Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 225 (2007) 785–807
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

A conservative level set method for two phase flow II

Elin Olsson [a,*], Gunilla Kreiss [b], Sara Zahedi [a]

[a] Department of Numerical Analysis and Computer Science, Royal Institute of Technology, 100 44 Stockholm, Sweden
[b] Division of Scientific Computing, Department of Information Technology, Uppsala University, 751 05 Uppsala, Sweden

Received 10 January 2006; received in revised form 21 December 2006; accepted 28 December 2006
Available online 2 February 2007

Abstract

In this paper, we continue to develop and study the conservative level set method for incompressible two phase flow with surface tension introduced in [J. Comput. Phys. 210 (2005) 225–246]. We formulate a modification of the reinitialization and present a theoretical study of what kind of conservation we can expect of the method. A finite element discretization is presented as well as an adaptive mesh control procedure. Numerical experiments relevant for problems in petroleum engineering and material science are presented. For these problems the surface tension is strong and conservation of mass is important. Problems in both two and three dimensions with uniform as well as non-uniform grids are studied. From these calculations convergence and conservation is studied. Good conservation and convergence are observed.

Keywords: Level set method; Two phase flow; Conservative method

1. Introduction

To study the extraction of petroleum and the construction of hard metals through liquid phase sintering it is necessary to be able to model incompressible two phase flow with surface tension accurately. If no phase change occurs the mass of each of the two fluids should be conserved. Since the flow is incompressible this implies that the volume occupied by any of the fluids should be preserved as well. It is therefore of importance that the method used to simulate the flow conserves these volumes.

The most commonly used methods to model incompressible two phase flow with surface tension are the volume of fluid method (VOF) [1,2], the level set method [3,4], the front tracking method [5] and phase field methods [6]. Here we focus on the volume of fluid method and the level set method. There are advantages and disadvantages of both of these methods.

The VOF method has the main advantage of conserving the volumes of the two fluids exactly. The interface is however represented by a discontinuity of a globally defined function. Because of the discontinuity it is hard

☆ DOI of original article: 10.1016/j.jcp.2005.04.007.
* Corresponding author. Tel.: +46 8 790 71 87.
E-mail addresses: elin@comsol.se (E. Olsson), gunillak@it.uu.se (G. Kreiss), sara7@kth.se (S. Zahedi).

0021-9991/$ - see front matter © 2007 Published by Elsevier Inc.
doi:10.1016/j.jcp.2006.12.027

both to move the interface as well as calculating surface tension, which depends on the mean curvature of the interface, accurately. Specially designed methods have to be used to advect the interface. Standard finite element discretizations will for example not work.

In level set methods the interface is defined by the zero contour of a signed distance function, the level set function. Since this function is smooth across the interface it is easier to advect the interface as well as to calculate the curvature with high order of accuracy. Both finite element and finite difference approximations of the level set method have been used successfully to advect the interface. There is however no built in volume conservation. A small amount of mass is thus lost or gained in each time step. As time evolves these errors will typically accumulate.

In [7] we constructed a modified level set method with built in conservation. Here, the level set function Φ was a regularized characteristic function. A reinitialization procedure, formulated as a conservation law, was used to preserve the smooth profile of the regularized characteristic function. Conservative methods could then be used for both advection and reinitialization. Because of this $\int_\Omega \Phi \, d\Omega$ was conserved exactly. Since Φ was a regularized characteristic function, this implied a good conservation of the volume bounded by the sharp interface $\Phi = 0.5$. The numerical calculations showed very good conservation of the volume bounded by the sharp interface. For the test cases with given velocity fields good convergence was obtained. A rising air bubble in water was also studied. Mass conservation was very good, but the convergence as the grid was refined was slow.

We believe that a possible reason for the slow convergence for the rising bubble was the formulation of the reinitialization. In the reinitialization there was diffusion. The diffusion in the direction normal to the interface was balanced by a compressive term, such that the resulting motion of the interface could be expected to be small. The diffusion tangential to the interface was however not balanced by any compression. This tangential diffusion might thus have moved the interface slightly. In this paper, we propose the diffusion in the reinitialization to be only in the direction normal to the interface. We obtain good convergence with an estimated order of accuracy of two for two phase flow calculations with strong surface tension.

The convergence to steady state in the reinitialization step will be studied. We will see that we can expect this convergence to be exponential in time on a fast time scale. This implies that we can expect the computational expenses due to the reinitialization to be small. We will also perform a theoretical analysis on what kind of conservation we can expect of the method. The numerical tests agree well with our theoretical results.

We also propose a finite element discretization of the method. An adaptive procedure is also presented, such that the grid can be refined close to the interface. Using an existing finite element package this could easily be implemented for both two and three dimensional problems. Results for a few different problems are presented to investigate the conservation and convergence of the method.

2. The conservative level set method

A conservative level set method for motion of interfaces in a given divergence free velocity field was introduced in [7]. We summarize the basic idea of the method here. Assume an arbitrary domain Ω divided into subdomains Ω_1 and Ω_2 such that $\Omega_2 = \Omega \setminus \Omega_1$. Let Γ be the internal boundary between Ω_1 and Ω_2. The idea of the conservative level set method is to define the boundary Γ implicitly by a function Φ being a regularized characteristic function. This means that Φ goes rapidly from zero to one across the interface Γ and that $\Phi \approx 1$ for $x \in \Omega_1$ and $\Phi \approx 0$, $x \in \Omega_2$ away from the boundary Γ. Γ can be sharply defined as the 0.5 contour of Φ. The normal, $\hat{n}$, and curvature, κ, of the interface can easily be calculated from $\hat{n} = \nabla\Phi/|\nabla\Phi|$ and $\kappa = -\nabla \cdot \hat{n}$. As the interface moves we want to keep the shape of the profile constant. Small perturbations in the shape should be damped, since in numerical computations there will always be perturbations. How the shape is preserved will be motivated by a study of the one dimensional case in the following section.

2.1. Stable traveling wave solutions in one dimension

Consider advection in one dimension for some constant velocity v:

$$\Phi_t + v\Phi_x = 0. \tag{1}$$

Assume that initially Φ is the regularized characteristic function given by

$$\Phi_0(x) = \frac{1}{1 + e^{-x/\varepsilon}}. \tag{2}$$

Next assume an initial perturbation $\delta(x)$, i.e.

$$\Phi(x, 0) = \Phi_0(x) + \delta(x).$$

The solution to (1) is then

$$\Phi(x, t) = \Phi_0(x - vt) + \delta(x - vt).$$

Perturbations will thus not be damped but simply advected with v. For numerical calculations, where new perturbations are constantly introduced, the shape will become more and more distorted as time evolves. All stable numerical methods also have some artificial diffusion that will smear the profile. We thus have to stabilize (1) such that $\Phi(x, t) \to \Phi_0(x - vt)$ as $t \to \infty$.

To achieve this, we study the partial differential equation (PDE)

$$\Phi_t + v\Phi_x = \frac{1}{\mu}(\varepsilon\Phi_{xx} - (\Phi(1 - \Phi))_x). \tag{3}$$

If $\Phi(x, 0) = \Phi_0$, where Φ_0 is given by (2), then $\Phi(x, t) = \Phi_0(x - vt)$, i.e. the analytical solution is identical to the solution of (1). What happens now if we perturb the initial data? Introducing $\xi = x - vt$, $\tau = t/\mu$, and $\Psi(\xi, \tau) = 1 - 2\Phi(\xi, \tau)$, in (3) yields the well known Burgers equation for Ψ:

$$\frac{1}{2}(\Psi^2)_\xi + \Psi_\tau = \varepsilon\Psi_{\xi\xi}.$$

Steady state solutions to this equations are $\Psi_0(\xi + \gamma)$ with $\Psi_0(\xi) = 1 - 2\Phi_0(\xi)$ for any shift γ. We can thus in general only hope for convergence in time to a shifted profile. The evolution of perturbations to the initial data $\Psi_0(\xi) = 1 - 2\Phi_0(\xi)$ this equation was studied in [8]. By assuming the perturbations $\delta(\xi, \tau)$ to be small, the evolution of $\delta(\xi, \tau)$ can be approximated by a linearized problem

$$\delta = \mathscr{L}\delta.$$

It was shown in [8] that the spectrum of the operator $\mathscr{L}$ consists of an isolated eigenvalue $\lambda = 0$ plus a continuous spectrum $-\infty < \lambda < -c/\varepsilon$ for some $c > 0$.

It can be shown that the eigenvalue 0 corresponds to a shift in Ψ. This means that we can expect small perturbations to $\Psi_0(\xi)$ to converge to a shifted Ψ, i.e. $\Psi(\xi, \tau) \to \Psi_0(\xi + \gamma)$, as $\tau \to \infty$. The convergence to the shifted steady state can be expected to be exponential in time on the time scale $\tau \sim \varepsilon$. The stability of viscous shocks was also studied in [9]. There it was shown that if the perturbation is of zero mass, i.e. if $\int_{-\infty}^{\infty} \delta(\xi, 0)\, d\xi = 0$, then there will be no shift, i.e. $\gamma = 0$. Since v is constant, (1) is also a conservation law. When using conservative numerical schemes, the truncation errors will be of zero mass. We thus expect these errors to decrease rapidly in time without shifting the profile.

The time scale on which the perturbations decrease will be $t \sim \mu\varepsilon$. We assume the velocity $v = \mathcal{O}(1)$. If $\varepsilon\mu \ll 1$ this time scale will be much faster than the time scale related to the advection. For a constant velocity it is reasonable to choose $\mu = 1$, since ε is assumed to be small.

Remark: We are only interested in divergence free velocity fields. For a velocity field with only one non-zero component, the divergence free condition implies that the velocity is constant. It does therefore not make sense to discuss variable velocities in one space dimension.

2.2. Stabilized advection in two and three dimensions

For divergence free velocities $\vec{u}$ in several space dimensions we want to stabilize the profile across the interface in the direction normal to the interface Γ. The stabilized advection can be expressed as

$$\Phi_t + \nabla \cdot (\vec{u}\Phi) = \frac{1}{\mu}\nabla \cdot (-\Phi(1 - \Phi)\hat{n} + \varepsilon(\nabla\Phi \cdot \hat{n})\hat{n}), \tag{4}$$

where $\hat{n} = \nabla\Phi/|\nabla\Phi|$. In several space dimensions, the divergence free condition on the velocity does not imply a constant velocity. Then also variations in the velocity will distort the shape of Φ across the interface. This implies that it might be necessary to make μ, smaller in order to keep the profile of Φ across the interface. Typically, the size of μ will depend on the gradient of the velocity at the interface. To handle this numerically we split the advection and the stabilization into a set of two PDEs. First one time step of

$$\Phi_t + \nabla \cdot (\Phi\vec{u}) = 0 \tag{5}$$

will be solved. Using the resulting Φ, which we denote by Φ_*^{n+1}, as initial condition one time step of

$$\Phi_t + \frac{1}{\mu}\nabla \cdot (\Phi(1-\Phi)\hat{n}) = \frac{1}{\mu}\varepsilon\nabla \cdot ((\nabla\Phi \cdot \hat{n})\hat{n}) \tag{6}$$

is solved. Here, $\hat{n} = \nabla\Phi_*^{n+1}/|\nabla\Phi_*^{n+1}|$. One time step of Eq. (6) can be replaced by taking μ timesteps of

$$\Phi_\tau + \nabla \cdot (\Phi(1-\Phi)\hat{n}) = \varepsilon\nabla \cdot ((\nabla\Phi \cdot \hat{n})\hat{n}), \tag{7}$$

where $\tau = t/\mu$ and the time steps are equal, $\Delta t = \Delta\tau$. Note that in both (6) and (7) the same $\hat{n}$ is used, and that $\hat{n}$ does not change during the stabilization step. We want to choose μ such that the shape of Φ across the interface is given by the steady state solution of (7). Thus, instead of explicitly determining μ, we solve (7) to steady state. The process of solving (7) to steady state will be referred to as the reinitialization step. In this way we do not have to determine how small μ, should be. Moreover, from the discussion in Section 2.1 we can expect the time for convergence of (7) to steady state to be of $\mathcal{O}(\varepsilon)$. As will be seen later we will choose the discretization parameters to $\Delta t \sim \Delta x \sim \varepsilon$. This implies that only a few time steps, $\mathcal{O}(1)$, will be needed to reach the steady state of (7).

Since both the advective step as well as the reinitilization step are formulated as conservation laws, $\int_\Omega \Phi \, dx$ is constant in time in the continuous case. By using conservative numerical methods, this quantity can also be conserved exactly in the discrete approximation of the PDEs. The conservative properties of the method will be discussed further in Section 4.

2.2.1. Modification of the reinitialization step

In the original formulation in [7] the reinitialization was given by

$$\Phi_\tau + \nabla \cdot (\Phi(1-\Phi)\hat{n}) = \varepsilon\nabla \cdot (\nabla\Phi). \tag{8}$$

If $\hat{n} = \nabla\Phi/|\nabla\phi|$, then $\nabla\Phi(\nabla\Phi \cdot \hat{n})\hat{n}$. This implies that the diffusion will result in a flux in the direction of $\hat{n}$ only. This flux will rapidly become balanced by the compressive flux $\Phi(1-\Phi)\hat{n}$. Hence, we can expect the motion of the interface during the reinitialization to be small. We will however fix $\hat{n}$ such that $\hat{n}(\tau) = \nabla\Phi_{\tau=0}/|\nabla\Phi_{\tau=0}|$ during each reinitialization step, $\hat{n} = \nabla\Phi/|\nabla\Phi|$ will then only hold for $\tau = 0$. As τ increases the diffusion might result in a small flux in the direction tangential to the interface. This flux will not be balanced by any compression. Because of this, the tangential diffusion might move the interface. This effect will be particularly strong if a large τ is needed to reach the steady state of (8). To avoid any tangential diffusion, we have replaced (8) by (7).

We will use finite elements to discretize the reinitialization step. For these to be stable (see for example [10]) diffusion is needed in the direction of the compression only. Because of this we will not have to add any artificial diffusion to stabilize the discretization of (7).

2.3. Discretization using finite elements

In order to be able to easily apply our method to problems with complex geometries as well as to simplify the use of adaptive grids we use a finite element discretization of the PDEs in space. In time finite differences are used.

For all finite element approximations we need to define finite dimensional function spaces. Throughout this paper we will denote

$$V_h = \{f(x) : f(x) \text{ is piecewise linear within } \Omega \text{ and } f(x) = 0 \ \forall x \in \Lambda \subset \partial\Omega\}.$$

We will not explicitly define Λ, but simply assume it to be the part of the boundary where Dirichlet boundary conditions have been given on the corresponding unknown. Vector valued function spaces with d components will be denoted by

$$W_h = \{\vec{f}(x) = [f_1(x), \ldots, f_d(x)]^{\mathrm{T}} : f_i(x) \text{ is piecewise linear within } \Omega \text{ and } f_i(x) = 0 \ \forall x \in \Lambda_i \subset \partial\Omega, \ i = 1, \ldots, d\}. \tag{9}$$

We will assume the boundaries to be walls, and impose no flux boundary conditions for Φ. Thus, no Dirichlet conditions on Φ are set, i.e. $\Lambda = \emptyset$ for Φ. The spatial finite element discretization of (5), the advection step, can now be formulated as: Find $\Phi \in V_h$ such that

$$\int_\Omega v\Phi_t \, \mathrm{d}x - \int_\Omega \nabla v \cdot (\Phi\vec{u}) \, \mathrm{d}x + \int_{\partial\Omega} v\phi\vec{u} \cdot \hat{v} \, \mathrm{d}S = 0 \quad \forall v \in V_h. \tag{10}$$

Here $\hat{v}$ is the normal of $\partial\Omega$. The boundary term vanishes if the boundaries are walls, since then $\vec{u} \cdot \hat{v} = 0$. The spatial discretization of the reinitialization (7) is given by: Find $\Phi \in V_h$ s.t.

$$\int_\Omega v\Phi_\tau \, \mathrm{d}x + \int_\Omega \nabla v \cdot (-\vec{f} + \varepsilon(\nabla\Phi \cdot \hat{n})\hat{n}) \, \mathrm{d}x + \int_{\partial\Omega} v(\vec{f} - \varepsilon(\nabla\Phi \cdot \hat{n})\hat{n}) \cdot \hat{v} \, \mathrm{d}S = 0 \quad \forall v \in V_h. \tag{11}$$

Here, $\vec{f} = \Phi(1 - \Phi)\hat{n}$. By omitting the boundary term from the system the condition that the flux of Φ through the boundaries is zero is imposed.

The temporal discretization of the advection equation (10) is discretized using forward Euler. Let $\Phi^n \approx (t_n)$ and denote the constant time step $\Delta t = t_{n+1} - t_n$. An intermediate Φ_*^{n+1} is then calculated. Find $\Phi_*^{n+1} \in V_h$ such that

$$\int_\Omega v \frac{\Phi_*^{n+1} - \Phi^n}{\Delta t} \, \mathrm{d}x - \int_\Omega \nabla v \cdot (\Phi^n\vec{u}^n) \, \mathrm{d}x = 0 \quad \forall v \in V_h. \tag{12}$$

If we do not perform any reinitialization, then the discretization (12) is unstable. With the reinitialization step applied, we did not observe any instabilities in our numerical experiments. If necessary, (12) can be stabilized by adding streamline diffusion.

To calculate the reinitialization step, the normal of the interface, $\hat{n}_*^{n+1}$, has to be approximated. This is also done using finite elements. $\hat{n}_*^{n+1} \in W_h$ is specified such that

$$\int_\Omega \vec{v} \cdot \frac{\nabla\Phi_*^{n+1}}{|\nabla\Phi_*^{n+1}|} \, \mathrm{d}x = \int_\Omega \vec{v} \cdot \hat{n}_*^{n+1} \, \mathrm{d}x \quad \forall \vec{v} \in W_h. \tag{13}$$

Note that the same normal vectors are used throughout one reinitialization step. Since the reinitialization equation is solved to steady state, the order of accuracy of the temporal discretization is not important. However, a first order explicit discretization gives severe restrictions on the time step for larger ε. A first order implicit discretization gives a nonlinear system to solve. We use a second order semi-implicit discretization in time with good stability properties. We start by letting $k = 0$ and $\Phi_c^0 = \Phi_*^{n+1}$. Then for $k = 0, \ldots, m$ we determine $\Phi_c^{k+1} \in V_h$ such that

$$\int_\Omega v \frac{\Phi_c^{k+1} - \Phi_c^k}{\Delta\tau} \, \mathrm{d}x - \int_\Omega \left(\frac{\Phi_c^k + \Phi_c^{k+1}}{2} - \Phi_c^{k+1}\Phi_c^k\right)\nabla v \cdot \hat{n}_*^{n+1} - \varepsilon\nabla\left(\frac{\Phi_c^k + \Phi_c^{k+1}}{2}\right) \cdot \hat{n}_*^{n+1}(\nabla v \cdot \hat{n}_*^{n+1}) \, \mathrm{d}x = 0 \quad \forall v \in V_h. \tag{14}$$

The iteration stops when

$$\frac{\|\Phi_c^{m+1} - \Phi_c^m\|}{\Delta\tau} < \delta \tag{15}$$

for some small δ. Following the discussion in Section 2.1 we expect only a few time steps to be needed to fulfill the condition (15). Finally, we set $\Phi^{n+1} = \Phi_c^{m+1}$.

2.4. Comments and restrictions on the parameter ε

There are several reasons to choose the parameter ε as small as possible. First of all, it is still unclear how the reinitialization step moves the 0.5 contour of Φ. We have seen in Section 2.1 that the addition of the stabilizing term does not alter the position of the 0.5 contour in the one dimensional case. This holds even for large ε. However, for two and three dimensions, it is still unclear how the 0.5 contour of Φ will move during the reinitialization. It will depend on the size of ε and in the limit $\varepsilon \to 0$, the reinitialization will not move the 0.5 contour at all. We intend to investigate the motion of the 0.5 contour of Φ during reinitialization in another paper.

As we shall see in Section 4, a smaller ε also gives better conservation of the area bounded by the 0.5 contour of Φ. As we couple the method with the incompressible Navier–Stokes equations, we will also reduce the smearing of density, viscosity and surface tension by choosing ε smaller.

There are however numerical restrictions on how small we can choose ε. The calculation of normal and curvature will be more accurate if the transition in Φ is smooth. A too small ε compared to the gridsize h will also create over or undershoots in the steady state solution of (14). The corresponding one dimensional problem is to find ϕ^n such that

$$\int_a^b \phi^n(1-\phi^n)v_x \,\mathrm{d}x = \varepsilon \int_a^b \phi_x^n v_x \,\mathrm{d}x \quad \forall v \in V_h.$$

Choose v such that $v_x = 1$ if $x \in [x_i, x_{i+1}]$ and zero otherwise. Denoting $h_i = x_{i+1} - x_i$ and $\phi_i = \phi^n(x_i)$ we get

$$h_i\left(\frac{\phi_i + \phi_{i+1}}{2} - \frac{\phi_i^2 + \phi_{i+1}^2 + \phi_i\phi_{i+1}}{3}\right) = \varepsilon(\phi_{i+1} - \phi_i).$$

If $\phi_i \approx \phi_{i+1} \approx 0$ we get by linearizing around zero that

$$\phi_{i+1} \approx \frac{h_i/2 + \varepsilon}{\varepsilon - h_i/2}\phi_i.$$

This implies that if $\varepsilon < h_i/2$, then ϕ_i and ϕ_{i+1} will be of opposite sign, i.e. ϕ^n will be oscillating. By assuming $\phi_i \approx \phi_{i+1} \approx 1$ we obtain the same restriction on ε. Note that on a non-uniform grid restrictions on ε will be different on different parts of the grid. A small ε is typically desired close to $\phi = 0.5$ to get a sharp transition. Far away from the interface we can allow a bigger ε. This can be achieved by having an adaptive grid refined close to the interface. In this way we can allow a smaller ε close to the interface without introducing oscillations. The adaptive procedure will be discussed further in Section 5.

We will let ε be proportional to the grid size. This means that we are not solving the same set of equations on different grids. One must be aware of that as the grid is refined, the profile of Φ will not become better resolved, and that we can therefore cannot expect pointwise convergence of Φ. At least for simple model problems we do expect convergence in L^2 norm, and that the 0.5 contour of Φ should converge. However, for general cases it is unclear what kind of convergence to expect. In our numerical experiments, we do obtain good convergence of the 0.5 contour of Φ. These results are presented in Section 6. If problems with convergence occur, use $\varepsilon = C(\Delta x)^{1-d}$, $1 > d > 0$ instead of $\varepsilon = C\Delta x$. In [7], $d = 0.1$ was needed to obtain convergence in one of the numerical tests.

3. The incompressible Navier–Stokes equations

The incompressible Navier–Stokes equations for two phase flow with surface tension are given by

$$(\rho\vec{u})_t + \nabla\cdot(\rho\vec{u}\vec{u}) = -\nabla p + \frac{1}{Re}\nabla\cdot(\mu(\nabla\vec{u} + (\nabla\vec{u})^{\mathrm{T}})) + \frac{\rho}{Fr^2}\vec{e}_g + \frac{1}{We}\vec{F}_{sv}, \tag{16}$$

$$\nabla\cdot\vec{u} = 0, \tag{17}$$

$$\Phi_t + \nabla\cdot(\Phi\vec{u}) = 0, \tag{18}$$

where $Re = \frac{\rho_{\text{ref}} u_{\text{ref}} l_{\text{ref}}}{\mu_{\text{ref}}}$, $Fr = \frac{u_{\text{ref}}}{\sqrt{l_{\text{ref}} g}}$ and $We = \frac{\rho_{\text{ref}} u_{\text{ref}}^2 l_{\text{ref}}}{\sigma}$, ρ_{ref}, μ_{ref}, l_{ref}, u_{ref} are constant reference density, viscosity, length and velocity, and σ is the surface tension. We will use a diffuse interface approach to model the force due to surface tension, introduced in [11]:

$$\vec{F}_{sv} = \kappa \nabla \Phi = -\left(\nabla \cdot \frac{\nabla \Phi}{|\nabla \Phi|} \right) \nabla \Phi.$$

The density and viscosity are constant within each fluid. To smooth the discontinuity across the interface, we let

$$\begin{aligned} \rho &= \rho_1 + (\rho_2 - \rho_1)\Phi, \\ \mu &= \mu_1 + (\mu_2 - \mu_1)\Phi. \end{aligned} \tag{19}$$

Here, ρ_1, ρ_2 and μ_1, μ_2 are the dimensionless densities and viscosities of the two fluids, respectively. We will assume that all boundaries are walls and impose $\vec{u} = 0$ there.

Approximations of the curvature and the gradient of Φ have to be calculated. Since Φ^{n+1} is piecewise linear, the gradient of Φ^{n+1} is discontinues and piecewise constant. In order to be able to calculate the curvature, we define $(\nabla \Phi)^{n+1}$ as the projection of the gradient of Φ^{n+1} onto a continues, piecewise linear space. In other words, we define $(\nabla \Phi)^{n+1} \in W_h$ such that

$$\int_\Omega (\nabla \Phi)^{n+1} \cdot \vec{v} \, \mathrm{d}x = \int_\Omega \nabla (\Phi^{n+1}) \cdot \vec{v} \, \mathrm{d}x \quad \forall \vec{v} \in W_h. \tag{20}$$

Note that no boundary conditions are necessary to solve (20). The mean curvature of the level sets of any Φ is given by

$$\kappa = -\nabla \cdot \frac{\nabla \Phi}{|\nabla \Phi|}.$$

Assuming a contact angle of 90° i.e. $\hat{n} \cdot \hat{v} = 0$ on $\partial\Omega$ this can be discretized by finding $\kappa^{n+1} \in V_h$ such that

$$\int_\Omega v \kappa^{n+1} \, \mathrm{d}x = \int_\Omega \nabla v \cdot \frac{(\nabla \Phi)^{n+1}}{|(\nabla \Phi)^{n+1}|} \, \mathrm{d}x \quad \forall \vec{v} \in V_h. \tag{21}$$

This might however produce spurious oscillations in space in κ^{n+1} with a frequency of about $1/h$. These high frequent parts can be damped if we instead calculate a regularized curvature $\tilde{\kappa}$ by approximating

$$\tilde{\kappa} - \varepsilon_2 \Delta \tilde{\kappa} = \kappa = -\nabla \cdot \frac{(\nabla \Phi)}{|(\nabla \Phi)|}.$$

Fourier analysis gives

$$\hat{\tilde{\kappa}}(\vec{\omega}) = \frac{1}{1 + \varepsilon_2 (\|\vec{\omega}\|^2)} \hat{\kappa}(\vec{\omega}),$$

where $\hat{\kappa}(\vec{\omega})$ and $\hat{\tilde{\kappa}}(\vec{\omega})$ are the Fourier coefficients of κ and $\tilde{\kappa}$ and $\vec{\omega}$ is a vector of the frequencies in the x, y and z directions. By letting $\varepsilon_2 = h\hat{\tilde{\kappa}} \approx \hat{\kappa}$ if $\|\vec{\omega}\| = \mathcal{O}(1)$ and $\hat{\tilde{\kappa}} \approx h\hat{\kappa}$ if $\|\vec{\omega}\| = \mathcal{O}(1/h)$. An approximation of the regularized curvature is: Find $\tilde{\kappa}^{n+1} \in V_h$ such that

$$\int_\Omega v \tilde{\kappa}^{n+1} \, \mathrm{d}x = \int_\Omega \nabla v \cdot \frac{(\nabla \Phi)^{n+1}}{|(\nabla \Phi)^{n+1}|} \, \mathrm{d}x - \varepsilon_2 \int_\Omega \nabla v \cdot \nabla \tilde{\kappa}^{n+1} \, \mathrm{d}x \quad \forall \vec{v} \in V_h. \tag{22}$$

Here we use the boundary condition $\nabla \tilde{\kappa}^{n+1} \cdot \hat{v} = 0$ on $\partial\Omega$.

There are a wide range of different methods available for the discretization of the Navier–Stokes equations. We use a projection method [12,13] similar to the unconditionally stable method proposed for one fluid incompressible flow by Guermond and Quartapelle [14]. We choose this method since it is well suited for low Reynolds number calculations (no severe time step restrictions). It also allows us to solve for each variable separately, and thereby reducing the amount of memory needed in the calculations. For more general problems, with for instance more complicated boundary conditions, a fully coupled solution strategy is more

suitable. For a thorough discussion on the advantages and disadvantages of projection methods, we refer to the recent review paper by Guermond et al. [15].

To begin with consider discretization in time only. The first step is to calculate an intermediate velocity $\vec{u}_*^{n+1}$ that is not divergence free, but satisfies $\vec{u}_*^{n+1} = 0$ on the boundaries.

$$\frac{1}{\Delta t}(\rho^{n+1}\vec{u}_*^{n+1} - \rho^n\vec{u}^n) + \nabla\cdot(\rho^{n+1}\vec{u}^n\vec{u}_*^{n+1}) = -\nabla p^n + \frac{1}{Re}\nabla\cdot(\mu(\nabla\vec{u}_*^{n+1} + (\nabla\vec{u}^n)^{\mathrm{T}})) + \frac{\rho^{n+1}}{Fr^2}\vec{e}_g + \frac{1}{We}\vec{F}_{sv}^{n+1}. \quad (23)$$

We then define $\vec{u}^{n+1}$ similarly, but with the pressure term implicitly:

$$\frac{1}{\Delta t}(\rho^{n+1}\vec{u}^{n+1} - \rho^n\vec{u}^n) + \nabla\cdot(\rho^{n+1}\vec{u}^n\vec{u}_*^{n+1}) = -\nabla p^{n+1} + \frac{1}{Re}\nabla\cdot(\mu(\nabla\vec{u}_*^{n+1} + (\nabla\vec{u}^n)^{\mathrm{T}})) + \frac{\rho^{n+1}}{Fr^2}\vec{e}_g + \frac{1}{We}\vec{F}_{sv}^{n+1}. \quad (24)$$

We subtract (23) from (24), rearrange the terms and obtain

$$\vec{u}^{n+1} = \vec{u}_*^{n+1} + \frac{\Delta t}{\rho^{n+1}}\nabla(-p^{n+1} + p^n). \quad (25)$$

We then use $\nabla\cdot\vec{u}^{n+1} = 0$ and get a Poisson equation for the pressure:

$$\frac{1}{\Delta t}\nabla\cdot\vec{u}_*^{n+1} = \nabla\cdot\left(\frac{\nabla(p^{n+1} - p^n)}{\rho^{n+1}}\right). \quad (26)$$

The corrected velocity can now be calculated from (25). Note that $\vec{u}^{n+1}\cdot\hat{v} = \vec{u}_*^{n+1}\cdot\hat{v} = 0$ on the boundaries. Thus it follows from (25) that we have to enforce the unphysical boundary condition $\nabla(p^{n+1-p^n})\cdot\hat{v} = 0$ in this projection method. This can reduce the accuracy of the method as discussed in [15]. For our problems we did however obtain sufficiently accurate results.

So far, we have only discussed the temporal discretization of the Navier–Stokes equations. In space, we use linear finite element approximations. Corresponding to (23), we first determine $\vec{u}_*^{n+1} \in W_h$ such that

$$\frac{1}{\Delta t}\int_\Omega(\rho^{n+1}\vec{u}_*^{n+1} - \rho^n\vec{u}^n)\cdot\vec{v}\,\mathrm{d}x - \int_\Omega(\vec{u}^n\cdot\nabla\vec{v})\cdot\rho\vec{u}_*^{n+1}\,\mathrm{d}x\int_\Omega(\nabla\cdot\vec{v})p^n\,\mathrm{d}x - \frac{1}{Re}\int_\Omega\mu^{n+1}\sum_i\nabla v_i\cdot(\nabla u_{*i}^{n+1} + \vec{u}_{xi}^n)\,\mathrm{d}x$$
$$+ \int_\Omega\vec{v}\cdot\left(\frac{\rho^{n+1}}{Fr^2}\vec{e}_g + \frac{1}{We}\vec{F}_{sv}^{n+1}\right)\mathrm{d}x \quad \forall\vec{v}\in W_h, \quad (27)$$

with

$$\vec{F}_{sv}^{n+1} = \tilde{\kappa}^{n+1}(\nabla\Phi)^{n+1}.$$

Then, corresponding to (26), find $p^{n+1} \in V_h$ such that

$$-\frac{1}{\Delta t}\int_\Omega q\nabla\cdot\vec{u}_*^{n+1}\,\mathrm{d}x = \int_\Omega\frac{\nabla q\cdot\nabla(p^{n+1} - p^n)}{\rho^{n+1}}\,\mathrm{d}x \quad \forall q\in V_h. \quad (28)$$

Note that all functions in W_h vanish on $\partial\Omega$, while there is no such restriction on V_h. As discussed earlier, we must enforce $\nabla(p^{n+1} - p^n)\cdot\hat{v} = 0$. Finally, as in (25), $\vec{u}^{n+1}$ can be solved for. Find $\vec{u}^{n+1} \in W_h$ such that

$$\int_\Omega\vec{v}\cdot\frac{\vec{u}^{n+1} - \vec{u}_*^{n+1}}{\Delta t}\,\mathrm{d}x - \int_\Omega\vec{v}\cdot\frac{\nabla(p^{n+1} - p^n)}{\rho^{n+1}}\,\mathrm{d}x \quad \forall\vec{v}\in W_h. \quad (29)$$

We summarize the algorithm for solving both the Navier–Stokes as well as the motion of the interface. In each time step the following steps are executed.

- Calculate Φ_*^{n+1} using (12).
- Calculate $\hat{n}_*$ from Φ_*^{n+1} using (13).
- Using Φ_*^{n+1} as initial data, solve (14) to steady state. This gives Φ^{n+1}.
- Calculate $(\nabla\Phi)^{n+1}$ from Φ^{n+1} using (20).
- Calculate κ^{n+1} using (22).

- Calculate $\vec{u}^{n+1}$ from (27).
- Calculate p^{n+1} from (28).
- Finally, calculate $\vec{u}^{n+1}$ using (29).

The implementations were done using the finite element tool Femlego [16,17].

4. Conservation

We can consider conservation of the method from two perspectives. These are the conservation of mass and the conservation of the area (volume for problems in three dimension) bounded by the 0.5 contour of Φ.

Exact mass conservation of the discrete solution for both two and three dimensional problems is easily verified by considering one time step of the method. The finite element discretizations given by (12) and (14) holds for any $v \in V_h$. In particular, they hold for $v \equiv 1$. The gradient of $v \equiv 1$ is zero. Eq. (12) therefore gives

$$\int_\Omega \frac{\Phi_*^{n+1} - \Phi^n}{\Delta t}\,\mathrm{d}x = 0,$$

or

$$\int_\Omega \Phi_*^{n+1}\,\mathrm{d}x = \int_\Omega \Phi^n\,\mathrm{d}x.$$

Equivalently, by letting $v \equiv 1$ in (14) we obtain

$$\int_\Omega \Phi_c^{k+1}\,\mathrm{d}x = \int_\Omega \Phi_c^{k}\,\mathrm{d}x.$$

Putting this together it is clear that

$$\int_\Omega \Phi^{n+1}\,\mathrm{d}x = \int_\Omega \Phi_c^{m+1}\,\mathrm{d}x = \int_\Omega \Phi_c^{m}\,\mathrm{d}x = \cdots = \int_\Omega \Phi_c^{0}\,\mathrm{d}x = \int_\Omega \Phi_*^{n+1}\,\mathrm{d}x = \int_\Omega \Phi^{n}\,\mathrm{d}x,$$

or simply

$$\int_\Omega \Phi^{n+1}\,\mathrm{d}x = \int_\Omega \Phi^{n}\,\mathrm{d}x.$$

Since the density is approximated by $\rho^n = \rho_1 + (\rho_2 - \rho_1)\phi^n$, it follows that $\int_\Omega \rho^n\,\mathrm{d}x = \int_\Omega \rho^{n+1}\,\mathrm{d}x$. Mass will thus be conserved exactly by the method. In a diffuse interface approach we can interpret Φ as the concentration of fluid one. The total mass of fluid one, m_1, will then also be conserved exactly, since $m_1^{n+1} = \int_\Omega \rho_1 \Phi^{n+1}\,\mathrm{d}x = \int_\Omega \rho_1 \Phi^n\,\mathrm{d}x = m_1^n$. Since the total mass of both fluids is conserved, the mass of the second fluid will also remain constant.

Next we study the conservation of the area bounded the 0.5 contour of Φ^n, which we denote by $A_{\Phi^n=0.5}$. Here we will only study the conservation of area for a two dimensional problem, although our derivation easily can be extended to volume conservation of three dimensional problems.

Assume that the reinitialization step is solved to steady state in each time step. Then Φ^n is a steady state solution of (14) with initial condition Φ_*^n:

$$\int_\Omega (\Phi^n - (\Phi^n)^2)\nabla v \cdot \hat{n}_*^n - \varepsilon \nabla v \cdot ((\nabla \Phi^n \cdot \hat{n}_*^n)\hat{n}_*^n)\,\mathrm{d}x = 0 \quad \forall v \in V_h. \tag{30}$$

Here, $\hat{n}_*^n \approx \frac{\nabla \Phi_*^n}{|\nabla \Phi_*^n|}$. Thus Φ^n is an approximation to a function Φ solving

$$\Phi - \Phi^2 == \varepsilon(\nabla \Phi \cdot \hat{n}), \tag{31}$$

where $\hat{n} = \hat{n}_*^n$ is independent of Φ. With Φ and Φ^n we associate subspaces of Ω according to the following definition.

Definition 1. Define Ω_1, Ω_2 and Γ as the subspaces of Ω where $\Phi < 0.5$, $\Phi > 0.5$ and $\Phi = 0.5$, respectively. Let Ω_1^n, Ω_2^n and Γ^n be the subspaces of Ω where $\Phi^n < 0.5$, $\Phi^n > 0.5$ and $\Phi^n = 0.5$. Define $\widetilde{\Phi}^n$ by

$$\widetilde{\Phi}^n(x) = \begin{cases} 1 & \text{if } x \in \Omega_2^n, \\ 0 & \text{otherwise.} \end{cases}$$

and correspondingly for $\widetilde{\Phi}$:

$$\widetilde{\Phi}(x) = \begin{cases} 1 & \text{if } x \in \Omega_2, \\ 0 & \text{otherwise.} \end{cases}$$

It follows directly from the definition of $\widetilde{\Phi}^n$ that

$$A_{\Phi^n=0.5} = \int_\Omega \widetilde{\Phi}^n(x)\,\mathrm{d}x.$$

To investigate the conservation of $A_{\Phi^n=0.5}$ we study the difference between $A_{\Phi^n=0.5}$ and the exactly conserved quantity $\int_\Omega \Phi^n\,\mathrm{d}x$:

$$\begin{aligned} \left|\int_\Omega \Phi^n\,\mathrm{d}x - A_{\Phi^n=0.5}\right| &= \left|\int_\Omega \Phi^n\,\mathrm{d}x - \int_\Omega \widetilde{\Phi}^n(x)\,\mathrm{d}x\right| \\ &= \left|\int_\Omega \Phi^n\,\mathrm{d}x - \int_\Omega \widetilde{\Phi}^n(x)\,\mathrm{d}x - \left(\int_\Omega \Phi\,\mathrm{d}x - \int_\Omega \widetilde{\Phi}\,\mathrm{d}x\right) + \left(\int_\Omega \Phi\,\mathrm{d}x - \int_\Omega \widetilde{\Phi}\,\mathrm{d}x\right)\right| \\ &\leqslant \|\Phi - \Phi^n\|_{L^1(\Omega)} + \|\widetilde{\Phi} - \widetilde{\Phi}^n\|_{L^1(\Omega)} + \left|\int_\Omega \left(\Phi - \widetilde{\Phi}\right)\mathrm{d}x\right|. \end{aligned} \tag{32}$$

As we estimate the size of the first two terms on the right-hand side we have to keep in mind that $\varepsilon \sim \Delta x$. This implies that the resolution of the transition layer will not increase as the grid is refined. The thickness of the transition layer will however be proportional to ε. We can thus estimate the size of the first two terms to be $C\Delta x$. The size of the constant C will depend on the order of the method and the number of gridpoints in the transition layer. The last term is only related to an exact solution of (31) and will not depend on the numerical method used. Bounds on the last term can be found, given suitable restrictions on the vector field $\hat{n}$.

Definition 2. For every point x_0 on Γ, let $X(x_0, s)$ denote a path in Ω such that

$$X(x_0, 0) = x_0, \quad \frac{\partial X}{\partial s} = \hat{n}.$$

Furthermore, let Ω_0 be the subspace of Ω where no unique x_0 and s exist such that $x = X(x_0, s)$.

Examples of paths are given in Fig. 1. Typically, we cannot assume $\Omega_0 = \{0\}$. Often, as in the example in Fig. 1, we can however assume $\int_{\Omega_0} \mathrm{d}\Omega = 0$.

Lemma 1. *Assume a function Φ of which the 0.5 level set forms a closed smooth curve Γ. Let L_Γ be the length of Γ and assume that the magnitude of the curvature is bounded by κ_∞. Also assume that Φ fulfills*

$$\Phi - \Phi^2 = \varepsilon(\nabla\Phi \cdot \hat{n})$$

in $\Omega \setminus \Omega_0$ where $\int_{\Omega_0} \Phi\,\mathrm{d}\Omega = 0$.

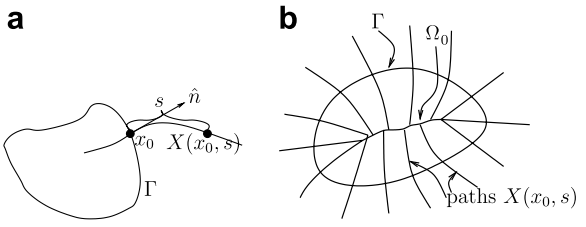

Fig. 1. Example of paths: (a) The path $X(x_0, s)$ for one $x_0 \in \Gamma$, (b) several paths corresponding to different x_0.

Then it holds that

$$\left|\int_\Omega (\Phi - \widetilde{\Phi})\,\mathrm{d}x\right| \leqslant L_\Gamma(\varepsilon\ln(2) + \kappa_\infty\varepsilon^2). \tag{33}$$

If the paths $X(x_0, s)$ are straight non intersecting lines perpendicular to Γ for all $|s| < b$, then

$$\left|\int_\Omega (\Phi - \widetilde{\Phi})\,\mathrm{d}x\right| \leqslant L_\Gamma(\kappa_\infty\varepsilon^2 + \varepsilon\ln(1 + \mathrm{e}^{-b/\varepsilon})). \tag{34}$$

Note that the second term of the bound given by (34) will decrease exponentially fast to zero as $\varepsilon \to 0$.

The proof is straight forward, and is found in Appendix A.

We can conclude that

$$A_{\Phi^n=0.5} = A_{\Phi^0=0.5} + 2\delta,$$
$$|\delta| \leqslant \|\Phi - \Phi^n\|_{L^1(\Omega)} + \|\widetilde{\Phi} - \widetilde{\Phi}^n\|_{L^1(\Omega)} + (\varepsilon\ln 2 + \kappa_\infty\varepsilon^2)$$

for general paths Ω, and for straight paths for $|s| < b$

$$|\delta| \leqslant \|\Phi - \Phi^n\|_{L^1(\Omega)} + \|\widetilde{\Phi} - \widetilde{\Phi}^n\|_{L^1(\Omega)} + L_\Gamma(\kappa_\infty\varepsilon^2 + \varepsilon\ln(1 + \mathrm{e}^{-b/\varepsilon})).$$

As will be seen in the section on numerical results, we experimentally obtained variations bounded by $L_\Gamma\kappa_\infty\varepsilon^2$. This suggests that the we can assume the paths to be straight and perpendicular to Γ sufficiently close to the interface. Paths perpendicular to Γ corresponds to that $\hat{n} = \frac{\nabla\Phi}{|\nabla\Phi|}$ on Γ.

Note that δ does not explicitly depend on t or Δt. This implies that the difference in area will not increase or decrease as t becomes larger unless the length or the maximum curvature of the interface is increasing or decreasing in time.

5. Adaptive mesh control

There are two reasons to refine the grid close to the interface. First of all, the position of the interface will be more accurately represented. Secondly, the smearing of the density, viscosity and surface tension across the interface is unphysical. It is therefore desirable to keep the thickness of the transition layer as small as possible. A restriction on how small ε can be was discussed in Section 2.4. If ε, and hence the thickness of the interface, is too small compared to the grid size there will be oscillations in Φ. In order to have a sharp transition, we need to have a refined grid close to the interface. We are therefore interested in an adaptive algorithm. We design our algorithm such that we can control the mesh with respect to the following parameters:

- δ: thickness of the interface (proportional to ε).
- m: number of elements across the interface.
- $\mathrm{Ind}(x)$: function indicating how to refine.

Here we will use the indicator function

$$\mathrm{Ind}(x) = \frac{1}{\varepsilon}\Phi(x)(1 - \Phi(x)),$$

and note that over an interface this is approximately $\nabla\Phi\cdot\hat{n} = |\nabla\Phi|$. Clearly, the indicator function is exponentially small away from the interface and has its maximum when $\Phi = 0.5$. The grid is then defined such that $\int_{K_i}\mathrm{Ind}(x)\,\mathrm{d}x \approx Tol$ for each element K_i. *Tol* can be chosen such that there will be m grid points across an interface of thickness δ. First we have to define the thickness of the interface. Assume $\Phi(\vec{x}_1) = 0.05$ and $\Phi(\hat{x}_2) = 0.95$ and that $\hat{x}_1 = X(x_0, s_1)$ and $\hat{x}_1 = X(x_0, s_2)$. We define the thickness of the interface by $\delta = |\hat{x}_2 - \hat{x}_1|$. Using that the profile is

$$\Phi(X(x_0, s)) = \frac{1}{1 + \mathrm{e}^{s/\varepsilon}}, \tag{35}$$

and assuming that the paths $X(x_0, s)$ are straight for $s_1 \leqslant s \leqslant s_2$ one easily sees that $\delta = \varepsilon \ln(0.95^2/0.05^2) \approx 6\varepsilon$. The grid size is given by $a(x) = (\varepsilon \cdot Tol)/(\Phi(1-\Phi))$, where $a(x)$ is the area of the elements. If we assume the elements to be isosceles right triangles (triangle with a right angle between two equal sides), then the length of the hypotenuse is $\mathrm{hyp}(x) = \sqrt{4a(x)}$. Introducing a coordinate system along the interface m, the minimum number of elements across the interface, can be estimated. To do this we again use that the profile of the interface is given by (35), obtaining

$$\begin{aligned} m &= \int_{-3\varepsilon}^{3\varepsilon} 1/\mathrm{hyp}(x)\,\mathrm{d}x = \frac{1}{2\sqrt{\varepsilon \cdot Tol}} \int_{-3\varepsilon}^{3\varepsilon} \sqrt{\Phi(1-\Phi)}\,\mathrm{d}x \\ &= \frac{1}{2\sqrt{\varepsilon \cdot Tol}} \int_{-3\varepsilon}^{3\varepsilon} \sqrt{\frac{\mathrm{e}^{-x/\varepsilon}}{(1+\mathrm{e}^{-x/\varepsilon})^2}}\,\mathrm{d}x \frac{\sqrt{\varepsilon}}{2\sqrt{Tol}} \int_{e^{-3\varepsilon}}^{e^{3\varepsilon}} \frac{1}{\sqrt{t}(1+t)}\,\mathrm{d}t \\ &= \frac{\sqrt{\varepsilon}}{\sqrt{Tol}}(\arctan(\sqrt{e^3})) - \arctan(\sqrt{e^{-3}}) = \mathrm{const}\frac{\sqrt{\varepsilon}}{\sqrt{Tol}} \approx 1.1\frac{\sqrt{\varepsilon}}{\sqrt{Tol}}. \end{aligned} \tag{36}$$

Thus $Tol = \frac{1.3\varepsilon}{m^2}$ will yield approximately m grid points over the interface. Correspondingly for three dimensional meshes, one easily sees that $Tol \sim \frac{\varepsilon^2}{m^3}$. In order not to make the grid too coarse away from the interface, we make sure $h \leqslant h_{\max}$ for some appropriate $h_{\max}$. We now have expressions on how ε and Tol should be chosen in terms of m and δ:

$$\varepsilon = \frac{\delta}{\ln(0.95^2/0.05^2)} \approx \delta/6, \tag{37}$$

$$Tol = \frac{0.2\delta}{m^2}. \tag{38}$$

As was shown in Section 2.4, we will get oscillations in Φ when $\Phi \approx 0$ or $\Phi \approx 1$ if ε is too small compared to the grid size h. Because of this we used a larger ε in the coarser parts of the grid such that $\varepsilon \geqslant Ch$ was fulfilled everywhere.

Since Φ will depend on time, the mesh will have to be adjusted in time. How often re-meshing is needed depends on how quickly the interface moves.

6. Numerical results

The performance and convergence of the method was tested on different problems where the motion was driven by surface tension. The gravitational forces are supposed to be small in our future applications. The gravity was therefore set to zero in all the computations presented here.

6.1. Oscillating droplet for small Reynolds number in 2D and 3D

In the first test problem the parameters in the Navier–Stokes were set such that the main forces were expected to be surface tension and viscous forces. The dimensionless parameters were set to $Re = 1,\ We = 1,\ \rho_1 = 10,\ \rho_2 = 1,\ \mu_1 = 10$ and $\mu_2 = 1$. The initial shape of the interface was constructed by a $1 \cdot 0.3$ rectangle and two half circles of radius 0.15 in 2D. For calculations in 3D a cylinder of radius 0.15 and length 1 with two half spheres of radius 0.15 at the ends was used.

6.1.1. Estimation of order of accuracy with respect to grid size

First 2D computations were performed to check the convergence of the method. The computational domain was $2 \cdot 2$ and was discretized using three different uniform meshes of $80 \cdot 80$, $116 \cdot 116$ and $160 \cdot 160$ nodes. The position of the 0.5 contour on the different grids for some time steps is shown in Fig. 2. On each grid we used: $\varepsilon = \Delta x \cdot \sqrt{2}$. Δt was proportional to Δx. The order of accuracy was calculated with respect to the average distance between the 0.5 contours of the exact solution Φ and a numerical solution Φ^n, i.e.

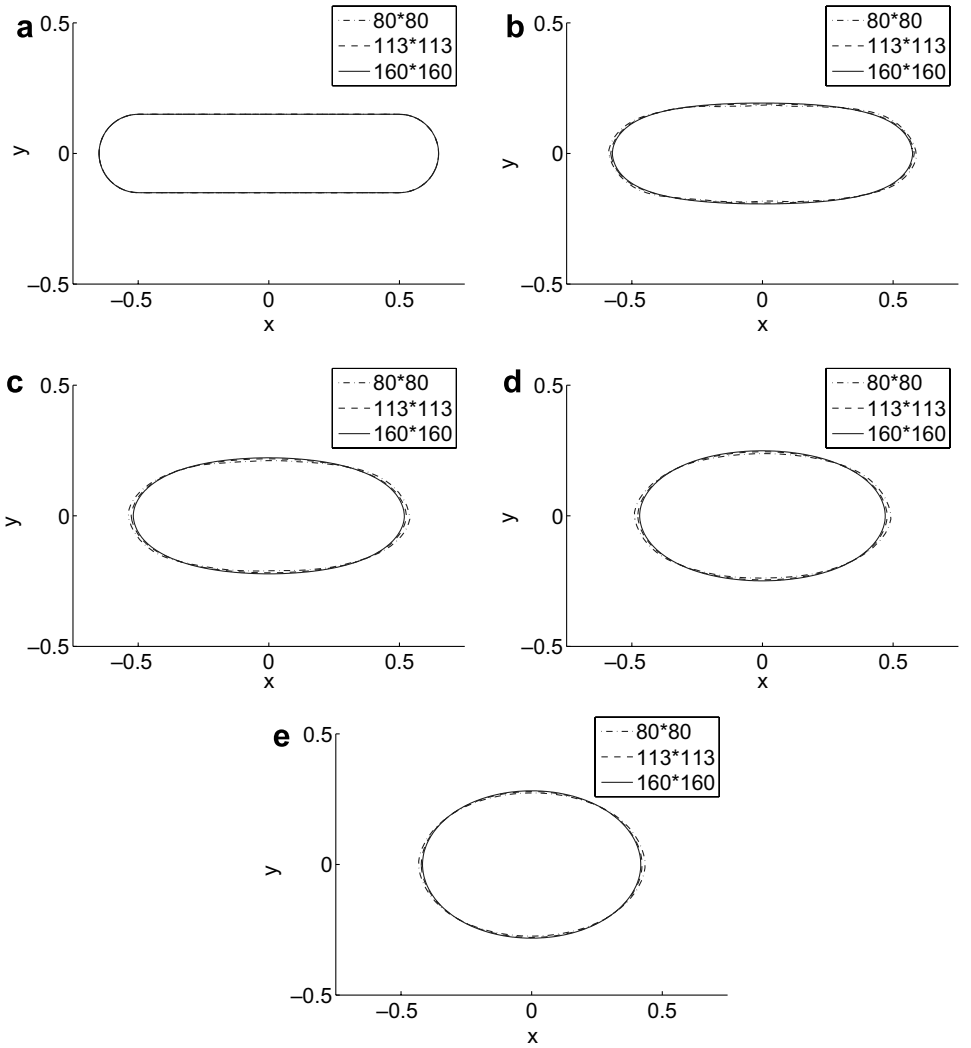

Fig. 2. Convergence study of oscillating droplet with $Re = 1$. Results using three different grids, each with $\varepsilon = \sqrt{2\Delta x}$. (a) $t = 0$, (b) $t = 3$, (c) $t = 5$, (d) $t = 7$, (e) $t = 10$.

$$E = \int_{\Omega} \frac{|\widetilde{\Phi}^n - \widetilde{\Phi}|}{L\Gamma} \, \mathrm{d}x \approx Ch^p, \tag{39}$$

where L_Γ is the length of the interface and $\widetilde{\Phi}^n$ and $\widetilde{\Phi}$ go sharply from zero to one at the 0.5 contour of Φ^n and Φ as in Definition 1. Since the exact solution of the problem is not known, (39) cannot be used directly to calculate the order of accuracy p. By comparing solutions obtained using three different grid sizes, $h_1,\ h_2 = kh_1,\ h_3 = k^2 h_1$, the order of accuracy can be estimated from

$$p \approx \frac{\ln\left(\int_\Omega |\widetilde{\Phi}_{h_2} - \widetilde{\Phi}_{h_3}| \, \mathrm{d}x - \int_\Omega |\widetilde{\Phi}_{h_1} - \widetilde{\Phi}_{h_2}| \, \mathrm{d}x\right)}{\ln k}.$$

The variation of the area bounded by the 0.5 contour of Φ and the estimated order of accuracy at different times are given in Figs. 3a and b. The area fluctuations are less than 1% on all grids. They are all also smaller than $\max_t(k_\infty L_\Gamma \varepsilon^2)$, indicating that the bound (34) is more adequate than (33). The order of accuracy is about two. This was the best we could hope for since we were using linear finite element discretizations. The convergence obtained here was much better than for the fluid flow tests in [7]. The reason of this improvement is the

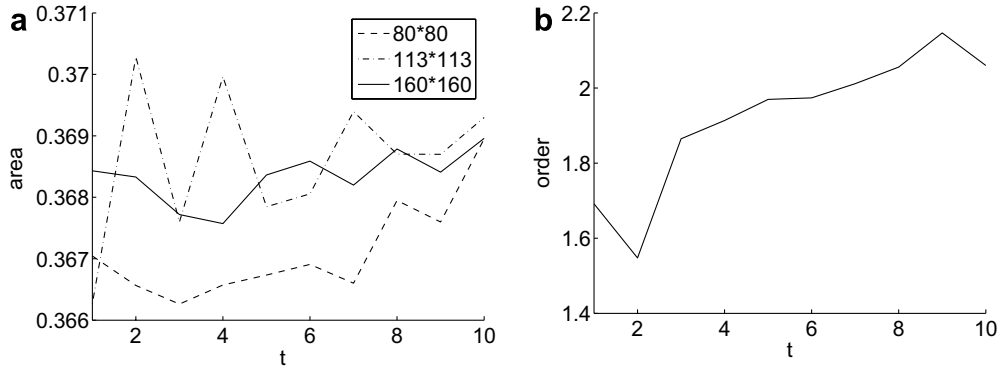

Fig. 3. Estimated order of accuracy and conservation of area bounded by the 0.5 contour of Φ. (a) Area conservation, (b) order of accuracy.

modification in the reinitialization step. In time, first order discretizations were used. Since our results show second order of accuracy, the error due to the temporal discretization appears to be negligible compared to the spatial errors. This justifies the use of only first order accurate discretizations in time.

6.1.2. Estimation of error with respect to ε

On the finest grid with $160 \cdot 160$ nodes, runs were also performed using different ε, see Fig. 4. This was done to investigate how the size of ε effects the solution. We are interested in the behavior of the error given by

$$E_\varepsilon = \int_\Omega \frac{|\widetilde{\Phi}_0 - \widetilde{\Phi}_\varepsilon|}{L_\Gamma}\,\mathrm{d}x \tag{40}$$

where the subscript of Φ denotes the size of ε. Several factors contribute to this error. First of all the smearing of density, viscosity and surface tension will result in an error. Secondly, the reinitialization will introduce some additional error. Because of the complexity it is hard to analyze the total effect of all these errors. We can how-

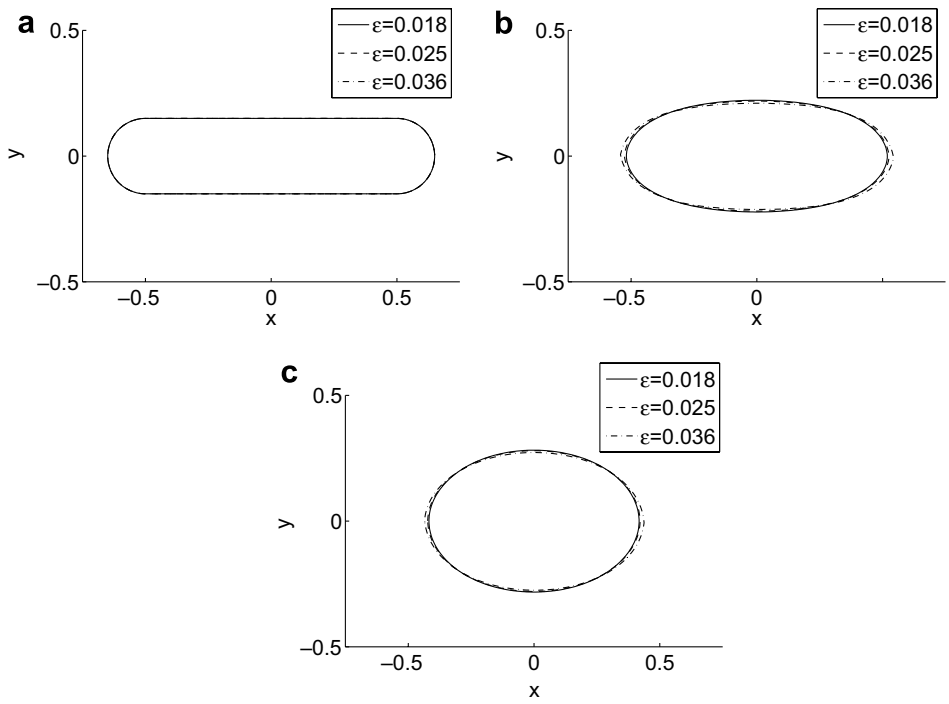

Fig. 4. 0.5 contour of Φ using different ε $(0.018, 0.025, 0.036)$ on $160 \cdot 160$ grid. (a) $t = 0$, (b) $t = 5$, (c) $t = 10$.

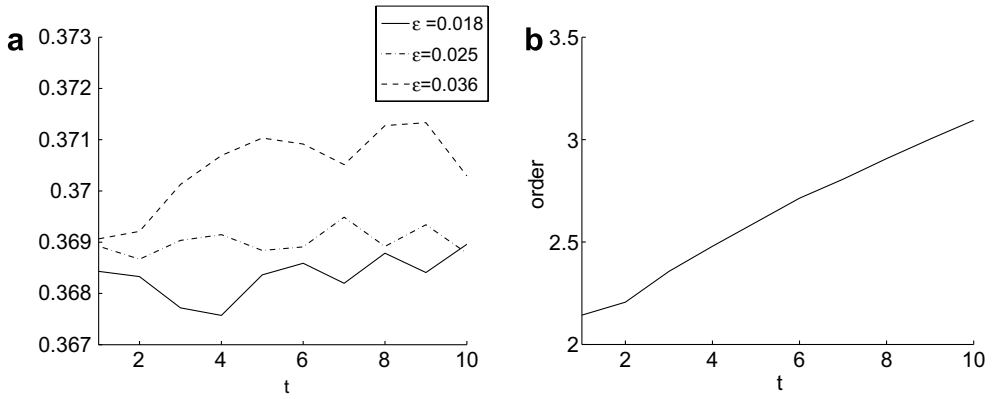

Fig. 5. Estimated order of accuracy with respect to ε and conservation of area bounded by the 0.5 contour of Φ. (a) Area conservation, (b) order of accuracy with respect to ε.

ever easily numerically estimate an order of accuracy p with respect to this error if we assume $E_\varepsilon \approx C\varepsilon^p$. Area conservation and estimated order of accuracy with respect to ε are given in Figs. 5a and b. Again, the variations in area suggest the bound (34) to be accurate. For this specific case it seems as if the error with respect to the average distance between the actual interface and the exact interface is $\mathcal{O}(\varepsilon^p)$ with $p \geqslant 2$.

6.1.3. Results using adaptive grids in two and three dimensions

Computations were also done using adaptive grids with ε set to 0.018. Since the grid was updated during the calculation the number of grid points varied, but was approximately 8000. An example of the grid is shown in Fig. 6. In Fig. 7 a comparison between results obtained using the adaptive grid and the uniform grids are given. Results on this adaptive grids compared well with the calculations performed using $160 \cdot 160 = 25,600$, i.e. using about three times as many nodes. The total computing time was thus reduced to 30%. The gain obtained by using adaptive grids can be expected to be larger for three dimensional computations.

Using the same parameters in the Navier–Stokes equations, three dimensional calculations were also performed. The initial shape of the interface was produced by a cylinder and two half spheres. Adaptive grids were used with $\varepsilon = 0.018$ and about 50,000 nodes. The volume bounded by the 0.5 isosurface varied with about one percent.

6.2. Small amplitude oscillations

To validate our method we have also simulated small amplitude oscillations. A droplet with density ρ_1 and viscosity μ_1 is surrounded by an infinite media with density ρ_2 and viscosity μ_2. In cylindrical coordinates (r, θ) the interface line $r = r(\theta)$ in the initial configuration is given by

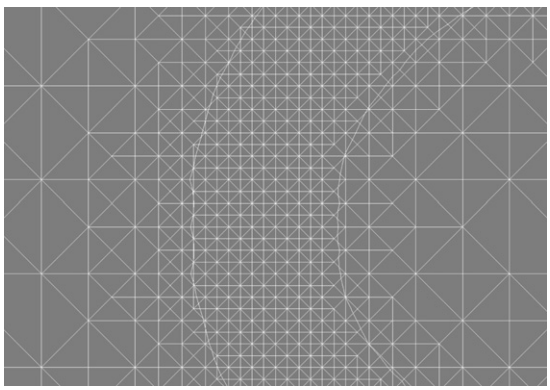

Fig. 6. Example of adaptive grid.

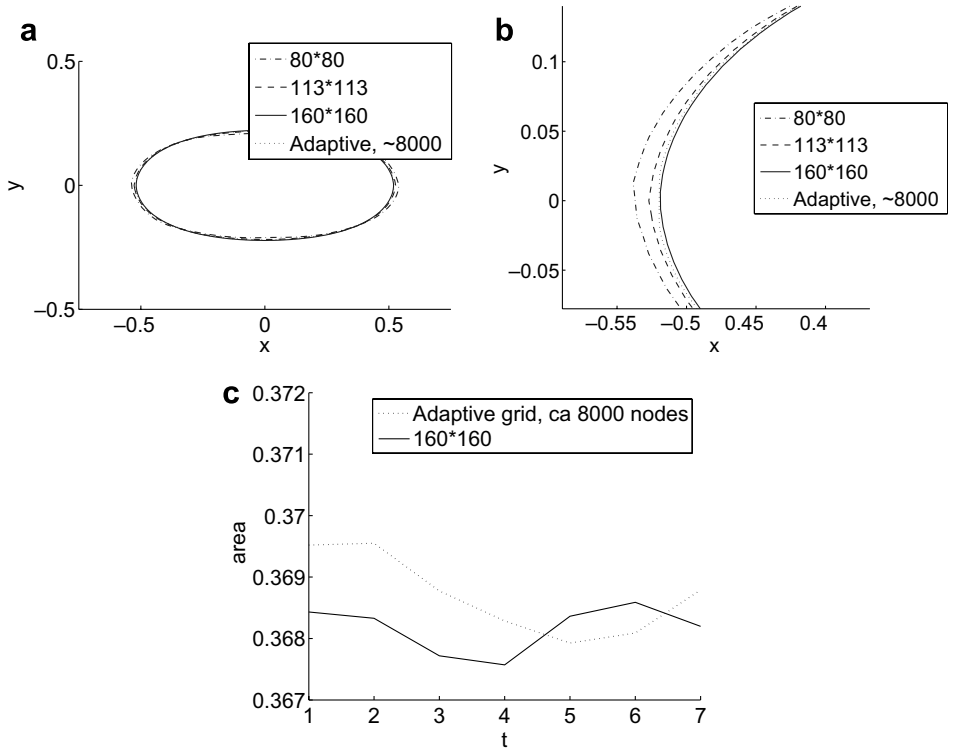

Fig. 7. Comparison of results using adaptive and uniform grids: (a) $t = 5$, (b) $t = 5$, (c) area conservation.

$$r = R_0(1 - \xi/4 + \xi P_2(\cos\theta)),$$

where P_2 is the Legendre polynomial of order 2, and $\xi \ll 1$. There are theoretical results on this problem. In [18,19] some analytical expressions for infinitesimal amplitude oscillations of an incompressible, inviscid droplet are derived. In particular, the angular frequency, ω_l, is given by

$$\omega_l^2 = \frac{(l^3 - l)\sigma}{(\rho_1 + \rho_2)R_0^3},$$

where l is the order of the Legendre polynomial. We set $R_0 = 1$, $\xi = 0.01$, $\alpha = 0.5$, $\mu_1 = \mu_2 = 5\times 10^{-3}$, $Re = 1$, $We = 2$ and $\rho_1 = 1$, and performed tests to determine the oscillating period T. Computations were done for different density ratios by changing the density ρ_2. We used homogeneous Neumann boundary conditions for the velocity. In Table 1 the analytical oscillation period $T = 2\pi/\omega_2$ and the results from our computations for the oscillation period and the relative error are shown. We have considered a fixed mesh where a quarter of the domain, a square with side equal to 2 is covered by $64 \cdot 64$ elements (see Fig. 8).

Table 1
Oscillation period T and relative error for different density ratios ρ_2/ρ_1

ρ_2/ρ_1	Analytical	$\Delta t = 0.0025$		$\Delta t = 0.001$	
	T	T	Error %	T	Error %
1	5.130	5.385	4.97	5.385	4.97
0.1	3.805	3.976	4.51	3.975	4.48
0.01	3.646	3.737	2.50	3.736	2.48
0.001	3.629	3.685	1.54	3.684	1.52
0.0001	3.628	3.678	1.38	3.677	1.36

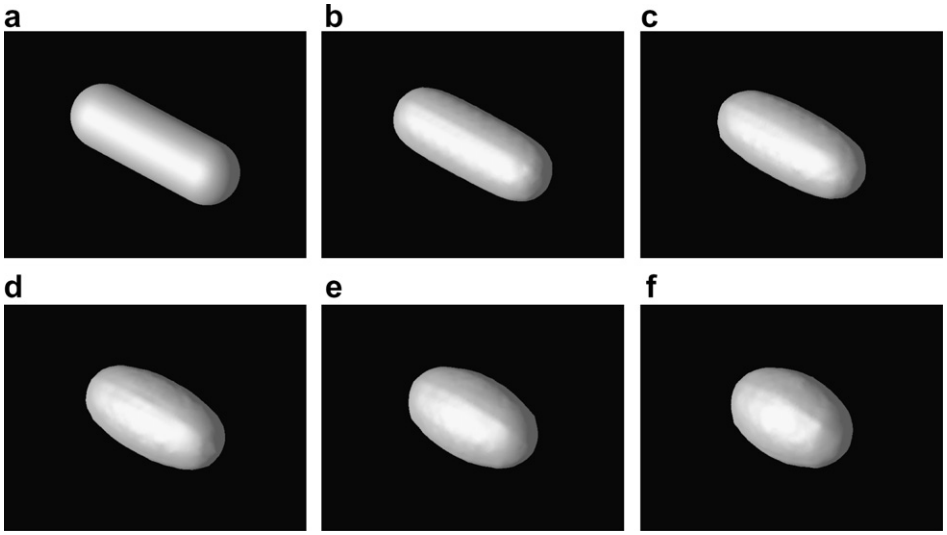

Fig. 8. Results from three dimensional calculations of viscous droplet ($Re = 1$): (a) $t = 0$, (b) $t = 2$, (c) $t = 4$, (d) $t = 6$, (e) $t = 8$, (f) $t = 10$.

Table 2
Oscillation period T and relative error for different mesh resolutions

Mesh	Analytical	$\Delta t = 0.0025$	
	T	T	Error %
64×64	3.646	3.737	2.50
128×128	3.646	3.707	1.68

We then keep the density ratio, $\rho_2/\rho_1 = 0.01$, and $\Delta t = 0.0025$ constant and in Table 2 we show how the oscillation period changes as a function of different mesh resolutions. In Fig. 9, we have plotted the position of the interface at $\theta = 0$ for the two different mesh resolutions. The period was determined by finding the first maximum of the curve.

In all the computations we have set $\epsilon = \sqrt{2}\Delta x$ and $\epsilon_2 = 0.001$.

In [20] computations for the same problem, using a front tracking scheme, are presented. For small values of ρ_2/ρ_1 our method performs better than their but for $\rho_2/\rho_1 \approx 1$ our method gives slightly poorer results. One should note that this problem is very well suited for a front tracking scheme, since the interphase undergoes only infinitesimal distortion. It is therefore not surprising that our method does not give as accurate results in all of the cases.

6.3. Oscillating liquid Cobalt droplet in air

Cemented carbides are extremely hard materials used for example in tools for steel cutting. The most common cemented carbide consist of Tungsten Carbide (WC) and Cobalt and is constructed in the following way. A powder of Tungsten Carbide and Cobalt is mixed. The mixture is heated such that the Cobalt becomes liquid. WC has an extremely high melting point and remains solid during the sintering. The liquid Cobalt glues the Tungsten Carbide grains together and densifies the material. One believes that the driving force during this process is surface tension. A brief description of the liquid phase sintering as well as properties of liquid Cobalt can be found in [21].

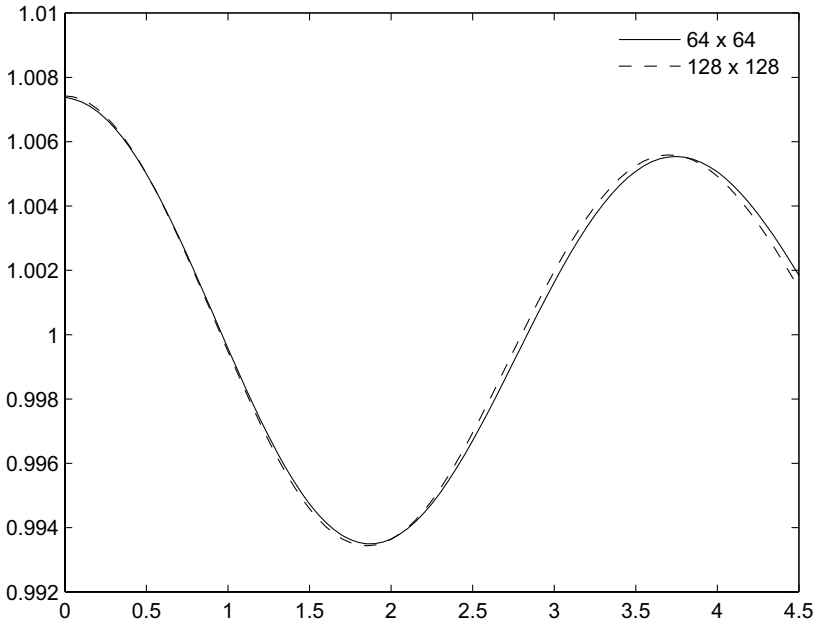

Fig. 9. Radius oscillation for the mesh resolutions 64×64 and 128×128.

In our third simulation we have set density, viscosity and surface tension to correspond to a droplet of liquid Cobalt in air. The length scale is proportional to the size of the powder grains. The dimensionless parameters become $Re = 1000,\ We = 1.2,\ \rho_1 = 0.0001,\ \rho_2 = 1,\ \mu_1 = 0.04$ and $\mu_2 = 1$. The shape of the initial droplet is a $0.6 \cdot 0.5$ rectangle with two half circles with radius 0.25 at the end. Calculations were performed on three different grids to check the convergence. The position of the interface on the different grids are shown in Fig. 10. The velocity field obtained on the finest grid are shown in Fig. 11.

The order of accuracy with respect to the average distance between the exact interface and the numerically obtained interface was calculated at $t = 0.1, 0.2, \ldots, 1$. The result is shown in Fig. 12a. The area of the 0.5 contour was also calculated at the same time points and is shown in Fig. 12b. We obtain good area conservation. From Fig. 10 it is clear that the shape of the droplet at $t = 0$ and $t = 0.8$ are very similar. Because of this we can expect the area bounded by the interface at $t = 0$ and $t = 0.8$ to be very similar. This is verified by our calculations as can be seen in Fig. 12b.

To make adaptive runs for this test case did not turn out to be useful. Because of the high Reynolds number the velocity field is varying rapidly in space. This means that we cannot have a very coarse grid away from the interface, since we need to resolve gradients in velocity properly. In an adaptive mesh control suitable for this problem, the criterion for refining should also depend on gradients of $\vec{u}$.

A three dimensional droplet of liquid Cobalt was also studied. In the three dimensional case velocities vary even more rapidly. Computations could therefore not be done using our simple adaptive approach. Instead we used a uniform mesh with $40 \cdot 40 \cdot 40 = 64,000$ nodes. Here $\varepsilon = 0.028$ was used. Because of the symmetry of the problem the size of the computational domain of $2 \cdot 2 \cdot 2$ could be reduced to $1 \cdot 1 \cdot 1$. As can be seen in Fig. 13 the motion of the interface is significantly different compared to the case with lower Reynolds number. The velocity field at a few different times is given in Fig. 14. At $t = 0.6$ steep gradients of the velocity across the interface can be observed. Eventually the grid is not fine enough to resolve the variations in the velocity field.

7. Conclusions

In this paper, we have continued to develop the conservative level set method introduced in [7]. An important improvement is the modification of the reinitialization step. The numerical results in this paper show

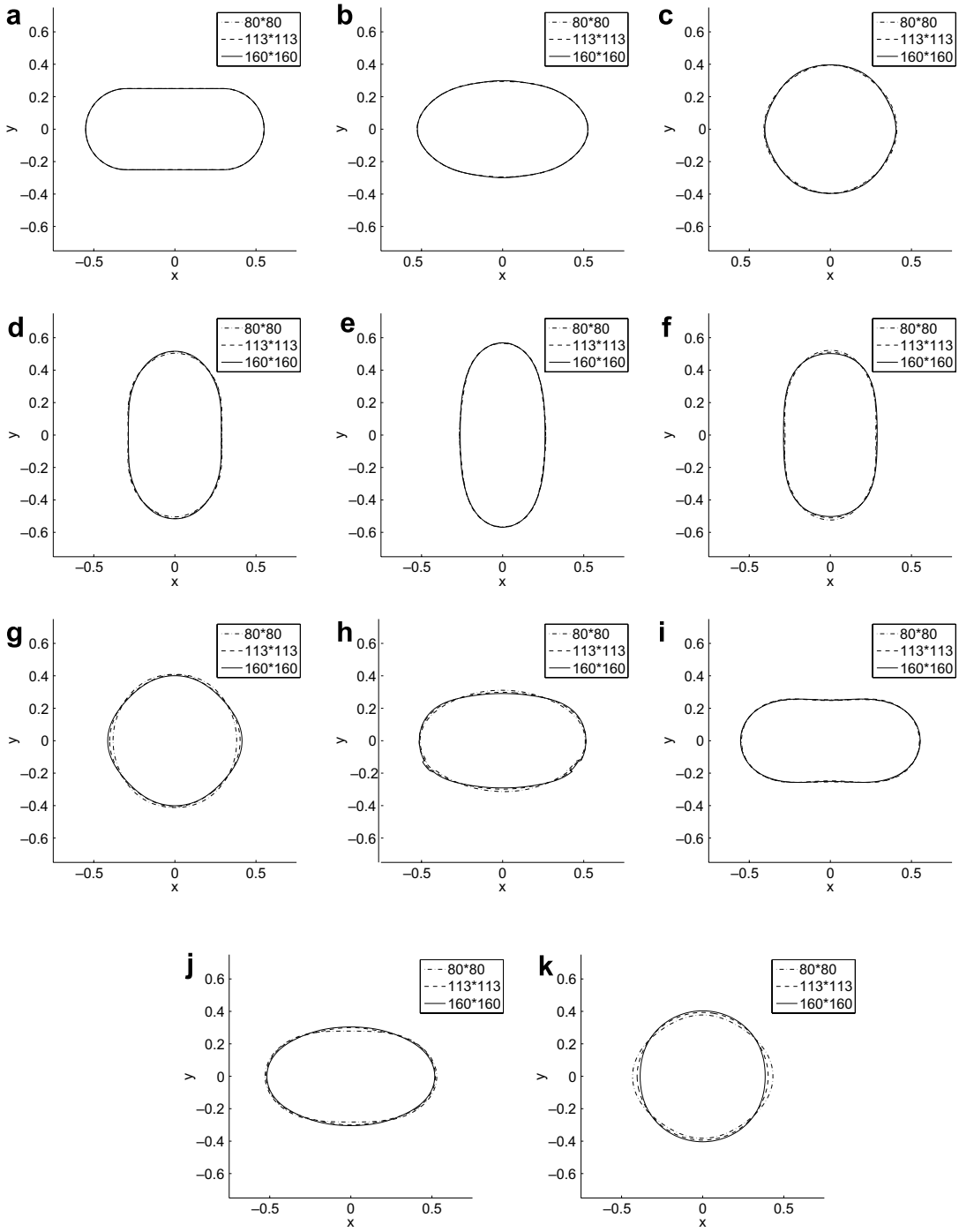

Fig. 10. Convergence study of numerical calculations of a two dimensional liquid Cobalt droplet in air. Results using three different grids, each with $\varepsilon = \sqrt{2}\Delta x$, are given. (a) $t = 0$, (b) $t = 0.1$, (c) $t = 0.2$, (d) $t = 0.3$, (e) $t = 0.4$, (f) $t = 0.5$, (g) $t = 0.6$, (h) $t = 0.7$, (i) $t = 0.8$, (j) $t = 0.9$, (k) $t = 1$.

much better convergence compared to the rising bubble in the previous paper. We are convinced that the reason for this is the modification of the reinitialization.

We have also given a theoretical analysis on how the area bounded by the appropriate level set can be expected to be conserved. The theory agrees well with results from our numerical computations.

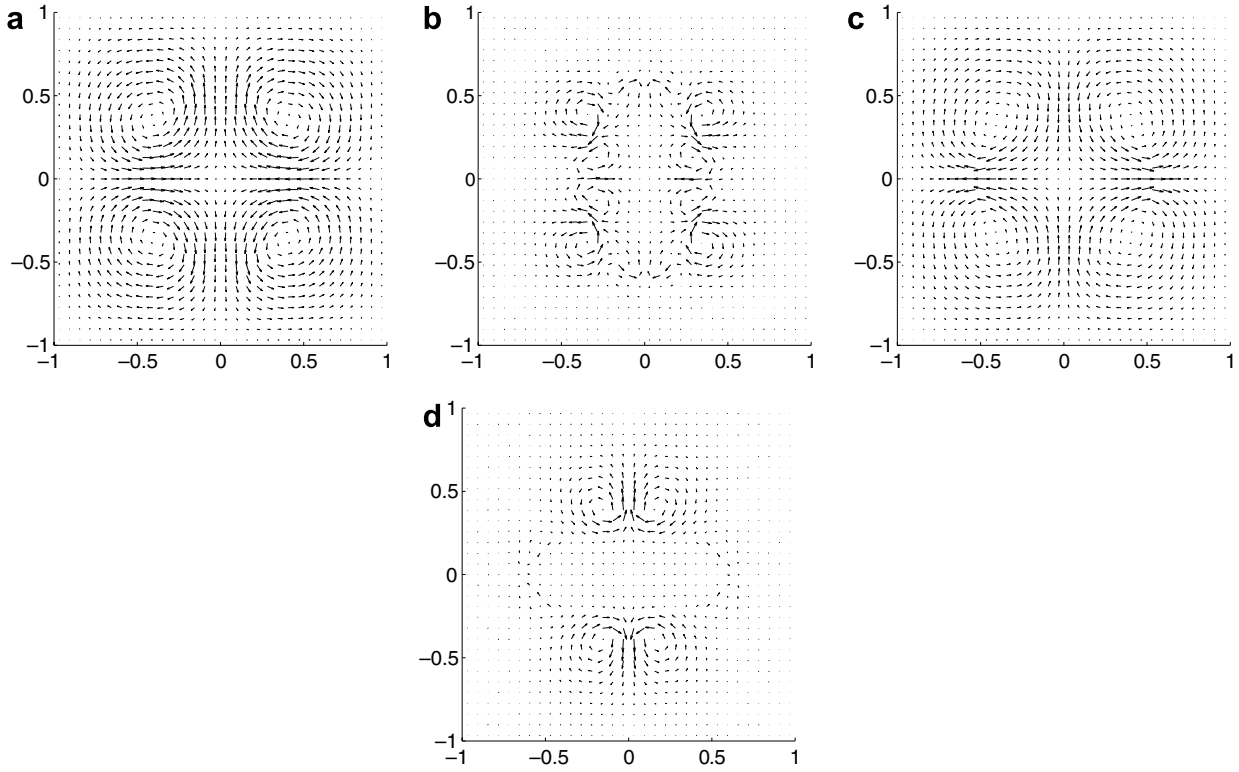

Fig. 11. Velocity field at different times for the two dimensional Cobalt droplet. (a) $t = 0.2$, (b) $t = 0.4$, (c) $t = 0.6$, (d) $t = 0.8$.

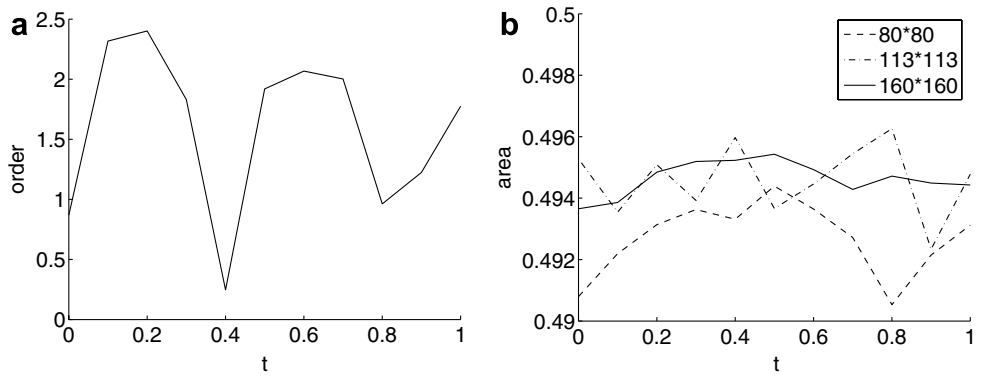

Fig. 12. Estimated order of accuracy and conservation of area bounded by the 0.5 contour for two dimensional liquid Cobalt droplet: (a) order of accuracy, (b) area conservation.

An adaptive finite element discretization of the method has been proposed and implemented in two and three dimensions. Several numerical experiments have been done to investigate the performance of the method. The results from the calculations showed good conservation and convergence, both with respect to the grid size and ε. The adaptive procedure presented here was only based on the distance to the interface. This turned out to work well for problems with a small Reynolds number. In the numerical experiments of the liquid Cobalt droplet the Reynolds number was higher. Because of high gradients in the velocity the criterion how to refine the grid was too crude. It should however be straightforward to take variations in the velocity into account in the adaptive process.

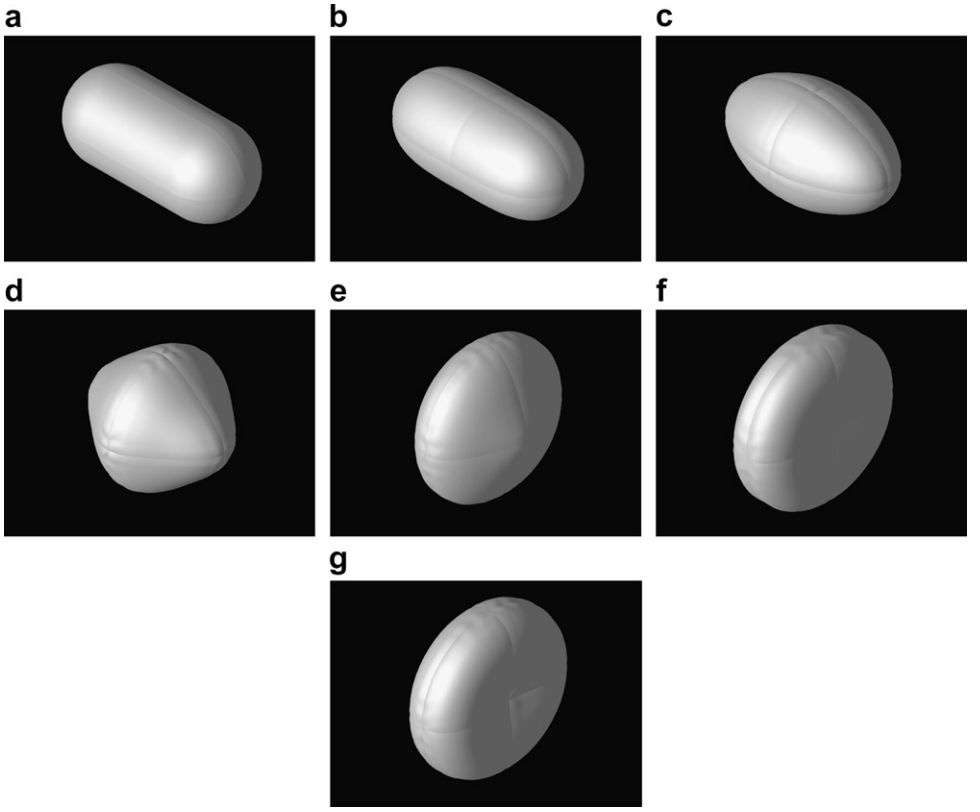

Fig. 13. Results obtained by three dimensional calculation for liquid Cobalt droplet in air: (a) $t = 0$, (b) $t = 0.1$, (c) $t = 0.2$, (d) $t = 0.3$, (e) $t = 0.4$, (f) $t = 0.5$, (g) $t = 0.6$.

Acknowledgments

The authors thank the members of the Computational Phase Transformation project at KTH (www2.mech.kth.se/phasetransf/index.htm). In particular, Walter Villanueva at KTH Mechanics is acknowledged for valuable help on how to use Femlego. This work was supported by SSF Grant A3 02:123.

Appendix A

Here we prove Lemma 1.

Proof. Since $\Phi(x)$ solves (31) it can easily be verified that $\Phi(X(x_0, s)) = g(s) = \frac{1}{1+e^{s/\varepsilon}}$ holds for any $x \in \Omega \setminus \Omega_0$. Define the signed distance function by $\Psi(x)$, i.e.

$$\Psi(x) = \begin{cases} \min\limits_{xI \in \Gamma} \|x - x_I\| & \text{if } x \in \Omega_1, \\ -\min\limits_{xI \in \Gamma} \|x - x_I\| & \text{if } x \in \Omega_2. \end{cases}$$

We have that

$$\int_\Omega \Phi \, \mathrm{d}x - \int_\Omega \widetilde{\Phi} \, \mathrm{d}x \int_{\Omega 1} \Phi \, \mathrm{d}x + \int_{\Omega 2} (\Phi - 1) \, \mathrm{d}x = I_1 + I_2.$$

Since $s(x)$ is the distance from x along a path to some point x_0 on the interface and $\Psi(x)$ is the shortest distance to the interface it is clear that $s(x) \geqslant \Psi(x)$ for any $x \in \Omega \setminus \Omega_0$ Since $g(s)$ is monotonically decreasing we have that $\Phi(x) \leqslant g(\Psi(x))$. It follows that

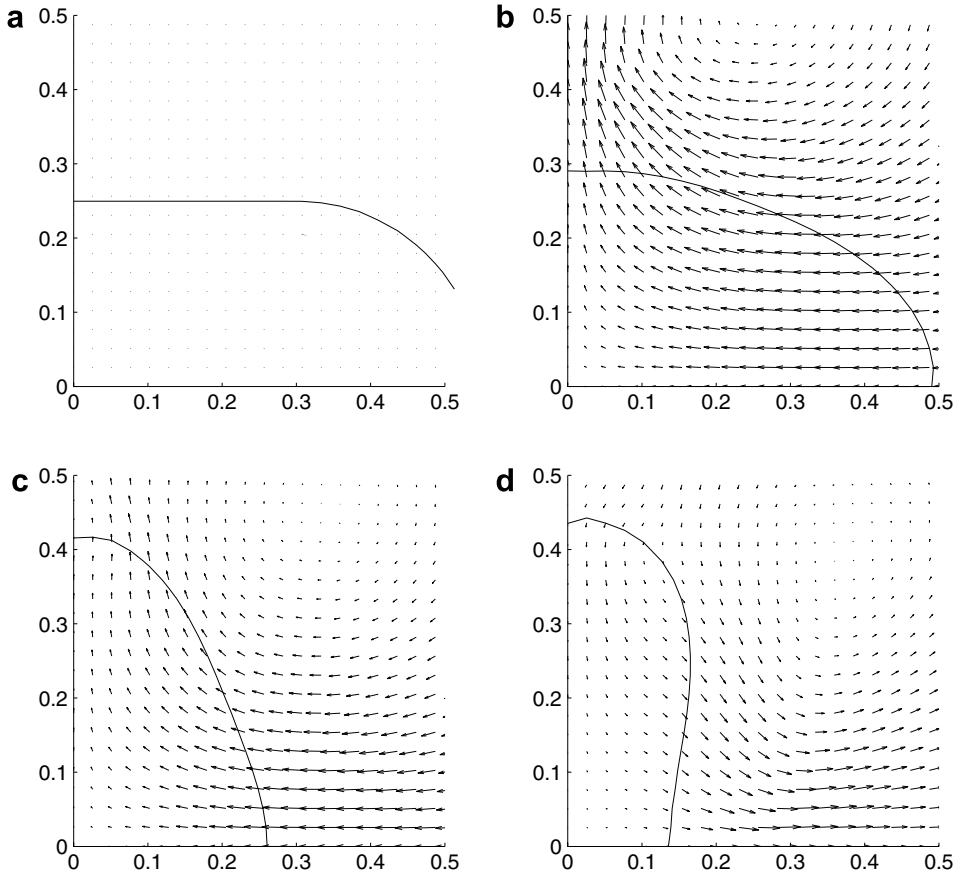

Fig. 14. The velocity field and the interface ($\Phi = 0.5$) for three dimensional Cobalt droplet: (a) $t = 0$, (b) $t = 0.2$, (c) $t = 0.4$, (d) $t = 0.6$.

$$0 < I_1 = \int_{\Omega_1} \Phi \, \mathrm{d}x = \int_{\Omega_1 \setminus \Omega_0} \Phi \, \mathrm{d}x \leqslant \int_0^\infty g(l) L(l) \, \mathrm{d}l, \tag{41}$$

where $L(l)$ is the length of the interface corresponding to $\Psi = l$. Since Γ is smooth there exists an a such that $\Psi(x)$ is smooth for all $x \in \{x : -a \leqslant \Psi(x) \leqslant a\}$. For any $|l| \leqslant a$

$$L(l) = \int_0^{L_\Gamma} 1 + \kappa l \, \mathrm{d}t. \tag{42}$$

Here $L_\Gamma = L(0)$ denotes the length of Γ. For any l it holds that

$$L(l) \leqslant L_\Gamma (1 + \kappa_\infty |l|), \tag{43}$$

where κ_∞ is the maximum magnitude of the curvature of Γ. (41) and (43) now gives

$$0 < I_1 \leqslant L_\Gamma \left(\int_0^\infty g(l) \, \mathrm{d}l + \kappa_\infty \int_0^\infty l g(l) \, \mathrm{d}l \right).$$

Similarly, one obtains bounds on I_2:

$$0 > I_2 \geqslant -L_\Gamma \left(\int_0^\infty g(l) \, \mathrm{d}l + \kappa_\infty \int_0^\infty l g(l) \, \mathrm{d}l \right).$$

Using that

$$\int_a^\infty g(l) \, \mathrm{d}l \int_a^\infty \frac{1}{1 + \mathrm{e}^{l/\varepsilon}} \, \mathrm{d}l = \varepsilon \ln(1 + \mathrm{e}^{-a/\varepsilon})$$

and

$$\int_0^\infty lg(l)\,\mathrm{d}l \int_0^\infty \frac{1}{1+\mathrm{e}^{l/\varepsilon}}\,\mathrm{d}l < \int_0^\infty \frac{l\,\mathrm{d}l}{\mathrm{e}^{l/\varepsilon}} = \varepsilon^2$$

we obtain the bound

$$|I_1 + I_2| < L_\Gamma\left(\varepsilon \ln 2 + \kappa_\infty \varepsilon^2\right).$$

If all the paths $X(x_0, S)$ are straight, non-intersecting lines for all $|s| < b$, then $\Phi(x) = g(\Psi(x))$ and cancellation of errors gives a stricter bound:

$$|I_1 + I_2| < L_\Gamma(\kappa_\infty \varepsilon^2 + \varepsilon \ln(1 + \mathrm{e}^{-b/\varepsilon})). \qquad \square \tag{44}$$

References

[1] W. Noh, P. Woodward, SLIC (simple line interface calculation), in: A. van de Vooren, P. Zandbergen (Eds.), Proceedings of the 5th International Conference in Fluid Dynamics, Lecture Notes in Physics, vol. 59, 1976, pp. 330–340.

[2] R. Scardovelli, S. Zaleski, Direct numerical simulation of free-surface and interfacial flow, Annu. Rev. Fluid Mech. 31 (1999) 567–603.

[3] M. Sussman, P. Smereka, S. Osher, A level set approach for computing solutions to incompressible two-phase flow, J. Comput. Phys. 114 (1994) 146–159.

[4] M. Sussman, E. Fatemi, P. Smereka, S. Osher, An improved level set method for incompressible two-phase flows, Comput. Fluids 27 (1998) 663–680.

[5] S. Unverdi, G. Tryggvason, A front-tracking method for viscous, incompressible, multi-fluid flows, J. Comput. Phys. 100 (1992) 25–37.

[6] W. Villanueva, G. Amberg, Some generic capillary-driven flows, Submitted to Int. J. Multiphase Flow.

[7] E. Olsson, G. Kreiss, A conservative level set method for two phase flow, J. Comput. Phys. 210 (2005) 225–246.

[8] D.H. Sattinger, On the stability of waves of nonlinear parabolic systems, Adv. Math. 22 (1976) 312–355.

[9] J. Goodman, Nonlinear asymptotic stability of viscous shock profiles for conservation laws, Arch. Rat. Mech. Anal. 95 (1986) 325–344.

[10] C. Johnson, Numerical solutions of partial differential equations by the finite element method, Lund: Studentlitteratur, 1987.

[11] J.U. Brackbill, D. Kothe, C. Zemach, A continuum method for modeling surface tension, J. Comput. Phys. 100 (1992) 335–353.

[12] A. Chorin, Numerical solution of the Navier–Stokes equations, Math. Comput. 22 (1968) 745–762.

[13] A. Chorin, On the convergence of discrete approximations to the Navier–Stokes equations, Math. Comput. 23 (1969) 341–353.

[14] J.-L. Guermond, L. Quartapelle, Calculation of incompressible viscous flows by an unconditionally stable projection FEM, J. Comput. Phys. 132 (1997) 12–33.

[15] J.-L. Guermond, P. Minev, J. Shen, An overview of projection methods for incompressible flows, Comp. Meth. Appl. Mech. Eng. 195 (2006) 6011–6045.

[16] G. Amberg, R. Tönhardt, C. Winkler, Finite element simulations using symbolic computing, Math. Comput. Simul. 49 (1999) 257–274.

[17] M. Do-Quang, I. Singer-Loglnova, W. Villanueva, G. Amberg, Problem solving environment for parallel adaptive computation, Submitted to Parallel Computing.

[18] S. Shin, D. Juric, Modeling three-dimensional multiphase flow using a level contour reconstruction method for front tracking without connectivity, J. Comput. Phys. 180 (2002) 427–470.

[19] D. Fyfe, E. Oran, M. Fritts, Surface tension and viscosity wit lagrangian hydrodynamics on a triangular mesh, J. Comput. Phys. 76 (1988) 349–384.

[20] E. Aulisa, S. Manservisi, R. Scardovelli, A novel representation of the surface tension force for two-phase flow with reduced spurious currents, Comp. Meth. Appl. Mech. Eng. 195 (2006) 6239–6257.

[21] C.J. Smithells, Smithells metals reference book, 7th ed., Butterworth-Heinemann, Oxford, 1992.