AN IMPROVED METHOD FOR CALCULATION OF INTERFACE PRESSURE FORCE IN PLIC-VOF METHODS

M. Sefollahi
Department of Mechanical Engineering, Isfahan University of Technology, Isfahan, Iran

and

E. Shirani\textsuperscript{1}
Department of Mechanical Engineering, Isfahan University of Technology, Isfahan, Iran
and
The Abdus Salam International Centre for Theoretical Physics, Trieste, Italy.

MIRAMARE – TRIESTE
August 2004

\textsuperscript{1} Regular Associate of ICTP. eshirani@cc.iut.ac.ir
Abstract

Conventional methods for the modeling of surface tension force in Piecewise Linear Interface Calculation-Volume of Fluid (PLIC-VOF) methods, such as Continuum Surface Force (CSF), Continuum Surface Stress (CSS) and also Meier’s method, convert the surface tension force into a body force. Not only do they include the force in the interfacial cells but also in the neighboring cells. Thus they produce spurious currents. Also the pressure jump, due to the surface tension, is not calculated accurately in these methods. In this paper a more accurate method for the application of interface force in the computational modeling of free surfaces and interfaces which use PLIC-VOF methods is developed. This method is based on the evaluation of the surface tension force only in the interfacial cells and not the neighboring cells. Also the normal and the interface surface area needed for the calculation of the surface tension force is calculated more accurately.

The present method is applied to a two-dimensional motionless drop of liquid and a bubble of gas as well as a non-circular two-dimensional drop, which oscillates due to the surface tension force, in an initially stagnant fluid with no gravity force. The results are compared with the results of the cases when CSF, CSS and Meier’s methods are used. It is shown that the present method calculates pressure jump at the interface more accurately and produces less spurious currents comparing to CSS and CSF models.
Introduction

In simulation of interfacial flows with fixed mesh, determination of the interface pressure and surface tension have been one of the most troublesome and challenging issues. Surface tension forces appear in equations by imposing a jump condition across the interface. This condition is difficult to apply numerically and has been the center of attention of many researchers.

Surface tension force may be included in the overall solution through the use of one of the following two methods. The first method is to apply the surface tension effect as a boundary condition along the free surface. This requires that the exact location of the interface be known at all times. Since the surface tension is related to the surface pressure via Laplace's equation, by applying this boundary condition, the pressure jump across the surface may be computed. For a staggered fixed mesh, an interpolation scheme is used to ensure that the computed surface pressure is correct in relation to the cell centered pressure. If it is not, the solution is iterated until the final pressure field is within some tolerance value of the previous iteration. If it is, the correct pressure solution has been obtained, and the overall Navier-Stokes solution is advanced to the next time level. This technique is not popular for two reasons. First, the cost of multiple iterations at each time-step is high. In most cases the time-step restrictions which would ensure stability require very small values. Therefore, these additional iterations at each time-step would increase the overall computational time needed to obtain a solution. Secondly, in order to properly implement this method, the exact location of the free surface at the next time step is required. Although the interface location can be determined through various methods (i.e., using VOF [1, 2] or level set reconstruction techniques [3]), its value is only known at the existing time-level. In fact, the exact location of the interface at the next-time level is not known a priori.

Several approaches and methods are generated to reconstruct an interface by using volume fraction data. One early algorithm is the Simple Line Interface Calculation (SLIC) method which was first introduced by Noh and Woodward [4]. Among the other methods are the piecewise constant stair stepped interface method [5,6,7], the piecewise linear interface method [8,9,10,11]; and piecewise second order interface method [12,13,14,15,16]. The popular method for interface advection and reconstruction is the Youngs method [9], which uses a stencil of 3×3 cells in order to fit a line segment inside of the central cell. Ashgriz and Poo [10] developed a method, referred to as FLAIR, based on fitting a line at the common side of two neighboring cells such that the liquid fluxes between the two cells are related with the actual slope and location of the interface.

In order to circumvent these problems, Brackbill et al. [17] developed a method referred to as the Continuum Surface Force (CSF) model. This model replaces the need to know the exact location of free surface by converting the surface tension effect into an equivalent volume force which is simply added to the Navier-Stokes equations as an additional body force. This force has smoothed 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:
\begin{equation}
F_{st}^{i,j,k} = \sigma \kappa \left( \frac{A_{i,j,k}}{V_{i,j,k}} \right) n_{i,j,k},
\end{equation}

where, $\sigma$ is the coefficient of the surface tension, $\kappa$ is the surface curvature, $n$ is the unit normal to the surface with the outward direction regarded as positive, $\delta (x-x_s)$ is the Dirac delta function, and $x_s$ 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. [17] led to the formation of artificial velocities (the so-called “spurious” or “parasitic” currents) due to an inaccurate representation of surface tension terms and associated pressure jump. These currents are strongly growing vertical flows in the transition region. In a paper by Brackbill and Kothe [18], they 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 [19] suggested another formulation of $F_{st}$, which has been adopted by Bussmann et al. [20]. The surface force per unit volume is computed only within each surface cell and is placed at the center of the cell:

\begin{equation}
\tilde{F}_{st}^{i,j,k} = K F_{st}^{i,j,k}. \tag{3}
\end{equation}

Significant effort has been put on improving the surface tension force predictions by using higher order kernels for more accurate estimations of curvature, $\kappa$, and unit normal vector, $n$, to the interface. In the original CSF method [17], the kernel is a quadratic $B$-spline. There are several other kernels [21-23], but the most widely used smoothing kernel is that proposed by Peskin [24]. Lafaurie et al. [25] 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 force are presented as a stress tensor, which is tangential to the interface.

The CSS method also produces numerical spurious effects as stated in [25]. In fact neither CSF nor CSS models produce very accurate numerical solution 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. New methods are needed to deal with this problem. There have been some attempts to reduce the spurious currents.
Popinet and Zaleski [26] 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. For more details, see ref. [26]. Their work is limited to the markers method.

Meier et al. [27] (and also Meier [28]) developed a new method to reduce the spurious currents. They have tried to improve 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 and by means of some weighted averaging; they calculated the surface tension force in the x- and y-momentum cells in their staggered grid configurations. Our method of calculation of the surface tension force is similar to what Meier et al. have done but we do not need to make the average of the forces in the two neighboring cells to get the force at the centers of x- and y-momentum cells. Also we calculate the value of normal vector to the interface more accurately. They indicated that their method is three to seven times more accurate than the CSF method of Brackbill et al. [17]. 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 [29] 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. [30] 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 surface and kinetic energies, the energy is conserved and the parasitic currents are reduced drastically.

Here, we present a method based on the calculation of the surface tension force at each x- and y-momentum cell centers of the interfacial cells as a volume force. This force is zero in any non-interfacial cells. The interface surface in each of the cells is assumed to be a straight line for two-dimensional case (or a plane surface in the three-dimensional case). The new method is then tested on the time evolution of a static drop, static bubble, and oscillating non-circular drop.

**Problem Formulation**

We consider 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:

\[
\frac{\partial u_i}{\partial x_i} = 0 , \tag{4}
\]

\[
\frac{\partial F}{\partial t} + u_i \frac{\partial F}{\partial x_i} = 0 , \tag{5}
\]

and

\[
\frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_j u_j}{\partial x_j} = - \frac{\partial \rho}{\partial x_i} + F^{st} \hat{i} + \mu \frac{\partial^2 u_i}{\partial x_j^2} + \rho g_i , \tag{6}
\]

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

\[
\rho = \rho_2 + F(\rho_1 - \rho_2) , \tag{7}
\]

and

\[
\mu = \mu_2 + F(\mu_1 - \mu_2) , \tag{8}
\]

where, \( \rho \) and \( \mu \) are density and viscosity of fluids, respectively and the subscripts 1 and 2 denote the two fluids involved.

If a pressure based numerical method is used (as is the case in this paper), then the Poisson equation needs to be solved. The Poisson equation is obtained by taking the divergence of the momentum equation, Eq. (6), and then simplified by the continuity equation, Eq. (4). The resulting equation is:

\[
\frac{\partial}{\partial x_i} \left( \frac{\partial p}{\partial x_i} \right) = - \frac{\partial}{\partial x_i} \left[ \frac{\partial}{\partial x_j} \left( \rho u_j u_j - 2\mu S_j - \frac{F^{st} \hat{i}}{\rho} \right) \right] \tag{9}
\]
Surface Tension Force

Following Brackbill et al.’s [17] continuum surface force (CSF) model, the interface curvature and the surface tension may be calculated as:

\[
\kappa(x) = -\nabla \cdot \hat{n}(x) = -\nabla \cdot \left( \frac{\nabla F(x)}{|\nabla F(x)|} \right),
\]

(10)

and

\[
F_{\nu}^{st} = \sigma \kappa \delta_x n = \sigma \kappa n \left[ \frac{\nabla F}{F} \right],
\]

(11)

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 \( F_{\nu}^{st} \) 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. [17, 31] 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. (11):

\[
F_{\nu}^{st} = \sigma \kappa \delta_x n = \sigma \kappa n \left[ \frac{\nabla F}{F} \right] \rho(x),
\]

(12)

where, \( \rho(x) \) is the local value of the density obtained by Eq. (7) and \([\rho]\) is the difference between the density of the heavier and the lighter fluids. The density correction term (the second fraction in Eq. (12)) is added to correct the force in the momentum equation. This damps 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 pressure jump when the jump in pressure is in the lighter fluid such as the bubble case. 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 [25, 32, 33, 34]. In this model, Eq. (12) is replaced with:

\[
F_{\nu}^{st} = -\nabla \cdot \mathbf{T} = \sigma \nabla \cdot \left( \left[ \nabla F \right] \mathbf{I} - \frac{\nabla F \otimes \nabla F}{|\nabla F|} \right).
\]

(13)

where, \( \otimes \) is tensor product.
Meier et al. [27] and Meier [28] 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 and then by means of some weighted averaging, as explained below, they calculated the force at the center of the x- and y-momentum cells. In Fig. (1), i and i+1 are the centers of the continuity and volume fraction cells and i+½ is the center of the x-momentum cell. Meier calculated the x-component of the volumetric surface tension force in cell i, \( F_{\text{vx},i,j}^{\text{st}} \), and in cell i+1, \( F_{\text{vx},i+1,j}^{\text{st}} \). Then he calculated the volumetric surface tension force for the x-momentum cell i+½, from the following relation.

\[
F_{\text{vx},i+1/2,j}^{\text{st}} = (0.5 + O_{x,i,j}) F_{\text{vx},i,j}^{\text{st}} + (0.5 + O_{x,i+1,j}) F_{\text{vx},i+1,j}^{\text{st}} 
\]

(14)

where, \( O_i \)'s are shown in Fig. (2). He used a similar formulation to calculate the volume surface tension force in the y direction, \( F_{\text{vy},i+1/2,j}^{\text{st}} \).

Our model is based on the calculation of the surface tension force directly in the x- and y-momentum cells. So we reconstruct the interface for the x- and y-momentum cells. In each of the cells the interface is assumed to be a straight line (for the two-dimensional problem), as shown in Fig. (1) for the x-component of the surface tension force. In this figure, the dashed line is the approximated interface location in the x-momentum cell located at i+½,j. To do this, we need to calculate the values of \( H_{x,i} \) and \( H_{x,i+1} \). To show the formulation, let’s start with the volumetric surface tension force acting on the cell i+½,j. By using Eq. (2), we have:

\[
F_{\text{vx},i+1/2,j}^{\text{st}} = \frac{\sigma \kappa_{i+1/2,j} A_{x,i+1/2,j}}{\Delta x \Delta y} 
\]

(15)

where, \( A_{x,i+1/2,j} \) is the projection of the interface surface area in the cell i+½,j, in the x-direction (that is on the y-axis) and is (see Fig. (1)):

\[
A_{x,i+1/2,j} = H_{x,i+1,j} - H_{x,i,j} 
\]

(16)

Thus, Eq. (15) becomes:

\[
F_{\text{vx},i+1/2,j}^{\text{st}} = \frac{\sigma \kappa_{i+1/2,j} (H_{x,i+1,j} - H_{x,i,j})}{\Delta x \Delta y} 
\]

(17)

or one may be able to deduce the above equation in the following form which is cell size free.

\[
F_{\text{vx},i+1/2,j}^{\text{st}} = \sigma \kappa_{i+1/2,j} \frac{\partial H_i'}{\partial x} 
\]

(18)
where, $H'_x$ is the relative length of $H_x$, that is $H_x/\Delta y$. However, Eq. (19) may not be true for general cases therefore we use Eq. (17) in this work. Similarly, for the $y$-component of the surface tension force in the $y$-momentum cell centered at $i,j+\frac{1}{2}$, we have:

$$F_{vy,i,j+1/2}^{st} = \frac{\sigma \kappa_{i,j+1/2} (H_{y,i,j+1} - H_{y,i,j})}{\Delta x \Delta y}$$  

or, similar to equation (18), we have

$$F_{vy,i,j+1/2}^{st} = \sigma \kappa_{i,j+1/2} \frac{\partial H'_y}{\partial y}$$  

It is obvious that the interface functions $H_x$ and $H_y$ are such that the values of $F_{v}^{st}$'s in equations (17) and (19) are only 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, Eq. (14), the entire neighboring interfacial cells in Meier’s method will have non-zero surface tension force.

**Calculation of Interface Function $H_k$**

The values of $H_x$ and $H_y$ are related to the location of interface at a cell and since we are using the PLIC technique, in cells $i,j$, the interface shape is a straight line and its formulation is known. So the calculation of the values of $H_{xi,j}$ and $H_{yi,j}$ is straightforward and can be done with negligible computational cost.

Let’s start with the reconstruction process in the piecewise linear interface calculation PLIC technique. In the PLIC method, the interface is approximated by a straight line of an appropriate inclination in each cell. The straight lines are not connected to each other at the cell faces. That is, the interface line at each cell is determined independent of the neighboring interface lines, and their ends need not necessarily be connected at the cell faces. Each line is determined so that it is perpendicular to an interface normal vector, and it divides the cell surface into two regions that match the given $F$ for the cell. For more details see Youngs [9]. Therefore, the interface normal vector $n$, (a unit vector perpendicular to the interface) is to be determined for each cell. This is achieved using the gradient of $F$:

$$n = -\frac{\nabla F}{|\nabla F|},$$  

where, the gradient of $F$ at each point is calculated using the values of $F$ in its immediate nine neighboring points. The nine neighboring points of point $i,j$ are:
\[ F_{i,j} = \begin{bmatrix} F_{i-1,j+1} & F_{i,j+1} & F_{i+1,j+1} \\ F_{i-1,j} & F_{i,j} & F_{i+1,j} \\ F_{i-1,j-1} & F_{i,j-1} & F_{i+1,j-1} \end{bmatrix}. \] (22)

Assuming \( \Delta x = \Delta y = h \), then the \( x \) and \( y \) components of the gradient of \( F_{i,j} \) are:

\[ m_{x,i,j} = \left[ F_{i+1,j+1} - F_{i-1,j+1} + 2(F_{i+1,j} - F_{i-1,j}) + F_{i+1,j-1} - F_{i-1,j-1} \right]/h, \] (23)

and

\[ m_{y,i,j} = \left[ F_{i+1,j+1} - F_{i+1,j-1} + 2(F_{i,j+1} - F_{i,j-1}) + F_{i+1,j+1} - F_{i-1,j+1} \right]/h. \] (24)

And the \( x \) and \( y \) components of the unit normal vector are:

\[ n_{x,i,j} = -\frac{m_{x,i,j}}{\sqrt{m_{x,i,j}^2 + m_{y,i,j}^2}}, \] (25)

and

\[ n_{y,i,j} = -\frac{m_{y,i,j}}{\sqrt{m_{x,i,j}^2 + m_{y,i,j}^2}}, \] (26)

where, \( n_x \) and \( n_y \) are components of \( n \). Once the normalized unit vector \( n \) is calculated, a straight line (the solid declined lines in Fig. (1)) is positioned perpendicular to it in such a way that it matches with the value of \( F \) in the cell.

Figure (3) shows that depending on the orientation of interface, eight different cases may occur. The normal vector angle, \( \theta \), (shown in Fig. (4)) can take any value between zero and \( 2\pi \). When \( n \) is in the first octant \( (0 \leq \theta \leq \pi/4) \), the different cases that may occur are shown in Fig. (4). All the other cases can be obtained in an equivalent situation with \( n \) in the first octant by mirroring appropriately on x-axis, y-axis and the bisector between them. In Fig. (4), \( a \) and \( b \) are vectors and by knowing them, \( H_k \) can easily be obtained (see Fig. (5)). In order to calculate \( H_k \), 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 \( n \times \vec{a} \vec{b} \) is positive. To calculate the components of vectors \( a \) and \( b \), we need to specify the limiting values of \( F \) for a particular \( n \).
\[ F_{\lim,1} = \frac{n_{\min}}{2n_{\max}} \quad \text{and} \quad F_{\lim,2} = 1 - F_{\lim,1}, \quad (27) \]

where,

\[ n_{\min} = \min(|n_x|,|n_y|) \quad \text{and} \quad n_{\max} = \max(|n_x|,|n_y|). \quad (28) \]

For simplicity, assume \( \Delta x = \Delta y = 1 \), then the components of vectors \( \mathbf{a} \) and \( \mathbf{b} \) can be defined as:

a) For \( F \leq F_{\lim,1} \) (triangle):

\[ a_x = \sqrt{2F \frac{n_{\min}}{n_{\max}}} \quad a_y = 0 \quad b_x = 0 \quad b_y = \frac{2F}{a_x} \quad (29) \]

b) For \( F_{\lim,1} < F \leq F_{\lim,2} \) (quadrilateral):

\[ a_x = F + \frac{n_{\min}}{2n_{\max}} \quad a_y = 0 \quad b_x = F - \frac{n_{\min}}{2n_{\max}} \quad b_y = 1. \quad (30) \]

c) For \( F > F_{\lim,2} \) (pentagon):

\[ a_x = 1 \quad a_y = 1 - \sqrt{2(1-F) \frac{n_{\max}}{n_{\min}}} \quad b_x = 1 - \frac{2(1-F)}{1-a_y} \quad b_y = 1. \quad (31) \]

Now we can calculate \( H_x \) and \( H_y \) easily. As an example, for the cell shown in Fig. (5), \( H_x \) is:

\[ H_x = \frac{a_x \frac{1}{2}}{a_x - b_x} b_y \quad (32) \]

Similarly for the same cell \( H_y \) is:

\[ H_y = \frac{a_x + a_y}{2} \quad (33) \]
Results and Discussions

Three cases are examined here. A circular drop of water in the air, bubble of air in water and a non-circular drop of water in the air. The first two cases are static tests and drop, bubble and their surrounding fluids are initially at rest and the gravity is absent. The exact solutions for these two cases are known. That is, no fluid motion has to be generated and the pressure difference inside of the drop or bubble and their surrounding must obey Laplace's equation. But due to numerical errors, especially inaccurate calculation of curvature and the surface tension force, some spurious currents, especially in the lighter fluid, are produced. They increase with time. The currents very much depend on the density ratio of the two fluids; as the density ratio (density of heavier fluid to the lighter fluid) increases, the magnitude of the currents increases drastically; see Jafari et al. [35]. The third case is a dynamic case and the oval shape drop of water due to the unsymmetric surface tension force starts to oscillate until it becomes circular.

For all cases, a two-dimensional SURFER [25, 32 and 33] code has been used. This code is based on the two-fluid PLIC-VOF method and is pressure based with Chorin’s projection method for a semi-implicit Navier-Stokes solver, which uses staggered equally spaced grid and advances the Poisson equation with a multigrid solver. The properties of air and water was used for T=20°C. That is $\rho_1=1000 \text{ kg/m}^3$, $\mu_1= 0.001 \text{ N.s/m}^2$, $\rho_2 = 1.204 \text{ kg/m}^3$, $\mu_2 = 1.82\times10^{-5} \text{ N.s/m}^2$ and $\sigma = 0.073 \text{ N/m}$, where the subscript 1 denotes for water and subscript 2 stands for air. The relative diameter of the drop or bubble (D/L, L is the domain length) was assumed 0.25 and for the dynamic case, the larger and smaller relative diameters of the oval are assumed 0.4 and 0.12. The number of grid points is 64×64 and the time step was kept constant and it was chosen such that the stability criteria as well as the required accuracy obtained. The results, for the present method as well as for CSS, CFS and Meier’s technique, are presented in this paper. The method of calculation of the curvature for all cases is the same and the Meier’s technique for the calculation of the curvature was not used in this work. The results of different models are compared and some discussions about the results are presented.

a. Static case

In order to show the accuracy of the pressure jump obtained in each calculation, the following equation, which is the relative error in the calculation of pressure jump, is used.

$$\text{error} = \frac{(\bar{p}_1 - \bar{p}_2) - \sigma}{\frac{\sigma}{R}}$$

(34)

where, $\bar{p}_1$ and $\bar{p}_2$ are the mean values of pressure inside and outside of the drop or bubble and $R$ is the radius of drop or bubble. Note that if the pressures $\bar{p}_1$ and $\bar{p}_2$ were calculated exactly, the value of error would be zero.
Figure (6) shows the evolution of error in pressure jump for the drop when our model as well as Meier’s model and the CSF model with the density correction are used. As is shown, our model has a maximum of 5% error whereas Meier’s model produces about 15% error and the CSF model with density correction produces about -16% error. In figure (7) the error for CSS model and CSF model without the density correction are plotted as a function of time. In this figure the results are shown for only up to 0.8s. The reason is that as time increases the results become unphysical, the error become 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. (8) and (9). The behavior of our model for the bubble case is much better than the other models. The maximum error for our model is about 8%, whereas for the Meier model it is -50% and for CSF with the density correction it 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 and the maximum error becomes about -10%. The reason that the CSF model with the density correction for the bubble case produces more inaccurate results is that, in the bubble case, unlike the drop case, 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.

Figures (10) and (11) show the contour lines at $F=0.5$ for drop after 10 seconds and bubble respectively. The results for all five cases -our model, Meier’s model, CSF with and without density correction and CSS model- are shown. For the drop case only our model and Meier’s model produce reasonable results. Also the CSF model with the density correction produces nearly physical results. For the bubble case only our model produces physical results. Finally Figs. (12) and (13) show the maximum and norm spurious velocities for the drop case, where norm spurious current is defined as the average of absolute velocity in the whole flow field. As shown in these figures, the spurious currents produced by our model, Meier’s model and the CSF model with density correction are about two orders of magnitude less than that the CSS and CSF without the density correction. The reason for the distortion of the drop shape when CSS and CSF models without the density correction are used, as shown in Fig. (10), is due to the large parasitic currents that these models produce. Similarly, the maximum and norm velocity of spurious currents for bubble are shown in Figs. (14) and (15) and the trend of the spurious currents are similar to the drop.

b. Dynamic case

Figures (16), (17), (18), (19) and (20) show the oscillation of the initially oval shape drop, when our model, Meier’s model, CSF model with the density correction, CSS model and CSF model without the density correction are used, respectively. The results for the first three cases are shown for up to 38.8 seconds, whereas the other two cases are shown up to 4 seconds. The reason for not showing the results of the last two cases for a longer period is that they become unphysical and the drop starts to break up. Our model and Meier’s model produce similar results, but since the CSF model with density correction under predicts the
pressure jump (the driving force for the oscillation) the surface tension force is lower than what it should be and that is why the oscillations obtained from the application of this model take place with delay compared to our own results. To see the accuracy of our result for the dynamic case, we compared the period of oscillations with the analytical solution obtained for small perturbation of a column of liquid with zero viscosity. Such analytical solution was obtained by Lamb [36] and is:

\[ \omega^2 = m(m^2 - 1) \frac{\sigma}{\rho a^3} \quad (35) \]

where, \( a \) is the mean radius, \( m \) is the mode and \( \omega \) is the frequency of the oscillations and is related to the period of oscillations \( (T = 2\pi / \omega) \). From this relation, the period of oscillations for the mode corresponding to the oval shape \( m=2 \) and \( a=0.109 \) (which is correspond to our test case), is 10.88s. Our method gives \( T=12.93s \) (18.8% difference compared to analytical results) and CSF model with the density correction gives \( T=16.4s \) (50.7% difference compared to analytical results). We see that our result is much closer that the analytical results and the differences are mostly due to the assumptions (inviscid flow and small perturbation) made to obtain the analytical solution.

Conclusions

A new method for implementing the surface tension force in the PLIC-VOF method when stagger grid is used was introduced. In this method, the interface locations and the interface surface area at the x- and y-momentum control volume are determined. Then the surface tension force as a volumetric force for the interfacial cells which are used to solve the momentum equations are calculated and added to the momentum equations.

The model was used to simulate static drop and bubble and also an oscillating initially oval shape drop. Similar problems are also simulated using Meier’s model of surface tension force, 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 shape of the drop or 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 when the density correction factor is used in the bubble case, the pressure jump produced by this model has about 100% error. The reason is that by multiplying the density factor to the surface tension force, we actually make this force to be nearly zero inside the lighter fluid. This is despite of the fact that the pressure jump occurs inside the bubble and one should not make the surface tension force in this area to be too small. Finally, the results for the dynamic case show that although the CSS and CSF model without the density correction factor do not produce physical results for this case, the new model is able to produce reasonable results.

Acknowledgments

This work was done within the framework of the Abdus Salam International Centre for Theoretical Physics, Trieste, Italy.
References


Figure 1 – Interface in a x-momentum cell (dotted line) and interface function $H_y$

Figure 2 – Offset vectors $o$ used to weight the partition of the surface tension body force on the staggered x-momentum cell

Figure 3 - Different configurations for an interface in a cell
Figure 4 - Different configurations for an interface in a cell.

Figure 5 - $H_x$ for a unit length cell

Figure 6 - Error for pressure jump calculation of drop as a function of times

Figure 7 - Error for pressure jump calculation of drop as a function of times
Figure 8 - *error* for pressure calculation of bubble as a function of time

Figure 9 - *error* for pressure calculation of bubble as a function of time
Figure 10- F=0.5 contour lines for drop.

a) at t=0;  b) new model;  c) Meier’s mode;  d) CSF with rho correction;
   e) CSF without rho correction;  f) CSS
Figure 11 - $F=0.5$ contour lines for bubble
a) at $t=0$;  b) new model;  c) Meier’s model;  d) CSF with rho correction;
e) CSF without rho correction;  f) CSS
Figure 12 - Maximum spurious velocity for drop

Figure 13 - Norm spurious velocity for drop

Figure 14 - Max spurious velocity for bubble

Figure 15 - Norm spurious velocity for bubble
Figure 16 - Oscillation of non-circular drop of water in air, within 38.8s, new model

Figure 17 - Oscillation of non-circular drop of water in air, within 38.8s, Meier’s model

Figure 18 - Oscillation of a drop of water in air within 38.8s, CSF with rho correction
Figure 19 - Oscillation of non-circular drop of water in air, within 4s, CSS

Figure 20 - Oscillation of a water drop in air, CSF w/o rho correction within 4s