Available online at www.sciencedirect.com

SCIENCE DIRECT®

International Journal of Heat and Mass Transfer 49 (2006) 740–754

International Journal of HEAT and MASS TRANSFER

www.elsevier.com/locate/ijhmt

Comparison of volume-of-fluid methods for surface tension-dominant two-phase flows

D. Gerlach [a], G. Tomar [b], G. Biswas [b,*], F. Durst [a]

[a] Lehrstuhl für Strömungsmechanik, Universität Erlangen–Nürnberg, Cauerstr. 4, D-91058 Erlangen, Germany
[b] Department of Mechanical Engineering, Indian Institute of Technology, Kanpur 208 016, India

Received 28 February 2005; received in revised form 11 July 2005
Available online 11 October 2005

Abstract

The capabilities of three volume-of-fluid methods for the calculation of surface tension-dominant two-phase flows are tested. The accurate calculation of the interface remains a problem for the volume-of-fluid method if the surface tension force plays an important role and the density ratios of the fluids in different phases are high. The result can be an artificial velocity field at the interface (“parasitic currents”), which can destabilize the interface significantly. The three different algorithms compared can be distinguished by their methods to compute the surface tension force, namely, the method using a kernel function for smoothing the discontinuity at the interface, a combined level-set and volume-of-fluid approach and a parabolic reconstruction of surface tension. The test cases consist of an equilibrium rod, a capillary wave and the Rayleigh–Taylor instability. The analytical solutions for each problem serve to examine the accuracy and the convergence behavior of each approach. Finally, the slow formation of a gas bubble at an underwater orifice was computed with the combined level-set and volume-of-fluid method and the results are compared with an analytical solution based on the Young–Laplace equation.

1. Introduction

Numerical methods for computing fluid flows with material interfaces have become very popular in recent decades owing to their capability to give a deep insight into the physical mechanisms. These methods can be divided into two groups depending on the type of grids used: *moving* or *fixed* grids. In the first group, the interface is treated as a sharp boundary whose motion is followed, i.e., the interface is identified with control volume boundaries dividing the computational domain into more than one domain. In the second group, the interface is moved through a fixed (Eulerian) grid. The interface position is computed at each time step and in all cells, which are partially filled. The fixed-grid methods have the advantage that they can handle strong topological deformations of the interface such as merging and fragmentation, but may fail to calculate the surface tension force at the interface accurately under certain circumstances. The aim of the present study was to examine three existing methods on fixed grids and assess their applicabilities in surface tension-dominant flows.

Two important approaches of fixed-grid methods, namely the *volume-of-fluid* and the *level-set* approaches, are mentioned in this context. In the level-set method [1], the interface is defined by a level-set function ϕ. Typically this function is initialized as a signed distance function from the interface, positive on one side and negative on the other side of the interface. The interface itself is represented by the zero level of ϕ. Based on the calculated velocity field, the front evolves as a solution of a transport equation for ϕ, whereby, in general, ϕ does not remain a distance function at later times. This leads to interface smearing and difficulties in preserving the mass conservation. Re-initialization techniques can be employed as introduced first by Sussman et al. [2] to overcome these problems. The level-set method has the inherent strength

[*] Corresponding author. Tel.: +91 0512 597616; fax: +91 0512 597408. *E-mail address:* gtm@iitk.ac.in (G. Biswas).

0017-9310/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ijheatmasstransfer.2005.07.045

Nomenclature

E	error
F	void fraction function
$\mathbf{f}_{\text{sa}}$	surface tension force per unit area
$\mathbf{f}_{\text{sv}}$	surface tension force per unit volume
g	gravitational acceleration
G	error function
h	grid spacing
H_ϵ	smoothed Heaviside function
K	wavenumber
$\mathbf{K}_8$	smoothing kernel
l	shift length of an interface to the cell center
$\mathbf{n}$	normal vector
Oh	Ohnesorg number
P	pressure
r	radial component of cylindrical polar coordinates
t	time
u	velocity component in x-direction
v	velocity component in y-direction
$\mathbf{v}$	velocity vector (u, v)
V	liquid volume fluxed through a cell face
x	horizontal coordinate
$\mathbf{x}$	position vector
y	vertical coordinate
z	vertical component of cylindrical polar coordinates

Greek symbols

δ_{s}	interface Dirac delta function
δV	void fraction fluxed through a cell face
Δt	time step
Δx, Δy	grid spacing in x and y directions
ϵ	half width of the transition region
κ	mean curvature
λ	wavelength
μ	dynamic viscosity
ρ	density
σ	surface tension
ϕ	level-set function

Subscripts

g	gas property
i, j	ith and jth computational cell in x and y directions
l	liquid property
x, y	derivative with respect to x or y

Superscripts

¯	approximated value
ˆ	unit vector
˜	smoothed field
*	intermediate value
n	nth time step
x, y	components in x or y direction

that ϕ varies smoothly across the interface, which leads to convenient formulae for the curvature and the interface normal vector. Owing to its simplicity, the level-set method has been used in solving a wide variety of problems [3].

In the volume-of-fluid method, the volume of each fluid is tracked in all cells containing portions of the interface, rather than the interface itself. A volume (void) fraction function F is defined for one particular fluid inside a cell as its material volume divided by the total cell volume. Hence F is zero or unity in pure fluid cells and has a value of $0 < F < 1$ in multi-fluid cells. Volume-of-fluid algorithms consist of three major parts, that is, the *interface reconstruction* method, which finds an explicit description of the interface in each multi-fluid cell based on void fractions at this time step, the *advection algorithm*, which calculates the distribution of F at the next time step by solving an advection equation using the reconstructed interface and the underlying velocity field, and the *surface tension model*, which takes account of surface tension effects at the interface.

In an incompressible formulation of the governing equations, conservation of mass is equivalent to conservation of volume. Hence a volume-of-fluid advection algorithm to advance the front can be found, which conserves the void fraction (volume). This procedure requires a non-diffusive solution of the advection equation, which is obtained by a geometrically based calculation technique of the void fraction fluxes at the cell faces based on the reconstructed interface. The conservation of the volume of each fluid is an important property of the volume-of-fluid method. In contrast, in the level-set formulation, ϕ has to be re-initialized after each time step in order to preserve ϕ as a distance function and to conserve mass, since conservation of ϕ and mass are not equivalent.

One can give a brief survey of the historical development of the volume-of-fluid formulation by considering the methods for reconstructing the interface. Two early approaches are the SLIC algorithm (simple line interface calculation) of Noh and Woodward [4] and the volume-of-fluid algorithm of Hirt and Nichols [5], in which the interface is represented by a piecewise-constant line in each two-fluid cell, either vertically or horizontally. A significant improvement of the interface representation was achieved by Youngs [6] by introducing a piecewise-linear method (piecewise-linear interface calculation, PLIC). The method of Youngs was shown to be very robust and efficient, but only of first-order accuracy. An improved version of phase interface representation (LVIRA) was devised by Puckett and co-workers [7] and implemented successfully by Welch and Rachidi [8] and Agarwal et al. [9] to predict bubble formation in film boiling. Here the orientation of the piecewise-linear line segment is computed more carefully leading to second-order accuracy if the interface is sufficiently smooth. Finally, the recently published method of Renardy and Renardy [10] should be mentioned, which uses piecewise-parabolic segments to capture the interface.

In contrast to the interface representation methods, the methods for introducing surface tension effects at the interface into the physical model remain a problem. In other words, volume-of-fluid formulations still lack an accurate method for calculation of surface tension predominantly in problems with high density ratios. Many recently published investigations related to this topic can be found in the literature [10–13]. Generally, the influence of surface tension is incorporated into the momentum equation following the continuum surface force (CSF) model of Brackbill et al. [14]. For doing so, the local curvature and the interface normal vector have to be calculated. This task is difficult, since the discontinuous void fraction function disallows the application of ordinary discretization schemes. An inconsistent calculation of the surface tension force can then result in the well-known phenomena of so-called "parasitic currents" [15]. These artifacts are a serious problem for the capability of the volume-of-fluid approach, since it is a restriction of the approach rather than a numerical inaccuracy. Under some circumstances, it may result in strong vortices at the interface despite the absence of external forcing, which may lead to catastrophic instability of the interface or even to break-up. Usually problems with parasitic currents arise when flows with high density ratios are considered. Unfortunately, a large range of applications belong to this group, e.g. two-phase flows of air bubbles in water or water drops in air. In such problems surface tension effects often play a dominant role, which are difficult to access with standard surface tension models. The amplitude of the spurious currents was estimated by Lafaurie et al. [15] for the case of a static drop to be of the order of $0.01\sigma/\mu$, where σ and μ are the surface tension coefficient and the viscosity, respectively. The Reynolds number of the spurious currents is therefore proportional to $1/Oh^2$, where Oh is the Ohnesorg number, defined as $Oh = \mu/(\sigma\rho L)^{1/2}$, ρ and L being the density and a characteristic length.

The present study is an attempt to examine the capabilities of recently published volume-of-fluid methods in surface tension-dominant flow problems. The two-dimensional incompressible Navier–Stokes equations for two immiscible fluids are solved for the test cases of an equilibrium rod, a capillary wave and the Rayleigh–Taylor instability. Analytical solutions are provided for each test case, making a rigorous validation possible.

Three different surface tension models are considered, namely (i) an improved version of the method of Brackbill et al. [14] due to Williams et al. [16], (ii) a combined level-set and volume-of-fluid approach based on Sussman et al. [17] and Son [18,19] and (iii) a method using a piecewise-parabolic representation of the interface based on the work of Renardy and Renardy [10]. In the following section, the volume-of-fluid formulation is given and the three algorithms under consideration are described in detail. To validate the interface reconstruction and advection methods, two test cases are presented in Section 3. The behavior of each algorithm under surface tension-dominant problems are discussed in Section 4. In Section 5 the combined level-set and volume-of-fluid approach is applied to the problem of the bubble formation at orifices. Water and air with a density ratio of about 1000 were used as working fluids. Under low flow rates, where the acting forces can be reduced to pressure and capillary forces, a comparison of the full numerical simulation with analytically predicted bubble shapes during the formation process is possible. The analytical solution was calculated based on the Young–Laplace equation for given boundary conditions [20,21]. This comparison presents a new and useful test case of engineering relevance.

2. Numerical description

The computational domain is uniformly divided into control volumes of spacing $\Delta x = \Delta y = h$. Each control volume is denoted by the indices i, j in the x, y Cartesian coordinate system. The staggered grid arrangement of Harlow and Welch [22] is applied, in which the scalar quantities such as the pressure, the void fraction F and the level-set function ϕ are defined at the cell centers (i,j) and the vector quantities such as the velocity components at the cell faces, i.e. the x-component denoted at the right cell face as $u_{i+1/2,j}$ and the y-component denoted at the top face as $v_{i,j+1/2}$.

The interface separates the two fluids, say liquid and gas. Numerically the particular fluid is defined by the void fraction $F_{i,j}$ in each control volume (i,j) as the fraction of the liquid inside a cell. It follows that

$$F = \begin{cases} 0 & \text{if there is a gas cell,} \\ 1 & \text{if there is a liquid cell} \end{cases} \tag{1}$$

and $0 < F < 1$ represents a two-phase cell including a portion of the interface.

Based on the void fraction, a single set of equations can be written for the entire computational domain consisting of the mass and momentum equations for an incompressible Newtonian fluid, given by

$$\nabla \cdot \mathbf{v} = 0, \tag{2}$$

$$\rho(F)\left(\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v}.\nabla\mathbf{v}\right) = -\nabla P + \rho(F)\mathbf{g} + \nabla \cdot \left[\mu(F)(\nabla\mathbf{v} + (\nabla\mathbf{v})^T)\right] + \mathbf{f}_{\mathrm{sv}}, \tag{3}$$

where $\mathbf{v} = (u, v)$ is the velocity vector, t is the time, P is the pressure, $\mathbf{g} = (0, -g)$ is the gravitational acceleration, $\mathbf{f}_{\mathrm{sv}}$ is the surface tension force and $\rho(F)$ and $\mu(F)$ are the density and viscosity, defined by

$$\rho(F) = \rho_l F + \rho_g(1 - F) \tag{4}$$

$$\mu(F) = \mu_l F + \mu_g(1 - F). \tag{5}$$

The term $\mathbf{f}_{sv}$ is the surface tension force per unit volume, which can be introduced in the momentum equation following the continuum surface force model of Brackbill et al. [14] as

$$\mathbf{f}_{sv} = \sigma\kappa\hat{\mathbf{n}}\delta_s, \tag{6}$$

where σ is the (constant) surface tension coefficient, κ is the mean curvature of the interface, $\hat{\mathbf{n}}$ is the unit normal vector of the interface and δ_s is the interface delta function. Therefore, it is apparent that Eq. (3) simplifies to a general one-phase momentum equation in the bulk region of either liquid or gas.

Since the fluid type remains constant along particle paths, the void fraction F is passively advected by

$$\frac{\partial F}{\partial t} + \mathbf{v} \cdot \nabla F = 0. \tag{7}$$

Using the continuity equation (3), the advection equation (7) can be reformulated into a conservative form as

$$\frac{\partial F}{\partial t} + \nabla \cdot (\mathbf{v}F) = 0. \tag{8}$$

By solving Eq. (8) numerically, overshoots and undershoots can occur, i.e. the void fraction can lie outside the interval $[0, 1]$. Although these effects are negligibly small, the interval of the void fraction was ensured by truncation if necessary. Based on the updated void fraction field, the fluid properties (Eqs. (4) and (5)) and the surface tension force (Eq. (6)) are calculated and the interface is reconstructed using a method described in Section 2.2.

2.1. Advection algorithm

Suppose that the velocity field $(u^n_{i+1/2,j}, v^n_{i,j+1/2})$ is known at the time $t^n = n\Delta t$. Also it is assumed that the interface is reconstructed in all two-phase cells, i.e., cells which have $0 < F_{i,j} < 1$, in such a way that the interface defines a liquid and gas region in the cell of volume $F^n_{i,j}h^2$ and $(1 - F^n_{i,j})h^2$. In order to advance $F^n_{i,j}$ to $F^{n+1}_{i,j}$ at time $t^{n+1} = (n + 1)\Delta t$, the advection equation (8) is solved. This algorithm is generally referred to as *volume-of-fluid advection algorithm*.

Eq. (8) can be discretized by finite-differencing methods as

$$F^{n+1}_{i,j} = F^n_{i,j} + \frac{\Delta t}{\Delta x}(\delta V_{i-1/2,j} - \delta V_{i+1/2,j}) + \frac{\Delta t}{\Delta y}(\delta V_{i,j-1/2} - \delta V_{i,j+1/2}), \tag{9}$$

where $\delta V_{i+1/2,j} = (uF)_{i+1/2,j}$ is, for example, the amount of liquid void fraction fluxed though the right cell face. To solve Eq. (9), the operator split approach is applied, resulting in two equations:

$$F^*_{i,j} = F^n_{i,j} + \frac{\Delta t}{\Delta x}(\delta V_{i-1/2,j} - \delta V_{i+1/2,j}) + \frac{\Delta t}{\Delta x}(u_{i+1/2,j} - u_{i-1/2,j}), \tag{10}$$

$$F^{n+1}_{i,j} = F^*_{i,j} + \frac{\Delta t}{\Delta y}(\delta V^*_{i,j-1/2} - \delta V^*_{i,j+1/2}) + \frac{\Delta t}{\Delta y}(v_{i,j+1/2} - v_{i,j-1/2}), \tag{11}$$

where the star (*) denotes the values at the intermediate stage after the sweep in the first direction. Here it is important to explain the third term on the right-hand side of Eqs. (10) and (11). It can be shown that more accurate results are obtained by employing this "divergence correction". In other words, we split $\partial F/\partial t + \nabla \cdot (\mathbf{v}\ F) = F\nabla \cdot \mathbf{v}$ rather than Eq. (8) into two equations [7,23].

The task is now to estimate δV for the discretized Eqs. (10) and (11) in such a way that the solution of the advection equation conserve the void fraction exactly, i.e. no numerical diffusion should be introduced. This is obtained by using a geometrically based calculation of δV. Consider the right face of a two-phase control volume as depicted in Fig. 1, where the liquid is located beneath the gas. Based on the velocity at the right cell face, a portion $(u_{i+1/2,j}\Delta t\ h)$ of the control volume can be calculated, whose fluids will be fluxed into the right neighboring cell, supposing $u_{i+1/2,j}$ is positive. Knowing the interface position, the amount of liquid in the fluxed rectangle $V_{i+1/2,j}$ is computed. Hence the flux is given by

$$\delta V_{i+1/2,j} = \frac{u_{i+1/2,j}V_{i+1/2,j}}{u_{i+1/2,j}\Delta t\Delta y} = \frac{V_{i+1/2,j}}{\Delta t\Delta y}. \tag{12}$$

After solving Eq. (10) with help of δV calculated as in Eq. (12), the interface is again reconstructed based on $F^*_{i,j}$. For the vertical fluxes the same method is followed, now calculating geometrically the fluxes of the void fractions crossing the top and bottom cell face. Hence the new void fraction field $F^{n+1}_{i,j}$ is obtained. The approach can be made second-order accurate by alternating the sweeping directions in each time step [24].

Care has to be take for the discretization of the "divergence correction" terms of the right-hand side of Eqs. (10)

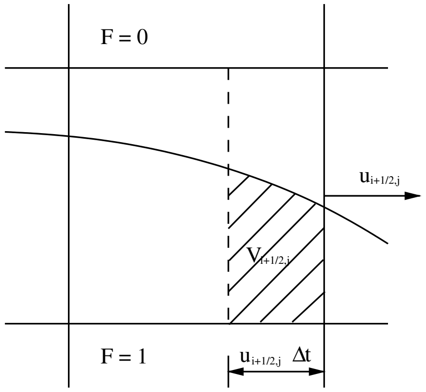

Fig. 1. Calculation of the fluxed void fraction at the right cell face.

and (11). By employing an implicit scheme in the first sweeping direction and an explicit scheme in the second direction as suggested by Puckett et al. [7], the conservation of F is maintained.

2.2. Interface reconstruction

As seen in the previous section, the definition of the interface position is required to advance the void fraction field in time. In the context of the volume-of-fluid approach, methods for doing this are called *interface reconstruction algorithms*. Three different methods are presented here, which together constitute with their surface tension model (Section 2.3) three different approaches for solving two-phase flows including surface tension effects. The interface reconstruction methods are (i) the least-squares volume-of-fluid interface reconstruction algorithm (LVIRA) [7], (ii) a method based on the parabolic reconstruction of surface tension method (PROST) [10] and (iii) a combined level-set and volume-of-fluid approach (CLSVOF) [17,19].

Common to all reconstruction methods described here is that the interface is represented by piecewise-linear interface segments. The position of the interface segment is uniquely defined by an interface unit vector $\hat{\mathbf{n}}$ pointing into the liquid region and a distance l from the cell center to the interface as indicated in Fig. 2. The aim of the reconstruction algorithm is to determine $\hat{\mathbf{n}}_{i,j}$ and $l_{i,j}$ in such a way that the approximation of the interface reproduces the updated void fraction $F_{i,j}$ in each cell exactly. This is realized by optimizing $l_{i,j}$ for a given $\hat{\mathbf{n}}_{i,j}$ by means of a standard root-finding method. The methods for obtaining $\hat{\mathbf{n}}$ are given below.

2.2.1. LVIRA

LVIRA involves the void fractions of a 3×3 block around the cell of interest to optimize the interface normal vector. Based on the current approximation $\hat{\mathbf{n}}_{i,j}$ and $l_{i,j}$ of the center cell, a straight line is defined through the 3×3 block. The optimum value of $\hat{\mathbf{n}}_{i,j}$ is the one corresponding to the minimum of the function

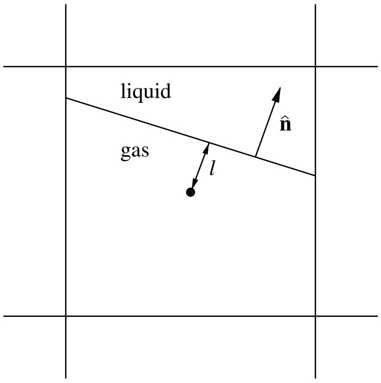

Fig. 2. Definition of the piecewise-linear interface.

$$G_{i,j}(\hat{\mathbf{n}}_{i,j}) = \sum_{k,l=-1}^{1} (F_{i+k,j+l} - \bar{F}_{i+k,j+l}(\hat{\mathbf{n}}_{i,j}))^2, \tag{13}$$

where $F_{i+k,j+l}$ is the updated (given) void fraction and $\bar{F}_{i+k,j+l}$ is the approximation due to the linear interface through the block. The optimization algorithm of Eq. (13) requires an initial guess. An improved version of Youngs' method [6] has been used to provide the initial value of $\hat{\mathbf{n}}_{i,j}$ as given by Rudman [25] or also known as the Green–Gauss gradient [26]:

$$n^x_{i,j} = \frac{1}{\Delta x}(F_{i+1,j+1} + 2F_{i+1,j} + F_{i+1,j-1} - F_{i-1,j+1} - 2F_{i-1,j} - F_{i-1,j-1}), \tag{14}$$

$$n^y_{i,j} = \frac{1}{\Delta y}(F_{i+1,j+1} + 2F_{i,j+1} + F_{i-1,j+1} - F_{i+1,j-1} - 2F_{i,j-1} - F_{i-1,j-1}). \tag{15}$$

These stencils themselves have been demonstrated to be very robust and efficient [25], but only of first-order accuracy. In our tests of the interface capturing methods (interface advection together with interface reconstruction) Youngs' method alone for interface reconstruction is also considered. More details about the LVIRA method are given elsewhere [7].

2.2.2. PROST

In the PROST method of the present work the interface is reconstructed using a piecewise-parabolic curve. Similarly to the approach in the LVIRA method, the parabolic curve is fitted over 3×3 block. The parameters of the curve defined in the center cell of the block are found such that the difference between known void fractions and those based on the parabolic reconstruction is minimized in the 3×3 block under consideration.

The parabolic curve is represented by a quadratic equation of the form

$$k + \hat{\mathbf{n}}\cdot(\mathbf{x}-\mathbf{x}_0) + (\mathbf{x}-\mathbf{x}_0)\cdot\mathbf{A}(\mathbf{x}-\mathbf{x}_0) = 0. \tag{16}$$

This equation describes a parabola if $\mathbf{A}$ is a symmetric matrix with the property that $\mathbf{A}\hat{\mathbf{n}} = 0$, which ensures that the axis of the paraboloid is along $\hat{\mathbf{n}}$. The equation thus obtained involves three parameters for our two-dimensional case: k, one for $\mathbf{n}$, since $\hat{\mathbf{n}}$ can be expressed as $\sin(\theta)\hat{\imath} - \cos(\theta)\hat{\jmath}$, and one for $\mathbf{A}$. The quadratic equation in two dimensions can be written in a simplified form as

$$k + n^x(x-x_0) + n^y(y-y_0) + a[(x-x_0)(y-y_0)]\begin{bmatrix}(n^y)^2 & -n^x n^y\\ -n^x n^y & (n^x)^2\end{bmatrix}\begin{bmatrix}(x-x_0)\\ (y-y_0)\end{bmatrix} = 0. \tag{17}$$

These parameters take values such that an error function defined as

$$G_{i,j}(\theta,k,a) = \sum_{k,l=-1}^{1} w(F_{i+k,j+l} - \bar{F}_{i+k,j+l}(\theta,k,a))^2 \tag{18}$$

is minimized. Summation here is over the 3×3 block, which is expected to have at least three and at most five two-phase cells. As in the LVIRA algorithm, $F_{i+k,j+l}$ denotes the given void fraction and $\bar{F}_{i+k,j+l}$ is the approximation due to the parabolic interface through the block. Two weight factors $w_1 \cdot w_2 = w$ are used as suggested [10] to force the optimization method to fit better in cells of F near to 0 and 1 and to suppress the formation of "checkerboards" at the interface.

For optimization of the three parameters, the void fractions $\bar{F}$ are computed by area calculations. Intersection points between the current curve fitted and the cell faces are first identified and two more points are added inside the cell. The area of the polygon is then calculated by collecting the intersection points together with the cell vertices lying on the liquid side of the interface in counter-clockwise direction, so that the area is given by

$$\frac{1}{2}\sum_{i=1}^{n}(x_i y_{i+1} - x_{i+1} y_i), \tag{19}$$

where n is the number of vertices of the polygon with $i = 0$ corresponding to $i = n$.

To minimize the least-squares function (Eq. (18)), the Hooke–Jeeves pattern search algorithm [27] was applied, which is a gradient-free algorithm. Parameters k, θ and a thus found define the curve which represents the interface at the center cell. It is important to note the following difference between the present implementation and the original algorithm [10]. The PROST method presented here also uses a piecewise-linear interface representation as in the other two algorithms. The resultant unit normal vector of the parabola $\hat{\mathbf{n}} = (\sin(\theta), \cos(\theta))$ is utilized as the interface normal vector, defining the position of the interface by adjusting the distance l. These two values are subsequently utilized for the advection algorithm (Section 2.1). Hence the PROST method is used here as a tool for computing the surface tension force and the surface normal vector rather than as a complete volume-of-fluid algorithm.

2.2.3. CLSVOF

The CLSVOF method is an attempt to combine the advantages of the level-set and the volume-of-fluid approaches. That is, the volume-of-fluid advection algorithm is applied, because it is mass conserving, and the level-set function is computed additionally, because its smoothness allows one to use simple finite-differencing schemes to calculate the interface normal vector and the curvature of the interface [17–19].

Suppose that the position of the interface is initially known and the level-set function $\phi_{i,j}$ is given in each cell as the signed distance function from the reconstructed interface. The coupled approach for capturing the interface is realized by the following steps:

(1) The first sweep, say in the x-direction, of the volume-of-fluid advection algorithm is performed by solving Eq. (10). As the result, $F^*_{i,j}$ is known as an intermediate distribution.

(2) ϕ is advected in the same sweeping direction by solving the split form of

$$\frac{\partial \phi}{\partial t} + \mathbf{v} \cdot \nabla \phi = 0. \tag{20}$$

That is, for the sweep in the x-direction we have

$$\frac{\partial \phi}{\partial t} + u\frac{\partial \phi}{\partial x} = 0. \tag{21}$$

The convective term in Eq. (21) is discretized by employing a second-order ENO scheme as suggested in [18,28].

(3) Since the level-set field is smooth, the interface normal vector can simply be calculated by a central finite-differencing scheme for discretization of

$$\hat{\mathbf{n}} = \frac{\nabla \phi}{|\nabla \phi|} \tag{22}$$

in all two-phase cells [19]. As given above, the length l is adjusted to match the given void fraction with that of the reconstructed interface.

(4) The second sweep in the y-direction (Eq. (11)) leads to the final distribution $F^{n+1}_{i,j}$ in the current time step.

(5) Also the intermediate level-set function is advanced in the y-direction by solving

$$\frac{\partial \phi}{\partial t} + v\frac{\partial \phi}{\partial y} = 0. \tag{23}$$

The discretization scheme is the same as described in step 2.

(6) The interface is again reconstructed based on ϕ (see step 3).

(7) Since the finite-differencing scheme for advancing ϕ (steps 2 and 5) does not conserve the amount of liquid in the domain exactly, the level-set function has to be reinitialized in all cells based on the reconstructed interface. The algorithm to reinitialize ϕ was implemented following the method of Sussman et al. [17].

The accuracy of the advection equations of F and ϕ can be made to second order when the sweeping directions are changed in each time step (see Section 2.1).

2.3. Surface tension model

In the continuum surface force model (CSF) [14] the force of interfacial tension is introduced as a surface volume force $\mathbf{f}_{\text{sv}}$ (force per unit volume), written as

$$\mathbf{f}_{\text{sv}} = \mathbf{f}_{\text{sa}}\delta_{\text{s}}, \tag{24}$$

where δ_{s} is the Dirac delta function, which is zero everywhere except at the interface, and $\mathbf{f}_{\text{sa}}$ is the surface tension force per unit interfacial area, given as

$$\mathbf{f}_{\text{sa}} = \sigma\kappa\hat{\mathbf{n}} + \nabla_{\text{s}}\sigma, \tag{25}$$

where σ is the surface tension coefficient, κ and $\hat{\mathbf{n}}$ are the curvature and the unit normal vector of the interface and ∇_s is the surface gradient. In the present work, the tangential component $\nabla_s\sigma$ is omitted, since σ remains constant without temperature or concentration gradients. The first term in Eq. (25) is the force acting normal to the interface, proportional to the curvature and the surface tension coefficient. The normal force tends to smooth regions of high values of κ. The surface tension force per unit volume is then

$$\mathbf{f}_{\mathrm{sv}} = \sigma\kappa\hat{\mathbf{n}}\delta_{\mathrm{s}}. \tag{26}$$

This expression can be simplified using the definition of the Dirac delta function [14]

$$\delta_{\mathrm{s}} = \frac{|\nabla c|}{[c]} = |\nabla c| = |\mathbf{n}|, \tag{27}$$

where c is a characteristic color function defining each fluid involved and $[c]$ is the jump of the color function $(c_2 - c_1)$ across the interface. As will be seen in subsequent sections, $[c]$ is unity, simplifying δ_{s} to $|\mathbf{n}|$. Therefore, the surface tension force introduced in the momentum equation (3) has the form

$$\mathbf{f}_{\mathrm{sv}} = \sigma\kappa\mathbf{n}. \tag{28}$$

The aim of the surface tension model is to calculate the curvature κ and the interface normal vector $\mathbf{n}$. Three methods are implemented for the three algorithms under consideration.

2.3.1. K_8 kernel

In this approach, the void fraction function F is chosen as the color function. Since F changes abruptly across the interface, large inaccuracies would be introduced by applying standard discretization schemes for the spatial derivatives of F. Hence, as in the original CST method (Brackbill et al. [14]), F is smoothed in a region of finite thickness concentrated around the interface (transition region). This mollified void field $\tilde{F}$ is obtained by convolving F with the radially symmetric kernel K_8, as suggested by Williams et al. [16], defined by

$$\mathbf{K}_8(r,\epsilon) = \begin{cases} A[1-(r/\epsilon)^2]^4 & \text{if } r < \epsilon, \\ 0 & \text{otherwise,} \end{cases} \tag{29}$$

where ϵ is the size of the support Ω_K of the kernel and A is a normalization constant to ensure $\int_{\Omega_K}\mathbf{K}_8(r,\epsilon)\,\mathrm{d}r = 1$, that is,

$$\widetilde{F}(\mathbf{x}) = \mathbf{K}_8 * F(\mathbf{x}) = \int_{\Omega_K} F(\mathbf{x}')\mathbf{K}_8(\mathbf{x}'-\mathbf{x})\,\mathrm{d}\mathbf{x}'. \tag{30}$$

The normal to the interface $\mathbf{n} = \nabla\widetilde{F}$ is computed by a central differencing schema on the smooth void fraction field $\tilde{F}$. The curvature

$$\kappa = -\nabla\cdot\hat{\mathbf{n}} = -\nabla\cdot\frac{\mathbf{n}}{|\mathbf{n}|} \tag{31}$$

is found by applying a finite-difference stencil defined on a 3×3 matrix (Eqs. (14) and (15)). Since $\mathbf{n}$ in Eq. (28) is nonzero only in the transition region, the surface tension force acts only in the transition region, which is of finite extent 2ϵ around the interface. The value of ϵ should be chosen with caution. Too little smoothing can lead to noise in the curvature field, whereas too much smoothing can violate the locality of the surface tension force.

2.3.2. PROST

Following the PROST algorithm [10], the local curvature results directly from the optimization procedure described in Section 2.2.2. The curvature κ at the apex of the parabola can be found to be $\kappa = 2\mathrm{tr}(\mathbf{A}) = 2a$, with A being the matrix from Eq. (16).

In contrast to the K_8 and the following CLSVOF method, the surface tension force (Eq. (28)) is calculated only in two-phase cells and not in a transition region. This is a natural choice and an important feature, since no dependence of the calculated surface tension force exists on the width of the transition region. For doing so, the curvature κ and the normal vector $\mathbf{n}$ of Eq. (28) have to be known at the cell faces of the scalar control volumes. The product $\kappa\mathbf{n}$, for example, at the east cell face is given by

$$\kappa\left.\frac{\partial F}{\partial x}\right|_{i+1/2,j} = \left(\frac{w_1\kappa_{i,j} + w_2\kappa_{i+1,j}}{w_1 + w_2}\right)\cdot\left(\frac{F_{i+1,j} - F_{i,j}}{\Delta x}\right), \tag{32}$$

where $w_1 = F_{i,j}(1 - F_{i,j})$ and $w_2 = F_{i+1,j}(1 - F_{i+1,j})$ [10].

2.3.3. CLSVOF

In the combined level-set and volume-of-fluid approach, advantage is taken of the smooth level-set function ϕ. In contrast to the discontinuous F field, ordinary discretization schemes can be applied to ϕ. The normal vector $\mathbf{n}$ is calculated following Sussman et al. [17] as

$$\mathbf{n} = \nabla H_\epsilon(\phi), \tag{33}$$

where H_ϵ is a smoothed Heaviside function, defined as

$$H_\epsilon(\phi) = \begin{cases} 0 & \text{if } \phi < -\epsilon, \\ \frac{1}{2}\left[1 + \frac{\phi}{\epsilon} + \frac{1}{\pi}\sin\left(\frac{\pi\phi}{\epsilon}\right)\right] & \text{if } |\phi| \leqslant \epsilon, \\ 1 & \text{if } \phi > \epsilon. \end{cases} \tag{34}$$

The choice of $H_\epsilon(\phi)$ instead of ϕ for computing $\mathbf{n}$ guarantees a finite thickness of the transition region of 2ϵ.

The curvature κ is given by

$$\kappa = -\nabla\cdot\hat{\mathbf{n}} = \nabla\cdot\frac{\nabla\phi}{|\nabla\phi|} = -\frac{\phi_y^2\phi_{xx} - 2\phi_x\phi_y\phi_{xy} + \phi_x^2\phi_{yy}}{(\phi_x^2 + \phi_y^2)^{3/2}}. \tag{35}$$

Eqs. (33) and (35) are discretized by ordinary central-differencing schemes.

Instead of using expression (26) for the calculation of the surface tension force based on the continuum surface force model of [14], the continuous surface stress model (CSS) [15,29] can also be used, i.e. $\mathbf{f}_{\mathrm{sv}} = \nabla\cdot[(\mathbf{1} - \hat{\mathbf{n}}\otimes\hat{\mathbf{n}})\sigma\delta_{\mathrm{s}}]$. Both expressions can be shown to be equivalent for constant σ [15]. Tests of the CSS method in the context of the CLSVOF approach applied to the problem of an equilibrium rod as discussed in Section 4.1 shows that the CLSVOF-CSF method works slightly better than the CLSVOF-CSS approach. Hence the results presented below are based on the CLSVOF-CSF method.

2.4. Outline of the numerical solution procedure

The methods in Sections 2.1–2.3 describe the special features of volume-of-fluid methods compared with common one-fluid finite-differencing flow solvers. These methods are incorporated into a solution process as follows. A staggered MAC grid [22] is used as the basis for the numerical algorithm. The convective term in the momentum equation (3) is discretized by an essentially non-oscillatory (ENO) scheme of second order (see, e.g, [28]). All other space derivatives are centered. Suppose the void fraction distribution F^n at the time $t^n = n\Delta t$ is known, the densities and viscosities at t^n are calculated from Eqs. (4) and (5) based on F in the PROST method, on the smoothed void fraction field $\tilde{F}$ for the K_8 method (Eq. (30)) and on the Heaviside function (Eq. (34)) for the CLSVOF method. Also the surface tension force (Eq. (28)) is computed using the methods in Section 2.3. Then the discretized form of the momentum equation (3) is solved explicitly, resulting in an intermediate velocity field, which is, in general, not divergence free (Eq. (3)), because the pressure gradient is discretized using values at the old time step t^n. Therefore, the velocity at the new time step is eliminated from the discretized continuity and momentum equations in such a way that the solution of the resulting pressure equation assures that the velocity field satisfies the continuity equation. An iterative method based on a preconditioned conjugate gradient scheme of Van der Vorst [30] is used for solving the pressure equation. Based on the velocity field at the new time step, the advection equations (10) and (11) (for the combined level-set and volume-of-fluid approach also Eqs. (21) and (23)) are solved followed by a reconstruction of the interface.

The solution scheme described above is second order in space and first order in time. Time step restrictions exist for the present procedure owing to the advection algorithm and the explicit treatment of the convection term in Eq. (3). Of these two CFL constraints, the first one is more restrictive, which requires that the sum of the volume fluxed over the cell faces must be smaller than the total cell volume. This can be ensured by

$$\Delta t_c \leqslant \frac{h}{2\max(|\mathbf{v}|)}. \tag{36}$$

Furthermore, the explicit treatment of the surface tension term also results in a restriction as given in Ref. [14] as

$$\Delta t_s < \left[\frac{(\rho_1+\rho_2)h^3}{4\pi\sigma}\right]^{1/2}. \tag{37}$$

The time step is chosen to be smaller than Δt_c and Δt_s.

Finally, the three different algorithms compared in Section 4 have to be defined. Since the advection algorithm is the same for all approaches, only the interface reconstruction methods and the surface tension models are listed.

- K_8: LVIRA (interface reconstruction, Section 2.2.1), K_8 kernel (surface tension model, Section 2.3.1).
- *PROST*: Section 2.2.2 (interface reconstruction) and 2.3.2 (surface tension model).
- *CLSVOF*: Section 2.2.3 (interface reconstruction) and 2.3.3 (surface tension model).

3. Validation of the interface tracking algorithms

Before the three methods are applied to solve the Navier–Stokes equations including surface tension effects, the interface reconstruction and advection algorithms are validated. Two test cases were chosen from Rudman [25], namely the advection of a hollow square and a circle in shear flow. For the simulation of such test cases, the initial field of F is carefully provided and the initial interface is reconstructed. The velocity field is known in advance and is held constant with respect to time. The time cycle itself consists only of the advection sweeps of F (and ϕ for the CLSVOF method) followed by an interface reconstruction. Analytical solutions allow a rigorous check of the methods.

3.1. Advection of a hollow square

The configuration of this test case is a hollow square, which is aligned with the coordinate axes as given in Ref. [25] and outlined in Fig. 3. The square is exposed to a constant, unidirectional velocity field $\mathbf{v} = (2, 1)$. Using a mesh size of 200×200, the hollow square is advected 500 time steps with a cell Courant number (CFL) of 0.25. The methods tested are distinguished by their interface reconstruction method (Section 2.2), since the advection algorithm is identical in all methods. The accuracy of each method can be measured by the error E, given as

$$E = \frac{\sum_{i,j}|F^{\text{comp}}_{i,j} - F^{\text{exact}}_{i,j}|}{\sum_{i,j}F^0_{i,j}}, \tag{38}$$

where $F^{\text{comp}}_{i,j}$ and $F^{\text{exact}}_{i,j}$ are the computed and the exact void fraction after 500 time steps and $F^0_{i,j}$ is the initial distribution. The results are provided in Table 1. In addition to Rudman's value using his modified Youngs' method,

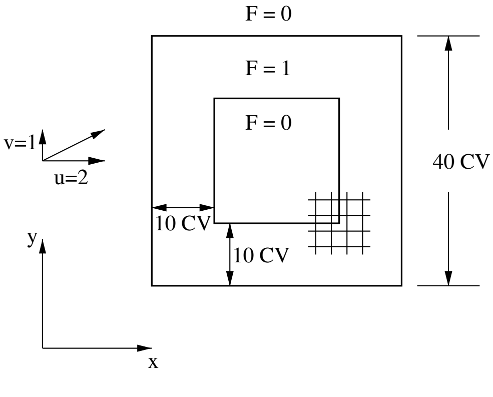

Fig. 3. Configuration of a hollow square aligned with the coordinate axes.

Table 1
Results of the hollow square after 500 time steps with CFL = 0.25

Method	E
Youngs	2.691×10^{-2}
LVIRA	3.340×10^{-2}
CLSVOF	2.410×10^{-2}
PROST	2.789×10^{-2}
Upwind O(1)	0.733
ENO O(2)	0.285
Youngs (Rudman)	2.58×10^{-2}

results of advecting F with a first- and second-(ENO) order upwind discretization scheme are added.

The present results are in good agreement with the literature value. A more detailed examination of the simulations shows that all methods face the problem of resolving the corners of the hollow square. This leads to rounding of the corners. At least for the LVIRA method, it is known that second-order accuracy can only be obtained if the interface is sufficiently smooth [7]. Hence improvements to Youngs' first-order method are small or the results are even slightly worse. As known in the literature, ordinary upwind discretization schemes are not appropriate for the present purpose.

3.2. Circle in shear flow

The second advection test represents a more realistic problem than the translation test in the previous section. In such translation test cases, the deformation of the interface is missing, i.e. the velocity field satisfies not only $\nabla \cdot \mathbf{v} = 0$, but also $\partial u/\partial x \neq 0$ and $\partial v/\partial y \neq 0$. This was pointed out by Rider and Kothe [31], showing that simple translation and rotation problems are not sufficient to verify the ability of interface tracking methods. Typically interfaces in real problems undergo strong topological changes, including merging and fragmentation. A test problem taking such shear effects into account is the circle placed in a single vortex [31]. In the present study, parameters of this test case are taken from Ref. [25].

The velocity field is defined by

$$u = \sin(x)\cos(y) \quad \text{and} \quad v = -\cos(x)\sin(y). \tag{39}$$

The computational domain has a square shape of edge length π resolved by 100×100 control volumes. A circle of radius $\pi/5$ is placed at $(\pi/2, \pi/4)$. In order to check the results with a known solution, the simulations are integrated in time $N = 1000$ or 2000 time steps forward followed by the same number of time steps backward, in which the signs of the velocity components (Eq. (39)) are reversed. Hence the final void fraction distribution should be exactly the same as the initial one. The error is calculated by Eq. (38). The mesh Courant number is 0.25.

The results of the error norm are provided for the two cases $N = 1000$ and 2000 in Table 2. The development of the interface for $N = 1000$ is depicted in Fig. 4 after $1000\Delta t$ forward and after $1000\Delta t$ forward and backward integrations. The interface represents the true interface computed by the interface reconstruction method, i.e. the midpoint of each linear segment is shown. Using contour lines of the void fraction field would mean to smooth the interface representation.

Table 2
Results of a circle in shear flow after $1000\Delta t$ and $2000\Delta t$ integrated forwards and backwards

Method	$E(N = 1000)$	$E(N = 2000)$
Youngs	8.389×10^{-3}	3.725×10^{-2}
LVIRA	6.550×10^{-3}	3.340×10^{-2}
CLSVOF	5.084×10^{-3}	2.648×10^{-2}
PROST	6.215×10^{-3}	3.191×10^{-2}
Youngs (Rudman)	8.60×10^{-3}	3.85×10^{-2}

The CFL number is 0.25.

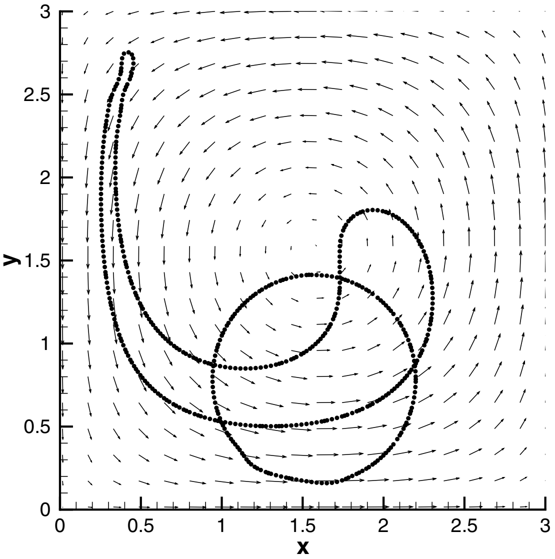

Fig. 4. Computed interfaces after $1000\Delta t$ forward and after $1000\Delta t$ forward and backward time integrations using the CLSVOF method.

The results of the present study prove the reliability of the interface capturing algorithms. It should be noted that the mass conservation, which can be defined as the change of liquid in the domain during computations as $\Delta V_1 = (\sum_{i,j} F^0_{i,j} - \sum_{i,j} F_{i,j})\Delta x \Delta y$, of the shear test case is smaller (10^{-4}) than that in the simple translation case (10^{-9}) for all methods. This effect of strong vorticity can be improved by grid refinement.

4. Results and discussion

4.1. Equilibrium rod

The comparison of the three surface tension methods described in Section 2.3 starts with the standard test case of a static rod. Since the gravity and other external forces are absent and an inviscid fluid is assumed, only the surface tension force is balanced by the pressure force. This results in a pressure jump at the interface given by the Laplace equation from zero outside the rod to the value of

$$P_{\mathrm{drop}} = \sigma\kappa = \frac{\sigma}{R}, \tag{40}$$

inside the rod, where R is the radius of the circle.

We chose a 6×6 cm domain, divided into equally spaced control volumes of $h = 0.2$, 0.1, and 0.05 cm (resolution of 30×30, 60×60 or 120×120). The rod of radius $R = 2$ cm has a density of $\rho_l = 1$ g/cm^3, placed at the point (3, 3) inside the background fluid of density $\rho_g = 0.5$ or 0.001 g/cm^3. The surface tension coefficient is given as 23.61 g/s^2. According to the Laplace equation (40), the pressure jump is 11.805 g/s^2. This exact value is compared with the computed pressure inside the drop determined by

$$\langle P \rangle = \frac{1}{N_{\mathrm{d}}} \sum_{i,j=1}^{N_{\mathrm{d}}} P_{i,j}, \tag{41}$$

where the sum is over N_{d} cells having a density $\rho \geqslant 0.99 \cdot \rho_l$. We can also estimate an L_1 and L_2 error norm given by

$$L_1 = \left| \frac{\sum_{i,j=1}^{N_{\mathrm{d}}} P_{i,j} - P_{\mathrm{drop}}}{N_{\mathrm{d}} P_{\mathrm{drop}}} \right| \quad \text{and}$$
$$L_2 = \left[\frac{\sum_{i,j=1}^{N_{\mathrm{d}}} (P_{i,j} - P_{\mathrm{drop}})^2}{N_{\mathrm{d}} P_{\mathrm{drop}}^2} \right]^{0.5}. \tag{42}$$

Ideally, the pressure field should be constant inside the drop and should be exactly balanced by the surface tension force at the interface. Hence any velocity field is an unphysical numerical artifact, which is known as "parasitic currents" [15]. A measure of the spurious currents is the maximal velocity in the computational field, which is given in the present work after one and 50 time steps ($\Delta t = 10^{-5}$ s).

First we recalculate a case considered by Brackbill et al. [14] using $\rho_g = 0.5$ g/cm, i.e. $\rho_l/\rho_g = 2$. The results are summarized in Table 3 for the medium grid resolution ($R/\Delta x = 20$). The value of Brackbill et al. [14] is their best result obtained by employing the unsmoothed density field at the interface for computing the interface normal vector and a smoothed field for the curvature. Furthermore, a convergence study was performed for the case of $\rho_l/\rho_g = 1000$. The radius of the rod was resolved by 10, 20 and 40 control volumes. The data are provided in Table 4.

The results show the excellent properties of the PROST method compared with the K_8 and CLSVOF methods, which can be clearly seen for the more demanding case of a high density ratio (Table 4). A convergence rate of about one or higher is found for PROST. Our experience is that the performance of the CLSVOF and K_8 methods are dependent on the width of the transition region, which is not the case for the PROST method. We chose a width of three control volumes for both methods (CLSVOF and K_8), which gave good results for the other test problems as well.

Also, it is very important to examine the development of the spurious currents resulting from inconsistent calculation of the surface tension force. Since we consider an inviscid fluid, inaccuracies of the surface tension force introduce

Table 3
Comparison of the surface tension models with the result of Brackbill et al. [14] using $\rho_l/\rho_g = 2$

Method	$\frac{\langle P \rangle}{P_{\mathrm{drop}}}$	L_2	L_1	$\lvert u_{\max,1} \rvert$	$\lvert u_{\max,50} \rvert$
$R/\Delta x = 20$, $\rho_l/\rho_g = 2$					
PROST	1.0007	3.02×10^{-3}	7.18×10^{-4}	1.14×10^{-7}	5.69×10^{-6}
K_8	0.9844	5.72×10^{-2}	1.56×10^{-2}	7.45×10^{-7}	3.59×10^{-5}
CLSVOF	0.9951	3.84×10^{-2}	4.83×10^{-3}	9.69×10^{-7}	4.71×10^{-5}
Brackbill	1.016	2.82×10^{-2}	–	–	–

Table 4
Convergence study of the surface tension models for a density ratio $\rho_l/\rho_g = 1000$ on three different grid refinement levels

$\frac{R}{\Delta x}$	$\frac{\langle P \rangle}{P_{\mathrm{drop}}}$	L_2	L_1	$\lvert u_{\max,1} \rvert$	$\lvert u_{\max,50} \rvert$
PROST, $\rho_l/\rho_g = 1000$					
10	1.0048	4.83×10^{-3}	4.81×10^{-3}	7.82×10^{-8}	3.91×10^{-6}
20	1.0009	9.79×10^{-4}	9.48×10^{-4}	1.70×10^{-7}	8.53×10^{-6}
40	1.00007	5.25×10^{-4}	7.04×10^{-5}	4.34×10^{-7}	2.17×10^{-5}
K_8, $\rho_l/\rho_g = 1000$					
10	0.9783	7.60×10^{-2}	2.17×10^{-2}	2.28×10^{-6}	1.12×10^{-4}
20	0.9867	4.90×10^{-2}	1.33×10^{-2}	5.33×10^{-6}	2.66×10^{-4}
40	0.9881	3.86×10^{-2}	1.19×10^{-2}	4.62×10^{-5}	2.31×10^{-3}
CLSVOF, $\rho_l/\rho_g = 1000$					
10	0.9886	4.99×10^{-2}	1.14×10^{-2}	1.16×10^{-6}	5.68×10^{-5}
20	0.9925	3.17×10^{-2}	7.53×10^{-3}	5.97×10^{-6}	2.92×10^{-4}
40	0.9971	2.22×10^{-2}	2.92×10^{-3}	1.91×10^{-5}	9.37×10^{-4}

an undamped velocity field in the vicinity of the interface. Again, the PROST methods minimize successfully the spurious currents compared with the other methods. A comparison of Tables 3 and 4 indicates the weak dependence of PROST on the density ratio compared with K_8 and CLSVOF. However, for all three methods the spurious currents increase with grid refinement.

4.2. Capillary wave

This test case deals with the damped oscillation of an interface between two viscous fluids without external forces. The interface has initially a cosine-shaped perturbation and the two fluids being quiescent. The parameters of the test are the same as given by Popinet and Zaleski [11], who used a different approach for interface tracking. In contrast to the volume-of-fluid methods used in the present study, which can be called an implicit or front-capturing method, they used an explicit or front-tracking method, because the interface is given explicitly by interface marker particles. Hence the capabilities of our methods can be directly compared with another approach well established in the literature [32,33]. Furthermore, an analytical solution based on the work of Prosperetti [34] serves as a measure of the absolute deviation compared with the true solution.

The computational domain is a square box of width $H = \lambda = 2\pi/K$, where λ and K are the wavelength and wavenumber, respectively. The viscosities and densities of the two fluids are the same, resulting in a non-dimensional viscosity $\epsilon = \mu K^2/(\rho\omega_0) = 6.472 \times 10^{-2}$, where ω_0 is defined by the dispersion relation $\omega_0^2 = \sigma K^3/(2\rho)$ of the linear theory of an interface oscillation between two inviscid fluids. The Ohnesorg number $\mathrm{Oh} = \mu/(\sigma\rho\lambda)^{1/2} = 1/\sqrt{3000}$ and the initial interface perturbation is $0.01H$.

The time evolution of the capillary wave is given in Fig. 5 for the three methods under consideration up to a non-dimensional time $\tau = t\omega_0 = 25$. The amplitude is normalized by the height of the initial perturbation. The results of the PROST and CLSVOF methods are close and can be better examined by calculating the relative error with the analytical solution (see Table 5). The error is defined by the rms of the difference between the normalized solutions. For comparison, the values of Popinet and Zaleski [11] for this test case are added. The PROST and CLSVOF methods are shown to perform well and the results are comparable to those of the front-tracking method of Popinet and Zaleski.

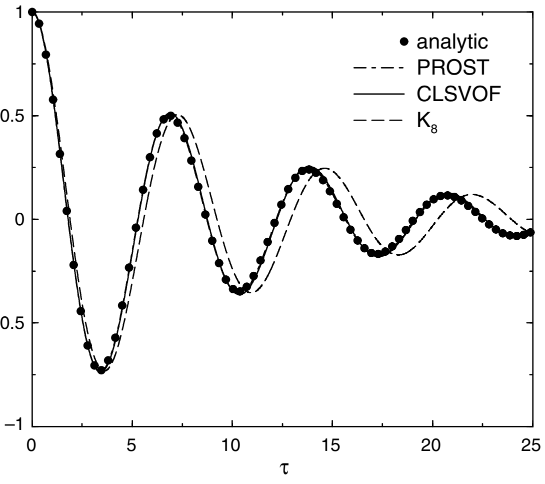

Fig. 5. Amplitude of the capillary wave in the viscous case compared with the analytical solution of Prosperetti [34]. The results of the PROST and CLSVOF methods are not distinguishable in this figure.

4.3. Rayleigh–Taylor instability

The instability of an interface between two fluid layers is considered when the heavier fluid lies over a lighter fluid and acceleration is directed from the heavier to the lighter fluid. If the interface is disturbed by a small perturbation, the instability grows exponentially as $\exp(nt)$ with time. Under the assumptions that the fluids are inviscid, incompressible and of infinite depth and the non-linear terms are small, Bellman and Pennington [35] extended the work of Rayleigh and Taylor by including surface tension in their analytical solution. The growth rate n was given by them as

$$n^2 = Kg\left[A - \frac{K^2\sigma}{g(\rho_1 + \rho_2)}\right], \tag{43}$$

where K is the wavenumber of the perturbation, g is the gravitational acceleration perpendicular to the interface, $A = (\rho_2 - \rho_1)/(\rho_1 + \rho_2)$ and ρ_1 and ρ_2 are the densities of the lighter and heavier fluid, respectively. Following Daly [36], it is helpful to introduce a ratio

$$\Phi = \frac{\sigma}{\sigma_c} = \frac{\sigma K^3}{(\rho_2 - \rho_1)g}, \tag{44}$$

where σ_c is a critical value of the surface tension coefficient for which $n = 0$. This ratio is a useful measure of the importance of the surface tension compared with

Table 5
Relative error between our volume-of-fluid methods and the analytical solution for different grid refinement levels

Grid	PROST	CLSVOF	K_8	Popinet and Zaleski
8^2	0.2960	0.3169	0.4542	0.2972
16^2	0.0818	0.0991	0.2433	0.0778
32^2	0.0069	0.0131	0.1095	0.0131
64^2	0.0018	0.0033	0.0456	0.0098

For comparison the results of Popinet and Zaleski [11] are added.

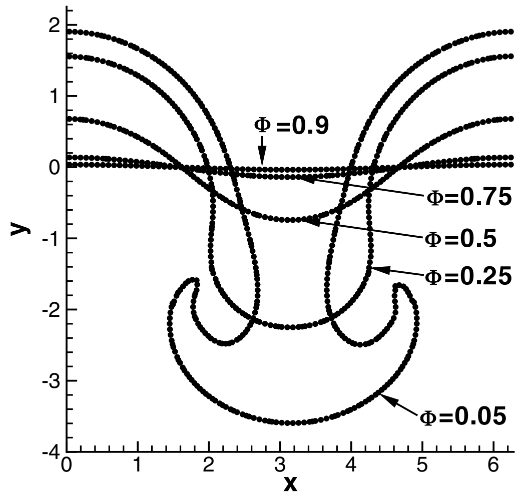

Fig. 6. Shape of the interface for different values of Φ depicted at $\tau = 10$.

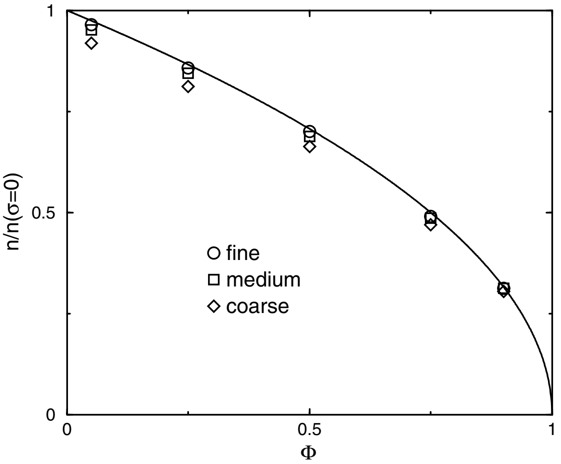

Fig. 7. Convergence study of the growth rate n and comparison with the analytical solution.

acceleration, implying instability for $\Phi < 1$. Numerical simulations are performed in the range $0.05 < \Phi < 0.9$. K is chosen to be unity, leading to a domain width of $L = (2\pi/K) = 2\pi$. The Atwood number is $A = 0.6$. For normalization, the characteristic length K^{-1} and time $n(\sigma = 0)^{-1}$ are used. The non-dimensional time is denoted by τ.

In the following, attempts have been made to examine the influence of the surface tension (Φ) on the evolution of the instability and the influence on the efficiency of the surface tension algorithms. The interface shapes for different values of Φ at the particular time instant $\tau = 10$ are depicted in Fig. 6 for computations on an 80×240 grid using the PROST method. It can be seen that the growth rate of the instability is significantly delayed if Φ is increased.

The growth rate n can be examined by a comparison with the analytical solution. Therefore, a dependence $n = f(\Phi)$ can be obtained by combining Eqs. (43) and (44). Numerically, the growth rate is calculated by observing the interface amplitude at the middle of the domain width with time. A linear region can be identified, when the logarithm of the amplitude is plotted, which will give the growth rate by applying linear regression. This was done for the three algorithms for $\Phi = 0.05$, 0.25, 0.5, 0.75 and 0.9. The results using the PROST method for three different grid levels (20×60, 40×120 and 80×240) are compared with the analytical solution in Fig. 7. The agreement with the analytical curve is excellent and the results can be seen to be improved compared with those of Brackbill et al. [14].

The three methods can be compared more accurately by calculation of the error between the computed and analytical results as provided in Table 6. For high values of Φ $(0.75, 0.9)$, the K_8 method diverges or fails to predict the growth rate of the Rayleigh–Taylor instability. This can be ascribed to the strong spurious currents occurring at the interface in the case of the K_8 method, as already shown by the equilibrium rod test (Section 4.1). The resulting interface oscillations disturb the evolution of the interface. However, for the remaining values of Φ converging results were obtained. Additionally, an increase in the error with increasing Φ for the CLSVOF and K_8 methods can be identified. In contrast, the behavior of the PROST method can be seen to be independent of Φ.

Table 6
Error of the growth rate n between the numerical and analytical results for the three methods considered (PROST, CLSVOF, K_8) depending on the relative importance of the surface tension Φ

	Grid	$\Phi = 0.05$	$\Phi = 0.25$	$\Phi = 0.5$	$\Phi = 0.75$	$\Phi = 0.9$
PROST	20×60	5.7%	6.2%	6.1%	6.0%	3.5%
	40×120	2.3%	2.4%	2.7%	2.7%	0.9%
	80×240	1.0%	1.0%	0.9%	1.9%	0.9%
CLSVOF	20×60	7.4%	7.7%	8.5%	10.1%	15.3%
	40×120	2.7%	3.4%	4.4%	5.2%	7.5%
	80×240	0.8%	1.5%	2.1%	2.7%	3.5%
K_8	20×60	8.7%	8.1%	9.1%	1.4%	26.0%
	40×120	3.6%	3.4%	3.8%	2.2%	26.0%
	80×240	1.5%	1.0%	2.1%	3.0%	29.0%
Exact n		2.365	2.101	1.716	1.213	0.767

5. Bubble formation at an underwater orifice

The last problem considered here is of bubble formation and its detachment from an underwater orifice. The formation process of gas bubbles from orifices is of importance in many industrial applications. In the case of inter-phase transport processes, one of the aims is to increase the interfacial area between the two phases to obtain greater transfer rates. Thus the bubble size distribution produced at the orifices is of interest. In this section the combined level-set and volume-of-fluid method is applied to simulate the bubble formation process at a single orifice with the aims, firstly, to demonstrate the capability of the CLSVOF method to simulate engineering problems and, secondly, to present a test case for the comparison and validation between the numerical and analytical results for the bubble formation at low flow rates.

For the present case of very low flow rate ($\dot{Q} = 1\,\mathrm{ml/min}$), the formation can be predicted analytically by means of the solution of the Young–Laplace equation, as it was done, for example, in [20]. Thus the formation is considered to happen quasi-static, so that the force balance at the bubble interface consists only of capillary and pressure forces. The resulting ordinary differential equation still contains the radius of curvature at the bubble apex R_0 as a free parameter and has to be solved numerically. The result of the solution of the Young–Laplace equation is an axisymmetric bubble shape and thus its volume can be determined for the given boundary conditions. The complete evolution of a static formation can be predicted by solving the differential equation for varying values of R_0 in such a way that the bubble volume and height increases in each step consecutively. This procedure can be continued until the largest bubble volume in equilibrium is found under the given boundary conditions [20]. It may be mentioned that after this stage, the bubble detaches dynamically. For the present theoretical calculations, the method described in [21] has been applied and a good agreement was found between the analytical and experimental results for the quasi-static bubble formation.

For the numerical simulation of the bubble formation, the CLSVOF method (Section 2) was transformed to an axisymmetrical (r, z) coordinate system. Water and air at 20 °C were chosen as the working fluids for the present calculations, i.e. the density ratio is about 1000. At the orifice of diameter 3 mm the gas flows with a defined parabolic velocity profile. A flow rate of 1 ml/min into the computational domain is considered. Symmetry boundary conditions are defined at the sidewalls and outflow conditions at the top boundary. The bubble is forced to be attached to the orifice rim. Variable time stepping was employed to overcome the problem of different time scales during quasi-static formation and highly dynamic detachment.

In Fig. 8 the comparison between the CLSVOF simulation and the analytical bubble shapes is shown. For the CLSVOF method the bubble contour is given by dots, where each dot corresponds to the midpoint of a line segment in a two-phase cell, and the analytical solution is represented by a few points of the calculated contour (circles). The time difference between the shapes shown is 0.5 s , except the last one, which presents the final equilibrium shape obtainable with the analytical approach. From this time instant, the formation continues dynamically, as shown by the further results of the CLSVOF method in Fig. 9, where a time gap of about 0.01 s persists between the figures in sequence.

In order to calculate theoretically the bubble volume after the detachment, one typically takes the neck volume (bubble volume above the bubble neck) of the last static bubble [20], since it cannot be assumed that the gas above the orifice detaches completely (see, e.g., Fig. 9). The theoretically predicted time period for detachment at the 3 mm orifice is 3.06 s. The time period measured by the CLSVOF method is 2% higher. Further comparisons have been done for a 2 mm orifice, where the same level of accuracy has been obtained.

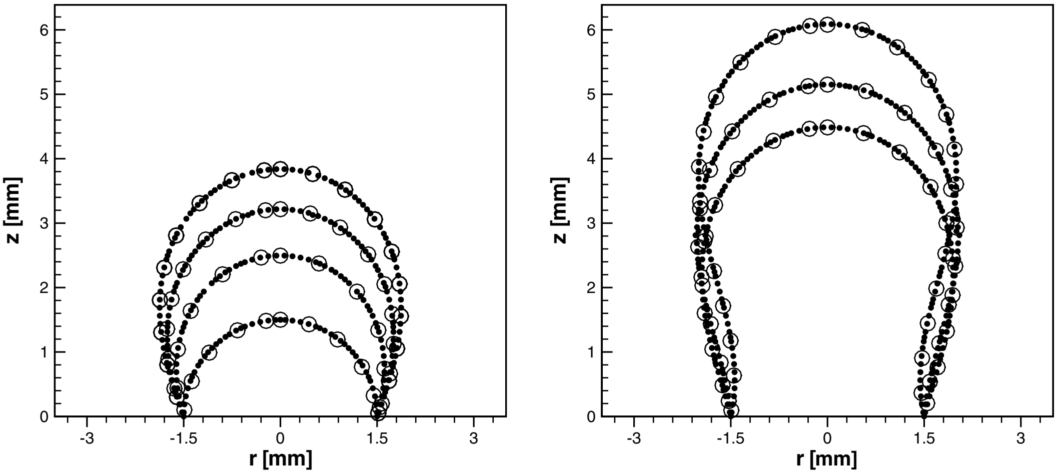

Fig. 8. Comparison of the bubble shapes at formation between the CLSVOF method (dots) and the analytical solution (open circle).

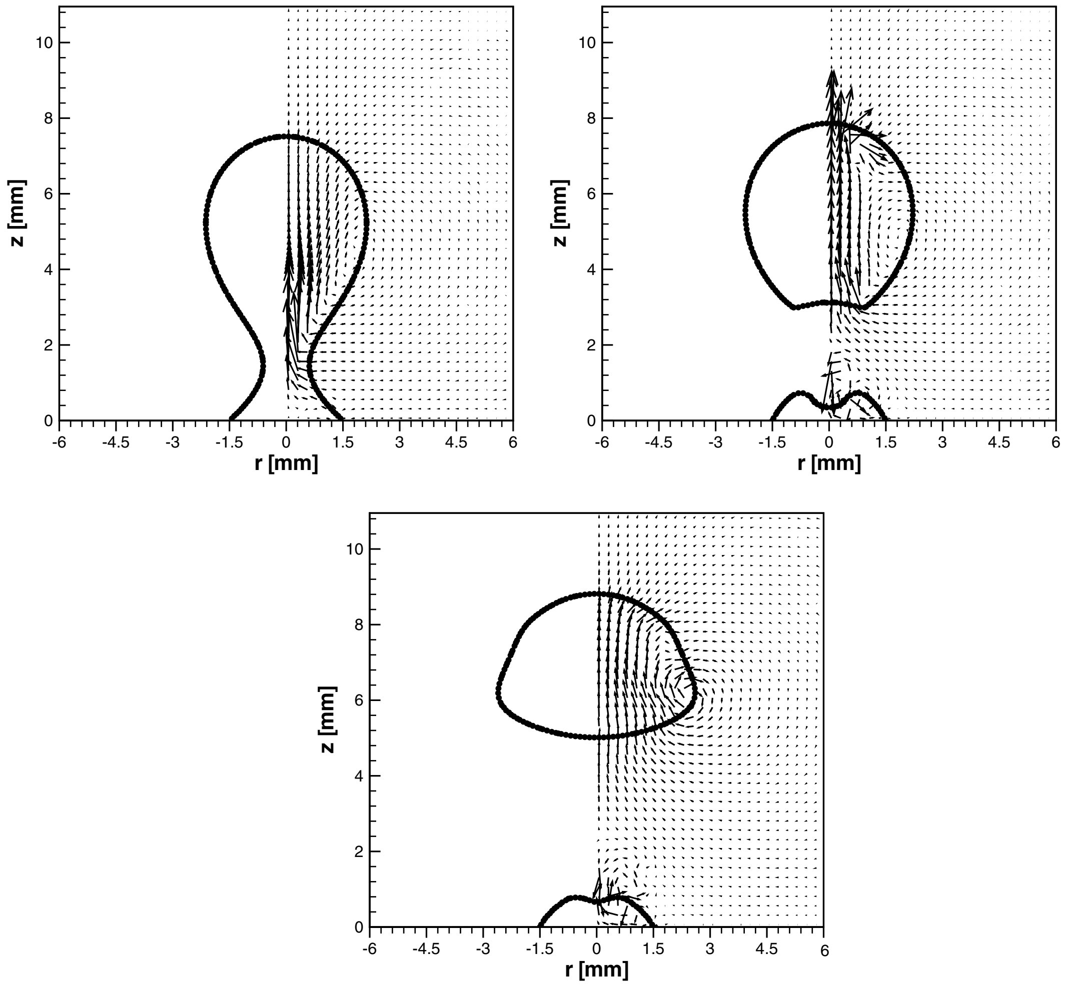

Fig. 9. Detachment of a bubble computed with the CLSVOF method.

The agreement between the theory and simulation confirms the quality of the present CLSVOF approach also for high density ratios.

6. Conclusions

Two-dimensional numerical simulations of an equilibrium rod, a capillary wave and the Rayleigh–Taylor instability were performed by means of three different volume-of-fluid algorithms. The three methods considered are an improved version of the method of Brackbill et al. [14] due to Williams et al. [16], a combined level-set and volume-of-fluid approach based on the work of Sussman et al. [17] and Son [18,19] and a method using a piecewise-parabolic representation of the interface for the surface tension and the interface normal calculation due to Renardy and Renardy [10]. The methods are named K_8, CLSVOF and PROST, respectively. The accuracy, the convergence behavior and the influence of “parasitic currents”, which are a consequence of inaccurate calculations of the surface tension force, were examined in detail for each method.

The study revealed that the PROST method is superior to the CLSVOF and K_8 methods. High accuracy and a successful reduction of the parasitic currents compared with the other two methods feature this approach. However, the optimization of the least-squares function (Eq. (18)) is computationally intensive, and from this perspective, CLSVOF presents a true alternative to PROST. The implementation of the CLSVOF method is straightforward and the algorithm is the fastest considered here. Recent improvements [37] of this method are highly promising. However, the results obtained here are not as good as for the PROST method, but might be sufficient for a large number of applications. In contrast, the K_8 method was demonstrated to have weaknesses in the surface force calculation. The parasitic currents at the interface led to diverging or wrong results for the case of the Rayleigh–Taylor instability (Table 6).

Finally, the present CLSVOF method was used to simulate the slow formation of a gas bubble at an underwater orifice. The calculations have been done for the water–air system, for which the density ratio is about 1000. The low flow rate considered here allows a comparison of the computed bubble shapes during formation with an analytical solution. Good agreement was found between the numerically calculated bubble contours with their analytical counterpart.

Acknowledgement

One of the authors (D.G.) gratefully acknowledges the support received from the Bayerische Forschungsstiftung, Germany.

References

[1] S. Osher, L.A. Sethian, Fronts propagating with curvature-depending speed: algorithms based on Hamilton–Jacobi formulation, J. Comput. Phys. 79 (1988) 12–49.

[2] 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.

[3] S. Osher, R.P. Fedkiw, Level set methods: an overview and some recent results, J. Comput. Phys. 169 (2001) 463–502.

[4] W.F. Noh, P.R. Woodward, SLIC (simple line interface calculation), in: A. van der Vooren, P. Zandberger (Eds.), Lecture Notes in Physics, vol. 59, Springer-Verlag, New York/Berlin, 1976, p. 330.

[5] C.W. Hirt, B.D. Nichols, Volume of fluid (VOF) method for the dynamics of free boundaries, J. Comput. Phys. 39 (1981) 201–225.

[6] D.L. Youngs, Time-dependent multi-material flow with large fluid distortion, in: K. Morton, M. Baines (Eds.), Numerical Methods for Fluid Dynamics, Academic Press, New York, 1982, pp. 273–285.

[7] E.G. Puckett, A.S. Almgren, J.B. Bell, D.L. Marcus, W.J. Rider, A high-order projection method for tracking fluid interfaces in variable density incompressible flows, J. Comput. Phys. 130 (1997) 269–282.

[8] S.W.J. Welch, T. Rachidi, Numerical computation of film boiling including conjugated heat transfer, Numer. Heat Transfer, Part B 42 (2002) 35–53.

[9] D.K. Agarwal, S.W.J. Welch, G. Biswas, F. Durst, Planar simulation of bubble growth in film boiling in near-critical water using a variant of the VOF method, J. Heat Transfer (ASME) 126 (2004) 329–338.

[10] Y. Renardy, M. Renardy, PROST: a parabolic reconstruction of surface tension for the volume-of-fluid method, J. Comput. Phys. 183 (2002) 400–421.

[11] S. Popinet, S. Zaleski, A front-tracking algorithm for accurate representation of surface tension, Int. J. Numer. Meth. Fluids 30 (1999) 775–793.

[12] M. Rudman, A volume-tracking methods for incompressible multifluid flows with large density variations, Int. J. Numer. Methods Fluids 28 (1998) 357–378.

[13] D. Jamet, D. Torres, J.U. Brackbill, On the theory and computation of surface tension: the elimination of parasitic currents through energy conservation in the second-gradient method, J. Comput. Phys. 182 (2002) 262–276.

[14] J.U. Brackbill, D.B. Kothe, C. Zemach, A continuum method for modeling surface tension, J. Comput. Phys. 100 (1992) 335–354.

[15] B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, G. Zanetti, Modeling merging and fragmentation in multiphase flows with SURFER, J. Comput. Phys. 113 (1994) 134–147.

[16] M.W. Williams, D.B. Kothe, E.G. Puckett, Accuracy and convergence of continuum surface-tension models, in: W. Shyy, R. Narayanan (Eds.), Fluid Dynamics at Interfaces, Cambridge University Press, Cambridge, 1998, pp. 294–305.

[17] M. Sussman, E.G. Puckett, A coupled level set and volume-of-fluid method for computing 3D and axisymmetric incompressible two-phase flows, J. Comput. Phys. 162 (2000) 301–337.

[18] G. Son, N. Hur, A coupled level-set and volume-of-fluid method for the buoyancy-driven motion of fluid particles, Num. Heat Transfer B 42 (6) (2002) 523–542.

[19] G. Son, Efficient implementation of a coupled level-set and volume-of-fluid method for 3-dimensional incompressible two-phase flows, Numer. Heat Transfer B 43 (6) (2003) 549–565.

[20] M.S. Longuet-Higgins, B.R. Kerman, K. Lunde, The release of air bubbles from an underwater nozzle, J. Fluid Mech. 230 (1991) 365–390.

[21] D. Gerlach, G. Biswas, F. Durst, V. Kolobaric, Quasi-static bubble formation on submerged orifices, Int. J. Heat Mass Transfer 48 (2005) 425–438.

[22] F.H. Harlow, J.E. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface, Phys. Fluids 8 (12) (1965) 2182–2189.

[23] W.J. Rider, D.B. Kothe, Reconstructing volume tracking, J. Comput. Phys. 141 (1998) 112–152.

[24] G. Strang, On the construction and comparison of difference schemes, SIAM J. Numer. Anal. 5 (1968) 506–517.

[25] M. Rudman, Volume-tracking methods for interfacial flow calculations, Int. J. Numer. Methods Fluids 24 (1997) 671–691.

[26] D.B. Kothe, W.J. Rider, S.J. Mosso, J.S. Brock, J.I. Hochstein, Volume tracking of interfaces having surface tension in two and three dimensions, AIAA Paper 96-0859.

[27] K. Deb, Optimization for Engineering Design, Prentice-Hall, New Delhi, 1995.

[28] Y.C. Chang, T.Y. Hou, B. Merriman, S. Osher, A level set formulation of Eulerian interface capturing methods for incompressible fluid flows, J. Comput. Phys. 124 (1996) 449–464.

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

[30] H.A.V. der Vorst, BI-CGSTAB: A fast and smoothly converging variant of BI-CG for the solution of nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. 13 (2) (1992) 631–644.

[31] W.J. Rider, D.B. Kothe, Stretching and tearing interface tracking methods, AIAA Paper 95-1717.

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

[33] D. Juric, G. Tryggvason, Computations of film boiling, Int. J. Multiphase Flow 24 (3) (1998) 387–410.

[34] A. Prosperetti, Motion of two superimposed viscous fluids, Phys. Fluids 24 (7) (1981) 1217–1223.

[35] R. Bellman, R.H. Pennington, Effects of surface tension and viscosity on Taylor instability, Quart. Appl. Meth. 12 (1954) 151–162.

[36] B.J. Daly, Numerical study of the effect of surface tension on interface instability, Phys. Fluids 12 (7) (1969) 1340–1354.

[37] M. Sussman, A second order coupled level set and volume-of-fluid method for computing growth and collapse of vapor bubbles, J. Comput. Phys. 187 (2003) 110–136.