARY

This paper presents a finite element method for incompressible multiphase flows with capillary interfaces based on a (formally) second-order projection scheme. The discretization is on a fixed Eulerian grid. The fluid phases are identified and advected using a level set function. The grid is temporarily adapted around the interfaces in order to maintain optimal interpolations accounting for the pressure jump and the discontinuity of the normal velocity derivatives. The least-squares method for computing the curvature is used, combined with piecewise linear approximation to the interface. The time integration is based on a formally second order splitting scheme. The convection substep is integrated over an Eulerian grid using an explicit scheme. The remaining generalized Stokes problem is solved by means of a formally second order pressure-stabilized projection scheme. The pressure boundary condition on the free interface is imposed in a strong form (pointwise) at the pressure-computation substep. This allows capturing significant pressure jumps across the interface without creating spurious instabilities. This method is simple and efficient, as demonstrated by the numerical experiments on a wide range of free-surface problems. Copyright © 2004 John Wiley & Sons, Ltd.

KEY WORDS: finite element method; Navier–Stokes equations; multiphase flows

1. INTRODUCTION

Multiphase flows occur frequently in nature and engineering practice. Nevertheless, they still pose a major research challenge from both, theoretical and computational points of view. The present study proposes a finite element splitting technique for a direct simulation of multifluid flows, based on the Eulerian approach. In order to maintain optimal interpolation rates, however, we also adapt the grid locally around the free boundaries. Based on the

*Correspondence to: P. D. Minev, Department of Mathematical Sciences, University of Alberta, Edmonton, Alberta, Canada T6G 2G1.
†E-mail: tc8@ualberta.ca
‡E-mail: minev@ualberta.ca
§E-mail: kumar.nandakumar@ualberta.ca

Contract/grant sponsor: National Science and Engineering Research Council of Canada

Received 18 November 2002
Accepted 15 April 2003

pioneering works of Osher and Sethian [1] and Unverdi and Tryggvason [2], the most popular techniques use fixed Cartesian grids using the finite difference method (FDM). The reader is referred to Reference [3, 4] for recent reviews on the so-called front capturing and front tracking techniques. The former approach includes the level set and the volume-of-fluid (VOF) methods that can be combined for optimal numerical efficiency [5]. Commonly, the density and viscosity jumps and the surface tension (a Dirac delta function) are regularized/smoothed across the interface in order to facilitate the application of FDM discretizations (e.g. Reference [6]). The interface is no longer sharp but has a finite thickness. It suffers of at least two obvious drawbacks. Firstly, the regularization/smoothing must be over a few grid cells, resulting in a requirement for a high grid resolution. Secondly, such regularization is only first order accurate in space [7], and inherently reduces the accuracy of any formally higher-order schemes when the singularities are significant.

Most finite element methods (FEM) for free boundary problems are based on the so-called Eulerian-Lagrangian approach. The reader is referred to Reference [8] and the references therein for a brief recent review of the works with this method. One of the most recent attempts was conducted by Pillapakkam and Singh [9] to study the bubble hydrodynamics in viscoelastic two-phase flows. However, the approach they formulated is a direct extension of the level-set method in the finite difference context. Another direct application of the level set method was formulated by Quecedo and Pastor [10].

The most important advantages of the FEM with respect to multifluid flows are: (i) it allows for an easy local fitting of the grid to the interfaces and therefore for an optimal interpolation of the velocity and pressure; (ii) it allows for an easy and accurate incorporation of the surface tension avoiding the need of regularization of the Dirac delta function there. This surface force, however, leads to a pressure jump across the interfaces that cannot be optimally approximated on fixed Eulerian grids. Even if the pressure is eliminated from the system, using for example a penalty method [11] or the streamline function/curl projection method [12], this difficulty still remains. Moreover, the former method usually leads to a badly conditioned linear system, and therefore, to a poor efficiency for unsteady problems, while the latter one is very inefficient when generalized to three-dimensional problems and encounters difficulties in the enforcement of the boundary conditions. The authors [13] have presented a 3D finite element technique based on a dynamic basis enrichment for pressure and velocity and using $P_2 - P_1$ elements. The local correction of the approximations leads to a significant improvement of the mass conservation properties of the algorithm. In the present paper, we extend the second-order (in time) projection scheme of Kim and Moin [14] to multiphase incompressible flows. The resulting technique can be considered as a compromise between the arbitrary-Lagrangian–Eulerian (ALE) approach and the fixed grid, Eulerian approach. It possesses the major approximation advantage of the former one, temporarily fitting the grid to the boundary at each time step, but avoids the need of regriding. The local grid adaptation captures the discontinuities of the pressure, and the velocity derivatives. It also facilitates the imposition of the pressure boundary conditions at the interface, at the pressure computation substep of the projection scheme. The adaptation is much easier if first-order elements are used and therefore we applied a stabilized first-order approximation for both, the pressure and velocity.

The rest of the paper is organized as follows. In Section 2 we present the mathematical model and a discussion about the interface capturing. The discretization procedure is presented in Section 3, including the time splitting, spatial discretization procedure and the enforcement

of the boundary conditions. The algorithms for the curvature computation and local grid adaptation are also described. The method is validated in Section 4 using available experimental data and convergence tests. The conclusions are summarized in Section 5.

2. PROBLEM FORMULATION

We consider a two-dimensional flow domain Ω that contains two different Newtonian fluids Ω_1 and Ω_2, with constant densities (ρ_1, ρ_2) and viscosities (μ_1, μ_2). The fluids are assumed to be homogeneous, immiscible and incompressible, and separated by an interface Γ. In each single-phase domain, the fluid motion is governed by the Navier–Stokes equations which, in a Cartesian coordinate system $\mathbf{x}=(x, y)$ with the y-axis pointing opposite to gravity, read

$$\rho_i \frac{D\mathbf{u}_i}{Dt} = \mu_i \nabla^2 \mathbf{u}_i - \nabla P_i, \quad \text{in } \Omega_i \tag{1}$$

$$\nabla \cdot \mathbf{u}_i = 0, \quad \text{in } \Omega_i,\ i=1,2 \tag{2}$$

in the time interval $t \in [0, T]$. Here the hydrodynamic pressure $P = p + \rho g y$ is the total pressure p minus the hydrostatic pressure, $\mathbf{u}=(u, v)$ is the velocity and g is the acceleration due to gravity. The total derivative in an Eulerian reference frame is $D/Dt = \partial/\partial t + (\mathbf{u} \cdot \nabla)$. Let us define $\Omega = \Omega_1 \cup \Omega_2$ and $\Gamma = \partial\Omega_1 \cap \partial\Omega_2$. We consider the case of droplets or bubbles in a viscous liquid and therefore we suppose that $\partial\Omega_2 \cap \partial\Omega = \varnothing$.

We assume Dirichlet boundary conditions on $\partial\Omega$, $\mathbf{u}_i|_{\partial\Omega} = \mathbf{u}_b$. Using the classical hypothesis that the surface tension is proportional to the mean curvature κ of Γ, we have the following boundary conditions on Γ:

$$[P - (\boldsymbol{\sigma}\mathbf{n}) \cdot \mathbf{n}] = \gamma\kappa + [\rho] g y \tag{3}$$

Here, $\boldsymbol{\sigma} = \mu[\nabla\mathbf{u} + (\nabla\mathbf{u})^{\mathrm{T}}]$ is the deviatoric stress, $[\cdot]$ represents the jump between the limiting values from the two sides of Γ, $\mathbf{n}$ is the unit vector normal to the interface pointing towards the interior of Ω_2, and γ is the coefficient of surface tension. The mean curvature is $\kappa = \nabla \cdot \mathbf{n}$. For the purpose of the projection scheme introduced below, it is more convenient to split the condition (3) as given by

$$[P] = \gamma\kappa + [\rho] g y \tag{4a}$$

$$\left[\mu \frac{\partial \mathbf{u}}{\partial \mathbf{n}}\right] = 0 \tag{4b}$$

This splitting introduces a splitting error at the free boundary whose implications are not studied yet. It deserves a careful analysis that is beyond the scope of this paper. Note, however, that in the context of a projection scheme, it is very difficult to implement (3) exactly because the major goal of the projection schemes is to split the pressure and velocity problems and therefore these conditions are almost exclusively used in the literature when projections schemes are employed (e.g. Reference [15]). Also, Equations (4a) and (4b) are not equivalent

to (3) but they do enforce it. The split set of boundary conditions determines the properties of the solution around the interface. The total pressure is discontinuous across Γ, due to the surface tension if $\gamma \neq 0$ and/or the buoyancy if $[\rho] \neq 0$. The normal derivatives of the velocity across Γ can be discontinuous if $[\mu] \neq 0$ on Γ.

The fluid properties (ρ_i, μ_i) are identified using a continuous indicator function ϕ that satisfies

$$\frac{D\phi}{Dt} = 0 \tag{5}$$

Following the concepts of the pseudo-concentration by Thompson [16] and the level set function by Osher and Sethian [1], it is convenient if the fluid interface Γ corresponds to the zero level set $\phi = 0$, so that

$$\mathbf{x} \in \begin{cases} \Omega_1 & \text{if } \phi(\mathbf{x}) > 0 \\ \Gamma & \text{if } \phi(\mathbf{x}) = 0 \\ \Omega_2 & \text{if } \phi(\mathbf{x}) < 0 \end{cases} \tag{6}$$

The normal direction to Γ is given by $\mathbf{n} = \nabla\phi/|\nabla\phi|$ at $\phi = 0$. It automatically takes care of merging and breaking of the interface. The present method differs from the level set method in the finite difference context, where ϕ must be a signed distance function over the entire time period T (see Reference [3]). Here, this is unnecessary because we use the level set function only for identification of the front and the different phases.

3. DISCRETIZATION

3.1. Time splitting

For the time discretization we use a classical splitting of the operators involved in the Navier–Stokes equations. Starting at time level $n \in [0, N]$ where $T = N\Delta t$, the linearized fractional substeps are summarized as

$$\tilde{\mathbf{u}}_i = \mathbf{u}_i^n - \Delta t[\tfrac{3}{2}(\mathbf{u}_i^n \cdot \nabla)\mathbf{u}_i^n - \tfrac{1}{2}(\mathbf{u}_i^{n-1} \cdot \nabla)\mathbf{u}_i^{n-1}] \tag{7a}$$

$$\rho_i \mathbf{u}_i^* = \rho_i \tilde{\mathbf{u}}_i + \tfrac{1}{2}\Delta t \mu_i \nabla^2(\mathbf{u}_i^n + \mathbf{u}_i^*), \quad \mathbf{u}_i^*|_{\partial\Omega} = \mathbf{u}_b^{n+1} + \rho_i^{-1}\nabla\Theta_i^n|_{\partial\Omega} \tag{7b}$$

$$\rho_i \mathbf{u}_i^{n+1} = \rho_i \mathbf{u}_i^* - \nabla\Theta_i^{n+1}, \quad \nabla \cdot \mathbf{u}_i^{n+1} = 0, \quad \mathbf{n} \cdot \mathbf{u}_i^{n+1}|_{\partial\Omega} = \mathbf{n} \cdot \mathbf{u}_b^{n+1} \tag{7c}$$

Here, $\tilde{\mathbf{u}}_i$ is the convected velocity, $\mathbf{u}_i^*$ is the intermediate velocity and Θ_i is an auxiliary variable related to the pressure. Note that the advection substep (7a) and the diffusion substep (7b) can, in general, be unified in a single advection-diffusion substep. However, in the present method, as it will become clear below, the spatial discretization of (7a) and (7b) are performed differently and therefore we prefer to keep the two substeps separated. Moreover, the convection of the level set function ϕ, the Cauchy problem for a scalar, can be done in a manner akin to that of $\tilde{\mathbf{u}}_i$, which makes it nearly computationally free. Equation (7c) clearly

implies that

$$\nabla \cdot (\rho_i^{-1} \nabla \Theta_i^{n+1}) = \nabla \cdot \mathbf{u}_i^* \tag{8}$$

subject to an equivalent boundary condition $\nabla(\Theta_i^{n+1} - \Theta_i^n) \cdot \mathbf{n}|_{\partial\Omega} = 0$. The actual pressure, which is not explicitly computed during time integration, is related to Θ_i via

$$P_i^{n+1} = \Theta_i^{n+1}/\Delta t - \tfrac{1}{2}\,\mu \nabla \cdot \mathbf{u}_i^* \tag{9}$$

The mass conservation, as demonstrated by the numerical examples, is satisfactory, although the convection is solved in a non-conservative form. As claimed by Kim and Moin [14], this scheme is second-order accurate in time for the velocity. Actually, it can be shown (see Reference [17]) that, in case of a single-phase flow, this scheme is equivalent to the so-called rotational form of the pressure correction scheme whose analysis is presented by Guermond and Shen [18]. In case of multifluid flows, however, the splitting of the boundary condition (3) and an artificial interface condition on Θ that will be discussed below will deteriorate the time accuracy of the scheme.

An optimal spatial approximation for multifluid problems can be achieved only if the interpolants for pressure and velocity have the same singularities across the interface as the pressure and velocity themselves. It is clear from point of view of the interpolation theory that at least one order of convergence will be lost if the points of discontinuity of a piecewise continuous function are not guaranteed to be points of discontinuity of a piecewise linear interpolant. Li and Lai [19] developed an incremental projection scheme with the immersed interface method in the finite difference context and reported second-order accuracy (in L_∞ norm) for a certain free-surface problems with a constant density and viscosity ($[\rho] = [\mu] = 0$). To the best of our knowledge, this is the only scheme claimed to be second order (in space). Note that if the scheme is incremental then the pressure P^n from the previous time step will be involved in computing the intermediate velocity at time level $n+1$. However, unlike in the case of single-phase flows, if the point $\mathbf{x}$ moves at the time step $n+1$ from one of the phases into the other then the pressure prediction P^n at $\mathbf{x}$ will be $O(1)$ away from the pressure at level $n+1$ there. This is because the jump of the pressure across the free boundary is in general a non-zero $O(1)$ function with respect to the time step and the grid size. This will certainly deteriorate the splitting error of the scheme and may result in the development of instabilities around the free boundary. Unfortunately, this problem is not addressed explicitly in the literature. We conjecture that in order an incremental scheme to achieve a higher than first-order accuracy in time (for the velocity), in case of multiphase flows on Eulerian grids, it is necessary to take as a pressure predictor in a given point $\mathbf{x}$ the previous level pressure at the *foot of the characteristic* originating at this point. This may also shed some light on the development of high-order incremental schemes for single phase flows. The present form of projection involves the inconvenience of the boundary condition on the intermediate velocity given by the second equation of (7b). It requires computing explicitly the gradient of Θ on $\partial\Omega$ which is a discontinuous function in the finite element context. Therefore, we project it onto the space of $\mathbf{u}^*$ as discussed in Section 3.3. This projection makes the scheme equivalent to one of the stabilized schemes discussed by Codina [20] and therefore allows for the use of an equal (first in the present case) order approximation for the pressure and velocity. The incorporation of the internal boundary conditions on Γ will be discussed in the next section.

3.2. Weak formulation

The weak formulation of (7a) is quite straightforward and therefore we present only the formulations of the split generalized Stokes problem (7b), (7c) and (8), with the special treatment for the interfacial boundary conditions.

Let us choose the test functions $\mathbf{v}$ for the velocity from $H_0^1(\Omega)$ (the Sobolev space of functions with square integrable spatial derivatives and vanishing on $\partial\Omega$). Then summing the Galerkin formulations in Ω_i $(i=1,2)$ we obtain the following weak formulation of (7b):

$$\int_\Omega [\rho(\mathbf{u}^* - \tilde{\mathbf{u}}) \cdot \mathbf{v} + \tfrac{1}{2}\,\Delta t \mu \nabla(\mathbf{u}^* + \mathbf{u}^n) \colon \nabla \mathbf{v}]\,\mathrm{d}\Omega = 0 \tag{10}$$

Here it is convenient to define

$$\mu = \mu_i,\ \rho = \rho_i,\ \Theta = \Theta_i,\ \mathbf{u}^* = \mathbf{u}_i^* \quad \text{and} \quad \tilde{\mathbf{u}} = \tilde{\mathbf{u}}_i \text{ in } \Omega_i\ (i=1,2)$$

Note that the surface integral that appear after the integration by parts of the second-order terms is equal to zero along Γ because of (4b) which is applied to the intermediate velocity $\mathbf{u}^*$.

Since Θ is discontinuous across Γ, we cannot derive such a unified weak formulation for Equation (8). Therefore, we derive two formulations (one for each subdomain) and apply the internal boundary conditions (4) for the pressure after the spatial discretization is performed (as explained below). Choosing the test functions for the pressure $q_i \in H^1(\Omega_i)$, multiplying the first equation of (7c) by ∇q_i, integrating the velocity terms by parts and taking into account the second equation of (7c) we end up with

$$\int_{\Omega_i} \rho_i^{-1} \nabla \Theta_i^{n+1} \cdot \nabla q_i \,\mathrm{d}\Omega = -\int_{\Omega_i} \nabla \cdot \mathbf{u}_i^* q_i \,\mathrm{d}\Omega + \int_{\partial\Omega_i} \rho_i^{-1} \nabla \Theta_i^{n+1} \cdot \mathbf{n} q_i \,\mathrm{d}s,\ (i=1,2) \tag{11}$$

The pressure related quantities Θ_i^{n+1} are discontinuous functions and cannot be used for imposing the boundary condition onto the next intermediate velocity $\mathbf{u}^*$. Therefore, we project them (via an L^2 projection) onto a subspace of $H^1(\Omega_i)$. This is done by multiplying the first equation of (7c) by the test functions for the velocity $\mathbf{v}$ and integrating over Ω_i. Summing the two equations $(i=1,2)$ we obtain

$$\int_\Omega \rho(\mathbf{u}^{n+1} - \mathbf{u}^*) \cdot \mathbf{v} \,\mathrm{d}\Omega = -\int_{\Omega_1} \nabla \Theta_1^{n+1} \cdot \mathbf{v} \,\mathrm{d}\Omega - \int_{\Omega_2} \nabla \Theta_2^{n+1} \cdot \mathbf{v} \,\mathrm{d}\Omega \tag{12}$$

Note that the integrals in the right-hand side cannot be unified as a single integral over Ω because Θ_1^{n+1} and Θ_2^{n+1} generally do not match on the internal boundary Γ. This additional projection step which is clearly equivalent to an L^2 projection of the (generally) discontinuous gradients of Θ_i^{n+1} onto a subspace of $H^1(\Omega_i)$, is actually proposed by Codina [20] to stabilize a set of projection methods (with respect to the pressure). With it, the present projection scheme becomes identical (at least in case of a single-phase flow) to one of the stabilized projection schemes analysed by Codina [20] who proved that if the time step is chosen to be sufficiently large this scheme should not generate spurious pressure modes if equal-order pressure–velocity approximation is used. The numerical tests with two-phase flows, presented below, showed that the stability of the scheme is preserved in this case as well.

3.3. Spatial discretization

The spatial discretization is performed on a fixed, Eulerian finite element grid (*reference grid*) which, however, is locally refined (*refined grid*) at each time step so that the interface Γ is always aligned with element faces i.e. it never crosses the interior of a finite element at any given discrete time instant t^n. An element on the reference grid that intersects with the interface and is to be reconnected to construct the refined grid is called a *front element*. Details of the local refinement are given later. The velocity space is approximated by means of a P_1 approximation on the refined grid. Equation (7a) as well as the equation for the indicator ϕ (5) are discretized on the reference grid while (7b) and (7c) are discretized on the refined grid. Let Φ_a be the P_1 shape function for the velocity and Φ_a for the pressure corresponding to a point a, the weak formulation (10) reads

$$\int_\Omega [\rho(\mathbf{u}_h^* - \tilde{\mathbf{u}}_h)\cdot\boldsymbol{\Phi}_a + \tfrac{1}{2}\,\Delta t\mu\nabla(\mathbf{u}_h^* + \mathbf{u}_h^n)\colon\ \nabla\boldsymbol{\Phi}_a]\,\mathrm{d}\Omega = 0 \tag{13}$$

Here, $\mathbf{u}_h^*$ is the discrete velocity subject to the external boundary condition $\mathbf{u}_h^* = \mathbf{u}_b^{n+1} + \rho^{-1}\nabla\Theta_{1,h}^n$. Since $\nabla\Theta_{1,h}^n$ is discontinuous across the element faces we use its L^2 projection already computed in the step (12).

As already mentioned, we discretize Θ_i^{n+1} using the same grid and basis Φ_a as the one used for the discretization of $\mathbf{u}^*$. The discrete versions of (11) in each phase then read (again the subscript h denotes a discrete quantity)

$$\int_{\Omega_1} \rho_1^{-1}\nabla\Theta_{1,h}^{n+1}\cdot\nabla\Phi_a\,\mathrm{d}\Omega = -\int_{\Omega_1}\nabla\cdot\mathbf{u}_1^*\Phi_a\,\mathrm{d}\Omega + \int_{\partial\Omega_1}\rho_1^{-1}\nabla\Theta_{1,h}^{n+1}\cdot\mathbf{n}\Phi_a\,\mathrm{d}s + \int_\Gamma \rho_1^{-1}\nabla\Theta_{1,h}^{n+1}\cdot\mathbf{n}\Phi_a\,\mathrm{d}s \tag{14a}$$

$$\int_{\Omega_2} \rho_2^{-1}\nabla\Theta_{2,h}^{n+1}\cdot\nabla\Phi_a\,\mathrm{d}\Omega = -\int_{\Omega_2}\nabla\cdot\mathbf{u}_2^*\Phi_a\,\mathrm{d}\Omega + \int_{\partial\Omega_2}\rho_2^{-1}\nabla\Theta_{2,h}^{n+1}\cdot\mathbf{n}\Phi_a\,\mathrm{d}s - \int_\Gamma \rho_2^{-1}\nabla\Theta_{2,h}^{n+1}\cdot\mathbf{n}\Phi_a\,\mathrm{d}s \tag{14b}$$

Note that if at $t=0$, $\nabla\Theta_{1,h}^0\cdot\mathbf{n} = 0$ on $\partial\Omega$ the first surface integral in the right-hand side of (16a) will be equal to zero at any further time level. We have already one boundary condition on Γ for this set of equations, following from (4a). Since these are the Galerkin formulations of two Poison equations (in Ω_1 and Ω_2, respectively) we need two boundary conditions to close the problem and the most convenient form for the second condition is $[\rho^{-1}\nabla\Theta_h\cdot\mathbf{n}] = 0$ on Γ because it would eliminate the interfacial integrals in the right-hand side of (14a) and (14b) if the two formulations are summed. This condition is certainly artificial but it is unavoidable in the context of projection schemes. Since $\mathbf{u}_h^*$ is continuous across Γ, it follows from the first equation of (7c) that this condition is equivalent to assume that the normal component of $\mathbf{u}_h^{n+1}$ is continuous across Γ. The implications of this condition are not studied although it is widely used (sometimes quite implicitly) in combination with projection schemes.

Let us consider one interface node denoted by J. The equations of the discrete systems (16a) and (16b) that correspond to this node are

$$\sum_{I \in N_J} \Theta_{1,I}^{n+1} \int_{\Omega_1} \rho_1^{-1} \nabla \Phi_I \cdot \nabla \Phi_J \, \mathrm{d}\Omega = -\int_{\Omega_1} \nabla \cdot \mathbf{u}_{1,h}^* \Phi_J \, \mathrm{d}\Omega \tag{15a}$$

$$\sum_{I \in N_J} \Theta_{2,I}^{n+1} \int_{\Omega_2} \rho_2^{-1} \nabla \Phi_I \cdot \nabla \Phi_J \, \mathrm{d}\Omega = -\int_{\Omega_2} \nabla \cdot \mathbf{u}_{2,h}^* \Phi_J \, \mathrm{d}\Omega \tag{15b}$$

N_J are all the nodes linked to J through the elements containing J, including J itself. Now we impose the internal boundary condition (4a), which appears to be a periodic-like boundary condition for these formulations. It reads

$$\Theta_{2,h}^{n+1}|_\Gamma = \Theta_{1,h}^{n+1}|_\Gamma + [\Theta] \tag{16}$$

where $[\Theta]$ is given by

$$[\Theta] = \Delta t(\kappa\gamma + [\rho] g y_\Gamma + \tfrac{1}{2}[\mu \nabla \cdot \mathbf{u}_h^*]).$$

Provided that we can compute $[\Theta]$ pointwise as explained in the next section, we can impose this condition after summing (15a) and (15b) which yields

$$\sum_{I \in N_J} \left(\Theta_{1,I}^{n+1} \int_{\Omega_1} \rho_1^{-1} \nabla \Phi_I \cdot \nabla \Phi_J \, \mathrm{d}\Omega + \Theta_{2,I}^{n+1} \int_{\Omega_2} \rho_2^{-1} \nabla \Phi_I \cdot \nabla \Phi_J \, \mathrm{d}\Omega \right) = -\int_\Omega \nabla \cdot \mathbf{u}_h^* \Phi_J \, \mathrm{d}\Omega \tag{17}$$

Note that in this equation, some of the nodes in N_J will be on Γ and therefore we will have two values $\Theta_{1,I}$ and $\Theta_{2,I}$ for the pressure related quantity Θ, corresponding to the two fluids in Ω_1 and Ω_2. We can eliminate (condense) one of these values using condition (16), say $\Theta_{2,I}$, and thus the discrete equation corresponding to the node J becomes

$$\sum_{I \in N_J} \Theta_I^{n+1} \int_\Omega \rho^{-1} \nabla \Phi_I \cdot \nabla \Phi_J \, \mathrm{d}\Omega = -\int_\Omega \nabla \cdot \mathbf{u}_h^* \Phi_J \, \mathrm{d}\Omega - \sum_{K \in N_J^\Gamma} [\Theta]_K \int_{\Omega_2} \rho_2^{-1} \nabla \Phi_K \cdot \nabla \Phi_J \, \mathrm{d}\Omega \tag{18}$$

where N_J^Γ is the subset of nodes in N_J that belong to Γ from Ω_2. This procedure is the same as the one used to impose periodic boundary condition for the pressure.

Likewise, we can obtain the discrete formulation for (12):

$$\int_\Omega \rho(\mathbf{u}_h^{n+1} - \mathbf{u}_h^*) \cdot \mathbf{\Phi}_a \, \mathrm{d}\Omega = -\int_{\Omega_1} \nabla \Theta_{1,h}^{n+1} \cdot \mathbf{\Phi}_a \, \mathrm{d}\Omega - \int_{\Omega_2} \nabla \Theta_{2,h}^{n+1} \cdot \mathbf{\Phi}_a \, \mathrm{d}\Omega$$

of which the right-hand side can also be rewritten similar to (18):

$$\int_\Omega \rho(\mathbf{u}_h^{n+1} - \mathbf{u}_h^*) \cdot \mathbf{\Phi}_J \, \mathrm{d}\Omega = -\int_\Omega \nabla \Theta_h^{n+1} \cdot \mathbf{\Phi}_J \, \mathrm{d}\Omega - \sum_{K \in N_J^\Gamma} [\Theta]_K \int_{\Omega_2} \nabla \Phi_K \cdot \mathbf{\Phi}_J \, \mathrm{d}\Omega \tag{19}$$

For the purpose of pressure condensation, we need to evaluate the jump $[\Theta]$ pointwise in all the nodes on the boundary. This includes the hydrostatic jump $[\rho]gy_\Gamma$, the interface curvature κ and the divergence correction term $\frac{1}{2}[\mu\nabla\cdot\mathbf{u}_h^*]$ that generally does not vanish if $[\mu]\neq 0$. We project the correction term $\Theta_\mu=\frac{1}{2}[\mu\nabla\cdot\mathbf{u}_h^*]$, which is also needed when retrieving the global pressure (9), using the following least squares approximation:

$$(\Theta_{\mu,h},\Phi_a)_{\Omega_i}=(\tfrac{1}{2}\,\mu_i\nabla\cdot\mathbf{u}_h^*,\Phi_a)_{\Omega_i} \tag{20}$$

in each of the subdomains Ω_i. Here $(.,.)$ denotes the usual L^2 inner product. By lumping the mass matrix we can localize this to the interface nodes only. Note that the so-computed velocity divergence will be discontinuous across the interface because we solve the two problems in (20) separately.

3.4. Interface curvature

The pressure condensation technique introduced above requires also an explicit pointwise computation of the curvature of the interface Γ. There are several methods that can be used for this purpose. Some authors use smoothing techniques such as the cubic spline interpolant method [21], for example. These techniques, however, are quite sophisticated if applied to three-dimensional problems and therefore we use directly the zeroth level set of ϕ for capturing of Γ i.e. we find all points $\mathbf{y}$ on the edges of the front elements such that $\phi(\mathbf{y})=0$. Then the curvature of an arbitrary level set is given by

$$\kappa(\mathbf{x})=\nabla\cdot\mathbf{n}_\phi=\nabla\cdot\frac{\nabla\phi(\mathbf{x})}{|\nabla\phi(\mathbf{x})|},\quad \text{for } \mathbf{x}\in\Omega \tag{21}$$

Probably the easiest way to compute the pointwise curvature is to approximate it using the following least squares approximation

$$\int_\Omega \kappa_h\Phi_a\,\mathrm{d}\Omega=-\int_\Omega\nabla\Phi_a\cdot\mathbf{n}_\phi\,\mathrm{d}\Omega+\int_{\partial\Omega}\mathbf{n}_\phi\cdot\mathbf{n}_{\partial\Omega}\Phi_a\,\mathrm{d}s \tag{22}$$

where $\mathbf{n}_\phi$ and $\mathbf{n}_{\partial\Omega}$ are the unit normal directions of the level sets and the external boundary, respectively. $\mathbf{n}_\phi$ is constant in case of linear triangular finite elements. Note that the right-hand side term of (21) is integrated by parts in (22). The last term of above equation is to be determined from the so-called contact angle condition that should be enforced when a contact point/line exists or this term should vanish for a closed surface. Similarly to the convection of level set functions (see Reference [22]), this mass-matrix problem can also be localized within the neighbourhood of the interface by lumping of the mass matrix. We resolve the curvature on the *reference grid* while the curvature on the interface (i.e. in the intersection points $\mathbf{y}$) is interpolated when reconnecting the front elements. The accuracy of the calculation of interface curvature is evaluated in Table I by placing a unit circle ($R=1$) at the centre of a 4×4 square using subsequent refinements of right angle triangles. The curvature approximation is clearly second-order accurate. This technique works reasonably well up to moderate surface tension. However, for large surface tension some sort of smoothing was necessary and we used the simple inverse area-weighted averaging as recommended by Gresho and Sani [23, p. 938].

Table I. Calculation of interface curvature: a unit circle ($R = 1$) placed at the centre of a 4×4 square discretized by right triangles.

Reference grid	Intersection points	Mean value (+1)	Standard deviation	$\|\|E(\kappa)\|\|_1$	$\|\|E(\kappa)\|\|_\infty$
21×21	62	1.064×10^{-2}	1.301×10^{-2}	1.502×10^{-2}	3.238×10^{-2}
41×41	134	2.719×10^{-3}	3.027×10^{-3}	3.520×10^{-3}	8.117×10^{-3}
81×81	270	7.434×10^{-4}	7.041×10^{-4}	8.593×10^{-4}	2.101×10^{-3}
161×161	542	1.952×10^{-4}	1.689×10^{-4}	2.135×10^{-4}	5.280×10^{-4}
Rate	/	/	/	4.1	3.9

3.5. Element refinement around the front

There are at least three different ways to introduce new degrees of freedom for both, the velocity and the pressure around the interface so as to account for their singularities there. These are:

(a) Construction of polygonal (with more than 3 sides eventually) elements by incorporating the intersection points into the front element nodes, resulting in a time-dependant mapping and non-standard element basis [24]. This technique is rather elaborate and has only been applied to 2D solid-particle flows where the domains are fictitious inside the particles. Its extension to 3D is also nontrivial.

(b) Dynamic enrichment of the element basis while keeping the standard mapping for the front elements [13, 25]. This method works on the permanent grid and does not require any reconnection of the front elements if the interface is represented by a set of Lagrangian surface markers/patches. The elemental basis is enriched to be compatible with the discontinuities of the pressure and the velocity derivatives. Due to these singularities at the interface, the Gaussian quadratures may not provide enough accuracy for the integration in the front elements and over the surface elements, and therefore one needs to use adaptive lower-order Newton–Cotes quadrature. Nevertheless, the dynamic basis enrichment can be used for 3D higher-order (Taylor–Hood) elements where the reconnection of the front elements is quite a demanding task.

(c) Local grid adaptation/reconnection (e.g. Reference [11]). In this method the front elements are refined so that the interface is always aligned with some element faces. Then the piecewise the singularity of the velocity derivatives is automatically enforced and using the pressure condensation mechanism proposed above one can enforce the singularity of the pressure. Moreover, exact (Gaussian) numerical integration can be used. The alignment/refinement of 2D triangular elements is quite an easy task and we discuss it below. The reader is referred to Reference [25] for the tetrahedron-partitioning algorithm developed for crack growth problem without regriding, although it was motivated by the need for an accurate numerical integration when implementing a similar enrichment of (b). The refinement of 3D higher-order elements is a more challenging task and the method discussed in (b) may be a better choice.

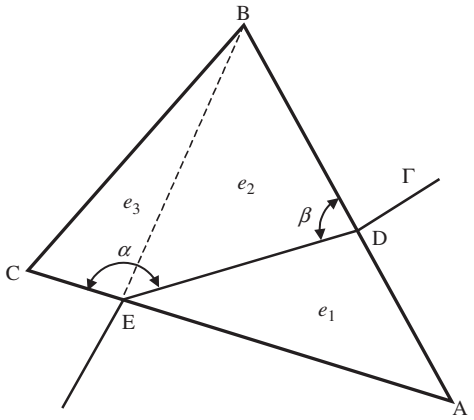

Figure 1. Reconnecting the front elements (pressure is double valued at D and E).

The identification of the front elements is quite straightforward because in such elements the level set function ϕ would have a different sign in two of the vertices. The position of the zero level set can be determined using the linear finite element interpolant of ϕ in the given front element. It is quite clear then (see Figure 1), that if the front triangle is intersected by a line so that exactly two of its sides are intersected, it can be subdivided into exactly three triangles without a violation of the connectivity rules for a finite element triangulation. Since this is the most common situation, it is algorithmically simpler if we do not allow the intersection to occur through a vertex of the front element and then this would be the only possible situation left (we assume that the reference grid is fine enough, so that the interface cannot intersect all the three sides of the triangle). This restriction is enforced in the following manner. If the distance between the zero level set point on some of the sides of the triangle and one of the vertices is smaller than a preset minimum allowed distance, $\varepsilon = O(h^2)$, than this point is fictitiously moved away from this vertex (without resetting the values of ϕ) on a distance ε. This mechanism introduces an $O(h^2)$ error in the approximation of the zeroth level set which is consistent with the overall accuracy of the method.

The extension of this refinement algorithm to 3D requires taking into account more possibilities for the intersection of Γ with the edges of the front elements. It will be presented in a forthcoming paper.

3.6. Implementation details

The implementation of the algorithm described above is a non-trivial task because of the introduction of dynamic (temporary) degrees of freedom around the interface Γ. They are stored separately from the degrees of freedom associated with the reference grid nodes. The matrices are also dynamically partitioned so that only the coefficients corresponding to the nodes in the front elements are recomputed at each time step. The overall algorithm proceeds as follows:

(a) Solve the pure convections for velocity $\tilde{\mathbf{u}}$ and level set function ϕ^{n+1} on the reference grid;

(b) Identify the front elements and compute interface curvature κ (localized in front element nodes only) on the reference grid based on the advected ϕ^{n+1};
(c) Reconnect the front elements to construct a new refined grid, interpolate $\mathbf{u}^n$, $\tilde{\mathbf{u}}$ and κ at the intersection points. Compute the 'pressure' jump $[\Theta]$ and condense the pressure;
(d) Solve the problems for $\mathbf{u}^{n+1}$ and Θ^{n+1} on the refined grid;
(e) Reload the velocity and pressure vectors and go to the next time level.

In all cases when the mass matrix is to be inverted, we used the lumped version of this matrix that greatly reduces the computational cost. Note that in case of first-order elements (used here) the mass lumping through inexact integration is equivalent to row-summing of the matrix. The mass lumping does not spoil the spatial accuracy of the method in case of first-order elements.

4. NUMERICAL RESULTS

4.1. Front reconnection/refinement test

Before applying it to multi-phase flows, we tested the proposed method, especially the front element reconnection, on a well-known single-phase problem, the lid-driven cavity flow. We set $\gamma = 0$, $\rho_1 = \rho_2 = 1.0$ and $\mu_1 = \mu_2 = 1/Re$, where $Re = 40$ is the Reynolds number based on the upper-lid velocity and cavity length. The unit cavity is subdivided into a grid of right triangles with 31^2 nodes. In the first case we resolve the steady flow without reconnection (on the reference grid only). In the second case we initially place a circular curve (properly initializing ϕ) in the middle of the cavity and perform the refinement around this (advected) fictitious interface at each time step. The two results are compared in Figure 2 that shows the reference/refined grids and velocity contours. At $t = 2.0$, when the front has undergone a severe deformation the results are essentially the same.

4.2. Curvature-driven flow

The next test case is the so-called equilibrium state test. Consider a circular fluid particle in another viscous fluid at rest, under a zero gravity condition. Even at non-zero surface tension the circular shape of the particle should be preserved and the fluids should remain at rest no matter how long we integrate the equations in time. This problem is much more demanding than it appears at a first glance. It is particularly troublesome if an Eulerian method is used to solve it. For strong surface tension or long time integration, a significant loss of mass may be accumulated in this case (see Reference [6]). We simulated the equilibrium state of a circular bubble of a radius $R = 0.25$ (constant curvature $\kappa = 1/R$) in a static fluid with $\rho_1 = \rho_2 = 1$, $\mu_1 = \mu_2 = 0.1$, $g = 0$, $\gamma = 1$ (Laplace number $La = \rho\gamma R/\mu^2 = 250$). The unit square domain Ω was meshed using 25^2 nodes and right angle triangles. The initial velocity and pressure are set to zero. The system reaches the steady state instantly, right after the first time step and the pressure becomes almost constant in each of the phases. The dimensionless velocity measured by the capillary number $Ca = ||\mathbf{u}||_\infty \mu/\gamma$ increases to $O(10^{-5})$ but does not grow in time. This, so-called 'parasitic flow', is caused by the approximation of the curvature of the interface. Note, however, that in dynamic problems, there will be an additional contribution due to the splitting error of the projection schemes that deteriorates the incompressibility of the velocity field. In spite of the piecewise-linear approximation of the interface, the

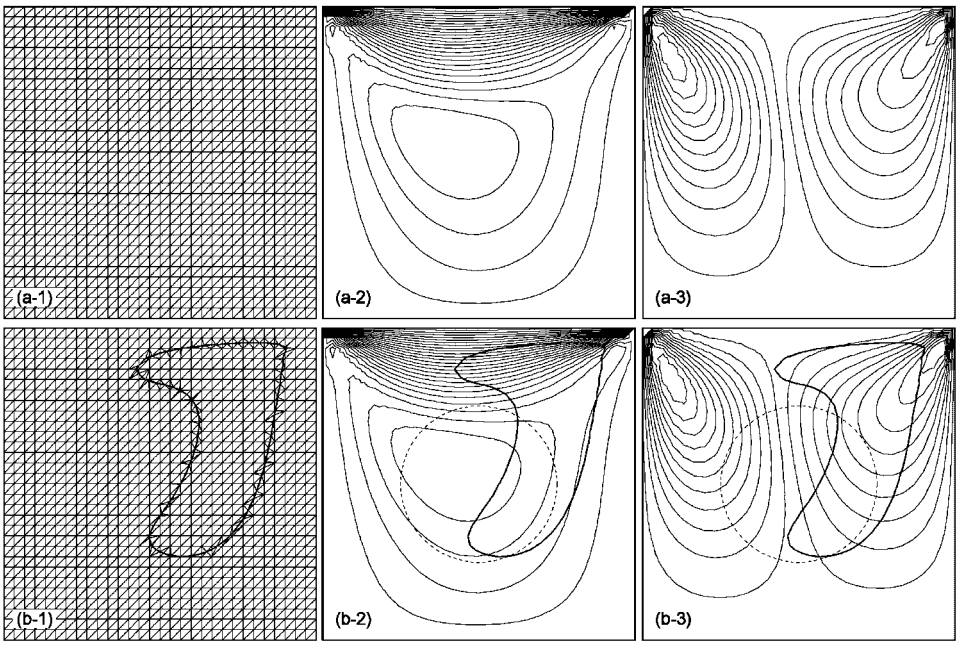

Figure 2. Single-phase cavity-driven flow test for the grid reconnection and skewness tolerance ($Re = 40,\ t = 2.0$): (a) without front; and (b) with front reconnection; (1) reference/refined grids, (2) u contours and (3) v contours. The dashed line indicates the initial shape.

magnitude of the 'parasitic flow' is within an acceptable range, one order of magnitude lower than in the front tracking method of Shin and Juric [26]. We used the same resolution to resolve the problem as in this latter study. The reader is referred to References [4, 21] for detailed discussions on this numerical phenomenon.

The next test problem that we considered is the so-called bubble relaxation test. At nonzero surface tension and zero gravity conditions any fluid particle whose shape deviates from circular (spherical in 3D) should recover the circular (spherical in 3D) static and stable shape. An ellipse was initially placed at the centre of a unit square cavity discretized with 31^2 nodes and right angle triangles. The ellipse axes are $5/16$ and $1/5$, and its area is equal to the area of a circle with a radius of $1/4$. The fluid properties outside and inside the ellipse are $\rho_1 = \rho_2 = 1.0,\ \mu_1 = \mu_2 = 0.1,\ g = 0$. With $\gamma = 1.0$, which is considered to be a large surface tension coefficient, the flow quickly reaches the equilibrium state. As shown in Figure 3(a), the ellipse restores a circular shape within 30 time steps ($\Delta t = 0.01$). One of the instantaneous velocity fields is shown in Figure 3(b). At the 100th time step when the simulation was terminated, the mass loss was less then 0.1%. The initial and equilibrium-state results are presented in Figure 3. In Table II we present the results for the same problem obtained on a series of successively refined grids. In order to eliminate the influence of the temporal error we chose a very small time step $\Delta t = 0.001$. The results are analysed at $t = 0.2$ when the bubble shape is still elliptical. The relative errors (between two successive grids) measured in various norms show slightly lower than quadratic convergence for both, the velocity and the pressure. The level set function converges with the same rate. It is also seen that the use

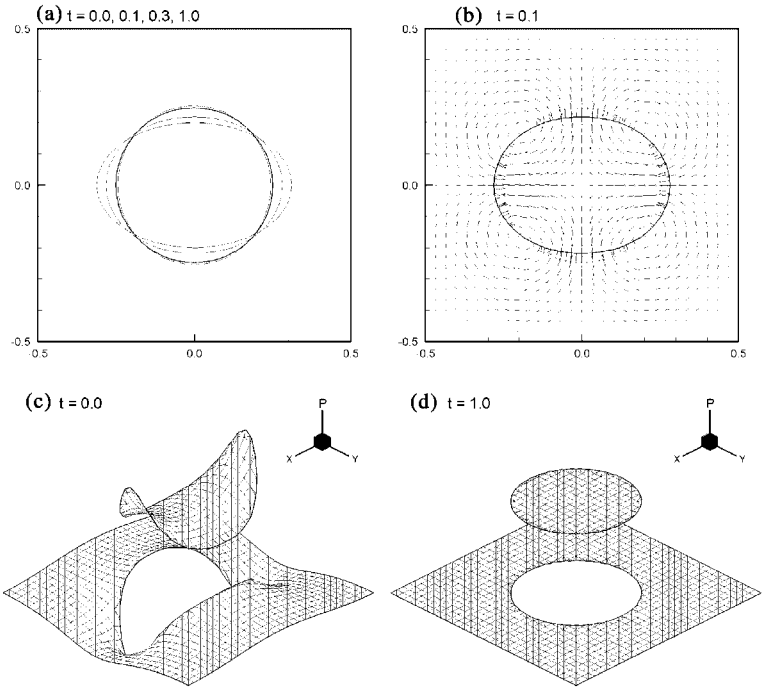

Figure 3. Relaxation of an initially elliptical bubble in a viscous fluid ($\rho_1 = \rho_2 = 1.0,\ \mu_1 = \mu_2 = 0.1,\ \gamma = 1.0,\ \Delta t = 0.01,\ 31^2$ nodes): (a) front evolution; (b) velocity field at $t = 0.1$; (c) the plot of Θ right after the first time step; and (d) the equilibrium-state Θ. Initial shape $\phi(x, y) = (16x/5)^2 + (5y)^2 - 1$.

Table II. Convergence test for the relaxation of an initially ellipsoid bubble driven by large surface tension, analysis at $t = 0.2$ with $\Delta t = 0.001$.

Reference grid	Change of mass (%)	$\|\|E(\mathbf{u})\|\|_1$	$\|\|E(P)\|\|_1$	$\|\|E(\mathbf{u})\|\|_\infty$	$\|\|E(P)\|\|_\infty$
21×21	0.22	/	/	/	/
41×41	0.09	2.503×10^{-2}	6.458×10^{-2}	2.852×10^{-2}	1.864×10^{-1}
81×81	0.02	7.166×10^{-3}	1.796×10^{-2}	8.028×10^{-3}	3.307×10^{-2}
Rate	/	3.5	3.6	3.6	5.6

of L_∞ norm for measuring pressure is problematic, since pressure is discontinuous across the interface.

Another test is the relaxation of a more sophisticated starfish-shaped droplet defined by the level-set function

$$\phi(r, \theta) = r - 0.2\sin(5\theta) - 0.5,\ \ (0 \leqslant \theta \leqslant 2\pi)$$

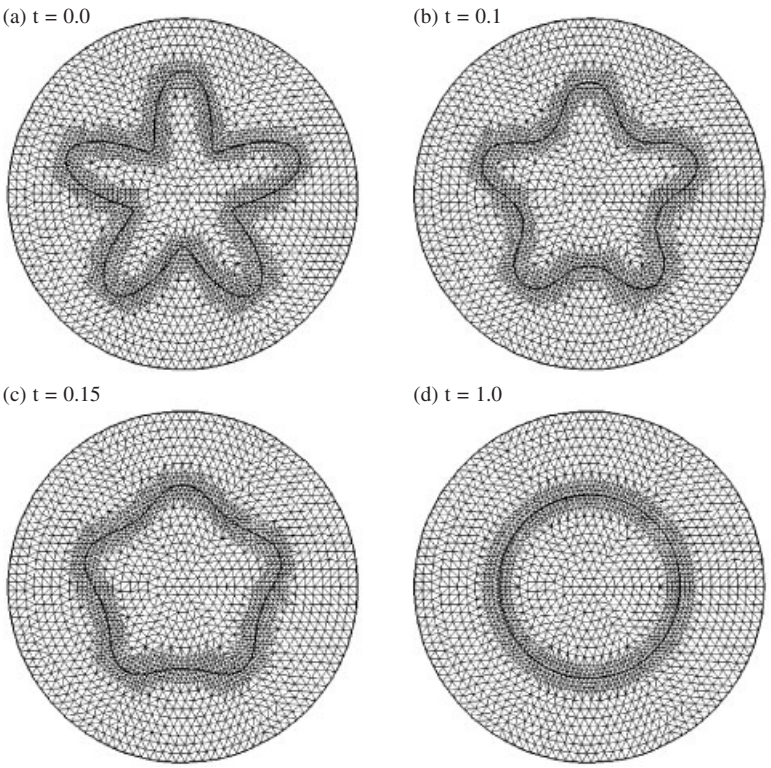

Figure 4. Relaxation of an initially starfish-shaped droplet in an otherwise stationary fluid ($\rho_1 = \rho_2 = 1.0$, $\mu_1 = \mu_2 = 0.1$, $\gamma = 1.0$, $\Delta t = 0.005$, effective grid size $\cong 1/40$). Initial shape $\phi(r, \theta) = r - 0.2\sin(5\theta) - 0.5$.

initially placed at the centre of a circular ($R = 1$) domain. The parameter setting is $\rho_1 = \rho_2 = 1.0$, $\mu_1 = \mu_2 = 0.1$, $g = 0$ and $\gamma = 1.0$. The domain is resolved with a general triangulation of element size $h \cong 1/20$ and $\Delta t = 0.005$. Prior to the front elements being reconnected at each time step, we conducted an adaptive h-refinement of the reference grid, which was done by quartering the front elements and their immediate neighboring elements. The effective element size near the interface was therefore $h_e = h/2$. Due to the large surface tension, the interface quickly approached the equilibrium state with a small-amplitude oscillation around it. The evolution of the interface and the adaptive h-refinement are presented in Figure 4. Only 0.12% of mass loss was accumulated during the entire simulation and it occurred mostly (0.08%) in the first few time steps. The same problem was solved by Li and Lai [19] using the immersed interface method, except that they used a finer uniform Cartesian grid (160^2 in a $[2 \times 2]$ domain) and a much smaller surface tension coefficient ($\gamma = 0.05$). Nevertheless, the loss of mass they reported was 1.61%.

4.3. A rising gas bubble

At the end, we present a quantitative comparison for the motion of a cylindrical gas bubble formed in a low-viscosity liquid with the experimental investigation by Walters and Davidson

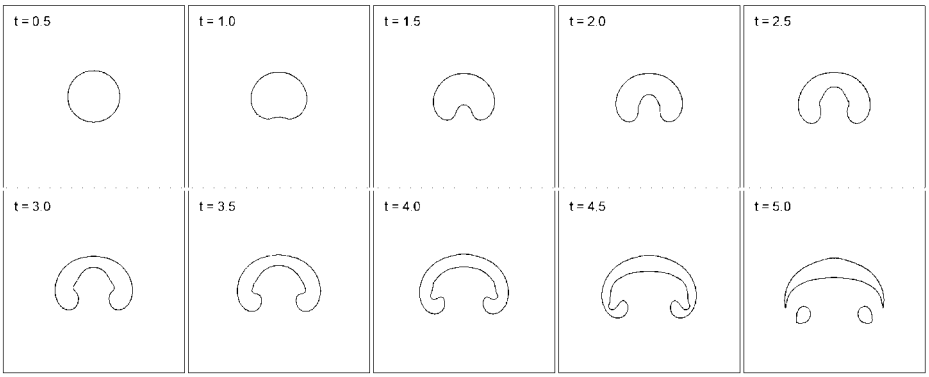

Figure 5. Evolution in time of a gas bubble in water ($Re = 1000$, $We = 200$, $Fr = 1$, $\rho_1/\rho_2 = 816$, $\mu_1/\mu_2 = 64$, $\Delta t = 0.001$).

[27]. Defining the reference velocity $U = (gR)^{1/2}$ where R is the initial bubble radius, we have the Reynolds number defined by $Re = \rho_1 UR/\mu_1$, Weber number $We = \rho_1 U^2 R/\gamma$ and unity Froude number $Fr = 1$. The centroid of the bubble and its speed are computed by

$$\mathbf{x}_b = \int_{\Omega_2} \mathbf{x}\, \mathrm{d}\Omega / \int_{\Omega_2} \mathrm{d}\Omega \quad \text{and} \quad \mathbf{v}_b = \int_{\Omega_2} \mathbf{u}\, \mathrm{d}\Omega / \int_{\Omega_2} \mathrm{d}\Omega$$

The values of the parameters are $Re = 1000$, $We = 200$, $\rho_1/\rho_2 = 816$ and $\mu_1/\mu_2 = 64$, which are similar to the experimental values in Reference [27]. The problem was solved in half of the domain imposing symmetry boundary condition at axis of symmetry. In order to prevent the bubble from leaving the computational domain we keep the bubble centroid fixed by subtracting $\mathbf{v}_b$ from the convection velocity in (7a). This idea is borrowed from the concept of grid velocity in the ALE. Instead of monitoring $\mathbf{x}_b$, which should be fixed at the origin, the characteristic trajectory of the centroid can be obtained by integrating $\mathbf{v}_b$ in time. The boundary conditions are no-slip for the vertical walls and open for the bottom. The computational domain is $[0,\ 3\frac{1}{2}] \times [-3\frac{1}{2},\ 3\frac{1}{2}]$ with the bubble centroid at the origin, discretized by effective 101×201 nodes and triangular elements.

The bubble evolution is presented in Figure 5. The formation of liquid tongue, skirt rolling, pinch off and spherical cap of the main bubble correlate well with the experimental pictures taken by Walters and Davidson [27]. The pinch off started at about $t = 4.42$ in their experiment (the 18th frame, 80 frame/s, $R = 2.54$ cm). The timing of pinch off also agrees with the numerical investigations by Baker and Moore [28] and Sussman *et al.* [6] obtained with different methods. Details of the detachment of two small bubbles from the main one at its lower edges are shown in Figure 6. In this case approximately 7% of the bubble mass was lost, mainly due to the unresolved sub-grid effect during the stretching and pinch off. If these effects are not presented the scheme preserves the mass reasonably well. The distance covered by the bubble centroid as a function of time is given in Figure 7, compared with the experimental measurements and the potential theory of Walters and Davidson [27]. The entire evolution of the velocity and the centroidal displacement are presented in Figure 8. After the initial acceleration ($t < 1$) the velocity decelerates to a minimum at $t \cong 3.0$ and then starts to increase until the detachment of two small bubbles.

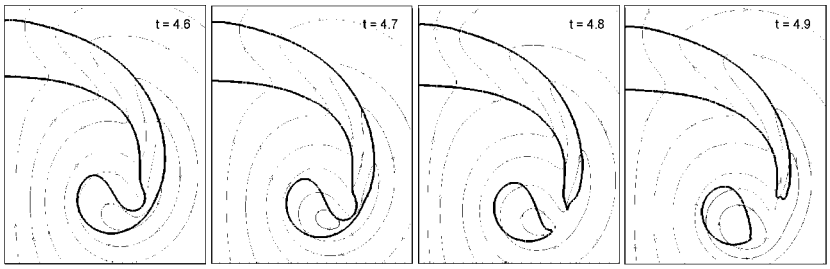

Figure 6. Pinch off of a gas bubble in water, detachment and streamtraces ($Re = 1000$, $We = 200$, $Fr = 1$, $\rho_1/\rho_2 = 816$, $\mu_1/\mu_2 = 64$).

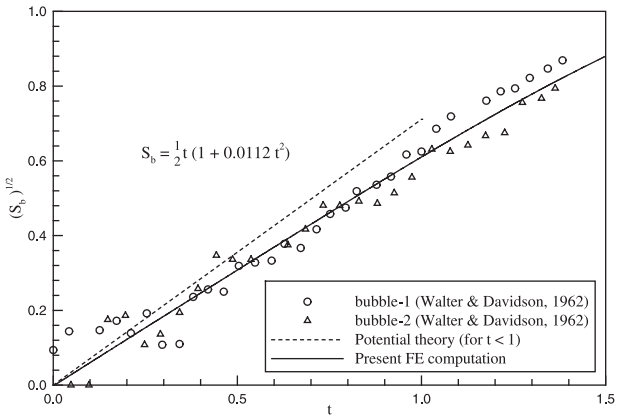

Figure 7. Initial motion of a cylindrical gas bubble in water: centroid displacement ($Re = 1000$, $We = 200$, $Fr = 1$, $\rho_1/\rho_2 = 816$, $\mu_1/\mu_2 = 64$).

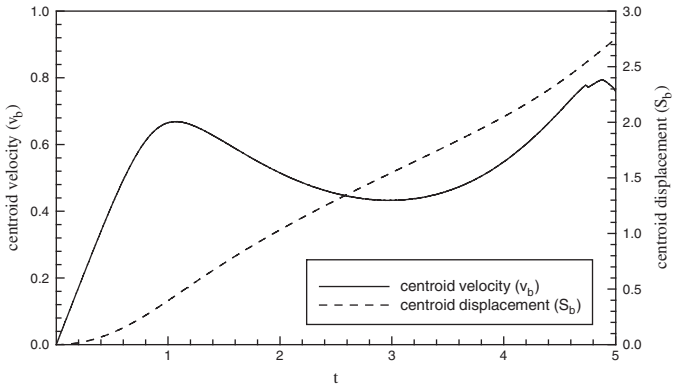

Figure 8. Time histories of velocity and displacement of the bubble centroid ($Re = 1000$, $We = 200$, $Fr = 1$, $\rho_1/\rho_2 = 816$, $\mu_1/\mu_2 = 64$).

5. CONCLUSIONS

A 2D finite element method for incompressible multifluid flows with capillary interfaces, based on a formally second-order projection scheme has been formulated in this paper. Its simplicity and effectiveness have been demonstrated for a wide range of free-surface problems. Extension of the present method to 3D is rather straightforward because the 3D-reconnection technique already exists. The fluid phases are advected and identified using the level set function. Using a front reconnection technique, the interfacial singularities are optimally approximated.

The least-squares method for computing the curvature enables the piecewise linear approximation for the interface. With time splitting, the convection step, including that for the level set function, is integrated over the permanent grid for a more relaxed restriction on numerical stability. The success of the projection method for the remaining generalized Stokes problem heavily relies on a proper pressure condensation. It will not only facilitate the 'one-field' solver, but also allows the pressure to instantly adjust itself to any significant jump across an interface due to the buoyancy and surface tension. The resultant convergence of pressure is much faster than otherwise ignoring the jump. Numerical experiments have shown that such an internal and implicit treatment of pressure will not lead to the spurious boundary layer known for the classic projections due to the splitting error. The pressure Laplacian matrix with a proper condensation for the jump is also a good preconditioner for the higher-order Uzawa method. Nevertheless, as we have pointed out, in order that a second-order incremental scheme works better with the multiphase flow, special care must be taken to insure that the guessed pressure and the current pressure will not unphysically fall into different phase domains across a significant jump.

ACKNOWLEDGEMENTS

This study was supported by a research grant of the National Science and Engineering Research Council of Canada.

REFERENCES

1. Osher S, Sethian JA. Fronts propagating with curvature dependent speed: algorithms based on Hamilton-Jacobi formulation. *Journal of Computational Physics* 1988; **79**:12–49.
2. Unverdi SO, Tryggvason G. A front-tracking method for viscous, incompressible, multi-fluid flows. *Journal of Computational Physics* 1992; **100**:25–37.
3. Osher S, Fedkiw RP. Level set methods: an overview and some recent results. *Journal of Computational Physics* 2001; **169**:463–502.
4. Tryggvason G, Bunner B, Esmaeeli A, Juric D, Al-Rawahi N, Tauber W, Han J, Nas S, Jan YJ. A front-tracking method for the computations of multiphase flow. *Journal of Computational Physics* 2001; **169**:708–759.
5. Sussman M, Puckett EG. A coupled level set and volume-of-fluid method for computing 3D and axisymmetric incompressible two-phase flows. *Journal of Computational Physics* 2000; **162**:301–337.
6. Sussman M, Smereka P, Osher S. A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational Physics* 1994; **114**:146–159.
7. Brackbill JU, Kothe DB, Zemach C. A continuum method for modelling surface tension. *Journal of Computational Physics* 1992; **100**:335–354.
8. Cruchaga M, Celentano D, Tezduyar T. A moving Lagrangian interface technique for flow computations over fixed grids. *Computer Methods in Applied Mechanics and Engineering* 2001; **191**:525–543.
9. Pillapakkam SB, Singh P. A level-set method for computing solutions to viscoelastic two-phase flow. *Journal of Computational Physics* 2001; **174**:552–578.
10. Quecedo M, Pastor M. Application of the level set method to the finite element solution of two-phase flows. *International Journal for Numerical Methods in Engineering* 2001; **50**:645–663.

11. Lock N, Jaeger M, Medale M, Occelli R. Local grid adaptation technique for front tracking problems. *International Journal for Numerical Methods in Fluids* 1998; **28**:719–736.
12. Torres DJ, Brackbill JU. The point-set method: front-tracking without connectivity. *Journal of Computational Physics* 2000; **165**:620–644.
13. Minev PD, Chen T, Nandakumar K. A finite element technique for multifluid incompressible flow using Eulerian grids. *Journal of Computational Physics* 2003; **187**:255–273.
14. Kim J, Moin P. Application of a fractional-step method to incompressible Navier-Stokes equations. *Journal of Computational Physics* 1985; **59**:308–323.
15. Li Z, Lubkin SR. Numerical analysis of interfacial two-dimensional Stokes flow with discontinuous viscosity and variable surface tension. *International Journal for Numerical Methods in Fluids* 2001; **37**:525–540.
16. Thompson E. Use of pseudo-concentrations to follow creeping viscous flows during transient analysis. *International Journal for Numerical Methods in Fluids* 1986; **6**:749–761.
17. Guermond JL, Minev P, Shen J. Numerical issues on projection methods. In preparation.
18. Guermond JL, Shen J. On the error estimates for the rotational pressure correction projection methods. *Mathematics of Computation*, submitted.
19. Li Z, Lai MC. The immersed interface method for the Navier-Stokes equations with singular forces. *Journal of Computational Physics* 2001; **171**:822–842.
20. Codina R. Pressure stability in fractional step finite element methods for incompressible flows. *Journal of Computational Physics* 2001; **170**:112–140.
21. Ginzburg I, Wittum G. Two-phase flows on interface refined grids modeled with VOF, staggered finite volumes, and spline interpolants. *Journal of Computational Physics* 2001; **166**:302–335.
22. Peng D, Merriman B, Osher S, Zhao H, Kang M. A PDE-base fast local level set method. *Journal of Computational Physics* 1999; **155**:410–438.
23. Gresho PM, Sani RL. *Incompressible Flow and the Finite Element Method* (V2). Wiley: New York, 2000.
24. Peskin AP, Hardin GR. Moving particles through a finite element grid. *Journal of Research National Institute Standards and Technology* 1998; **103**:77–91.
25. Sukumar N, Moes N, Moran B, Belyschko T. Extended finite element method for three-dimensional crack modelling. *International Journal for Numerical Methods in Engineering* 2000; **48**:1549–1570.
26. Shin S, Juric D. Modeling three-dimensional multiphase flow using a level contour reconstruction method for front tracking without connectivity. *Journal of Computational Physics* 2002; **180**:427–470.
27. Walters JL, Davidson JF. The initial motion of a gas bubble formed in an inviscid liquid, Part 1. The two-dimensional bubble. *Journal of Fluid Mechanics* 1962; **12**:409–417.
28. Baker GR, Moore DW. The rise and distortion of a two-dimensional gas bubble in an inviscid liquidine. *Physics of Fluids* 1989; **A 1**(9):1451–1459.

