Available online at www.sciencedirect.com
ScienceDirect
European Journal of Mechanics B/Fluids 27 (2008) 1–23

An improved method for calculation of interface pressure force in PLIC-VOF methods

M. Seifollahi [a], E. Shirani [a], N. Ashgriz [b,*]

[a] *Department of Mechanical Engineering, Isfahan University of Technology, Isfahan, Iran, 84156*
[b] *Department of Mechanical & Industrial Engineering, University of Toronto, Toronto, Ontario, Canada, M5S 3G8*

Received 29 August 2005; received in revised form 12 July 2006; accepted 11 January 2007

Available online 21 February 2007

Abstract

A method for the application of interface force in the computational modeling of free surfaces and interfaces which uses PLIC-VOF methods is developed. This method is based on evaluation of the surface tension force only in the interfacial cells with out using the neighboring cells. The normal and the interface surface area needed for the calculation of the surface tension force are calculated more accurately. This method is applied on a staggered grid and it is referred to as Staggered Grid Interfere Pressure calculation method or SGIP. The present method is applied to a two-dimensional motionless liquid drop and a gas bubble. In addition, oscillations of a non-circular two-dimensional drop and a bubble due only to the surface tension forces are considered. It is shown that the new method predicts the pressure jump at the interface more accurately and produces less spurious currents compared to CSF, CSS and Meier's methods when applied to the same cases.

Keywords: Volume-of-Fluid; Two-phase flow; Continuum surface force; Continuum surface stress; Interface; Surface tension force; Free surface flows; Spurious currents; Parasitic currents

1. Introduction

One of the challenges in numerical simulation of interfacial flows is modeling of the surface tension forces. In Piecewise Linear Interface Calculation (PLIC) methods using the Volume of Fluid (VOF) technique [1–8], special techniques are developed to calculate the surface tension forces, which are later converted to a body force. The more commonly used surface tension treatment methods are the Continuum Surface Force method (CSF) [9–18], the Continuum Surface Stress method (CSS) [19,20] and Meier's method [21,22]. One of the issues in the numerical modeling of surface tension forces is the production of so-called "spurious" or "parasitic" currents. These are small but growing vortical flows, which are generated due to the imbalance between stresses in the interfacial region. Spurious currents affect the interface shape, producing unphysical results. The main source for the production of spurious currents is the discretization errors such as those used to calculate the interface curvatures. The currents are produced mainly in the light fluid side near the interface, where the unbalanced stresses cause the light fluid to accelerate.

* Corresponding author. Tel.: +(416) 946 3408; fax: +(416) 978 7753.
E-mail address: ashgriz@mie.utoronto.ca (N. Ashgriz).

0997-7546/$ – see front matter
doi:10.1016/j.euromechflu.2007.01.002

Surface tension models cannot accurately calculate the pressure jump across the interface when the spurious currents grow.

In the Continuum Surface Force (CSF) model, the surface tension force is calculated by converting the surface tension effect into an equivalent volume force. This force is then added to the Navier–Stokes equations as an additional body force. This force has smoothing properties and acts only in a finite transition region across the interface. Note that the transition region is the region which contains the interfacial cells and their immediate neighboring cells. The CSF model reformulates surface tension into an equivalent volume force F^{st} as follows:

$$F^{\mathrm{st}}(x) = \sigma \int_S \kappa \mathbf{n} \delta(\mathbf{x} - \mathbf{x}_s)\, \mathrm{d}S, \tag{1}$$

where σ is the coefficient of the surface tension, κ is the surface curvature, $\mathbf{n}$ is the unit normal to the surface with the outward direction regarded as positive, $\delta(\mathbf{x} - \mathbf{x}_S)$ is the Dirac delta function, and $\mathbf{x}_S$ are points on the interface S. The integration is performed over the free surface area S. Surface tension is then incorporated into the flow equations simply as a component of the body force.

The original discretization of F^{st} proposed by Brackbill et al. [9] led to the formation of "spurious" currents. Brackbill and Kothe [10] showed that the original CSF formulation produces a vorticity source term and concluded that these currents will disappear as the transition region approaches zero. In an effort to reduce these effects, Aleinov and Puckett [23] suggested another formulation of F^{st}. The surface force per unit volume is computed only within each surface cell and is placed at the center of the cell:

$$F^{\mathrm{st}}_{i,j,k} = \sigma \kappa_{i,j,k} \frac{A_{i,j,k}}{V_{i,j,k}} n_{i,j,k}, \tag{2}$$

where $A_{i,j,k}$ is the surface area of the interface within the cell, $n_{i,j,k}$ is unit vector normal to the surface and $(V_{i,j,k})$ is the volume of the cell denoted by i, j, k. This force is then smoothed by convoluting it with a smoothing kernel, K:

$$\tilde{F}^{\mathrm{st}}_{i,j,k} = K F^{\mathrm{st}}_{i,j,k}. \tag{3}$$

Lafaurie et al. [19] converted the volumetric force used in the CSF method into stress form, the so-called continuum surface stress, CSS, method. In the CSS model, effects of capillary forces are presented as a stress tensor, which is tangential to the interface. The CSS method also produces numerical spurious effects as stated in [19]. In fact, neither CSF nor CSS models produce very accurate numerical solutions in capillary dominated fluid problems. In problems where the surface tension forces dominate the viscous forces, the spurious currents can cause interface oscillations and deform or destroy the interface.

There have been some attempts to reduce the spurious currents. Popinet and Zaleski [20] reduced the spurious currents considerably using a front capturing algorithm for the solution of two-dimensional flows. They used a Lagrangian advection marker to advect the interface and calculated the pressure force according to the location of the interface at each cell face. Two values for the pressure at each interfacial cell face are determined using the pressure at the two neighboring cells. They have also added a new source term in the Poisson equation to accurately calculate the pressure. Their method is more accurate when calculating the surface tension force and the associate pressure jump (see [20] for more details). Their work is limited to the markers method.

Meier et al. [21] and Meier [22] developed a new method to reduce the spurious currents by improving the interface curvature calculation. They used an estimator function, which is tuned with a least-squares-fit against reference data. They also calculated the surface tension force within each cell. By means of some weighted averaging, they calculated the surface tension force in the x- and y-momentum cells in their staggered grid configurations. They indicated that their method is three to seven times more accurate than the CSF method of Brackbill et al. [9]. They have been able to reduce the intensity of spurious currents by up to two orders of magnitude. Although their method calculates the interface curvature more accurately, the calculated pressure jump across the interface is still not accurate.

Renardy and Renardy [24] introduced another VOF based algorithm (referred to as PROST) for the calculation of the body force due to the surface tension. The advection of the volume fraction in their method is based on a Lagrangian scheme that allows no diffusion and produces a sharp interface. They used a least-square fit of a quadratic surface to the volume fraction function for each interface and its neighbors. No volume fraction smoothing was needed in their model. They were able to reduce the spurious currents with some success.

Jamet et al. [25] introduced a model to eliminate the parasitic currents through the conservation of energy in the second-gradient method. Therefore, by using the second-gradient method and the reduction of the truncation error in the computation of the energy exchanges between the surface and the kinetic energies, the energy is conserved and the parasitic currents are reduced drastically. Their method requires solution of an extra set of equations for the surface energies, which adds to the computational cost.

Shirani et al. [26] have introduced a new method for the interface pressure calculation based on the interface location, referred to as PCIL. In this method, first the intersection locations of an interface with the interfacial cell faces are determined. The area of a cell face, which is in contact with the heavier fluid, is normalized with the cell face area to obtain a factor H. Then, the two values of H's obtained from the cells on both sides of each internal cell face are averaged. Finally, the capillary force used in the momentum equation is corrected by multiplying it by the factor H determined for the cell. The new interface volumetric pressure force defined for each cell is then obtained from the following equation:

$$F^{\mathrm{st}} = H\sigma\kappa n\delta_s, \tag{4}$$

where δ_s is the Dirac delta function. This method was applied to some sample problems and it was shown that the velocities of the spurious currents decrease by more than two orders of magnitude compared with when the original CSS and CSF model were implemented. Variation of the velocity of the spurious currents with some non-dimensional parameters such as Ohnesorge number, density and viscosity ratios of the two fluids involved, were examined. It was shown that unlike the other methods used for modeling the surface tension force, in this method increasing the density ratio does not affect the level of spurious currents. The pressure jump obtained by this method is accurately calculated and only minor errors are produced when it is applied to concave surfaces such as a droplet. However, the method does not accurately calculate the pressure jump for convex surfaces such as a bubble, and similar issues as occurred in CSF method when the density correction factor is used, occurs in the PCIL method.

Finally, Francois et al. [18] proposed a new method for modeling surface tension which imposes an exact balance between pressure and surface tension forces. In this method, using a two-step projection method, the surface tension and pressure forces are both evaluated consistently at time $n + 1$. Volume tracking is done at the beginning of each timestep. Therefore, interface curvature and surface tension force become available at time $n + 1$. Then, the Poisson equation for pressure, where the surface tension force appears as a source term, is solved after evaluating the convective and viscous terms. Knowing the new pressure field, both surface tension and pressure forces are applied in a consistent manner. This method is used in continuous and sharp surface tension force (CSF and SSF) models. In CSF model, the surface tension force is evaluated and applied on the cell faces where the gradient of VOF function is non-zero. Note that the VOF function is not smoothed. However, in SSF model, the surface tension force is applied only on the cell faces across which the level set (or distance) function changes sign. The two models differ in the spatial distribution of the pressure jump across the interface. Francois et al. [18] showed that the CSF model is first-order accurate in pressure, whereas the SSF model is second-order. Also, both models yield spurious currents of the same order, which are machine-zero (1.e–15) when the exact interface curvature is imposed. They also found that errors in curvature estimates is the origin of spurious currents, regardless of the surface tension model employed. Their method is developed for collocated grid arrangement, whereas most of the presently used VOF based algorithms use staggered grid arrangement.

Here, we present a method for the calculation of the surface tension force based on a volume force calculation at each x- and y-momentum interfacial cell centers. This force is zero in non-interfacial cells. The interface in each of the cells is assumed to be a plane surface in three-dimensional cases (or a straight line in two-dimensional cases). This method is applied on a staggered grid and it is referred to as Staggered Grid Interface Pressure calculation method or SGIP. The new method is then tested on the time evolution of a two-dimensional static drop, static bubble, and oscillating non-circular drop and bubble. It is shown that this method can correctly calculate the pressure jump for any interface, and produce less spurious currents. The method presented in this paper is similar to Meier et al.'s [21] method except that forces in the two neighboring cells are not averaged in order to obtain the force at the centers of x- and y-momentum cells. In addition, the value of the normal vector to the interface is determined more accurately.

2. Problem formulation

We consider the two-dimensional, unsteady, incompressible Navier–Stokes equations. A volume-of-fluid (VOF) method along with a piecewise linear interface calculation (PLIC) is used to capture the fluid interfaces. It is assumed that the velocity field is continuous across the interface, but there is a pressure jump at the interface due to the presence of the surface tension. The governing equations describing this problem are:

$$\frac{\partial u_i}{\partial x_i} = 0, \tag{5}$$

$$\frac{\partial C}{\partial t} + u_i \frac{\partial C}{\partial x_i} = 0, \tag{6}$$

and

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_i u_j}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \mathrm{F}_{\mathrm{v}}^{\mathrm{st}} \cdot \hat{i} + \mu \frac{\partial^2 u_i}{\partial x_j^2} + \rho g_i, \tag{7}$$

where u_i's are the velocity components, and t and x_i are time and space coordinates, C is the volume fraction of fluid, which is zero where only one fluid exists and it is one where only the other fluid exists, p is the pressure, $\hat{i}$ is the unit vector in the ith direction, $\mathrm{F}_{\mathrm{v}}^{\mathrm{st}}$ is the surface tension force per unit volume and ρ and μ are the mixture density and absolute viscosity, respectively, and they depend on the densities and viscosities of each fluid as:

$$\rho = \rho_2 + C(\rho_1 - \rho_2), \tag{8}$$

and

$$\mu = \mu_2 + C(\mu_1 - \mu_2). \tag{9}$$

Here, ρ and μ are density and viscosity of fluids, respectively and the subscripts 1 and 2 denote the two fluids involved.

Here, the Poisson equation is solved to determine the pressure distribution. The Poisson equation is obtained by taking the divergence of the momentum equation, Eq. (7), and simplifying it using the continuity equation, Eq. (5). The resulting equation is:

$$\frac{\partial}{\partial x_i}\left(\frac{\partial p}{\partial x_i}\right) = -\frac{\partial}{\partial x_i}\left[\frac{\partial}{\partial x_j}\left(\rho u_i u_j - 2\mu S_{ij} - \frac{\mathrm{F}_{\mathrm{v}}^{\mathrm{st}} \cdot \hat{i}}{\rho}\right)\right]. \tag{10}$$

Following Brackbill et al.'s [9] continuum surface force (CSF) model, the interface curvature and the surface tension may be calculated as:

$$\kappa(\mathbf{x}) = -\nabla \cdot \tilde{\mathbf{n}}(\mathbf{x}) = -\nabla \cdot \frac{\nabla \tilde{C}(\mathbf{x})}{|\nabla \tilde{C}(\mathbf{x})|}, \tag{11}$$

and

$$\mathbf{F}_v^{\mathrm{st}} = \sigma \kappa \delta_s \mathbf{n} = \sigma \kappa \mathbf{n} \frac{|\nabla \tilde{C}|}{[C]}, \tag{12}$$

where the tilda denotes the filtered (smoothed) value, the square brackets denote the difference between the maximum and the minimum values of the function inside the brackets, and F_v^{st} is the volumetric surface tension force. The above model produces an artificial acceleration in the lighter fluid when the density ratio of the two fluids is large. This acceleration is the main source of producing spurious currents. Brackbill et al. [9], as well as Kothe et al. [27] recommended the addition of a density scaling factor in order to reduce the formation of such acceleration. Therefore, they proposed the following equation instead of Eq. (12):

$$\mathbf{F}_v^{\mathrm{st}} = \sigma \kappa \delta_s \mathbf{n} = \sigma \kappa \mathbf{n} \frac{|\nabla \tilde{C}|}{[C]} \frac{\rho(x)}{[\rho]}, \tag{13}$$

where $\rho(x)$ is the local value of the density obtained by Eq. (8) and $[\rho]$ is the difference between the density of the heavier and the lighter fluids. The density correction term (the second fraction in Eq. (13)) is added to correct the force in the momentum equation. This dampens the acceleration of the lighter fluid in the cells near the interface

that contain small amounts of heavier fluid. The newly added fraction is not directly obtained from any conservation law, but is only postulated. Although this fraction decreases the acting force in the lighter fluid and thus reduces the spurious currents, it causes inaccurate calculation of the pressure jump when the jump in pressure is in the lighter fluid such as that in a bubble. We will examine the effectiveness of this term later and will compare it with our model.

Another model which is widely used is that of Zaleski's Continuous Surface Stress (CSS) model [19,28–30]. In this model, Eq. (13) is replaced with:

$$\mathbf{F}_v^{\mathrm{st}} = -\nabla \cdot \mathbf{T} = \sigma \nabla \cdot \left(|\nabla C| \mathbf{I} - \frac{\nabla C \otimes \nabla C}{|\nabla C|} \right), \tag{14}$$

where $\otimes$ is a tensor product.

Meier et al. [21] and Meier [22] calculated the surface tension force within each cell. To do this, they used Eq. (2) to calculate the force at the center of each cell. Here we follow their notations and formulations which were given for two-dimensional case. They divided Eq. (2) by the density to obtain the body force in the cell:

$$\mathbf{e}_{i,j} = \frac{\sigma}{\rho_{i,j}} \kappa_{i,j} \frac{A_{i,j}}{V_{i,j}} n_{i,j}. \tag{15}$$

Since they used a staggered-mesh layout, they portioned $\mathbf{e}_{i,j}$ on the staggered momentum control volumes by means of some weighted averaging. In Fig. 1, i and $i+1$ are the centers of the continuity and volume fraction cells and $i+1/2$ is the center of the x-momentum cell. Meier [22] calculated the x-component of the body force in cell i, $e_{x,i,j}$, and in cell $i+1$, $e_{x,i+1,j}$. The body forces for the x-momentum cell $i+1/2$ is calculated from the following relation:

$$e_{x,i+1/2,j} = (0.5 + O_{x,i,j}) e_{x,i,j} + (0.5 - O_{x,i+1,j}) e_{x,i+1,j}, \tag{16}$$

O_x and O_y, shown in Fig. 1, are the x and y components of the offset vector of the interface center from the cell center. He used a similar formulation to calculate the body force in the y-direction, $e_{y,i,j+1/2}$.

For Meier's formulation, since $\mathbf{e}_{i,j}$ is calculated using the density at the center of the volume fraction cells, the acceleration depends on the value of C at the cell centers. For small values of C_{ij}, the values of $\mathbf{e}_{i,j}$ are large and thus strong parasitic velocities are induced. It should be noted that Meier also introduced another variant for the density calculation which we did not use in our work. In this variant, he suggested to use the average density instead of the cell density. The latter method produces less spurious currents but it is non-conservative (Meier recommended the first method in his thesis [22]).

To resolve the difficulties of CSF, CSS and Meier's methods, we have developed a new method to include surface tension in a way that keeps spurious currents low and calculates pressure differences more accurately. We will first present our model for three-dimensional cases and for staggered grids. We will then simplify the method for two-dimensional cases and provide more details for the calculation of their parameters. Finally, we will present some results on the application of the new method in two-dimensional drops and bubbles.

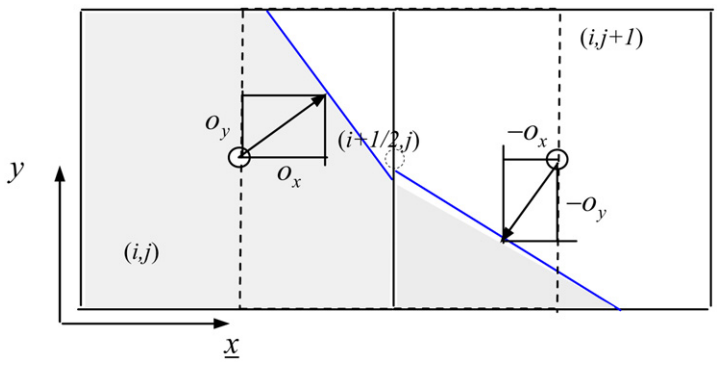

Fig. 1. Offset vectors O's used to weight the partition of the surface tension body force on staggered x-momentum cells.

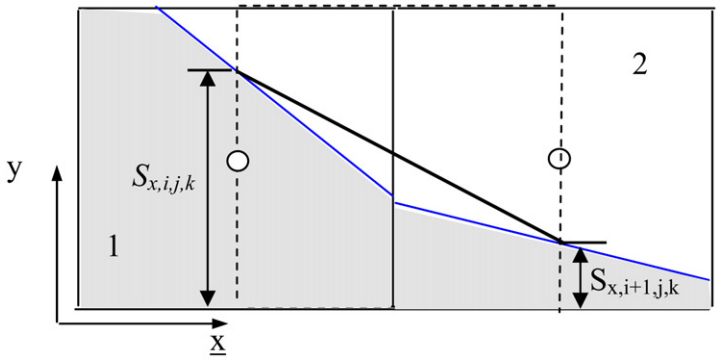

Fig. 2. Interface in a x-momentum cell (dotted line) and interface function S_x.

3. Surface tension calculation

The surface tension force is directly calculated in each of the x-, y- and z-momentum cells. This is convenient, since in staggered grid configuration, one needs to apply the x-, y- and z-components of the surface tension force in the x-, y- and z-momentum cells, respectively. In addition, this method avoids additional errors associated with the calculation of the surface tension force for each momentum cell as used in previous methods. To do so, we first reconstruct the interfaces for the x-, y- and z-momentum cells. In each of the cells, the interface is assumed to be a plane surface (in two-dimensional problems it is a straight line), as shown in Fig. 2 for the x-component of the surface tension force. In this figure, the thick solid inclined line in the x–y plane is the approximate interface location in the x-momentum cell located at $i + 1/2, j, k$. The values of so called interface functions, $S_{x,i,j,k}$ and $S_{x,i+1,j,k}$, are calculated next. The volumetric surface tension force acting on the cell $i + 1/2, j, k$ can be calculated by using Eq. (2):

$$F^{\mathrm{st}}_{vx,i+1/2,j,k} = \frac{\sigma \kappa_{i+1/2,j,k} A_{x,i+1/2,j,k}}{V_{i+1/2,j,k}}, \tag{17}$$

where $A_{x,i+1/2,j,k}$ is the projection of the interface surface area in $i + 1/2, j, k$ cell and in the x-direction (that is on the y–z plane). $A_{x,i+1/2,j,k}$ is given by (see Fig. 2):

$$A_{x,i+1/2,j,k} = S_{x,i+1,j,k} - S_{x,i,j,k}. \tag{18}$$

Thus, Eq. (17) becomes:

$$F^{\mathrm{st}}_{vx,i+1/2,j,k} = \frac{\sigma \kappa_{i+1/2,j,k} (S_{x,i+1,j,k} - S_{x,i,j,k})}{V_{i+1/2,j,k}}. \tag{19}$$

Similarly, for the y-component of the surface tension force in the y-momentum cell centered at $i, j + 1/2, k$, and for the z-component of the surface tension force in the z-momentum cell centered at $i, j, k + 1/2$, we have:

$$F^{\mathrm{st}}_{vy,i,j+1/2,k} = \frac{\sigma \kappa_{i,j+1/2,k} (S_{y,i,j+1,k} - S_{y,i,j,k})}{V_{i,j+1/2,k}} \tag{20}$$

and

$$F^{\mathrm{st}}_{vz,i,j,k+1/2} = \frac{\sigma \kappa_{i,j,k+1/2} (S_{z,i,j,k+1} - S_{z,i,j,k})}{V_{i,j,k+1/2}}, \tag{21}$$

respectively. It is obvious that the interface functions S_x, S_y and S_z are such that the values of F^{st}_v's in Eqs. (19)–(21) are non-zero in the interfacial cells. Thus, the surface tension force is applied only to the cells with interfaces, while due to the averaging procedure, the entire neighboring interfacial cells in Meier's method will have non-zero surface tension force (see Eq. (16)). Here, we use density of the momentum cells, because forces are calculated at the cell faces and not at the cell center. This provides a much better result compared with when the density at the cell center ($\rho_{i,j,k}$) is used.

4. Calculation of interface function S_k

The values of S_x, S_y and S_z are related to the location of interface in a cell and since we are using the PLIC technique, in cells i, j, k, the interface shape is a plane surface and its formulation is known. So the calculation of the values of $S_{x,i,j}$, $S_{y,i,j}$ and $S_{z,i,j}$ is straightforward and can be done with negligible computational cost.

We start with the reconstruction process using the piecewise linear interface calculation PLIC. In PLIC method, the interface is approximated by a plane surface of an appropriate inclination in each cell. The plane surfaces are not connected to each other at the cell faces. That is, the interface surface at each cell is determined independently of the neighboring interface, and their intersections with the cell faces need not necessarily be connected at the cell faces. Each plane is determined so that it is perpendicular to an interface normal vector, and it divides the cell volume into two regions that match the given C for the cell (see Youngs [4] for more details). Therefore, the interface normal vector $\mathbf{n}$ (a unit vector perpendicular to the interface) needs to be determined for each cell. This is achieved using the gradient of C:

$$\mathbf{n} = -\frac{\nabla C}{|\nabla C|}, \tag{22}$$

where the gradient of C at each point is calculated using the values of C in its immediate twenty six neighboring points. Now the interface at cell i, j, k is constructed and thus the values of $S_{x,i,j,k}$, $S_{y,i,j,k}$ and $S_{z,i,j,k}$ can be easily calculated. To avoid lengthy and tedious expressions, more details for calculation of S's are given below for two-dimensional cases.

5. Formulations for two-dimensional problems

The interface in two-dimensional problems is assumed to be a straight line rather than a plane surface, see Fig. 2 for the x-momentum cell. In this figure, the thick solid inclined line is the approximate interface location in the x-momentum cell located at $i + 1/2, j$. Here we replace the interface functions $S_{x,i,j,k}$ and $S_{x,i+1,j,k}$ with the two-dimensional interface functions $L_{x,i,j}$ and $L_{x,i+1,j}$. We now need to calculate the values of $L_{x,i,j}$ and $L_{x,i+1,j}$. To show the formulation, we start with the volumetric surface tension force acting on the cell $i + 1/2, j$. By using Eq. (2) for two-dimensional case, we have:

$$F^{\mathrm{st}}_{vx,i+1/2,j} = \frac{\sigma \kappa_{i+1/2,j} A_{x,i+1/2,j}}{\Delta x \Delta y}, \tag{23}$$

where $A_{x,i+1/2,j}$ is the projection of the interface surface line in $i + 1/2$, j cell and in the x-direction (that is on the y-axis). It is given by (see Fig. 2):

$$A_{x,i+1/2,j} = L_{x,i+1,j} - L_{x,i,j}. \tag{24}$$

Thus, Eq. (23) becomes:

$$F^{\mathrm{st}}_{vx,i+1/2,j} = \frac{\sigma \kappa_{i+1/2,j} (L_{x,i+1,j} - L_{x,i,j})}{\Delta x \Delta y}. \tag{25}$$

Similarly, for the y-component of the surface tension force in the y-momentum cell centered at i, $j + 1/2$, we have:

$$F^{\mathrm{st}}_{vy,i,j+1/2} = \frac{\sigma \kappa_{i,j+1/2} (L_{y,i,j+1} - L_{y,i,j})}{\Delta x \Delta y}. \tag{26}$$

We can use Eq. (22) to reconstruct the interface:

$$\mathbf{n} = -\frac{\nabla C}{|\nabla C|}. \tag{27}$$

The eight neighboring points of point i, j are:

$$\mathbf{C}_{i,j} = \begin{bmatrix} C_{i-1,j+1} & C_{i,j+1} & C_{i+1,j+1} \\ C_{i-1,j} & C_{i,j} & C_{i+1,j} \\ C_{i-1,j-1} & C_{i,j-1} & C_{i+1,j-1} \end{bmatrix}. \tag{28}$$

Vector **m** is defined as the gradient of $C_{i,j}$. To obtain **m** numerically, we first approximate its value at the cell-corners, for example at position $(i+1/2,\ j+1/2)$, we have:

$$\begin{aligned} m_{x,i+1/2,j+1/2} &= \frac{1}{2}\left(\frac{C_{i+1,j}-C_{i,j}}{x_{i+1,j}-x_{i,j}}+\frac{C_{i+1,j+1}-C_{i,j+1}}{x_{i+1,j+1}-x_{i,j+1}}\right),\\ m_{y,i+1/2,j+1/2} &= \frac{1}{2}\left(\frac{C_{i,j+1}-C_{i,j}}{y_{i,j+1}-y_{i,j}}+\frac{C_{i+1,j+1}-C_{i+1,j}}{y_{i+1,j+1}-y_{i+1,j}}\right). \end{aligned} \tag{29}$$

Then the x and y components of **m** at $i,\ j$ are obtained by means of averaging:

$$\begin{aligned} m_{x,i,j} &= \frac{1}{4}(m_{x,i-1/2,j-1/2}+m_{x,i-1/2,j+1/2}+m_{x,i+1/2,j-1/2}+m_{x,i+1/2,j+1/2}),\\ m_{y,i,j} &= \frac{1}{4}(m_{y,i-1/2,j-1/2}+m_{y,i-1/2,j+1/2}+m_{y,i+1/2,j-1/2}+m_{y,i+1/2,j+1/2}). \end{aligned} \tag{30}$$

Therefore,

$$\begin{aligned} m_{x,i,j} &= \frac{1}{8}\left(\frac{C_{i,j-1}-C_{i-1,j-1}}{x_{i,j-1}-x_{i-1,j-1}}+\frac{C_{i,j}-C_{i-1,j}}{x_{i,j}-x_{i-1,j}}+\frac{C_{i,j}-C_{i-1,j}}{x_{i,j}-x_{i-1,j}}+\frac{C_{i,j+1}-C_{i-1,j+1}}{x_{i,j+1}-x_{i-1,j+1}}\right.\\ &\quad\left.+\frac{C_{i+1,j-1}-C_{i,j-1}}{x_{i+1,j-1}-x_{i,j-1}}+\frac{C_{i+1,j}-C_{i,j}}{x_{i+1,j}-x_{i,j}}+\frac{C_{i+1,j}-C_{i,j}}{x_{i+1,j}-x_{i,j}}+\frac{C_{i+1,j+1}-C_{i,j+1}}{x_{i+1,j+1}-x_{i,j+1}}\right),\\ m_{y,i,j} &= \frac{1}{8}\left(\frac{C_{i-1,j}-C_{i-1,j-1}}{y_{i-1,j}-y_{i-1,j-1}}+\frac{C_{i,j}-C_{i,j-1}}{y_{i,j}-y_{i,j-1}}+\frac{C_{i-1,j+1}-C_{i-1,j}}{y_{i-1,j+1}-y_{i-1,j}}+\frac{C_{i,j+1}-C_{i,j}}{y_{i,j+1}-y_{i,j}}\right.\\ &\quad\left.+\frac{C_{i,j}-C_{i,j-1}}{y_{i,j}-y_{i,j-1}}+\frac{C_{i+1,j}-C_{i+1,j-1}}{y_{i+1,j}-y_{i+1,j-1}}+\frac{C_{i,j+1}-C_{i,j}}{y_{i,j+1}-y_{i,j}}+\frac{C_{i+1,j+1}-C_{i+1,j}}{y_{i+1,j+1}-y_{i+1,j}}\right). \end{aligned} \tag{31}$$

For the case of uniform mesh, $\Delta x=\Delta y=h$, the above equations become:

$$\begin{aligned} m_{x,i,j} &= \big[C_{i+1,j+1}-C_{i-1,j+1}+2(C_{i+1,j}-C_{i-1,j})+C_{i+1,j-1}-C_{i-1,j-1}\big]/(8h),\\ m_{y,i,j} &= \big[C_{i+1,j+1}-C_{i+1,j-1}+2(C_{i,j+1}-C_{i,j-1})+C_{i-1,j+1}-C_{i-1,j-1}\big]/(8h). \end{aligned} \tag{32}$$

And n_x, and n_y components of the unit normal vector, **n**, are:

$$n_{x,i,j} = -\frac{m_{x,i,j}}{\sqrt{m_{x,i,j}^2+m_{y,i,j}^2}}, \tag{33}$$

and

$$n_{y,i,j} = -\frac{m_{y,i,j}}{\sqrt{m_{x,i,j}^2+m_{y,i,j}^2}}. \tag{34}$$

Once the normalized unit vector **n** is calculated, a straight line (the thick solid declined lines in Fig. 2) is positioned perpendicular to it in such a way that it matches with the value of C in the cell.

Fig. 3 shows that depending on the orientation of interface, eight different cases may occur. The normal vector angle, θ, (shown in Fig. 4) can take any value between zero and 2π. When **n** is in the first octant ($0\leqslant\theta\leqslant\pi/4$), different cases may occur, which are shown in Fig. 4. All the other cases can be obtained by mirroring the equivalent situation with the first octant on the x-axis, the y-axis and the bisector between them. **a** and **b** vectors in Fig. 4 are used to determine L_k (see Fig. 5). In order to calculate L_k, the locations of **a** and **b**, the two ends of the straight line in each cell need to be determined. **a** and **b** are determined such that the cross product $\mathbf{n}\times\overrightarrow{ab}$ is positive. To calculate the components of vectors **a** and **b**, we need to specify the limiting values of C for a particular **n**.

$$C_{\mathrm{lim},1} = \frac{n_{\min}}{2n_{\max}} \quad\text{and}\quad C_{\mathrm{lim},2} = 1-C_{\mathrm{lim},1}, \tag{35}$$

where

$$n_{\min} = \min\big(|n_x|,|n_y|\big) \quad\text{and}\quad n_{\max} = \max\big(|n_x|,|n_y|\big). \tag{36}$$

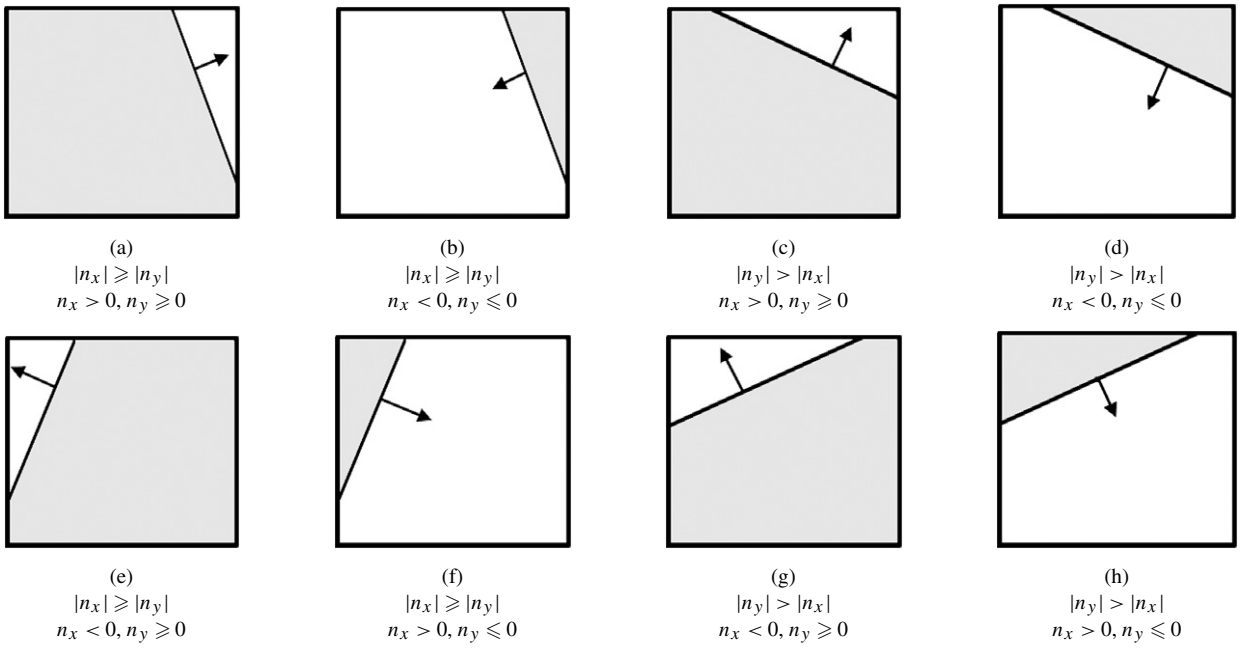

Fig. 3. Different configurations for an interface in a cell for the 2D case.

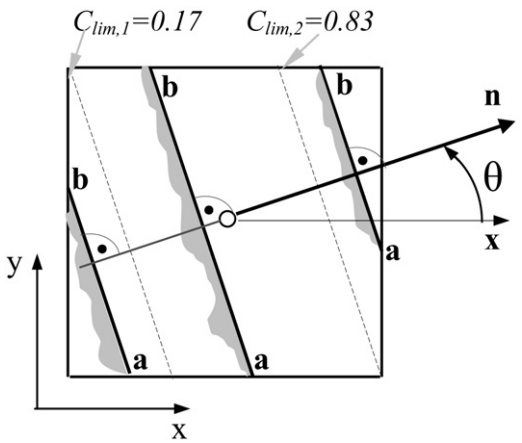

Fig. 4. Different locations of an interface in a cell for the 2D case.

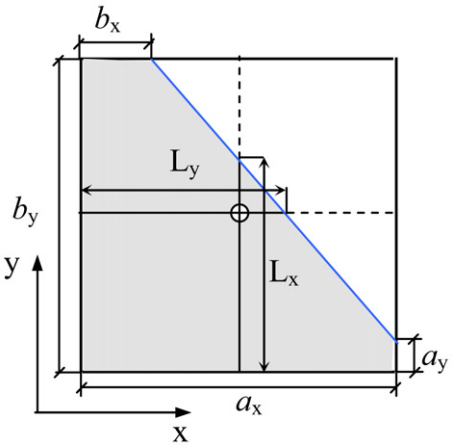

Fig. 5. L_x for a unit length cell for the 2D case.

Then:

$$C_{\text{lim},1} = \frac{h_{\min}}{2h_{\max}} \quad \text{and} \quad C_{\text{lim},2} = 1 - C_{\text{lim},1} \tag{37}$$

where

$$h_{\min} = \min\big(|n_x|\Delta x, |ny|\Delta y\big), \qquad h_{\max} = \max\big(|n_x|\Delta x, |ny|\Delta y\big). \tag{38}$$

The components of vector **a** and **b** in Fig. 4 can be defined as:

(a) For $C \leqslant C_{\text{lim},1}$ (triangle):

$$a_x = \sqrt{2C\Delta x\Delta y\frac{n_y}{n_x}}, \quad a_y = 0, \quad b_x = 0, \quad b_y = \frac{2C\Delta x\Delta y}{a_x}; \tag{39}$$

(b) For $C_{\lim,1} \leqslant C \leqslant C_{\lim,2}$ (quadrilateral):

$$a_x = C\Delta x + \frac{n_y}{2n_x}\Delta y, \quad a_y = 0, \quad b_x = C\Delta x - \frac{n_y}{2n_x}\Delta y, \quad b_y = \Delta y; \tag{40}$$

(c) For $C > C_{\lim,2}$ (pentagon):

$$a_x = \Delta y, \quad a_y = \Delta y - \sqrt{2(\Delta x\Delta y - C)\frac{n_x}{n_y}}, \quad b_x = \Delta x - \frac{2(\Delta x\Delta y - C)}{\Delta y - a_y}, \quad b_y = \Delta y. \tag{41}$$

L_x and L_y can now be calculated using **a** and **b**. For instance, for the cell shown in Fig. 5, L_x, is

$$L_x = a_y + 0.5a_x\frac{b_y - a_y}{a_x - b_x}. \tag{42}$$

Similarly for the same cell L_y is

$$L_y = b_x + 0.5b_y\frac{a_x - b_x}{b_y - a_y}. \tag{43}$$

6. Formulations for collocated grid

If the grid is collocated, the momentum cells and the C cell coincide at one location for each node point i, j, k. Therefore, $S_{x,i,j,k}$, $S_{y,i,j,k}$, $S_{z,i,j,k}$ are already available. In this case the surface tension force in the x-direction, for example, need to be calculated at i, j, k and Eq. (17) becomes:

$$F^{\mathrm{st}}_{vx,i,j,k} = \frac{\sigma\kappa_{i,j,k}A_{x,i,j,k}}{V_{i,j,k}}, \tag{44}$$

where

$$A_{x,i,j,k} = S_{x,i+1/2,j,k} - S_{x,i-1/2,j,k}. \tag{45}$$

For the two-dimensional case, the interface functions S's become L's, and thus:

$$A_{x,i,j,k} = L_{x,i+1/2,j,k} - L_{x,i-1/2,j,k}. \tag{46}$$

The values of L_x and L_y (see Fig. 6) can be obtained from Eqs. (42) and (43):

$$L_{x,i-1/2,j} = b_y \quad \text{and} \quad L_{y,i,j-1/2} = a_x. \tag{47}$$

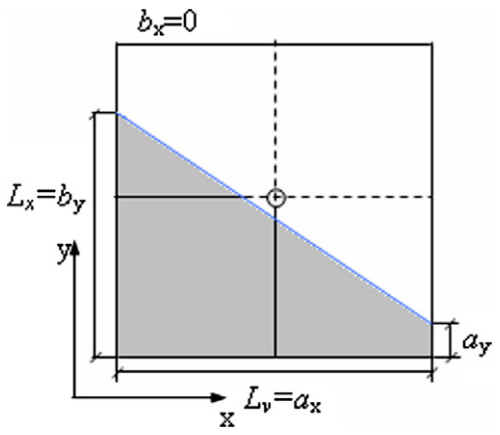

Fig. 6. L_x and L_y for a unit length cell for collocated grid for the 2D case.

7. Results and discussions

Four cases are examined here: (i) A circular drop of water in air, (ii) a bubble of air in water, (iii) a non-circular drop of water in air, and (iv) a non-circular bubble of air in water. The first two cases are static tests, in which the drop or the bubble, as well as their surrounding fluids are initially at rest. The third and the fourth cases are dynamic cases, in which an initially oval shaped drop or bubble starts to oscillate until it becomes circular. Gravity is absent in all cases.

7.1. Static case

An initially static circular drop or a circular bubble should remain stationary in time with a pressure difference across its interface that obeys Laplace's equation. However, due to numerical errors, especially inaccurate calculation of the curvature and, consequently, the surface tension force, some spurious currents are produced. These currents are

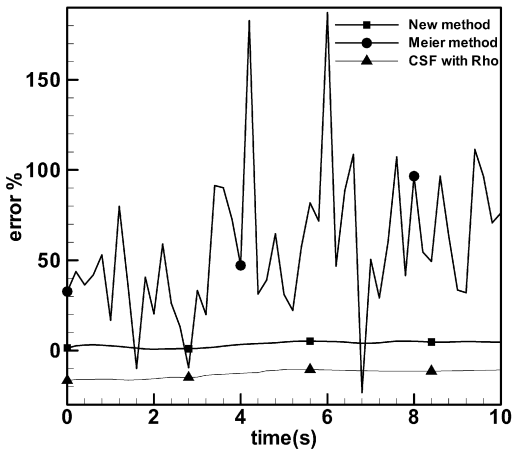

Fig. 7. Time evolution of percent of error for the pressure jump across the interface of a static drop using the new model, Meier model, and the CSF model with the density correction.

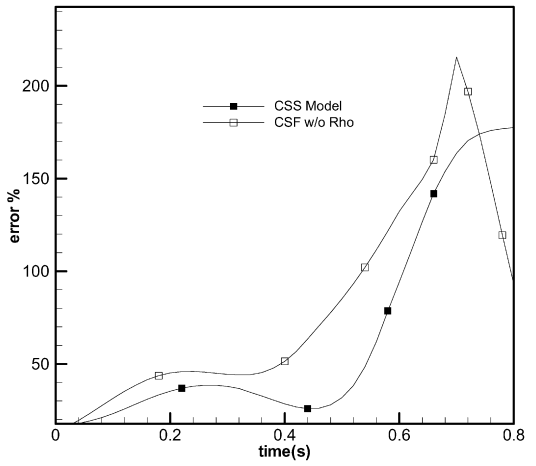

Fig. 8. Time evolution of percent of *error* for the pressure jump across the interface of a static drop using the CSS model and the CSF model without the density correction.

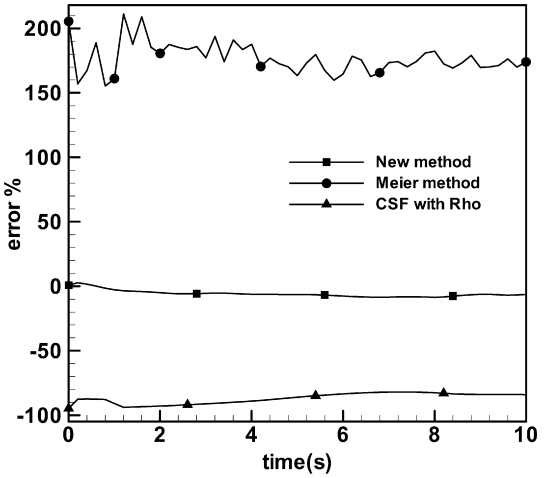

Fig. 9. Time evolution of percent of error for the pressure jump across the interface of a static bubble using the new model, Meier model, and the CSF model with the density correction.

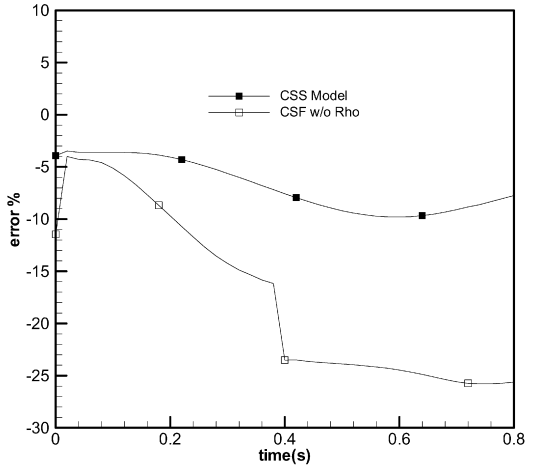

Fig. 10. Time evolution of percent of *error* for the pressure jump across the interface of a static bubble using the CSS model and the CSF model without the density correction.

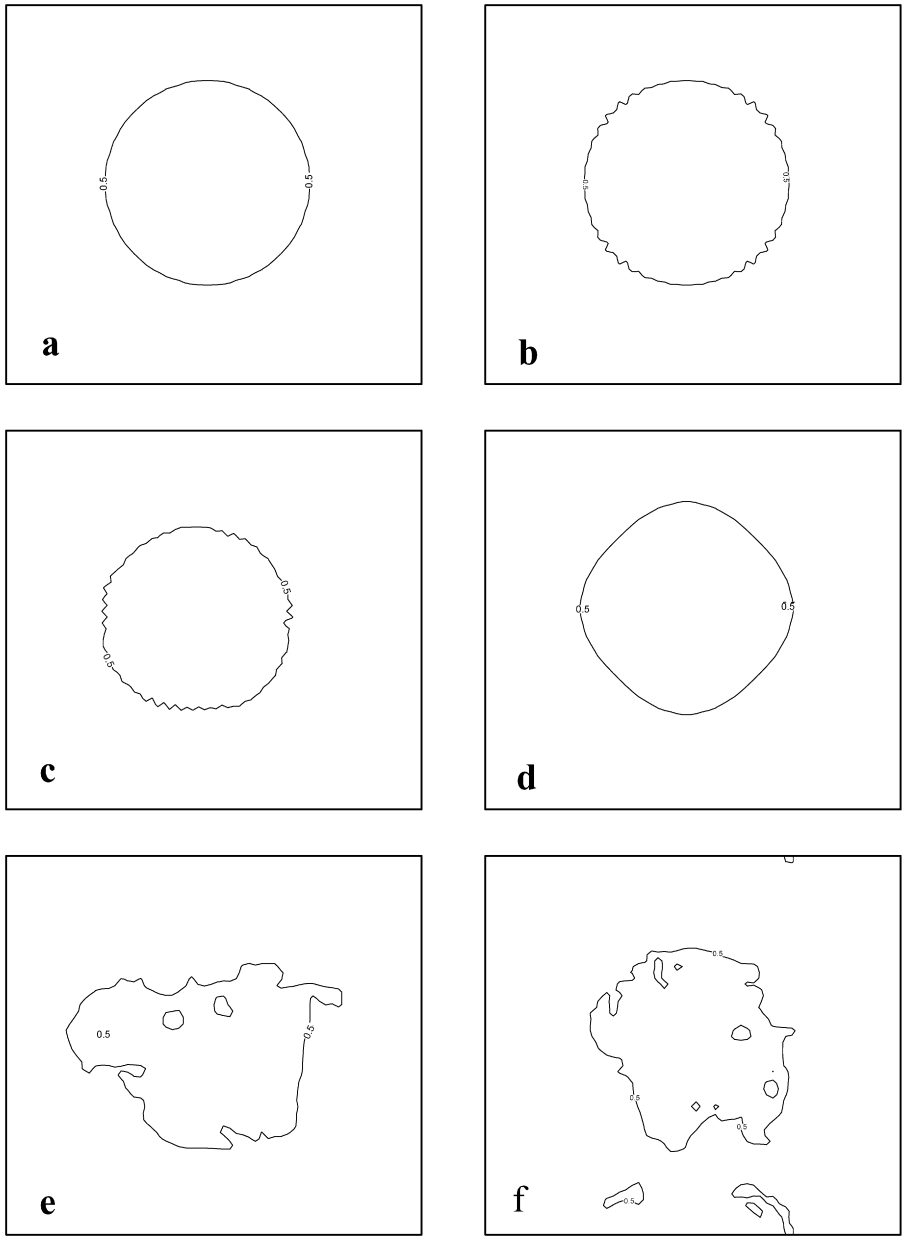

Fig. 11. $C = 0.5$ contour lines for a drop. (a) Initial shape, (b) new model, (c) Meier's mode, (d) CSF with density correction, (e) CSF without density correction; and (f) CSS model.

usually stronger in the lighter fluid, and they increase with time. The magnitude of these currents intimately depends on the density ratio of the two fluids. As the density ratio (density of the heavier fluid to the lighter fluid) increases, the magnitude of the currents increases drastically (see Shirani et al. [26]).

For all the cases presented here, the two-dimensional SURFER [19,28,29] code has been used. This code is based on the two-fluid PLIC-VOF method and it is pressure based with Chorin's projection method for a semi-implicit Navier–Stokes solver. It uses a staggered, equally-spaced grid and advances the Poisson equation with a multigrid solver. The two fluids used are (i) water with density of $\rho_1 = 1000\ \mathrm{kg/m^3}$ and viscosity of $\mu_1 = 0.001\ \mathrm{N\,s/m^2}$, and (ii) air with density of $\rho_2 = 1.204\ \mathrm{kg/m^3}$ and viscosity of $\mu_2 = 1.82\mathrm{E}{-5}\ \mathrm{N\,s/m^2}$. The surface tension coefficient is $\sigma = 0.073\ \mathrm{N/m}$. The diameter of the drop or the bubble is 1/4 of the computational domain. The number of grid

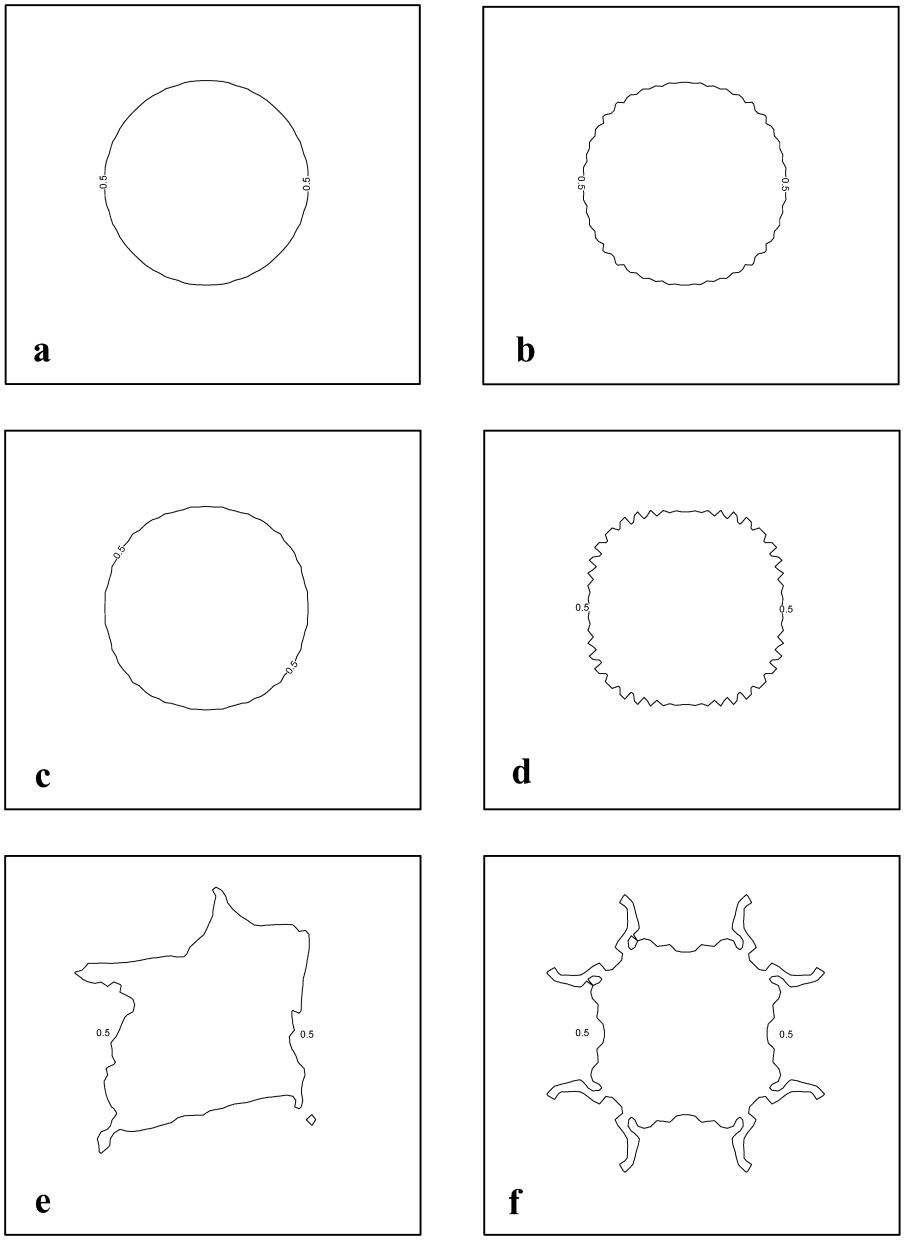

Fig. 12. $C = 0.5$ contour lines for a bubble. (a) Initial shape, (b) new model, (c) Meier's model, (d) CSF with density correction, (e) CSF without density correction; and (f) CSS model.

points is 66×66 and the time step is kept constant and it is chosen such that the stability criterion is satisfied and the required accuracy is obtained. The method of calculation of the curvature for all the cases is the same as the method given in the SURFER code. Meier's technique for the calculation of the curvature was not used in this work.

The *error* in the pressure jump calculation is determined based on the following equation:

$$error = \frac{(\bar{p}_1 - \bar{p}_2) - \sigma/R}{\sigma/R}, \tag{48}$$

where $\bar{p}_1$ and $\bar{p}_2$ are the mean values of pressure inside and outside of the drop or the bubble and R is its radius. If pressures $\bar{p}_1$ and $\bar{p}_2$ are calculated exactly, the value of the *error* would be zero.

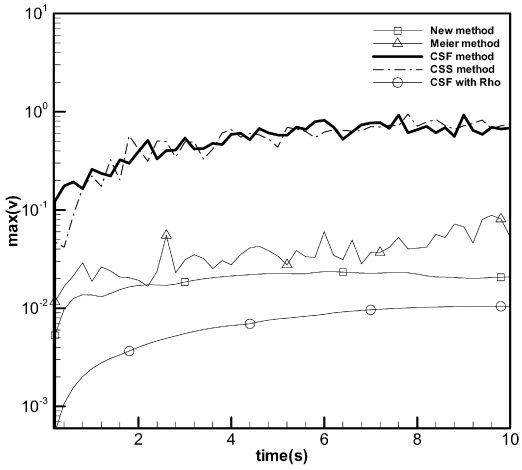

Fig. 13. Maximum spurious velocity for a drop.

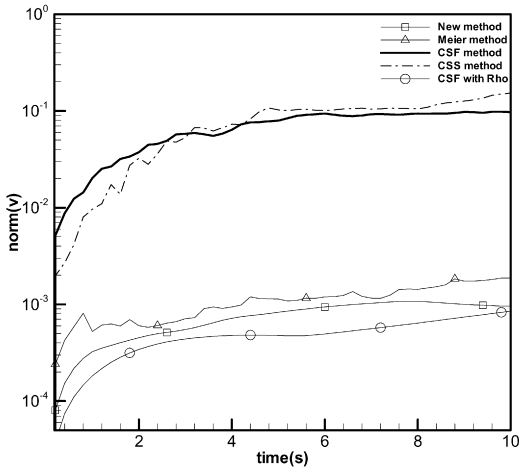

Fig. 14. Norm spurious velocities for a drop.

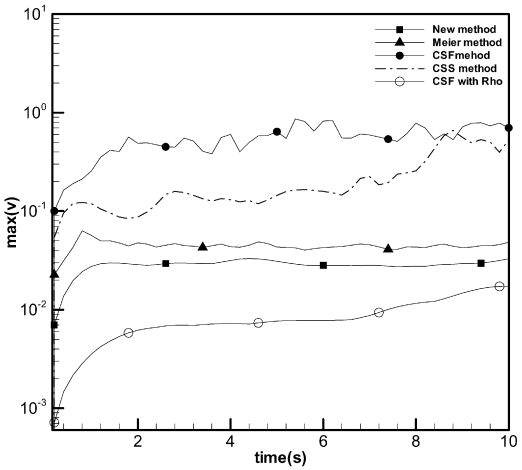

Fig. 15. Maximum spurious velocity for a bubble.

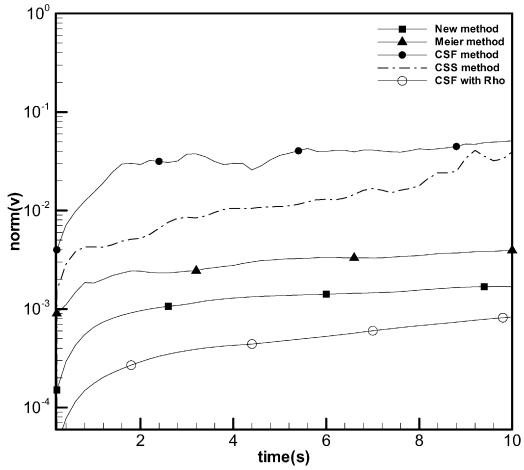

Fig. 16. Norm spurious velocities for a bubble.

Fig. 7 shows the evolution of the pressure jump error for the drop calculated using our model, referred to as SGIP, Meier's model and the CSF model with the density correction (the second fraction in Eq. (13)). The maximum error in SGIP model is 5%, in Meier's model is about 186% error, and in the CSF model with density correction (as defined in Eq. (13)) is about 16%. In Fig. 8, the *errors* for the CSS model and the CSF model without the density correction are plotted as a function of time. In this figure the results are shown for up to 0.8 s. The reason is that as time increases the results become unphysical, the *error* becomes very large, and the shape of the drop is no longer circular and even breaks up.

Similar results for the bubble case are shown in Figs. 9 and 10. The behavior of SGIP model for the bubble case is much better than the other models. The maximum *error* for SGIP model is about 8%, Meier's is 210% and that of CSF with the density correction is about 94%. The CSF model without the density correction works better for the simulation of the bubble and the maximum *error* is about 25%. Also the CSS model works better than the CSF model with a maximum *error* of about 10%. The CSF model with the density correction produces more error for the bubble case because the increase in pressure occurs inside the bubble, where the lighter fluid exists. When we multiply the density correction factor to the surface tension force, we actually force this term to become almost zero and thus the pressure jump does not occur and we get nearly 100% error.

Figs. 11 and 12 show the contour lines at $C = 0.5$ for the drop and the bubble after 10 seconds, respectively. The results for all five cases – SGIP model, Meier's model, CSF with and without density correction and CSS model –

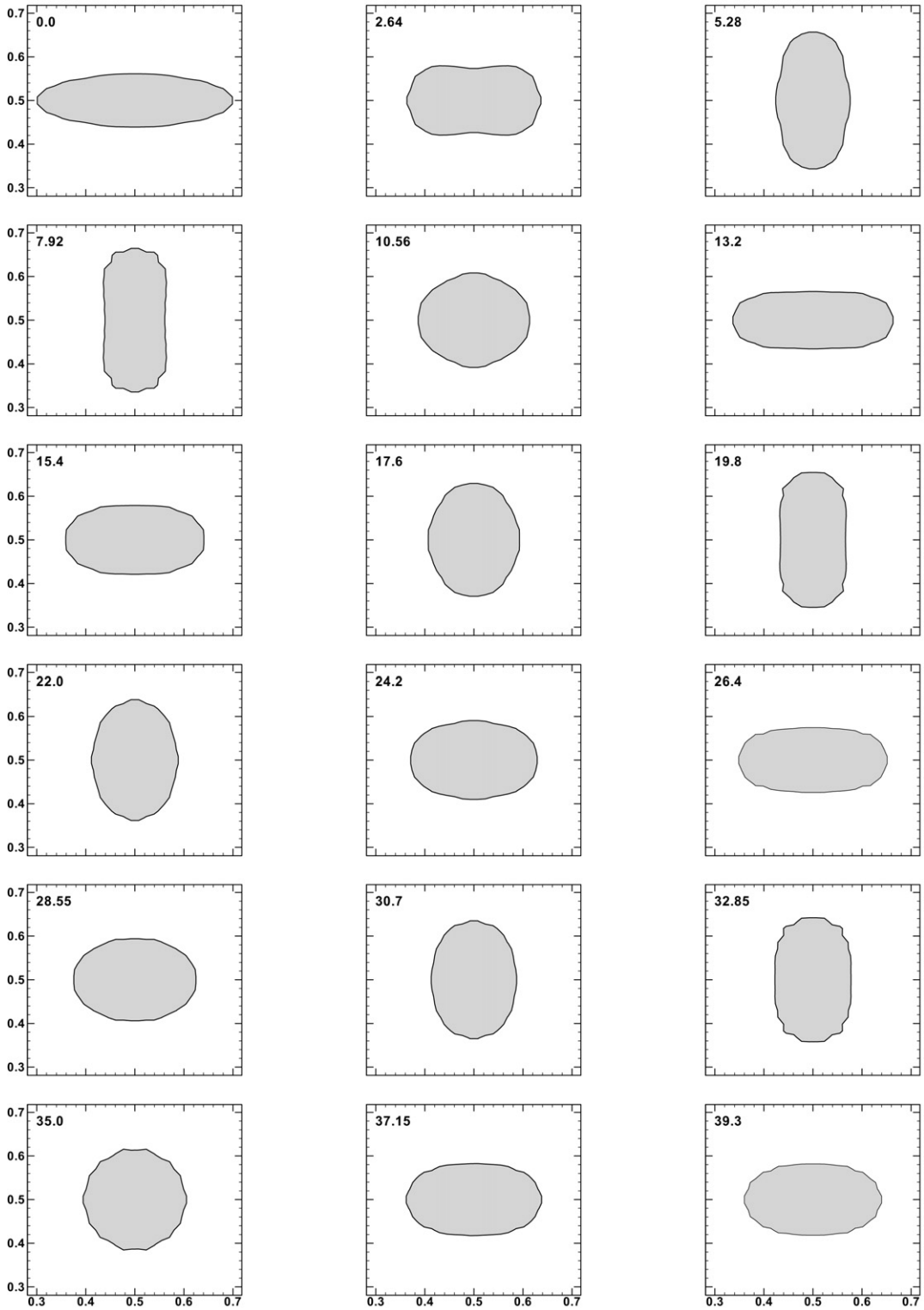

Fig. 17. Oscillation of a non-circular drop of water in air using the new model.

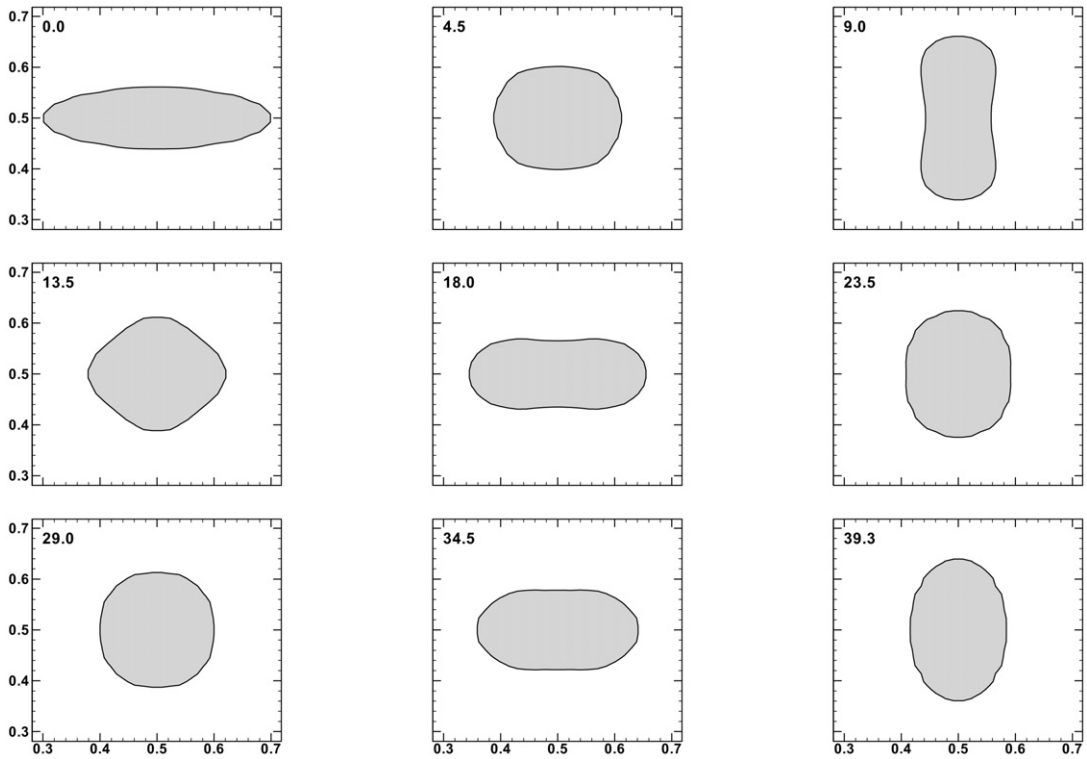

Fig. 18. Oscillation of a water drop in air using CSF method with density correction.

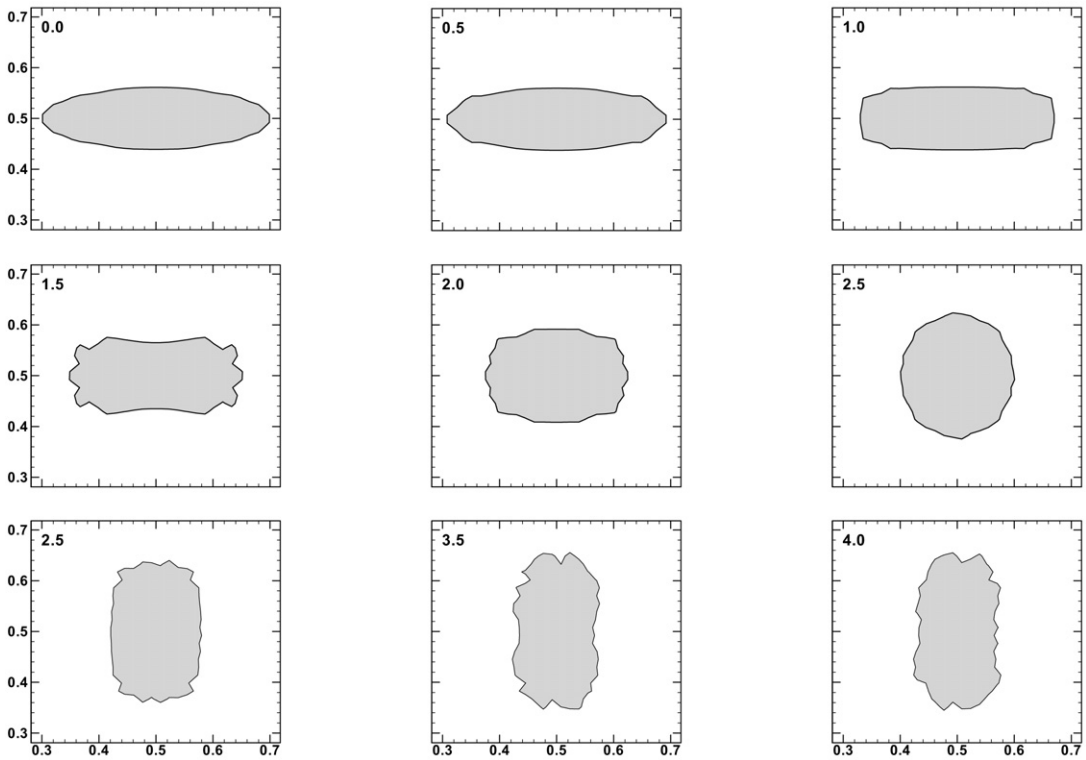

Fig. 19. Oscillation of a non-circular drop of water in air using Meier's model.

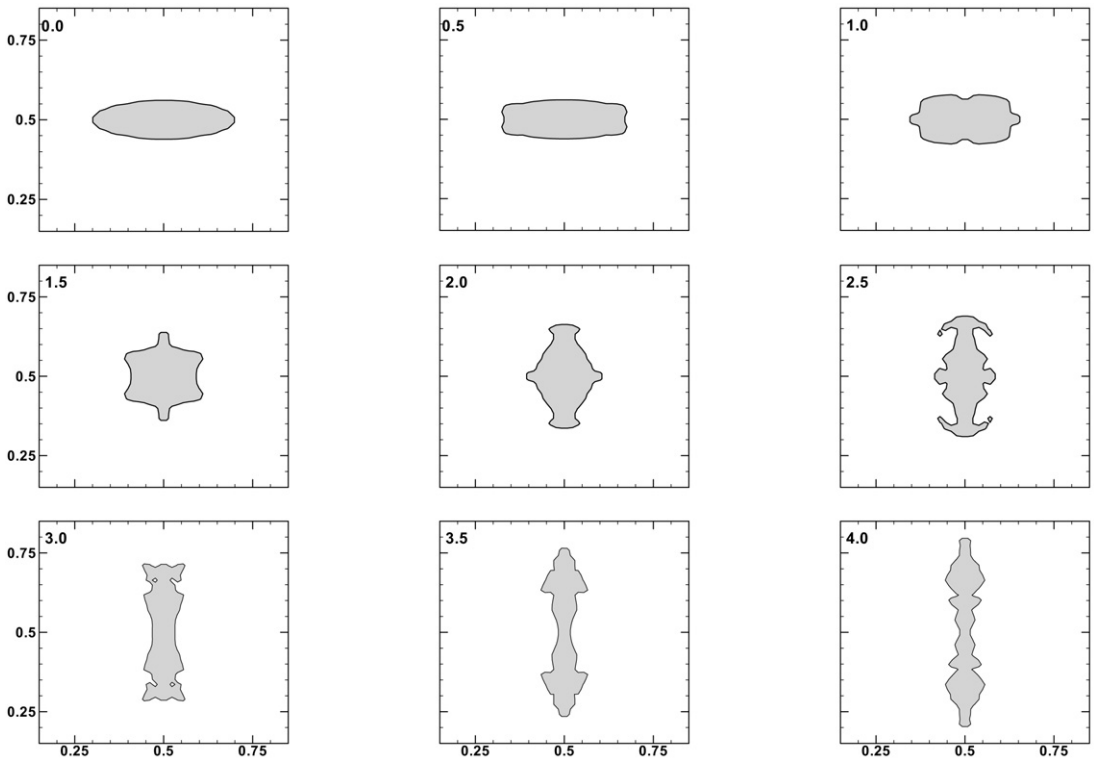

Fig. 20. Oscillation of a water drop in air within 4s using CSS method.

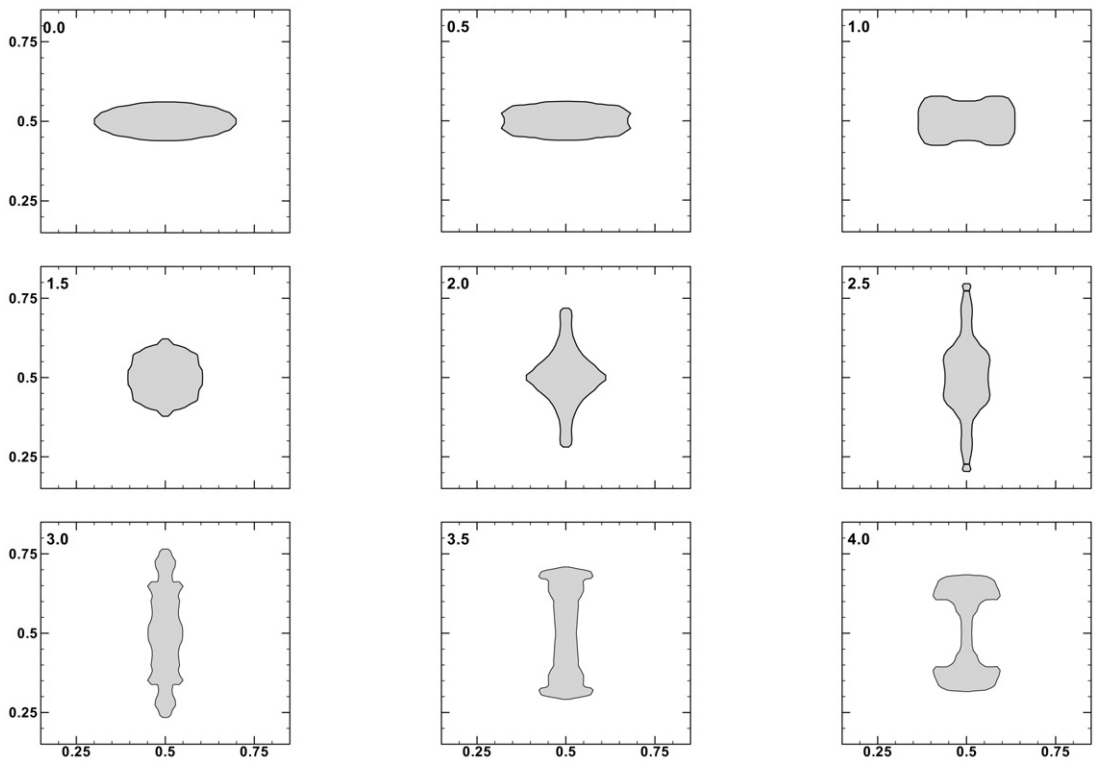

Fig. 21. Oscillation of a water drop in air using CSF method without density correction.

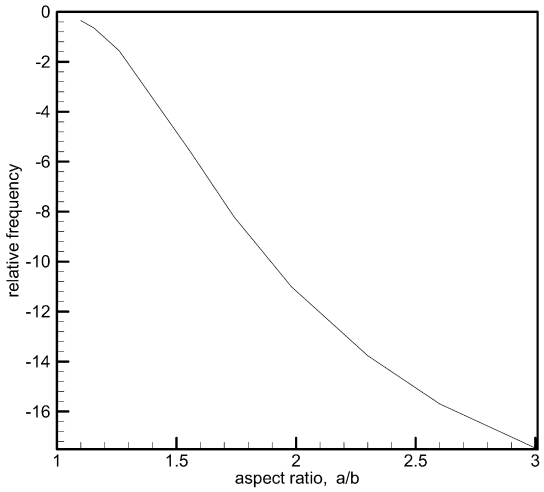

Fig. 22. Relative oscillation frequency of a water drop as a function of its initial aspect ratio.

are shown in these figures. Only SGIP model produces reasonable results for the drop case. Also Meier's model and the CSF model with the density correction produce nearly physical results. For the bubble case, however, only SGIP model and Meier's model produce physical results. Finally, Figs. 13 and 14 show the maximum and the norm of the spurious velocities for the drop case. The norm of the spurious currents is defined as the average of the absolute velocity in the whole flow field. These figures show that the spurious currents produced by SGIP model, Meier's model and the CSF model with density correction are about two orders of magnitude less than that of the CSS and CSF models without the density correction. The large shape distortions shown in Fig. 11, using CSS and the CSF models without the density correction, are due to large parasitic currents that these models produce. Similarly, the maximum and the norm velocity of spurious currents for the bubble are shown in Figs. 15 and 16, which show the same trends as in the drop case.

7.2. Dynamic case

In the dynamic cases, initially oval shape drops and bubbles are modeled. For the drop case, the larger diameter of the drop is 0.4 times the domain size and its smaller diameter is 0.12 times the domain size. Figs. 17–21 show the oscillation of the initially oval shape drop, when SGIP model, CSF model with the density correction, Meier's model, CSS model and CSF model without the density correction are used, respectively. The results for the first two cases are shown for up to 39.3 seconds, whereas the other two cases are shown up to 4 seconds. The CSF model with density correction underpredicts the pressure jump (lower surface tension force), therefore, its oscillations are somewhat delayed compared to SGIP results.

To check on the accuracy of our results, we have compared the period of oscillations with the analytical solution obtained for small perturbation of a column of liquid with zero viscosity provided by Lamb [31]:

$$\omega_0^2 = m\big(m^2 - 1\big)\frac{\sigma}{\rho a^3}, \tag{49}$$

where a is the mean drop radius, m is the oscillation mode, ω_0 is the frequency of the oscillation, and it is related to the period of oscillation ($T_0 = 2\pi/\omega_0$). From this relation, the period of oscillation for the mode corresponding to the oval shape, $m = 2$, and $a = 0.109$ (which corresponds to our test case), is 10.88 s. Our method results in a period of $T = 13.1$ s (20% difference compared to the analytical results) and CSF model with the density correction results in a period of $T = 16.4$ s (50.7% difference compared to the analytical results). We have also obtained results for different aspect ratios at a constant Reynolds number of $Re = 100$. The results are shown in Fig. 22. In this figure the relative frequency of the oscillations is defined as:

$$\textit{relative oscillations} = \frac{100(\omega - \omega_0)}{\omega_0}, \tag{50}$$

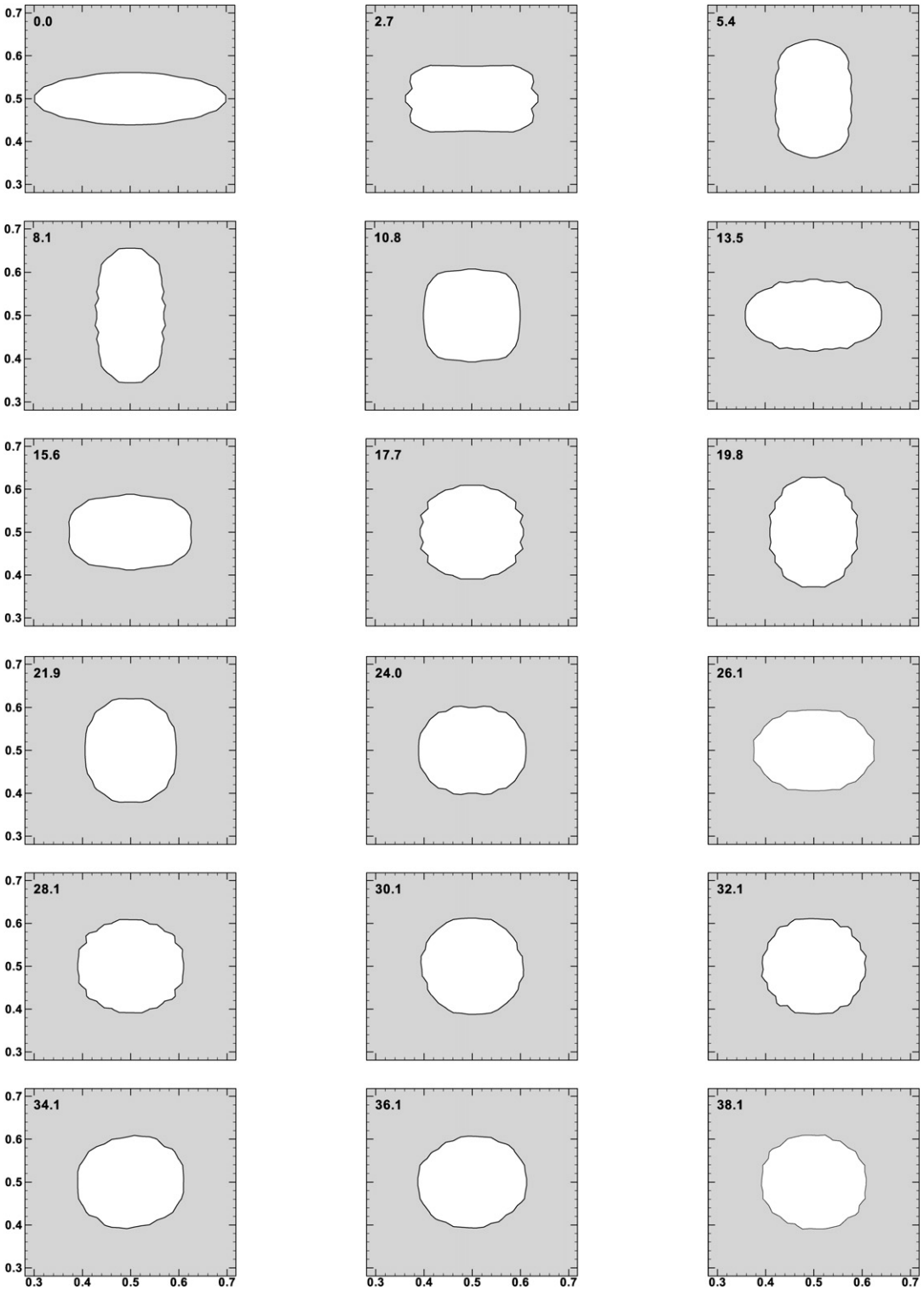

Fig. 23. Oscillation of a non-circular bubble of air in water using the new model.

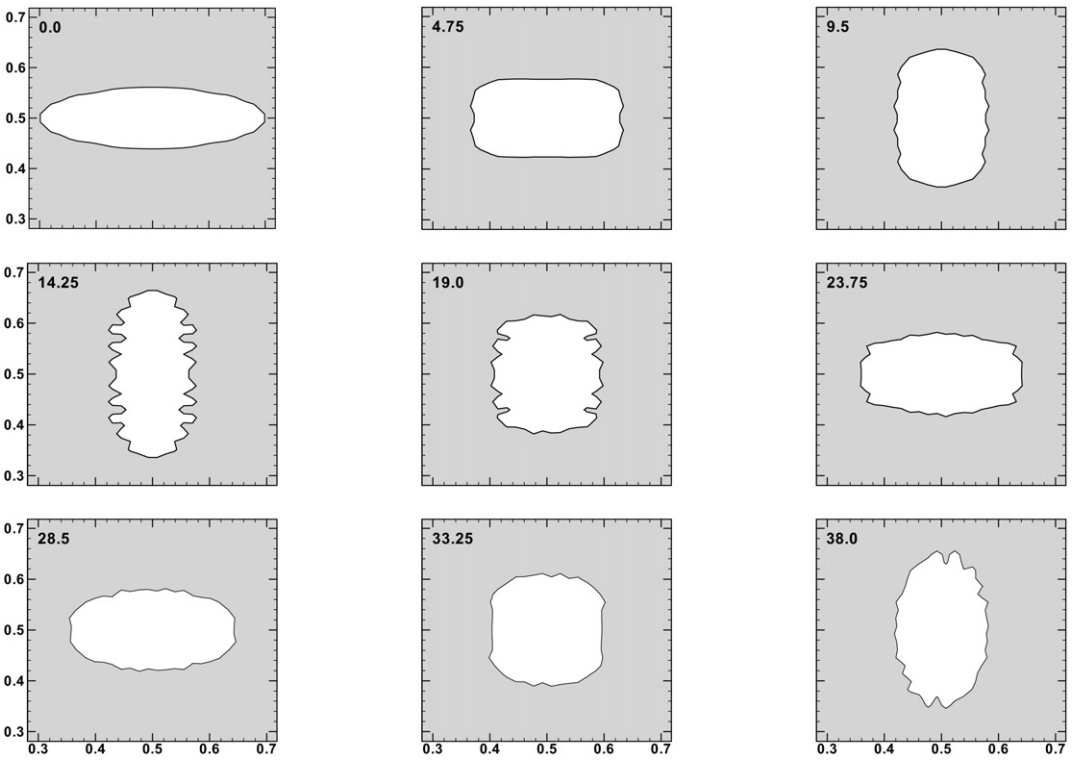

Fig. 24. Oscillation of a non-circular bubble of air in water using CSF model with density correction.

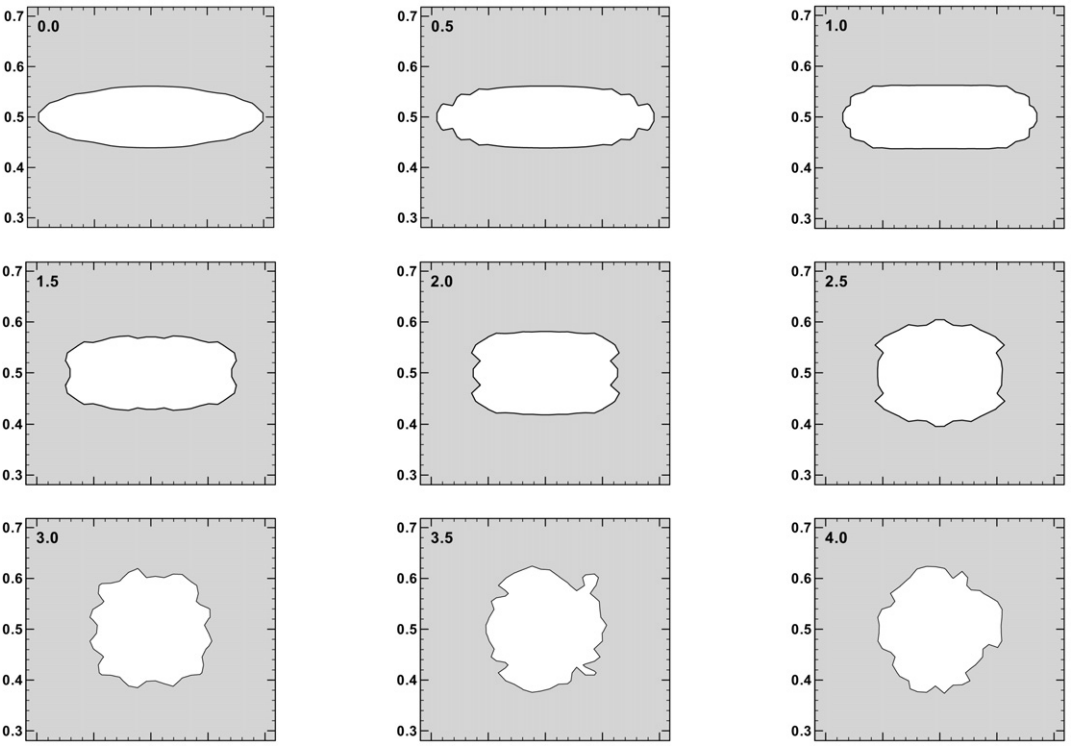

Fig. 25. Oscillation of a non-circular bubble of air in water using Meier's model.

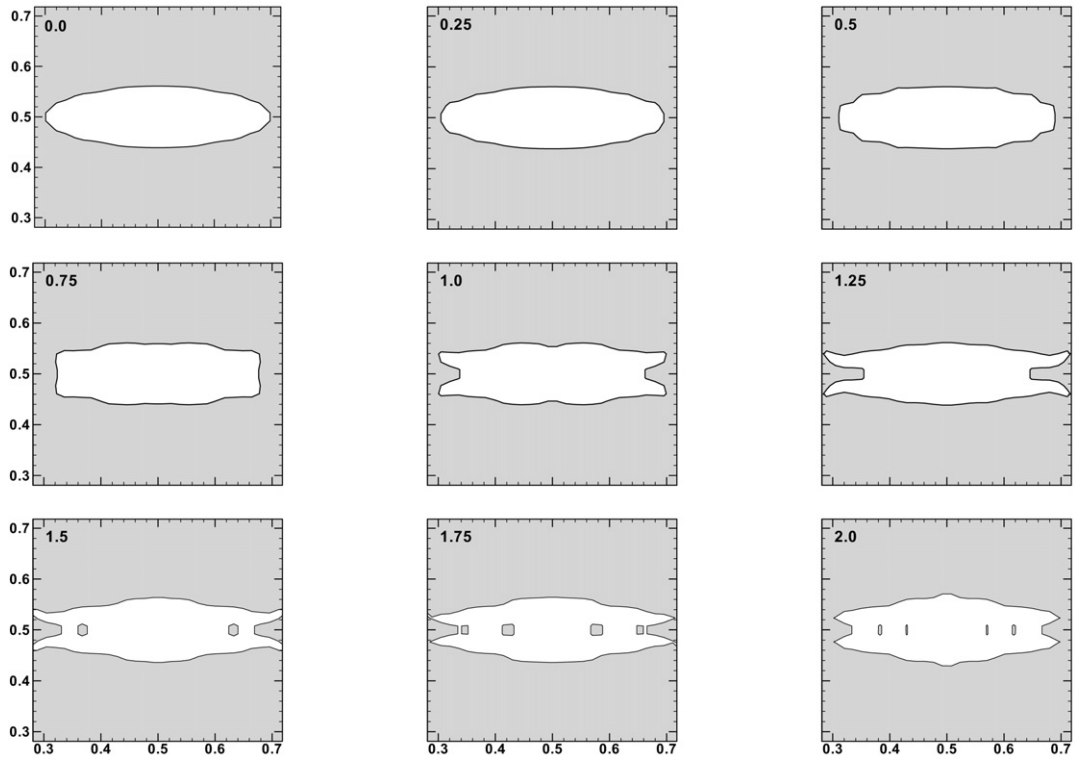

Fig. 26. Oscillation of non-circular bubble of air in water within 2s using CSS method.

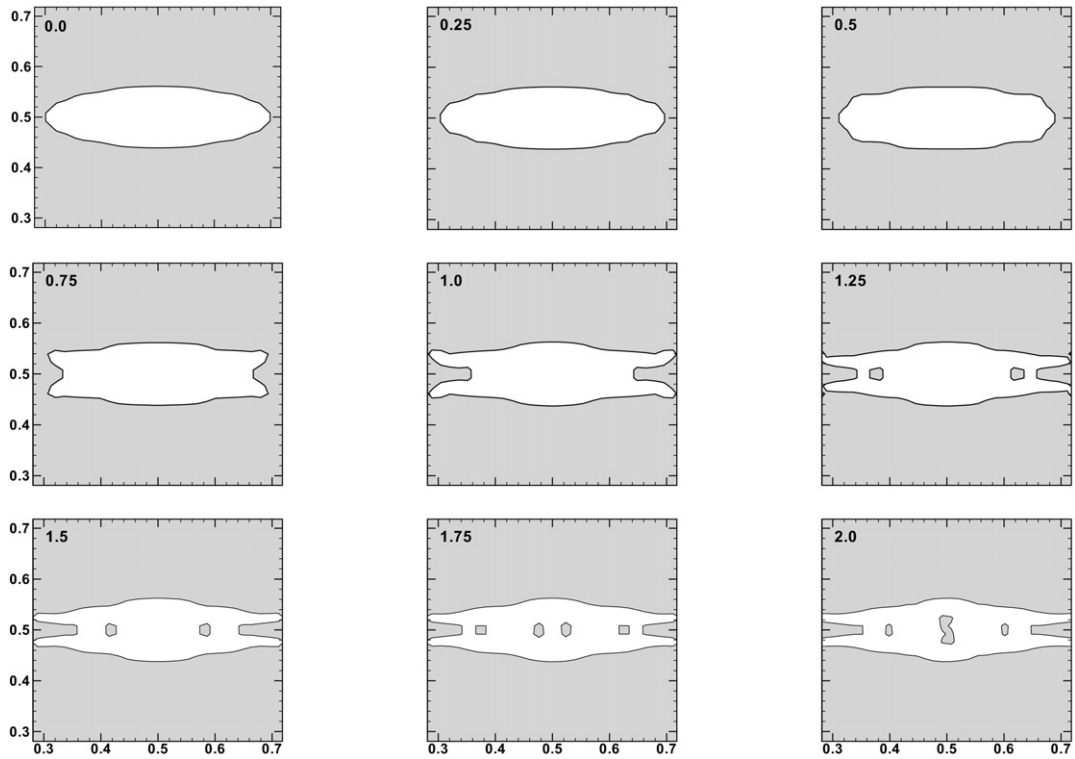

Fig. 27. Oscillation of non-circular bubble of air in water within 2s using CSF method without density correction.

where ω_0 and ω are the frequencies of the oscillations obtained by analytical and numerical solutions, respectively. From this figure, it is clear that as the aspect ratio becomes closer to one, the numerical solution reaches the analytical solution. This indicates that the new model has the correct asymptotic behavior.

Finally, the motion of an initially oval shape bubble of air in water is simulated. The larger diameter of the bubble is 0.2 times the domain size and its smaller diameter is 0.06 times the domain size. The properties are similar to the previous case. The number of grid points is 66×66 and the time step is kept constant and it is chosen such that the stability criterion is satisfied and the required accuracy is obtained. Figs. 23–27 show the oscillation of the initially oval shape bubble when our model, CSF model with the density correction, Meier's model, CSS model and CSF model without the density correction are used, respectively. The results obtained from SGIP model, Fig. 23, are reasonable and the shape of the initially oval shape bubble after 38.1 seconds and about three oscillations reaches its equilibrium state. The results obtained when CSF model with density correction is used, Fig. 24, show distorted interface shape of the oval and shows only one and half periods of oscillation after 38 seconds. The reason for the differences between the results of our model and this model in computing the oscillation period is that the CSF model with density correction underestimates the pressure jump across the interface and, therefore, there is not enough driving force for the oval to oscillate. The results, using Meier's and CSS models and CSF model without density correction, are not satisfactory and the shape of the oval deteriorates after short time as shown in Figs. 25–27.

8. Conclusions

A new method for implementing the surface tension forces in the PLIC-VOF method with staggered grid is introduced (SGIP). In this method, interface locations and interface surface areas at the momentum control volumes are determined. Surface tension forces are added to momentum equations as a volumetric force.

The model is used to simulate static drops and bubbles, as well as oscillating drops. The new model is compared with Meier's model, CSS model and CSF model with and without the density correction factor. It is shown that the present model produces better results in calculating the pressure jump across the interface, as well as the shapes of drops and bubbles. The spurious currents produced by the new model are two orders of magnitude less than the currents generated by CSS and CSF without the density correction factor. It is also shown that the CSF method with density correction factor produces about 100% error when applied to bubble calculation. In the bubble case, multiplying the density factor to the surface tension force makes the force nearly zero inside the lighter fluid. The new model can also produce fairly accurate results when applied to a drop and a bubble oscillation cases. Whereas, the Meier and CSS models and the CSF models with and without the density correction factor do not produce physical results for these cases.

References

[1] W.F. Noh, P.R. Woodward, SLIC (Simple Line Interface Method), in: Lecture Notes in Physics, vol. 59, Springer, 1976.
[2] B.D. Nichols, W.C. Hirt, R.S. Hotchkiss, A solution algorithm for transient fluid flow with multiple free boundaries, Technical report, La-8355, Los Alamos National Lab., 1980.
[3] C.W. Hirt, B.D. Nichols, Volume of Fluid (VOF) method for the dynamics of free boundaries, J. Comput. Phys. 39 (1981) 201–225.
[4] D.L. Youngs, Time-dependent multi-material flow with large fluid distribution, in: K.W. Morton, M.L. Norman (Eds.), Numerical Methods for Fluid Dynamics, 1986, pp. 187–221.
[5] N. Ashgriz, J.Y. Poo, FLAIR: Flux line-segment model for advection and interface reconstruction, J. Comput. Phys. 93 (1991) 449–468.
[6] S.O. Kim, H.C. No, Second order model for free surface convection and interface reconstruction, Int. J. Numer. Methods Fluids 26 (1) (1998) 79.
[7] J.E. Polliod, An analysis of piecewise linear interface reconstruction algorithm for volume of fluid methods, M.Sc. Thesis, Department of Mathematics, University of California, Davis, 1992.
[8] W.J. Rider, D.B. Kothe, Reconstructing volume tracking, J. Comput. Phys. 141 (1998) 112.
[9] J.U. Brackbill, D.B. Kothe, C. Zemach, A continuum method for modeling surface tension, J. Comput. Phys. 100 (1992) 335–354.
[10] J.U. Brackbill, D.B. Kothe, Dynamic modeling of the surface tension, in: Proceedings of the 3rd Microgravity Fluid Physics Conference, Cleveland, OH, 1996, pp. 693–698.
[11] M.W. Williams, D.B. Kothe, E.G. Puckett, Convergence and accuracy of continuum surface tension models, in: W. Shyy, R. Narayanan (Eds.), Fluid Dynamics at Interface, Cambridge University Press, Cambridge, 1999, pp. 294–305.
[12] X.-D. Liu, R.P. Fedkiw, M. Kang, A boundary condition capturing method for Poisson's equation on irregular domains, J. Comput. Phys. 160 (2000) 151–178.
[13] M.W. Williams, Numerical methods for tracking interfaces with surface tension in 3-D mold filling process, Ph.D. Dissertation, University of California, Davis, also Technical Report LA-13776-T, Los Alamos National Laboratory, 2000.

[14] I. Ginzburg, G. Wittum, Two-phase flows on interface refined grids modeled with VOF, staggered finite volumes, and spline interpolants, J. Comput. Phys. 166 (2001) 302–335.
[15] M.M. Francois, D.B. Kothe, E.D. Dendy, J.M. Sicilian, M.W. Williams, Balanced force implementation of continuum surface tension force method into a pressure correction algorithm, Technical Report FEDSM2003-45175, in: Proceedings of the Seventh ASME/JSME Joint Fluids Engineering Conference, July 6–11, Honolulu, HI, 2003.
[16] S. Shin, S.I. Abdel-Khalik, V. Daru, D. Juric, Accurate representation of surface tension using the level contour reconstruction method, J. Comput. Phys. 203 (2) (2005) 493–516.
[17] J. Kim, A continuous surface tension force formulation for diffuse-interface models, J. Comput. Phys. 204 (2005) 784–804.
[18] M.M. Francois, S.J. Cummins, E.D. Dendy, D.B. Kothe, J.M. Sicilian, M.W. Williams, A balanced-force algorithm for continuous and sharp interfacial surface tension models within a volume tracking framework, J. Comput. Phys. 213 (2006) 141–173.
[19] B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, G. Zanetti, Modelling merging and fragmentation in multiphase flows with SURFER, J. Comput. Phys. 113 (1994) 134–147.
[20] S. Popinet, S. Zaleski, A front-tracking algorithm for accurate representation of surface tension, Int. J. Numer. Methods Fluids 30 (1999) 775–793.
[21] M. Meier, G. Yadigaroglu, B.L. Smith, A novel technique for including surface tension in PLIC-VOF methods, Eur. J. Mech. B Fluids 21 (2002) 61–73.
[22] M. Meier, Numerical and experimental study of large steam-air bubbles injected in a water pool, PhD Dissertation no. 13091, Swiss Federal Institute of Technology, Zürich, Switzerland, 1999.
[23] I. Aleinov, E.G. Puckett, Computing surface tension with high-order kernels, in: H.A. Dwyer (Ed.), Proceedings of the Sixth International Symposium on Computational Fluid dynamics, Lake Tahoe, NV, 1995, pp. 13–18.
[24] Y. Renardy, M. Renardy, PROST: A Parabolic Reconstruction of Surface Tension for the volume-of-fluid method, J. Comput. Phys. 183 (2002) 400–421.
[25] 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.
[26] E. Shirani, N. Ashgriz, J. Mostaghimi, Interface pressure calculation based on conservation of momentum for front capturing methods, J. Comput. Phys. 203 (2005) 154–175.
[27] D.B. Kothe, W.J. Rider, S.J. Mosso, J.S. Brock, Volume tracking of interfaces having surface tension in two and three dimensions, AIAA 96-0859, 1996.
[28] S. Zaleski, J. Li, S. Succi, Two-dimensional Navier–Stokes simulation of deformation and break-up of liquid patches, Phys. Rev. Lett. 75 (1995) 244.
[29] S. Zaleski, Simulation of High Reynolds Number Breakup of Liquid–Gas Interface, Lecture Series, von Karman Institute for Fluid Dynamics, 1996.
[30] R. Scardovelli, S. Zaleski, Direct numerical simulation of free surface and interfacial flow, Annu. Rev. Fluid Mech. 31 (1999) 567–603.
[31] H. Lamb, Hydrodynamics, Cambridge University Press, 1932.