

# A sharp interface method for incompressible two-phase flows

M. Sussman <sup>\*,1</sup>, K.M. Smith, M.Y. Hussaini, M. Ohta, R. Zhi-Wei

*Department of Mathematics, Florida State University, 208 Love Bldg., 002C Love Bldg., 487 DSL, Tallahassee, FL 32306-451, United States*

Received 1 November 2005; received in revised form 8 June 2006; accepted 16 June 2006

Available online 27 July 2006

---

## Abstract

We present a sharp interface method for computing incompressible immiscible two-phase flows. It couples the level-set and volume-of-fluid techniques and retains their advantages while overcoming their weaknesses. It is stable and robust even for large density and viscosity ratios on the order of 1000 to 1. The numerical method is an extension of the second-order method presented by Sussman [M. Sussman, A second order coupled levelset and volume of fluid method for computing growth and collapse of vapor bubbles, *Journal of Computational Physics* 187 (2003) 110–136] in which the previous method treated the gas pressure as spatially constant and the present method treats the gas as a second incompressible fluid. The new method yields solutions in the zero gas density limit which are comparable in accuracy to the method in which the gas pressure was treated as spatially constant. This improvement in accuracy allows one to compute accurate solutions on relatively coarse grids, thereby providing a speed-up over continuum or “ghost-fluid” methods.

© 2006 Elsevier Inc. All rights reserved.

*MSC:* 65M06; 76D05; 76T05

*Keywords:* Incompressible flow; Immiscible fluids; Navier–Stokes equations; Multiphase flows; Numerical methods

---

## 1. Introduction

Efficient and accurate computation of incompressible two-phase flow problems has enormous value in numerous scientific and industrial applications. Applications include ship hydrodynamics, viscoelastic free surface flows, and liquid jets [10,14,15,48,36]. Current methods for the “robust” computation of immiscible two-phase flows [51,50,9,53,32,26,20,18] are all essentially spatially first-order accurate as the treatment of the interfacial jump conditions constrains the overall accuracy to first-order. Robustness is defined in terms of the ability of a numerical method to stably handle wide ranges of physical and geometrical parameters. We note that Hellenbrook et al. [21] developed a formally second-order level set method for two-phase flows, but the applications did not include large density ratios, surface tension, or complex geometries. It is unlikely that a straightforward application is possible to flow configurations with such wide parameter ranges. We also note that Ye et al. [60] presented a second-order Cartesian grid/front tracking method for two-phase flows, but

---

<sup>\*</sup> Corresponding author. Tel.: +1 850 644 7194; fax: +1 850 644 4053.

*E-mail address:* [sussman@math.fsu.edu](mailto:sussman@math.fsu.edu) (M. Sussman).

<sup>1</sup> Work supported in part by the National Science Foundation under contracts DMS 0108672, U.S. Japan Cooperative Science 0242524.

their results did not include complex geometries. Yang and Prosperetti [57] presented a second-order boundary-fitted tracking method for “single-phase” (free boundary problem) flows, but similarly as with Ye et al. [60], their results did not include complex geometries.

Although the formal order of accuracy of continuum approaches [53,9,51,38,29] or ghost-fluid approaches [26,28] is second-order, numerical dissipation at the free surface reduces the order to first-order. We propose a new method which extends the functionality of the method discussed in [45] from single-phase (pressure assumed spatially constant in the air) to multiphase (gas solution assumed incompressible). The resulting matrix system(s) are symmetric, guaranteeing robustness of the method, and are capable of stably handling wide parameter ranges (e.g. density ratio 1000:1, large Reynolds number) and geometries (e.g. topological merging and breaking). The method is consistent in that it captures the limiting cases of zero gas density and linear slip lines. Specifically, the present method reduces to the single fluid method [45] in the limit that the gas density and gas viscosity approach zero (i.e. the numerical solution of the gas phase approaches the condition of spatially constant pressure as the gas density approaches zero). The present method provides additional functionality over single fluid methods since one can accurately compute bubble entrainment, bubble formation, effect of wind on water, liquid jets, etc. Further, we demonstrate that the present method provides improved accuracy over existing two-fluid methods for a given grid, and provides a speed-up over existing methods for a given accuracy, as we can robustly compute flows on coarser meshes.

## 2. Governing equations

We consider the incompressible flows of two immiscible fluids (such as liquid/liquid or liquid/gas), governed by the Navier–Stokes equations:

$$\begin{aligned} \rho \frac{DU}{Dt} &= \nabla \cdot (-pI + 2\mu D) + \rho g \hat{z} \\ \nabla \cdot U &= 0 \end{aligned}$$

where  $U$  is the velocity vector,  $\rho$  is the density,  $p$  is the pressure,  $\mu$  is the coefficient of viscosity,  $g$  is the gravity,  $I$  is the unit tensor,  $\hat{z}$  is the unit vector in the vertical direction, and  $D$  is the deformation tensor defined by

$$D = \frac{\nabla U + (\nabla U)^T}{2}$$

At the interface,  $\Gamma$ , separating the two fluids, we have the normal continuity condition for velocity,

$$[U \cdot n] \equiv U^L \cdot n - U^G \cdot n = 0$$

we also have the tangential continuity condition for velocity (if viscous effects are present),

$$[U] = 0$$

and the jump condition for stress,

$$[n \cdot (-pI + 2\mu D) \cdot n] = \sigma \kappa$$

where  $n$  is the unit normal to the interface,  $\sigma$  is the coefficient of surface tension and  $\kappa$  is the local curvature.

Following the derivation in [12], we can rewrite the preceding governing equations in terms of the following equations based on the level set function  $\phi$ . In other words, analytical solutions to the following level set equations are also solutions to the original governing Navier–Stokes equations for two-phase flow. Our resulting numerical method will be based on the level set formulation.

If one defines the interface  $\Gamma$  as the zero level set of a smooth level set function,  $\phi$ , then the resulting equations are:

$$\rho \frac{DU}{Dt} = \nabla \cdot (-pI + 2\mu D) + \rho g \hat{z} - \sigma \kappa \nabla H \quad (1)$$

$$\nabla \cdot U = 0$$

$$\frac{D\phi}{Dt} = 0 \quad (2)$$

$$\begin{aligned} \rho &= \rho_L H(\phi) + \rho_G(1 - H(\phi)) \\ \mu &= \mu_L H(\phi) + \mu_G(1 - H(\phi)) \\ \kappa(\phi) &= \nabla \cdot \frac{\nabla \phi}{|\nabla \phi|} \end{aligned} \tag{3}$$

$$H(\phi) = \begin{cases} 1, & \phi \geq 0 \\ 0, & \phi < 0 \end{cases} \tag{4}$$

### 3. CLSVOF free surface representation

The free surface is represented by a “coupled level set and volume-of-fluid” (CLSVOF) method [50]. In addition to solving the level set equation (2), we also solve the following equation for the volume-of-fluid function  $F$ ,

$$\frac{DF}{Dt} = F_t + \mathbf{U} \cdot \nabla F = 0 \tag{5}$$

(5) is equivalent to

$$F_t + \nabla \cdot (\mathbf{U}F) = (\nabla \cdot \mathbf{U})F$$

Since  $\nabla \cdot \mathbf{U} = 0$ , we have

$$F_t + \nabla \cdot (\mathbf{U}F) = 0 \tag{6}$$

At  $t = 0$ ,  $F$  is initialized in each computational cell  $\Omega_{ij}$ ,

$$\Omega_{ij} = \{(x, y) | x_i \leq x \leq x_{i+1} \text{ and } y_j \leq y \leq y_{j+1}\}$$

as,

$$F_{ij} = \frac{1}{\Delta x \Delta y} \int_{\Omega_{ij}} H(\phi(x, y, 0)) \, dx \, dy$$

Here,  $\Delta x = x_{i+1} - x_i$  and  $\Delta y = y_{i+1} - y_i$ .

The reasons why we couple the level set method to the volume-of-fluid method are as follows:

- If one discretizes the level set equation (2), even in conservation form, the volume enclosed by the zero level set will not be conserved. This problem has been addressed by implementing global mass fixes [12], augmenting the level set equation by advecting massless particles [16,24], implementing adaptive mesh refinement techniques [43,47], and by implementing high order “spectral” methods [49,30]. In this paper, we preserve mass by coupling the level set method to the volume-of-fluid method [8,50,58]; effectively, by coupling the two, we are implementing a “local” mass fix instead of a “global” mass fix. The volume-of-fluid function  $F$  is used to “correct” the mass enclosed by the zero level set of  $\phi$  during the level set redistancing step (see Fig. 1).
- If one uses only the volume-of-fluid function  $F$  to represent the interface separating air and water, then one must be able to accurately extract the normal and curvature from  $F$ . Also, small pieces of volume might separate from the free surface which can pollute the solution for the velocity. Modern volume-of-fluid methods have addressed these problems [33,22,18] using second-order slope reconstruction techniques and calculating the curvature either from a “height fraction” or from a temporary level set function. The level set function in our implementation is used for calculating the interface normal and is used for calculating density and viscosity used by the Navier–Stokes equation. The level set function is *not* used for calculating the interface curvature; instead we use the volume-of-fluid function.

To clarify what information we extract from the level set function  $\phi$ , and what information we extract from the volume-of-fluid function  $F$ , we have:

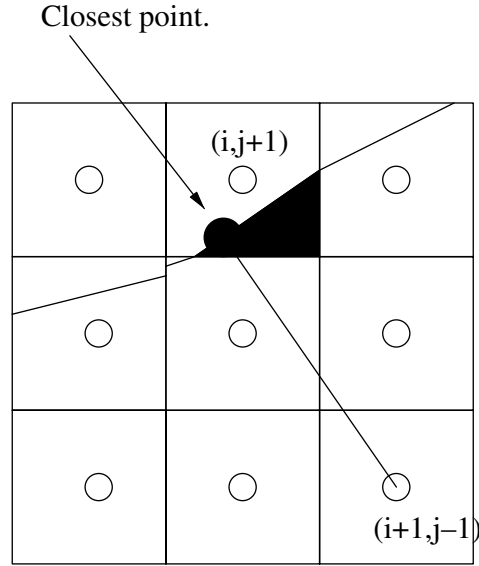


Fig. 1. After each time step, the level set function  $\phi$  is reinitialized as the closest distance to the piecewise linear reconstructed interface. The linear reconstruction encloses the volume given by  $F$  with its slope given by  $\mathbf{n} = \frac{\nabla\phi}{|\nabla\phi|}$ . In this diagram, the shaded area fraction is  $F_{i,j+1}$ , the distance from point  $\mathbf{x}_{i+1,j-1}$  to the closest point becomes the new value of  $\phi_{i+1,j-1}$ .

- The normals used in the volume-of-fluid reconstruction step are determined from the level set function (see e.g. Fig. 2).
- The “height fraction” (see Section 5.3) and velocity extrapolation calculations (see Section 5.6) both depend on the level set function. Therefore, cells in which  $F$  is very close to either 0 or 1 will not directly effect the accuracy of the solution to the momentum equations.
- The volume fractions are used, together with the slopes from the level set function, to construct a “volume-preserving” distance function along with providing “closest point” information to the zero level set (see Fig. 1).

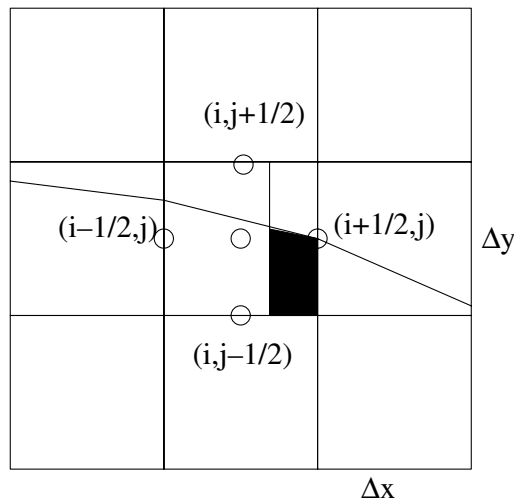


Fig. 2. In order to calculate the volume-of-fluid flux,  $F_{i+1/2,j}$ , one must first find the linear coupled level set and volume-of-fluid reconstructed interface. The flux across a face becomes the volume fraction of overall volume that is advected across a face. For this illustration,  $F_{i+1/2,j} = \frac{\text{shaded area}}{u_{i+1/2,j}\Delta t\Delta y}$ .

- The volume fractions are used to express the interfacial curvature to second-order accuracy (see Section A.2). We do not use the level set function for finding the curvature because our level set reinitialization step is only second-order accurate; the curvature as computed from the level set function will not provide the second-order accuracy that is provided directly from the volume fractions.

We observe that there are possibly more accurate representations of the interface [49,41,16,4,3,40]. However, it must be noted that the accuracy of the computations is limited by the order of accuracy of the treatment of the interfacial boundary conditions and not by the accuracy of the interface representation. Even if the interface representation is exact, if the velocity used to advance the interface is low order accurate, then the overall accuracy is constrained by the accuracy at which the velocity field (specifically, the velocity field at the interface) is computed. Our results in Sections 6.3 and 6.4 support this hypothesis. We demonstrate second-order accuracy for interfacial flows in which only first-order methods have been previously applied. We also show that we conserve mass to a fraction of a percent in our computations (e.g. largest mass fluctuation on coarsest grid in Section 6.3 was 0.08%).

When implementing the CLSVOF method, the discrete level set function  $\phi_{ij}^n$  and discrete volume fraction function  $F_{ij}^n$  are located at cell centers. The motion of the free surface is determined by the face centered velocities,  $u_{i+1/2,j}$  and  $v_{i,j+1/2}$ , which are derived from the momentum equation. A scalar quantity with the subscript  $ij$  implies that the quantity lives at the cell center  $(x_i, y_j)$ ,

$$\begin{aligned} x_i &= x_{i0} + (i + 1/2)\Delta x \\ y_j &= y_{j0} + (j + 1/2)\Delta y \end{aligned}$$

A scalar quantity with the subscript  $i + 1/2, j$  implies that the quantity lives at the right face center of cell  $ij$ ,

$$\begin{aligned} x_{i+1/2} &= x_{i0} + (i + 1)\Delta x \\ y_j &= y_{j0} + (j + 1/2)\Delta y \end{aligned}$$

A scalar quantity with the subscript  $i, j + 1/2$  implies that the quantity lives at the top face center of cell  $ij$ ,

$$\begin{aligned} x_i &= x_{i0} + (i + 1/2)\Delta x \\ y_{j+1/2} &= y_{j0} + (j + 1)\Delta y \end{aligned}$$

A vector quantity with the subscript  $i + 1/2, j$  implies that the first component lives at the right face center of a cell,  $(x_{i+1/2}, y_j)$ , and that the second component lives at the top face center of a cell,  $(x_i, y_{j+1/2})$ . A diagram illustrating where our discrete variables live is shown in Fig. 3.

The discrete face centered velocity field is assumed to satisfy the discrete continuity condition at every point in the liquid ( $\phi_{ij} \geq 0$ ):

$$(\text{Div } \mathbf{U})_{ij} = \frac{u_{i+\frac{1}{2},j} - u_{i-\frac{1}{2},j}}{\Delta x} + \frac{v_{i,j+\frac{1}{2}} - v_{i,j-\frac{1}{2}}}{\Delta y} = 0 \tag{7}$$

To integrate the solution for both the level set function  $\phi$  and the volume-of-fluid function  $F$ , we first simultaneously solve (6) and (2). Then, we reinitialize  $\phi$  by constructing a distance function that shares the same enclosed volume as determined from  $F$ , and the same slopes as determined from  $\phi$ .

Both the level set equation and the volume-of-fluid equation are discretized in time using second-order ‘‘Strang splitting’’ [44] where for one time step we sweep in the  $x$  direction then the  $y$  direction, then for the next time step, we sweep in the  $y$  direction, then the  $x$  direction. Assuming that the advective velocity is independent of time, this procedure is equivalent to solving for the  $x$  direction terms for  $\Delta t$  time, solving  $y$  direction terms for  $2\Delta t$  time, then solving for the  $x$  direction terms again for  $\Delta t$  time.

The spatial operators are split, where one alternates between sweeping in the  $x$  direction:

$$\begin{aligned} \frac{F_{ij}^* - F_{ij}^n}{\Delta t} + \frac{u_{i+1/2,j} F_{i+1/2,j}^n - u_{i-1/2,j} F_{i-1/2,j}^n}{\Delta x} &= F_{ij}^* \frac{u_{i+1/2,j} - u_{i-1/2,j}}{\Delta x} \\ \frac{\phi_{ij}^* - \phi_{ij}^n}{\Delta t} + \frac{u_{i+1/2,j} \phi_{i+1/2,j}^n - u_{i-1/2,j} \phi_{i-1/2,j}^n}{\Delta x} &= \phi_{ij}^* \frac{u_{i+1/2,j} - u_{i-1/2,j}}{\Delta x} \end{aligned} \tag{8}$$

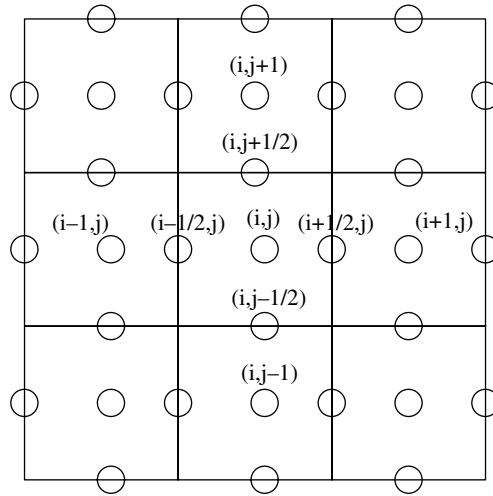


Fig. 3. Cell centered quantities,  $\phi_{ij}, F_{ij}, p_{ij}$ , live at the cell locations  $(i, j), (i + 1, j), (i, j + 1)$ , etc. The horizontal MAC velocity,  $u_{i+1/2,j}$ , lives at the vertical face centroids,  $(i - 1/2, j), (i + 1/2, j)$ , etc. The vertical MAC velocity,  $v_{i,j+1/2}$  lives at the horizontal face centroids,  $(i, j - 1/2), (i, j + 1/2)$ , etc.

and in the  $y$  direction:

$$\frac{F_{ij}^{n+1} - F_{ij}^*}{\Delta t} + \frac{v_{i,j+1/2}F_{i,j+1/2}^* - v_{i,j-1/2}F_{i,j-1/2}^*}{\Delta y} = F_{ij}^* \frac{v_{i,j+1/2} - v_{i,j-1/2}}{\Delta y} \tag{9}$$

$$\frac{\phi_{ij}^{n+1} - \phi_{ij}^*}{\Delta t} + \frac{v_{i,j+1/2}\phi_{i,j+1/2}^* - v_{i,j-1/2}\phi_{i,j-1/2}^*}{\Delta y} = \phi_{ij}^* \frac{v_{i,j+1/2} - v_{i,j-1/2}}{\Delta y}$$

The volume-of-fluid fluxes,  $F_{i+1/2,j}$  and  $F_{i,j+1/2}$ , are calculated as the fraction of liquid fluid to the overall fluid that is advected across a given cell face during a timestep (see Fig. 2). The level set fluxes,  $\phi_{i+1/2,j}$  and  $\phi_{i,j+1/2}$  are calculated by extrapolating the level set function in space and time to get a time-centered flux at given cell faces. Details are presented in [50,45].

If we add 8 to 9, then we have,

$$\frac{F_{ij}^{n+1} - F_{ij}^n}{\Delta t} + \frac{u_{i+1/2,j}F_{i+1/2,j}^n - u_{i-1/2,j}F_{i-1/2,j}^n}{\Delta x} + \frac{v_{i,j+1/2}F_{i,j+1/2}^* - v_{i,j-1/2}F_{i,j-1/2}^*}{\Delta y}$$

$$= F_{ij}^* \left( \frac{u_{i+1/2,j} - u_{i-1/2,j}}{\Delta x} + \frac{v_{i,j+1/2} - v_{i,j-1/2}}{\Delta y} \right) \tag{10}$$

If the right hand side of (10) is zero, then  $F$  shall be conserved since the left hand side of (10) is written in conservation form. In other words, if the discrete divergence free condition (7) is satisfied, then we have mass conservation. A key distinction between the two-phase algorithm we present here and previous sharp interface methods is that solutions derived from our method will approach the solutions of the corresponding one-phase method in the limit that the vapor is assumed to have uniform pressure. In order to achieve this goal, we implement a liquid velocity extrapolation procedure in which we extrapolate the liquid velocities into the gas (therefore, we shall store two separate velocity fields). The extrapolated liquid velocity may not satisfy (7) in vapor cells ( $\phi_{ij} < 0$ ). In order to maintain conservation of  $F$ , we have the additional step,

$$F_{ij}^{n+1} = F_{ij}^{n+1} - \Delta t F_{ij}^* \left( \frac{u_{i+1/2,j} - u_{i-1/2,j}}{\Delta x} + \frac{v_{i,j+1/2} - v_{i,j-1/2}}{\Delta y} \right) \tag{11}$$

The resulting advection procedure for  $F$  now becomes,

$$\frac{F_{ij}^{n+1} - F_{ij}^n}{\Delta t} + \frac{u_{i+1/2,j}F_{i+1/2,j}^n - u_{i-1/2,j}F_{i-1/2,j}^n}{\Delta x} + \frac{v_{i,j+1/2}F_{i,j+1/2}^* - v_{i,j-1/2}F_{i,j-1/2}^*}{\Delta y} = 0$$

We remark that in [45], we required that Eq. (7) hold in both liquid cells ( $\phi_{ij} \geq 0$ ) and extrapolated cells; this requirement necessitated an “extrapolation projection” step. In this work, we relax this condition and instead use (11).

**4. Temporal discretization: Crank–Nicolson/TVD Runge–Kutta, projection method**

Our temporal discretization procedure for approximating Eq. (1) is based on a combination of the Crank–Nicolson projection procedure (see e.g. [5,6]) for the viscous terms and the second-order TVD preserving Runge–Kutta procedure [42] for the nonlinear advective terms.

Our method follows loosely the outline below:

*Sweep 1*

$$U^{n+1,(0)} = U^n + \Delta t F(U^n) + \Delta t \frac{G(U^n) + G(U^{n+1,(0)})}{2} - \Delta t \text{Grad} P^n \tag{12}$$

*Sweep 2*

$$U^{n+1,(1)} = U^n + \Delta t F(U^{n+1,(0)}) + \Delta t \frac{G(U^n) + G(U^{n+1,(1)})}{2} - \Delta t \text{Grad} P^{n+1}$$

$$U^{n+1} = \frac{U^{n+1,(0)} + U^{n+1,(1)}}{2} \tag{13}$$

where  $F$  corresponds to the nonlinear advective terms,  $G$  corresponds to the viscous terms and  $\text{Grad} P$  corresponds to the pressure gradient term. To be more specific, we describe one sweep of our method below.

Prior to each timestep we are given a liquid velocity,  $U^{L,n}$ , and a total velocity  $U^n$ . The main distinction between our method and previous sharp interface methods is that we store  $U^{L,n}$  in addition to storing  $U^n$ . In a given time step, immediately after solving for  $U^{n+1}$ , we construct  $U^{L,n+1}$ ,

$$u_{i+1/2,j}^L = \begin{cases} u_{i+1/2,j} & \phi_{ij} \geq 0 \text{ or } \phi_{i+1,j} \geq 0 \\ u_{i+1/2,j}^{\text{extrapolate}} & \text{otherwise} \end{cases}$$

In other words,  $U^L$  corresponds to  $U$  except on gas faces, where we replace the gas velocity in  $U^L$  with the extrapolated liquid velocity.  $U^L$  is then used to calculate the nonlinear advective terms in the liquid, and also used to advance the free surface.

Prior to each time step, we are also given a “live pressure gradient”,

$$\text{Grad} P^n \approx \left( \frac{\nabla p + \sigma \kappa \nabla H}{\rho} \right)^n$$

a level set function,  $\phi^n$ , and a volume-of-fluid function,  $F^n$ . The “live pressure gradient”, level set function, and volume-of-fluid function are stored at cell centers. The velocity is stored at both cell centers and face-centers. As previously noted in Section 3, the subscript  $ij$  refers to the center of a computational cell, the subscript  $i + 1/2, j$  refers to the right face center of a cell, and the subscript  $i, j + 1/2$  refers to the top face center of a cell. A vector quantity with the subscript  $i + 1/2, j$  implies that the first component lives at the right face center of a cell and the second component lives at the top face center of a cell.

A representative outline of one sweep of our (two-phase) method follows.

*Step 1. CLSVOF [50,45] interface advection:*

$$\phi_{ij}^{n+1} = \phi_{ij}^n - \Delta t [U^L \cdot \nabla \phi]_{ij}$$

$$F_{ij}^{n+1} = F_{ij}^n - \Delta t [U^L \cdot \nabla F]_{ij}$$

Step 2. Calculate (cell centered) advective force terms:

$$\mathcal{A}_{ij}^L = [\mathbf{U}^L \cdot \nabla \mathbf{U}^L]_{ij}^n$$

$$\mathcal{A}_{ij} = [\mathbf{U} \cdot \nabla \mathbf{U}]_{ij}^n$$

Details for the calculation of these terms are presented in Section 5.1 below.

Step 3. Calculate (cell centered, semi-implicit) viscous forces:

$$\begin{aligned} \mathbf{U}_{ij}^n &= \begin{cases} \mathbf{U}_{ij}^{L,n} & \phi_{ij} \geq 0 \\ \mathbf{U}_{ij}^n & \phi_{ij} < 0 \end{cases} \\ \mathcal{A}_{ij} &= \begin{cases} \mathcal{A}_{ij}^L & \phi_{ij} \geq 0 \\ \mathcal{A}_{ij} & \phi_{ij} < 0 \end{cases} \\ \rho_{ij} &= \begin{cases} \rho_L & \phi_{ij} \geq 0 \\ \rho_G & \phi_{ij} < 0 \end{cases} \\ \frac{\mathbf{U}_{ij}^* - \mathbf{U}_{ij}^n}{\Delta t} &= -\mathcal{A}_{ij} + g\hat{\mathbf{z}} - \text{Grad} P_{ij}^n + \frac{1}{\rho_{ij}} \frac{\mathcal{L}_{ij}^* + \mathcal{L}_{ij}^n}{2} \end{aligned} \quad (14)$$

The discrete operator  $\mathcal{L}$  is a second-order approximation to  $\nabla \cdot 2\mu D$  (see Section 5.4).

Step 4. Interpolate cell centered forces to face centered forces:

$$\begin{aligned} \mathcal{A}_{i+1/2,j}^L &= \frac{1}{2}(\mathcal{A}_{i+1,j}^L + \mathcal{A}_{i,j}^L) \\ \mathcal{A}_{i+1/2,j} &= \frac{1}{2}(\mathcal{A}_{i+1,j} + \mathcal{A}_{i,j}) \\ \mathcal{A}_{i+1/2,j} &= \begin{cases} \mathcal{A}_{i+1/2,j}^L & \phi_{ij} \geq 0 \text{ or } \phi_{i+1,j} \geq 0 \\ \mathcal{A}_{i+1/2,j} & \text{otherwise} \end{cases} \\ \mathcal{L}_{i+1/2,j} &= \frac{1}{4}(\mathcal{L}_{ij}^* + \mathcal{L}_{ij}^n + \mathcal{L}_{i+1,j}^* + \mathcal{L}_{i+1,j}^n) \\ \mathbf{U}_{i+1/2,j}^n &= \begin{cases} \mathbf{U}_{i+1/2,j}^{L,n} & \phi_{ij} \geq 0 \text{ or } \phi_{i+1,j} \geq 0 \\ \mathbf{U}_{i+1/2,j}^n & \text{otherwise} \end{cases} \\ \mathbf{V}_{i+1/2,j} &= \mathbf{U}_{i+1/2,j}^n + \Delta t \left( -\mathcal{A}_{i+1/2,j} + \frac{2}{\rho_{i+1,j} + \rho_{i,j}} \mathcal{L}_{i+1/2,j} - \left[ \frac{\sigma \kappa \nabla H}{\rho} \right]_{i+1/2,j} + g\hat{\mathbf{z}} \right) \end{aligned} \quad (15)$$

See Section 5.2 for steps to discretize the surface tension force  $\frac{1}{\rho} \sigma \kappa \nabla H$ .

Step 5. Implicit pressure projection step:

$$\begin{aligned} \nabla \cdot \frac{\nabla p}{\rho} &= \nabla \cdot \mathbf{V} \\ \mathbf{U}_{i+1/2,j}^{n+1} &= \mathbf{V} - \left[ \frac{\nabla p}{\rho} \right]_{i+1/2,j} \end{aligned} \quad (16)$$

Section 5.5 provides the spatial discretization associated with the implicit pressure projection step. We solve the resulting linear system using the multigrid preconditioned conjugate gradient method (MGPCG) [52].

Step 6. Liquid velocity extrapolation; assign  $\mathbf{U}_{i+1/2,j}^{L,n+1} = \mathbf{U}_{i+1/2,j}^{n+1}$  and then extrapolate  $\mathbf{U}_{i+1/2,j}^{L,n+1}$  into the gas region (see Section 5.6).

Step 7. Interpolate face centered velocity to cell centered velocity:

$$\begin{aligned} \mathbf{U}_{ij}^{L,n+1} &= \frac{1}{2}(\mathbf{U}_{i+1/2,j}^{L,n+1} + \mathbf{U}_{i-1/2,j}^{L,n+1}) \\ \mathbf{U}_{ij}^{n+1} &= \frac{1}{2}(\mathbf{U}_{i+1/2,j}^{n+1} + \mathbf{U}_{i-1/2,j}^{n+1}) \end{aligned}$$



Step 8. Update the cell centered “live” pressure gradient term,

$$\text{Grad}P_{ij}^{n+1} = \frac{U_{ij}^* - U_{ij}^{n+1}}{\Delta t} + \text{Grad}P_{ij}^n$$

Often in this paper, we shall compare the “two-phase” algorithm just described, to the corresponding “one-phase” algorithm. So, in the appendix (Section A.1) we describe the “one-phase” equations and algorithm.

Remarks:

- At the very first time step, we initialize

$$\text{Grad}P^0 \equiv \text{Grad}P^{0(0)} \equiv 0$$

then we do five iterations of the Crank–Nicolson/Runge–Kutta procedure ((12) and (13)) in order to initialize an appropriate cell centered pressure gradient,

$$\text{Grad}P^0 = \text{Grad}P^{0(5)}$$

We have found empirically that the cell centered pressure gradient term sufficiently converges after 5 sweeps. For example, for the very first time step for the problem of the break-up of a cylindrical jet due to surface tension (Section 6.5), the relative error in the magnitude of  $\text{Grad}P^{0(5)}$  is 0.0008.

- If “Step 6” (velocity extrapolation) is ignored, then our method corresponds in spirit to the sharp interface “ghost-fluid” approach described in [26,28]. This is because, without velocity extrapolation,  $U^L = U$ . In this case, when  $U^L = U$ , the main difference separating our approach from previous sharp interface methods [26,28] is that we treat the viscosity jump conditions implicitly; therefore we have no time step constraints associated with viscosity. We shall label this method, where liquid velocity extrapolation is ignored, as the “semi-implicit ghost-fluid method”.
- If “Step 6” (velocity extrapolation) is not ignored, then our method has the property that, for the limiting case of zero gas density and zero gas viscosity, our two-phase method is discretely equivalent to the second-order “one-phase” approach [45] in which gas pressure is treated as spatially uniform; Section A.1 gives a review of the “one-phase” approach.

## 5. Spatial discretization

### 5.1. Nonlinear advective terms

The term,

$$(U \cdot \nabla U)_{ij}$$

is discretized as

$$\left( \begin{array}{l} u_{ij} \frac{\bar{u}_{i+1/2,j} - \bar{u}_{i-1/2,j}}{\Delta x} + v_{ij} \frac{\bar{u}_{i,j+1/2} - \bar{u}_{i,j-1/2}}{\Delta y} \\ u_{ij} \frac{\bar{v}_{i+1/2,j} - \bar{v}_{i-1/2,j}}{\Delta x} + v_{ij} \frac{\bar{v}_{i,j+1/2} - \bar{v}_{i,j-1/2}}{\Delta y} \end{array} \right)$$

The quantities  $\bar{u}_{i+1/2,j}$ ,  $\bar{v}_{i+1/2,j}$ ,  $\bar{u}_{i,j+1/2}$  and  $\bar{v}_{i,j+1/2}$  are constructed from the cell centered velocity field  $U_{ij}$  using upwind and slope-limited differencing; e.g.

$$\bar{u}_{i,j+1/2} = \begin{cases} u_{ij} + \frac{1}{2}u_{y,ij} & \text{if } v_{i,j+1/2} > 0 \\ u_{i,j+1} - \frac{1}{2}u_{y,i,j+1} & \text{if } v_{i,j+1/2} < 0 \end{cases}$$

The slopes  $u_{y,ij}$  are computed using second-order Van Leer slope limiting [54],

$$u_{y,ij} = \begin{cases} S \min(2|u_{i,j+1} - u_{i,j}|, 2|u_{i,j} - u_{i,j-1}|, \frac{1}{2}|u_{i,j+1} - u_{i,j-1}|) & \text{if } s > 0 \\ 0 & \text{otherwise} \end{cases}$$

where

$$S = \text{sign}(u_{i,j+1} - u_{i,j-1})$$

and

$$s = (u_{i,j+1} - u_{i,j})(u_{i,j} - u_{i,j-1})$$

## 5.2. Surface tension force

In this section, we describe the discretization of the face centered surface tension term,

$$\frac{\sigma \kappa_{i+1/2,j} (\nabla H)_{i+1/2,j}}{\rho_{i+1/2,j}}$$

which is found in Eq. (15).

The discretization of the face centered surface tension term at the face center,  $(i + 1/2, j)$ , is written as,

$$\frac{\sigma \kappa_{i+1/2,j} \frac{H(\phi_{i+1,j}) - H(\phi_{i,j})}{\Delta x}}{\rho_{i+1/2,j}} \quad (17)$$

where

$$H(\phi) = \begin{cases} 1 & \phi \geq 0 \\ 0 & \phi < 0 \end{cases}$$

and

$$\rho_{i+1/2,j} = \rho_L \theta_{i+1/2,j} + \rho_G (1 - \theta_{i+1/2,j})$$

The discretization of the height fraction,  $\theta_{i+1/2,j}$ , is given in Section 5.3 (also see [19,28]).

The curvature  $\kappa_{i+1/2,j}$  is computed with second-order accuracy directly from the volume fractions as described in Section A.2.

Our treatment of surface tension can be approached from two different perspectives: (1) the surface tension term is derived in order to enforce the pressure jump condition as with the ghost-fluid approach [26,28]; (2) the inclusion of the surface tension term (17) as a force term in the momentum equation (15) is *equivalent* to prescribing the second-order Dirichlet pressure condition of surface tension (39) that would occur if the gas pressure was treated as spatially constant and the gas was assumed to be a “void” ( $\rho_G = 0$ ).

### 5.2.1. “Ghost-fluid” perspective for the surface tension term

Here we give the “ghost-fluid” [26] derivation of the surface tension term for the inviscid Euler’s equations. Without loss of generality, we consider a free surface that is vertically oriented between cells  $(i, j)$  and  $(i + 1, j)$ , at the location  $(x_{i+1} - \theta \Delta x, y_j)$ , with liquid on the right and gas on the left (see Fig. 4). At the face separating cells  $(i, j)$  and  $(i + 1, j)$ , the updated velocity is given by,

$$\begin{aligned} \frac{u_{i+1/2,j}^{n+1} - u_{i+1/2,j}^*}{\Delta t} &= -\frac{\nabla p^L}{\rho^L} \\ \frac{u_{i+1/2,j}^{n+1} - u_{i+1/2,j}^*}{\Delta t} &= -\frac{\nabla p^G}{\rho^G} \end{aligned}$$

The continuity condition requires that,

$$\frac{\nabla p^L}{\rho^L} \cdot \mathbf{n} = \frac{\nabla p^G}{\rho^G} \cdot \mathbf{n} \quad (18)$$

and the pressure jump condition requires that,

$$p_i^L - p_i^G = -\sigma \kappa \quad (19)$$

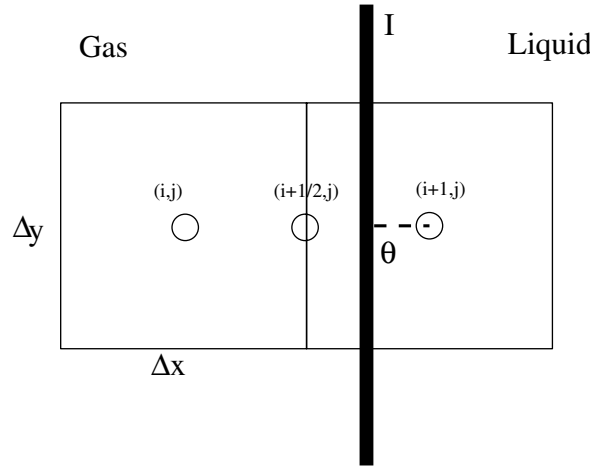


Fig. 4. “Ghost-fluid” treatment for vertical interface with liquid on the right and gas on the left.  $p_{i+1,j}$  is the liquid pressure at cell  $(i + 1, j)$  and  $p_{i,j}$  is the gas pressure at cell  $(i, j)$ .  $p_I^L$  is the liquid pressure on interface  $I$  and  $p_I^G$  is the gas pressure on interface  $I$ .  $\theta\Delta x$  is the distance from the interface,  $I$ , to the liquid cell  $(i + 1, j)$ .

where the gas and liquid pressure on the free surface are  $p_I^G$  and  $p_I^L$ , respectively. As a result of discretizing (18), one has,

$$\frac{p_{i+1,j}^L - p_I^L}{\rho^L \theta \Delta x} = \frac{p_I^G - p_{i,j}^G}{\rho^G (1 - \theta) \Delta x} \tag{20}$$

After one solves (19) and (20) for  $p_I^L$  and  $p_I^G$ , and substitutes the results back into the liquid and gas pressure gradients, one has,

$$\frac{u_{i+1/2,j}^{n+1} - u_{i+1/2,j}^*}{\Delta t} = -\frac{p_{i+1,j} - p_{i,j}}{\rho_{i+1/2} \Delta x} - \sigma \kappa_I \frac{H_{i+1,j} - H_{i,j}}{\rho_{i+1/2} \Delta x}$$

The surface tension term here is equivalent to (17). Without liquid velocity extrapolation, or if the gas density is not negligible, then this discretization will be first-order accurate since we had assumed in our derivation that the free surface was oriented either vertically or horizontally. Now suppose that we keep the liquid velocity separate from the gas velocity during the calculation of the nonlinear advection terms, and also suppose that the gas density is negligible, then one can relate the surface tension term to the Dirichlet boundary condition that one would impose for a “one-phase method”.

5.2.2. “One-phase” perspective for the surface tension

Here we give the “one-phase” [19,45] derivation of the surface tension term for the inviscid Euler equations. In contrast to the two-phase case where there were two boundary conditions at the interface, there is only one condition on pressure at the interface for the one-phase free boundary problem:

$$p_I^L = p_I^G - \sigma \kappa$$

The momentum equation in the liquid phase is,

$$\frac{\mathbf{u}^{L,n+1} - \mathbf{u}^{L,*}}{\Delta t} = -\frac{\nabla p^L}{\rho^L}$$

Suppose we are considering a free surface that passes between cells  $(i, j)$  and  $(i + 1, j)$ , at the location  $(x_{i+1} - \theta\Delta x, y_j)$ , with liquid on the right and gas on the left (see Fig. 4). Also, denote the gas and liquid pressure on the free surface as  $p_I^G$  and  $p_I^L$ , respectively. Then, discretely, one has,

$$-\frac{\nabla p^L}{\rho^L} = -\frac{p_{i+1}^L - (p_I^G - \sigma \kappa_I)}{\theta \Delta x \rho^L} = -\frac{p_{i+1} - p_i}{\theta \rho^L} - \sigma \kappa_I \frac{H_{i+1,j} - H_{i,j}}{\theta \rho^L}$$

This latter formulation is equivalent to the former when  $\rho^G = 0$  and we assume that gas pressure is spatially uniform. In other words, our treatment for surface tension corresponds to the treatment in a second-order “single-phase approach” (see Section A.1 or [45]). As mentioned by [19], this specification of the pressure boundary condition is second-order accurate; as opposed to the “ghost-fluid” perspective, we do not have to make assumptions regarding the orientation of the interface in order to get second-order.

5.3. Height fraction

The “height fraction”  $\theta_{i+1/2,j}$  [19,28,45] gives the one-dimensional fraction of water between cells  $(i,j)$  and  $(i+1,j)$ . Fig. 5 gives an illustration of the height fraction. The mixed face-centered density is expressed in terms of the height fraction. The height fraction  $\theta_{i+1/2,j}$  is derived from the level set function as follows:

$$\theta_{i+1/2,j}(\phi) = \begin{cases} 1 & \phi_{i+1,j} \geq 0 \text{ and } \phi_{i,j} \geq 0 \\ 0 & \phi_{i+1,j} < 0 \text{ and } \phi_{i,j} < 0 \\ \frac{\phi_{i+1,j}^+ + \phi_{i,j}^+}{|\phi_{i+1,j}^+| + |\phi_{i,j}^+|} & \text{otherwise} \end{cases}$$

The “+” superscript stands for the “positive part:” i.e.  $a^+ \equiv \max(a, 0)$ .

5.4. Semi-implicit viscous solve

An important property of our sharp-interface treatment for the viscous force terms is that resulting solutions of our two-phase algorithm approach solutions of the one-phase algorithm in the limit of zero gas density and zero gas viscosity (i.e. in the limit, in which the gas pressure is treated as spatially uniform).

The viscous force terms,  $\mathcal{L}_{ij}^*$  and  $\mathcal{L}_{ij}^n$ , appear in the discretized Navier–Stokes equations as shown below,

$$\frac{\mathbf{U}_{ij}^* - \mathbf{U}_{ij}^n}{\Delta t} = \mathcal{A}_{ij} + g\hat{\mathbf{z}} - \text{Grad} P_{ij}^n + \frac{1}{\rho_{ij}} \frac{\mathcal{L}_{ij}^* + \mathcal{L}_{ij}^n}{2} \tag{21}$$

$\mathcal{L}$  is a second-order discretization of the viscous force term,  $\nabla \cdot 2\mu D$ . In two dimensions, the rate of deformation tensor  $D$  is given by,

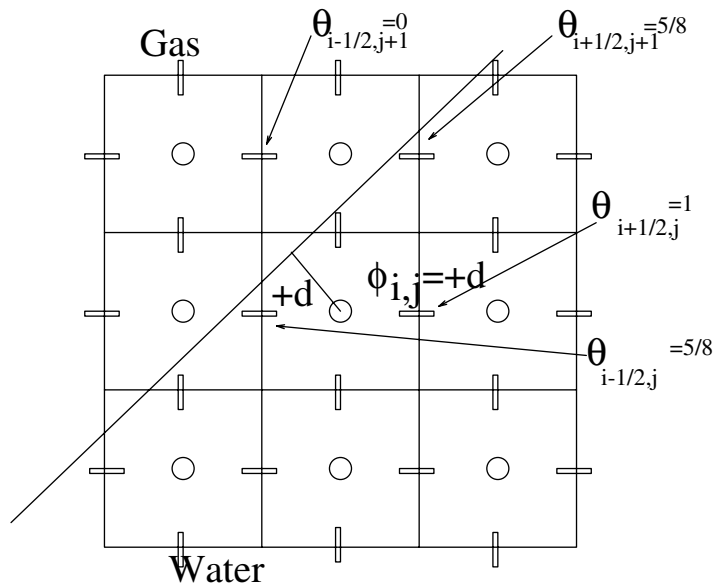


Fig. 5. Illustration of the face-centered height fraction,  $\theta_{i+1/2,j}$ ,  $\theta_{i,j+1/2}$ .

$$D = \begin{pmatrix} u_x & (u_y + v_x)/2 \\ (u_y + v_x)/2 & v_y \end{pmatrix}$$

In previous work [50], we found the rate of deformation tensor  $D$  at cell faces and used a finite volume discretization to approximate  $\nabla \cdot 2\mu D$ . In other words, in previous work we had,

$$(\nabla \cdot 2\mu D)_{ij} \approx \left( \frac{2\mu_{i+1/2,j}(u_x)_{i+1/2,j} - 2\mu_{i-1/2,j}(u_x)_{i-1/2,j}}{\Delta x} + \frac{\mu_{i,j+1/2}(u_y+v_x)_{i,j+1/2} - \mu_{i,j-1/2}(u_y+v_x)_{i,j-1/2}}{\Delta y} \right) \\ + \left( \frac{\mu_{i+1/2,j}(u_y+v_x)_{i+1/2,j} - \mu_{i-1/2,j}(u_y+v_x)_{i-1/2,j}}{\Delta x} + \frac{2\mu_{i,j+1/2}(v_y)_{i,j+1/2} - 2\mu_{i,j-1/2}(v_y)_{i,j-1/2}}{\Delta y} \right)$$

For a sharp interface method based on the finite volume discretization, the viscosity at a face is given by [26,28],

$$\mu_{i+1/2,j} = \begin{cases} \mu_L & \theta_{i+1/2,j} = 1 \\ \mu_G & \theta_{i+1/2,j} = 0 \\ 0 & \mu_G = 0 \text{ and } 0 < \theta_{i+1/2,j} < 1 \\ \frac{\mu_G \mu_L}{\mu_G \theta_{i+1/2,j} + \mu_L (1 - \theta_{i+1/2,j})} & \text{otherwise} \end{cases}$$

Unfortunately, with the above discretization for the viscosity term, the “two-phase” method does not correspond to the “single-phase” method (Section A.1) when  $\mu_G = 0$ . This is because velocities in gas cells could be accidentally included in the discretization of the coupling terms in liquid cells, even if  $\mu_G = 0$ . Fig. 6 gives an illustration of how gas velocities can be accidentally included in the discretization of the coupling terms (the term  $(\mu v_x)_y$  in the first equation and the term  $(\mu u_y)_x$  in the second). Therefore, we use the following “node based” discretization instead of the preceding finite volume discretization:

$$(\nabla \cdot 2\mu D)_{ij} = \left( \left( \frac{\partial(2\mu u_x)}{\partial x} \right)_{ij} + \left( \frac{\partial(\mu(u_y+v_x))}{\partial y} \right)_{ij} \right) \\ + \left( \left( \frac{\partial(\mu(u_y+v_x))}{\partial x} \right)_{ij} + \left( \frac{\partial(2\mu v_y)}{\partial y} \right)_{ij} \right)$$

where

$$\left( \frac{\partial(2\mu u_x)}{\partial x} \right)_{ij} \approx \left( 2\mu_{i+1/2,j+1/2}(u_x)_{i+1/2,j+1/2} - 2\mu_{i-1/2,j+1/2}(u_x)_{i-1/2,j+1/2} \right. \\ \left. + 2\mu_{i+1/2,j-1/2}(u_x)_{i+1/2,j-1/2} - 2\mu_{i-1/2,j-1/2}(u_x)_{i-1/2,j-1/2} \right) / (2\Delta x)$$

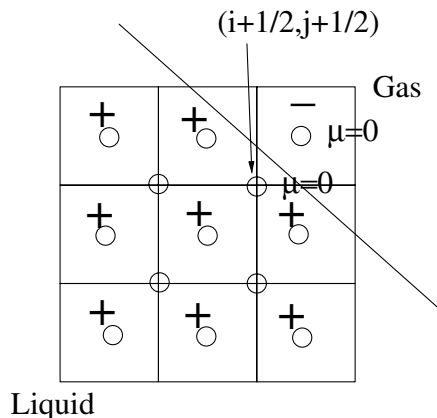


Fig. 6. Illustration of how the gas velocity at cell  $(i + 1, j + 1)$  is inadvertently included in the calculation of the coupling terms when  $\mu_G = 0$  and when the viscosity coefficient is given at the cell faces,  $\mu_{i+1/2,j}$  etc. The + and - signs refer to the sign of the level set function. In this scenario, all the face centered coefficients are equal to the liquid viscosity coefficient. If the viscosity coefficient is given at the nodes, and if  $\mu_G = 0$ , then  $\mu_{i+1/2,j+1/2} = 0$  and the gas velocity at  $(i + 1, j + 1)$  will not be included in the calculation of the viscous coupling terms.

$$\begin{aligned} \left(\frac{\partial(\mu(u_y + v_x))}{\partial y}\right)_{ij} &\approx \left(\mu_{i+1/2,j+1/2}(u_y + v_x)_{i+1/2,j+1/2} - \mu_{i+1/2,j-1/2}(u_y + v_x)_{i+1/2,j-1/2} \right. \\ &\quad \left. + \mu_{i-1/2,j+1/2}(u_y + v_x)_{i-1/2,j+1/2} - \mu_{i-1/2,j-1/2}(u_y + v_x)_{i-1/2,j-1/2}\right) / (2\Delta y) \\ \left(\frac{\partial(\mu(u_y + v_x))}{\partial x}\right)_{ij} &\approx \left(\mu_{i+1/2,j+1/2}(u_y + v_x)_{i+1/2,j+1/2} - \mu_{i-1/2,j+1/2}(u_y + v_x)_{i-1/2,j+1/2} \right. \\ &\quad \left. + \mu_{i+1/2,j-1/2}(u_y + v_x)_{i+1/2,j-1/2} - \mu_{i-1/2,j-1/2}(u_y + v_x)_{i-1/2,j-1/2}\right) / (2\Delta x) \\ \left(\frac{\partial(2\mu v_y)}{\partial y}\right)_{ij} &\approx \left(2\mu_{i+1/2,j+1/2}(v_y)_{i+1/2,j+1/2} - 2\mu_{i+1/2,j-1/2}(v_y)_{i+1/2,j-1/2} \right. \\ &\quad \left. + 2\mu_{i-1/2,j+1/2}(v_y)_{i-1/2,j+1/2} - 2\mu_{i-1/2,j-1/2}(v_y)_{i-1/2,j-1/2}\right) / (2\Delta y) \end{aligned}$$

The viscosity at a node is given by

$$\mu_{i+1/2,j+1/2} = \begin{cases} \mu_L & \theta_{i+1/2,j+1/2} = 1 \\ \mu_G & \theta_{i+1/2,j+1/2} = 0 \\ 0 & \mu_G = 0 \text{ and } 0 < \theta_{i+1/2,j+1/2} < 1 \\ \frac{\mu_G \mu_L}{\mu_G \theta_{i+1/2,j+1/2} + \mu_L (1 - \theta_{i+1/2,j+1/2})} & \text{otherwise} \end{cases}$$

where  $\theta_{i+1/2,j+1/2}$  is a “node fraction” defined as,

$$\theta_{i+1/2,j+1/2}(\phi) = \begin{cases} 1 & \phi_{i+1,j} \geq 0, \phi_{i,j} \geq 0, \phi_{i,j+1} \geq 0 \text{ and } \phi_{i+1,j+1} \geq 0 \\ 0 & \phi_{i+1,j} < 0, \phi_{i,j} < 0, \phi_{i,j+1} < 0 \text{ and } \phi_{i+1,j+1} < 0 \\ \frac{\phi_{i+1,j}^+ + \phi_{i,j}^+ + \phi_{i,j+1}^+ + \phi_{i+1,j+1}^+}{|\phi_{i+1,j}| + |\phi_{i,j+1}| + |\phi_{i,j}| + |\phi_{i+1,j+1}|} & \text{otherwise} \end{cases}$$

The “+” superscript stands for the “positive part:” i.e.  $a^+ \equiv \max(a,0)$ .

The components of the deformation tensor, e.g.  $(u_x)_{i+1/2,j+1/2}$ , are calculated using standard central differencing, i.e.

$$(u_x)_{i+1/2,j+1/2} = \frac{u_{i+1,j+1} + u_{i+1,j} - u_{i,j+1} - u_{i,j}}{2\Delta x}$$

The resulting linear system (21) for  $U^*$  is solved using the standard multigrid method.

*Remarks:*

- Our discretization of the viscous forces are second-order accurate away from the gas–liquid interface, but only first-order accurate at the gas–liquid interface. We observe first-order accuracy whether we are implementing our semi-implicit viscous solver as a part of the “single-phase” algorithm or as a part of the “two-phase” algorithm. Only in places where the free surface is aligned exactly with grid boundaries would our discretization be second-order accurate.
- Our proposed discretization of the node fraction,  $\theta_{i+1/2,j+1/2}$ , is not necessarily the only possible choice. The critical property that any discretization technique for the node fraction must have, is that  $\theta_{i+1/2,j+1/2} < 1$  if any of the surrounding level set values are negative.

5.5. Projection step

In this section, we provide the pertinent details for the discretization of the projection step found in Eq. (16),

$$\nabla \cdot \frac{\nabla p}{\rho} = \nabla \cdot \mathbf{V} \tag{22}$$

$$\mathbf{U} = \mathbf{V} - \frac{\nabla p}{\rho} \tag{23}$$

Eqs. (22) and (23) are discretized as

$$\text{Div} \frac{\text{Grad} P}{\rho} = \text{Div} \mathbf{V} \tag{24}$$

and

$$\mathbf{U} = \mathbf{V} - \frac{\text{Grad} P}{\rho}$$

respectively. Div is the discrete divergence operator defined by

$$(\text{Div} \mathbf{V})_{ij} = \frac{u_{i+1/2,j} - u_{i-1/2,j}}{\Delta x} + \frac{v_{i,j+1/2} - v_{i,j-1/2}}{\Delta y}, \tag{25}$$

and Grad represents the discrete gradient operator,

$$(\text{Grad} p)_{i+1/2,j} = \frac{p_{i+1,j} - p_{i,j}}{\Delta x} \tag{26}$$

$$(\text{Grad} p)_{i,j+1/2} = \frac{p_{i,j+1} - p_{i,j}}{\Delta y} \tag{27}$$

so that (24) becomes,

$$\frac{\frac{p_{i+1,j} - p_{i,j}}{\rho_{i+1/2,j}} - \frac{p_{i,j} - p_{i-1,j}}{\rho_{i-1/2,j}}}{\Delta x^2} + \frac{\frac{p_{i,j+1} - p_{i,j}}{\rho_{i,j+1/2}} - \frac{p_{i,j} - p_{i,j-1}}{\rho_{i,j-1/2}}}{\Delta y^2} = \text{Div} \mathbf{V}$$

The face centered density is defined by

$$\rho_{i+1/2,j} = \rho_L \theta_{i+1/2,j} + \rho_G (1 - \theta_{i+1/2,j}) \tag{28}$$

where the discretization of the height fraction,  $\theta_{i+1/2,j}$ , is given in Section 5.3.

At impenetrable boundaries, we give the Neumann boundary condition,

$$\nabla p \cdot \mathbf{n} = 0$$

and we also modify  $\mathbf{V}$  to satisfy,

$$\mathbf{V} \cdot \mathbf{n} = 0$$

At outflow boundaries, we give the Dirichlet boundary condition,

$$p = 0$$

i.e. if the top wall is outflow, then we have  $p_{i,j_{hi}+1} = -p_{i,j_{hi}}$ .

The resulting discretized pressure equation, (24), is solved for  $p$  using the multigrid preconditioned conjugate gradient method [52].

*Remark:*

- In the limit as  $\rho_G$  approaches zero, one recovers the second-order projection step described in [45]. In other words, in the limit of zero gas density, one recovers the second-order discretization of Dirichlet boundary conditions at the free surface. The discretization, using the height fractions  $\theta_{i+1/2,j}$ , corresponds to the second-order method described by [19] (in the zero gas density limit).
- By storing the velocity field at the cell faces and the pressure at the cell centers, we avoid the “checkerboard” instability while maintaining a discretely divergence free velocity field.
- We construct a temporary cell centered velocity field for calculating the advection and diffusion terms. Since at each timestep we interpolate the advective and diffusive forces from cell centers to cell faces in preparation for the next projection step, we avoid unnecessary numerical diffusion that would occur if we had interpolated the velocity itself from cell centers to cell faces.

5.6. Extrapolation of MAC velocities

The liquid velocity  $u_{i+1/2,j}^L$  is extended in a small “narrow band” about the zero level set of the level set function  $\phi$ . Extension velocities are needed on gas faces  $(i + 1/2,j)$  that satisfy  $\phi_{i,j} < 0$  and  $\phi_{i+1,j} < 0$ . We describe the initialization of  $u_{i+1/2,j}^L$  below; the case for  $v_{i,j+1/2}^L$  follows similarly. The extension procedure is very similar to that described in [45], except that (1) we choose an alternate, more stable, method for constructing our second-order linear interpolant and (2) we do not project the extended velocity field; in lieu of projecting the extended velocity field, we instead discretize the volume of fluid Eq. (6) in conservation form.

The steps for our liquid velocity extrapolation procedure are:

1. For each point where  $\phi_{i+1,j} < 0$  and  $\phi_{i,j} < 0$  and  $(1/2)(\phi_{ij} + \phi_{i+1,j}) > -K\Delta x$ , we already know the corresponding closest point on the interface  $\mathbf{x}_{closest, i+1/2,j} \equiv (1/2)(\mathbf{x}_{closest, ij} + \mathbf{x}_{closest, i+1,j})$ . The closest point on the interface has already been calculated during the CLSVOF reinitialization step (details found in [45], also see Fig. 1) since the distance at a gas cell  $\mathbf{x}_{ij}$  is,

$$d = -|\mathbf{x}_{ij} - \mathbf{x}_{closest,ij}|$$

2. Construct a  $7 \times 7$  stencil for  $u_{i+1/2,j}$  about the point  $\mathbf{x}_{closest, i+1/2,j}$ . A point  $\mathbf{x}_{i'+1/2,j'}$  in the stencil is tagged as “valid” if  $\phi_{i',j'} \geq 0$  or  $\phi_{i'+1,j'} \geq 0$ . A diagram of how this  $7 \times 7$  stencil is created for extending the horizontal velocity  $u_{i+1/2,j}^{extend}$  is shown in Fig. 7. Please see Fig. 8 for a diagram portraying the  $7 \times 7$  stencil used for constructing the vertical extension velocities  $v_{i,j+1/2}^{extend}$ .
3. Determine the *valid* cell  $(i^{crit} + 1/2, j^{crit})$  in the  $7 \times 7$  stencil that is closest to  $\mathbf{x}_{closest, i+1/2,j}$ .
4. Determine the slopes  $\Delta_x u$  and  $\Delta_y u$ . In the  $x$  direction, investigate the forward differences,

$$\Delta_x u = u_{i'+3/2,j^{crit}} - u_{i'+1/2,j^{crit}}$$

where  $(i' + 3/2, j^{crit})$  and  $(i' + 1/2, j^{crit})$  are valid cells in the  $7 \times 7$  stencil. In the  $y$  direction, investigate the forward differences,

$$\Delta_y u = u_{i^{crit}+1/2,j'+1} - u_{i^{crit}+1/2,j'}$$

where  $(i^{crit} + 1/2, j' + 1)$  and  $(i^{crit} + 1/2, j')$  are valid cells in the  $7 \times 7$  stencil. If any of the differences change sign in the  $x(y)$  direction, then the slope,  $\Delta_x u(\Delta_y u)$  is zero, otherwise the slope is taken to be the quantity  $\Delta_x u(\Delta_y u)$  that has the minimum magnitude.

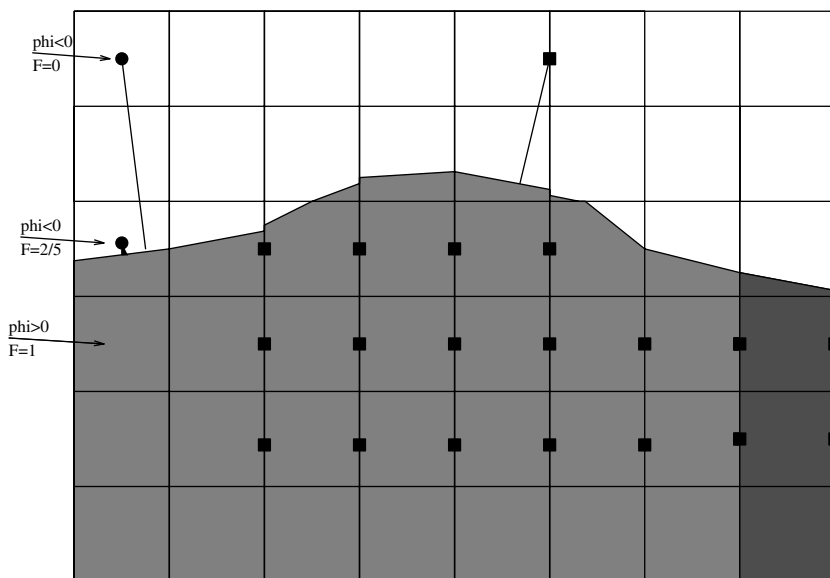


Fig. 7. Diagram hi-lighting the valid points in the  $7 \times 7$  stencil used for constructing the horizontal extension velocities.



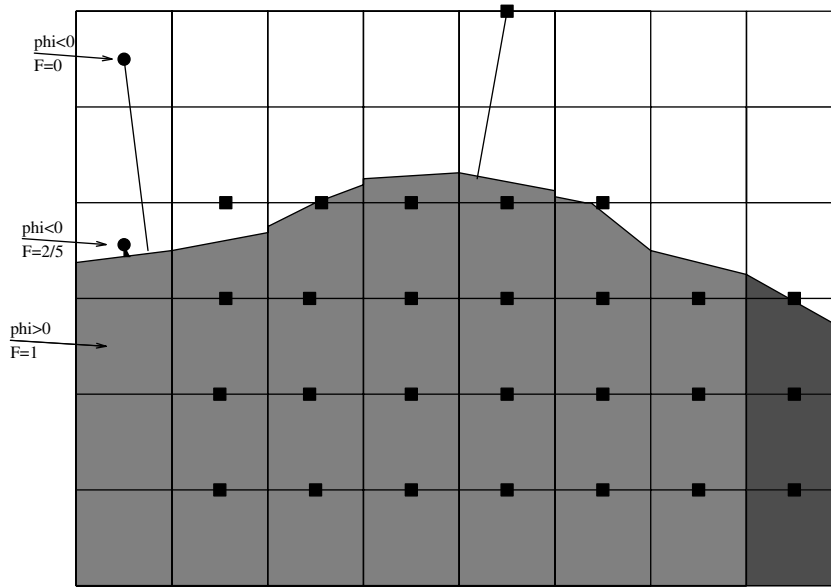


Fig. 8. Diagram hi-lighting the valid points in the 7 × 7 stencil used for constructing the vertical extension velocities.

5. Construct

$$u_{i+1/2,j}^{\text{extend}} = (\Delta_x u)(i - i^{\text{crit}}) + (\Delta_y u)(j - j^{\text{crit}}) + u_{i^{\text{crit}}+1/2,j^{\text{crit}}}$$

5.7. Timestep

The timestep  $\Delta t$  at time  $t^n$  is determined by restrictions due to the CFL condition, surface tension, and gravity:

$$\Delta t < \min_{i,j} \left( \frac{1}{2} \frac{\Delta x}{|U^n|}, \frac{1}{2} \sqrt{\frac{\rho^L}{8\pi\sigma}} \Delta x^{3/2}, \frac{1}{2} \frac{2\Delta x}{|u^n| + \sqrt{|u^n|^2 + 4g\Delta x}} \right)$$

The stability condition regarding gravity was determined “heuristically” in which we have the inequality,

$$(u + \Delta tg)\Delta t < \Delta x$$

The stability condition for surface tension is taken from [9,18]. Other references regarding stability conditions for incompressible flow are [2,31].

6. Results

In this section we test the accuracy of our numerical algorithm. In a few cases, we shall compare our sharp interface approach to the “semi-implicit ghost-fluid” approach. Also, we shall compare our two-phase sharp interface approach to our “one-phase sharp” interface method. In cases where the exact solution is unknown, we calculate the error by comparing the solutions on successively refined grids. The error in interface position is measured as

$$E_{\text{interface}} = \sum_{ij} \int_{\Omega_{ij}} |H(\phi_f) - H(\phi_c)| dx \tag{29}$$

where  $\phi_f$  and  $\phi_c$  correspond to the solutions using the fine resolution and coarse resolution grids, respectively.

The “average” error in liquid velocity is measured as (for 3d-axisymmetric problems),

$$E_{\text{Liquid}}^{\text{avg}} = \sum_{ij,\phi>0} \sqrt{(u_{f,ij} - u_{c,ij})^2 + (v_{f,ij} - v_{c,ij})^2} r_i \Delta r \Delta z \tag{30}$$

The “maximum” error in liquid velocity is measured as,

$$E_{\text{Liquid}}^{\text{max}} = \max_{ij,\phi>0} \sqrt{(u_{f,ij} - u_{c,ij})^2 + (v_{f,ij} - v_{c,ij})^2} \tag{31}$$

### 6.1. Parasitic currents

In this section we test our implementation of surface tension for the problem of a static two-dimensional (2d) drop with diameter  $D$ . We assume the density ratio and viscosity ratio are both one for this problem. The exact solution for such a problem is that the velocity  $\mathbf{u}$  is identically zero. If we scale the Navier–Stokes equations by the time scale  $T = D\mu/\sigma$ , and by the velocity scale  $U = \sigma/\mu$ , then the non-dimensionalized Navier–Stokes equations become,

$$\frac{D\mathbf{u}}{Dt} = -\nabla p + Oh^2 \Delta \mathbf{u} - Oh^2 \kappa \nabla H$$

where the Ohnesorge number  $Oh$  is defined as,

$$Oh = \frac{\mu}{\sqrt{\sigma \rho D}}$$

We investigate the maximum velocity of our numerical method for varying grid resolutions at the dimensionless time  $t = 250$ . The dimensions of our computational grid are  $5/2 \times 5/2$  with periodic boundary conditions at the left and right boundaries and reflecting boundary conditions at the top and bottom boundaries. A drop with unit diameter is initially located at the center of our domain  $(5/4, 5/4)$ . Our tolerance for the pressure solver and viscous solver is  $1.0E - 12$  (the error is measured as an absolute error and is the  $L^2$  norm of the residual). In Table 1 we display results of our grid refinement study for  $1/Oh^2 = 12,000$ . Our results indicate at least second-order convergence. These results are comparable to those in [35] where a front tracking method was used to represent the interface. Our results are also comparable to recent work by [18] in which a height fraction approach for surface tension was tested.

### 6.2. Surface tension driven (zero gravity) drop oscillations

In this section, we perform a grid refinement study for the problem of surface tension driven drop oscillations. In the previous example with parasitic currents, the density ratio was 1:1 and the viscosity ratio was 1:1; in this example, the density ratio is 1000:1 and the viscosity ratio is 1000:1.

According to the linearized results derived by Lamb [27, Section 275], the position of the drop interface is

$$R(\theta, t) = a + \epsilon P_n(\cos(\theta)) \sin(\omega_n t + \pi/2)$$

where

$$\omega_n^2 = \sigma \frac{n(n-1)(n+1)(n+2)}{a^3(\rho_l(n+1) + \rho_g n)}$$

Table 1  
Convergence study for static droplet with surface tension (parasitic currents test)

$\Delta x$	Maximum velocity
2.5/16	7.3E – 4
2.5/32	4.5E – 6
2.5/64	5.5E – 8

Maximum velocity at  $t = 250$  is shown.  $Oh^2 = 1/12000$ .

and  $P_n$  is the Legendre polynomial of order  $n$ .  $\theta$  runs between 0 and  $2\pi$ , where  $\theta = 0$  corresponds to  $r = 0$  and  $z = a$ . If viscosity is present, Lamb [27, Section 355] found that the amplitude is proportional to  $e^{-t/\tau}$ , where

$$\tau = \frac{a^2 \rho_L}{\mu_L (2n + 1)(n - 1)}$$

We compute the evolution of a drop with  $a = 1$ ,  $g = 0$ ,  $\mu_L = 1/50$ ,  $\mu_L/\mu_G = 1000$ ,  $\sigma = 1/2$ ,  $\rho_L = 1$  and  $\rho_L/\rho_G = 1000$ . The initial interface is given by  $R(\theta, 0)$ , with  $\epsilon = 0.05$  and  $n = 2$ . With these parameters we find  $\omega_2 = 2.0$  and  $\tau = 5.0$ . The fluid domain is  $\Omega = \{(r, z) | 0 \leq r \leq 1.5 \text{ and } 0 \leq z \leq 1.5\}$  and we compute on grid sizes ranging from  $32 \times 32$  to  $128 \times 128$ . The time step for each respective grid size ranges from 0.0007 to 0.000175. Symmetric boundary conditions are imposed at  $r = 0$  and  $z = 0$ .

In Table 2, we display the relative error between succeeding resolutions for the minor amplitude  $R_{\Delta x}(0, t)$  of the droplet. The average error  $E_{\text{Amplitude}}^{\text{avg}}$  is given by

$$E_{\text{Amplitude}}^{\text{avg}} \equiv \int_0^{3.5} |R_{\Delta x}(0, t) - R_{2\Delta x}(0, t)| dt$$

and the maximum amplitude error  $E_{\text{Amplitude}}^{\text{max}}$  is given by

$$E_{\text{Amplitude}}^{\text{max}} \equiv \max_{0 \leq t \leq 3.5} |R_{\Delta x}(0, t) - R_{2\Delta x}(0, t)|$$

In Fig. 9, we plot the minor amplitude versus time for the three different grid resolutions.

Table 2

Convergence study for zero gravity drop oscillations  $\sigma = 1/2$ ,  $\mu_L = 1/50$ ,  $\mu_L/\mu_G = 1000$ ,  $\rho_L/\rho_G = 1000$  and  $\alpha = 2$

$\Delta r$	$E_{\text{Amplitude}}^{\text{avg}}$	$E_{\text{Amplitude}}^{\text{max}}$
3/64	N/A	N/A
3/128	0.00076	0.00172
3/256	0.00021	0.00057

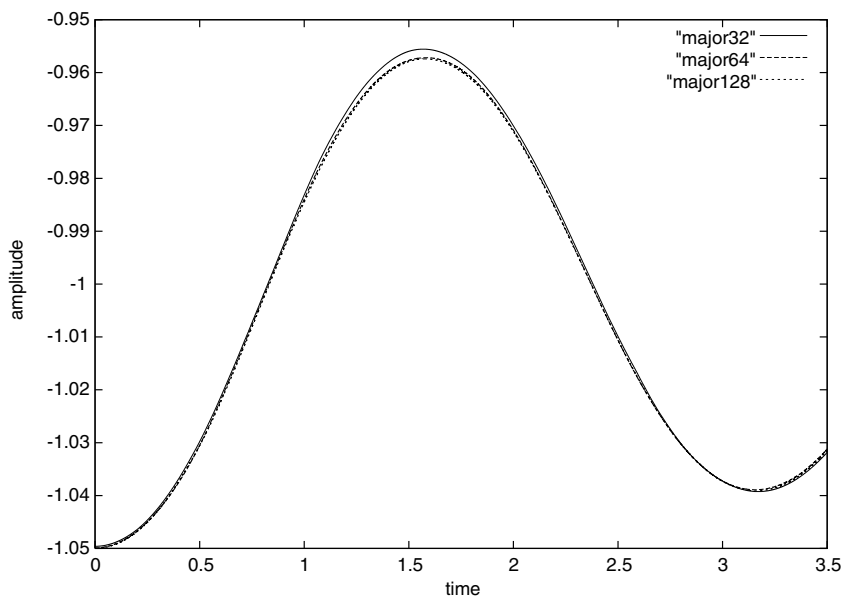


Fig. 9. Perturbation in minor amplitude for zero gravity drop oscillations (two-phase sharp interface method).  $\mu_L = 1/50$ ,  $\gamma = 1/2$ , density ratio 1000:1, viscosity ratio 1000:1.

### 6.3. Standing wave problem

For the standing wave problem, the free surface at  $t = 0$  is described by the equation

$$y = (1/4) + \epsilon \cos(2\pi x)$$

where  $\epsilon = 0.025$ . The gravitational force is  $g = 2\pi$ . We assume inviscid flow,  $\mu_L = \mu_G = 0$ , and the density ratio is 1000,  $\rho_L = 1$ ,  $\rho_L/\rho_G = 1000$ . The computational domain is a  $1/2$  by  $1/2$  box with symmetric boundary conditions at  $x = 0$  and  $x = 1/2$  and solid wall boundary conditions at  $y = 0$ . In Fig. 10 we compare the amplitude (at  $x = 0$ ) for 4 different grid resolutions:  $\Delta x = 1/64$ ,  $\Delta x = 1/128$ ,  $\Delta x = 1/256$  and  $\Delta x = 1/512$ . The timestep for each case is  $\Delta t = 0.02$ ,  $\Delta t = 0.01$ ,  $\Delta t = 0.005$  and  $\Delta t = 0.0025$ .

In Table 3, we show the relative error between the 4 graphs ( $0 \leq t \leq 10$ ). In Table 4, we provide the percent error for the maximum mass fluctuation for the time interval  $0 \leq t \leq 10$ ,

$$\max_{0 \leq t \leq 10} 100 \frac{|\text{mass}(t) - \text{mass}(0)|}{\text{mass}(0)}$$

In Fig. 11, we compare our proposed “two-phase” sharp interface method to the corresponding “one-phase” method described in Section A.1. They are almost identical, which is expected since our two-phase sharp interface approach becomes the one-phase approach in the limit of zero gas density  $\rho_G$  and zero gas viscosity  $\mu_G$ . Also in the same figure, we study the difference between our sharp interface approach with/without

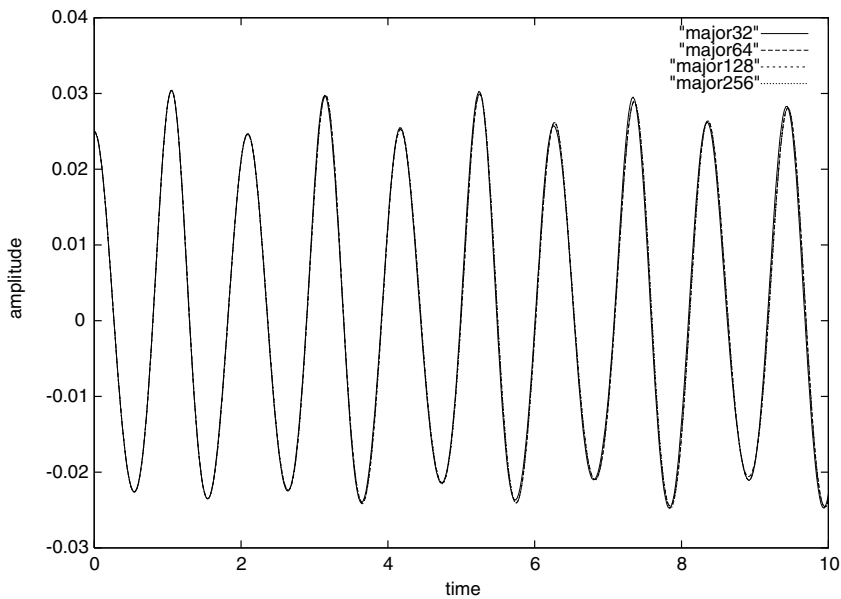


Fig. 10. Amplitude for inviscid standing wave problem. Density ratio 1000:1 (two-phase sharp interface method).

Table 3

Convergence study: relative error between coarse grid computations with cell size  $\Delta x_{\text{coarse}}$  and fine grid computations with cell size  $\Delta x_{\text{fine}}$  for amplitude at  $x = 0$  for standing wave problem

$\Delta x_{\text{coarse}}$	$\Delta x_{\text{fine}}$	Maximum error	Average error
1/64	1/128	2.4E – 3	6.2E – 4
1/128	1/256	6.5E – 4	1.5E – 4
1/256	1/512	2.8E – 4	4.9E – 5

Relative error measured for the period  $0 \leq t \leq 10$ . The physical domain size is  $1/2 \times 1/2$ .  $\Delta x$  is the mesh spacing which is  $\frac{1}{2n_x}$  where  $n_x$  is the number of cells in the  $x$  direction. For all our tests,  $\Delta x = \Delta y$ .

Table 4  
Convergence study: maximum mass fluctuation error measured as a percent of the initial mass

$\Delta x$	Mass error (%)
1/32	0.078
1/64	0.030
1/128	0.015
1/256	0.007

Mass error measured for the period  $0 \leq t \leq 10$ . The physical domain size is  $1/2 \times 1/2$ .  $\Delta x$  is the mesh spacing which is  $\frac{1}{2n_x}$  where  $n_x$  is the number of cells in the  $x$  direction. For all our tests,  $\Delta x = \Delta y$ .

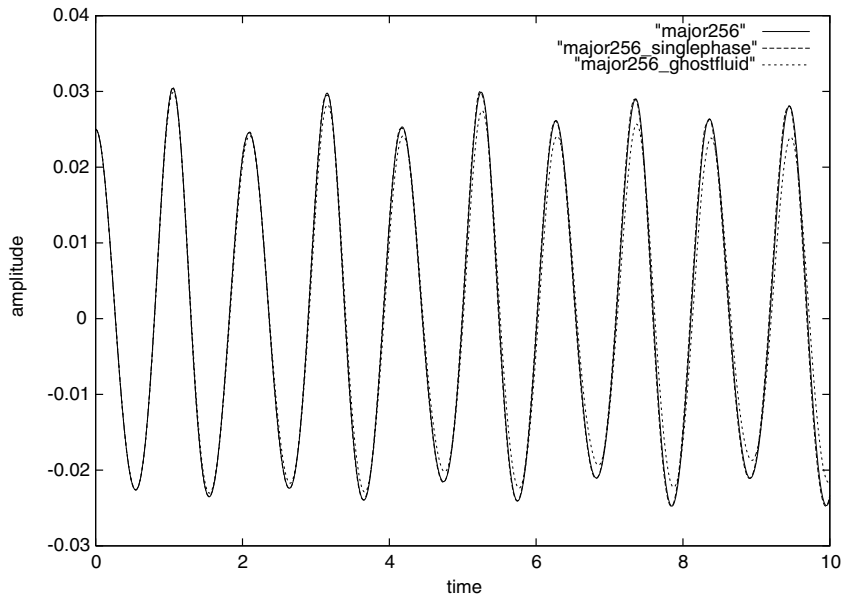


Fig. 11. Comparison of two-phase sharp interface method with “single-phase” method and “semi-implicit ghost-fluid” method. Density ratio 1000:1.

liquid velocity extrapolation (Step 6 in Section 4). Without liquid velocity extrapolation (a.k.a. the “semi-implicit ghost-fluid” approach), the results do not converge nearly as rapidly as with velocity extrapolation. The “no extrapolation” results with  $\Delta x = 1/512$  are more poorly resolved than the  $\Delta x = 1/64$  results corresponding to our sharp-interface approach with liquid velocity extrapolation.

We remark that in (3), we see that the order of accuracy is 1.6 on the finest resolution grids. The order is not 2 since our method is designed to approach a second-order method as  $\rho^G$  approaches zero. In this test, we believe that the error is so small, that the value of  $\rho^G$  is big enough to make itself the dominant contribution to the error. One can also look at  $\rho^G$  as being analogous to the cutoff used for  $\theta$  in the second-order discretization of the poisson equation on irregular domains [19].

#### 6.4. Traveling wave problem

In [56], experiments were conducted in which traveling waves were generated from wind. In this section we investigate the performance of our numerical algorithm for simulating traveling waves in the presence of wind. We shall validate our algorithm by way of a grid refinement test. We shall also compare results of our new algorithm to those results produced by our “semi-implicit ghost-fluid” method.

According to [56], a wind velocity of  $U = 5$  m/s will generate traveling waves with a wavelength  $\lambda = 15$  cm, a phase velocity  $C = 50$  cm/s, a wave period  $P = 0.3$  s, and a trough to peak wave height  $H = 1$  cm. Also, for a wind speed of  $U = 5$  m/s, the roughness length is  $z_{0a} = 0.3$  cm and the friction velocity is  $u_{*a} = 30$  cm/s.

We initialized our computational domain as a  $15 \times 30$  cm rectangular box with the initial position of the water surface given by,

$$y_{\text{surface}}(x) = 15.0 + \frac{H}{2} \cos(2\pi x/\lambda)$$

We shall assume periodic boundary conditions on the left and right walls, and “free-slip” boundary conditions on the upper and lower walls.

The initial velocity in the water is derived using a similar numerical procedure as found in [59]. We compute a stream function  $\psi$  which is defined in the whole computational domain. In the calculation of  $\psi$ , we assume the initial vorticity is zero everywhere except on the interface. The vortex sheet strength at the air–water interface is given by,

$$\Gamma = H\omega \cos(kx),$$

where  $k = 2\pi/\lambda$  is the wave number and  $\omega$  and  $k$  satisfy the following linear dispersion relation:

$$\omega^2 = gk + \frac{\sigma k^3}{\rho_L} \tag{32}$$

We have ignored the gas density  $\rho_G$  and the water depth (15 cm) in (32) since these values have a negligible effect on  $\omega$ . In our computations, we used the actual physical properties for air and water:  $\rho_L = 1.0 \text{ g/cm}^3$ ,  $\rho_G = 0.001229 \text{ g/cm}^3$ ,  $\mu_L = 0.0089 \text{ g/(cm s)}$ ,  $\mu_G = 1.73e - 4 \text{ g/(cm s)}$ ,  $g = 980.0 \text{ cm/s}^2$ , and  $\sigma = 72.8 \text{ dyne/cm}$ . Given these properties, we have  $\omega = 20.39$ . We remark that the linear dispersion relation predicts a period of  $P = 2\pi/\omega = 0.31$  which is very close to the experimental values reported by [56].

Given the vortex sheet strength, we solve for the stream function  $\psi$  using the following equation:

$$\psi_{xx} + \psi_{yy} = -\Gamma |\nabla H(\phi)| \tag{33}$$

$|\nabla H|$  is discretized as,

$$\sqrt{\left(\frac{H_{i+1/2,j} - H_{i-1/2,j}}{\Delta x}\right)^2 + \left(\frac{H_{i,j+1/2} - H_{i,j-1/2}}{\Delta y}\right)^2}$$

where,

$$H_{i+1/2,j} = \begin{cases} 1 & \phi_{ij} \geq 0 \text{ or } \phi_{i+1,j} \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

Once  $\psi$  is found, we have:

$$u_{ij} = \frac{\psi_{i,j+1} - \psi_{i,j-1}}{2\Delta y} \tag{34}$$

$$v_{ij} = -\frac{\psi_{i+1,j} - \psi_{i-1,j}}{2\Delta x} \tag{35}$$

The boundary conditions for  $\psi$  in (33) are homogeneous Dirichlet conditions at the top and bottom of the computational domain, and periodic boundary conditions on the left and right sides.

In [59], the velocity in the air as well as in the water was given by (34) and (35). In our test, we shall initialize the air velocity to have the characteristic logarithmic “wind” profile given by,

$$u(x, y) = \begin{cases} 0 & y < y_{\text{surface}}(x) + z_{0a} \\ \frac{u_{*a}}{K} \log\left(\frac{y - y_{\text{surface}}(x)}{z_{0a}}\right) & \text{otherwise} \end{cases}$$

$$v(x, y) = 0$$

where  $K = 0.4$  is von Karmon’s constant,  $z_{0a} = 0.3 \text{ cm}$  is the roughness length, and  $u_{*a} = 30.0 \text{ cm/s}$  is the friction velocity.

Given the cell centered initial velocity in the water and air, we interpolate these respective velocity fields from cell centers to cell faces and then we initialize the face centered velocity  $V$  as,

$$V_{i+1/2,j} = \begin{cases} U_{i+1/2,j}^L & \phi_{ij} \geq 0 \text{ or } \phi_{i+1,j} \geq 0 \\ U_{i+1/2,j}^G & \text{otherwise} \end{cases}$$

The initial velocity should be divergence free so we project  $V$  as described in Section 5.5 in order to insure a discretely divergence free initial velocity field. After the projection step, we initialize the liquid and gas velocity with the projected velocity  $U$  and then we extend the liquid velocity into the gas in order to construct  $U^L$ . In Fig. 12 we plot the initial velocity fields  $U^L$  and  $U$ .

In Fig. 13, we compare the amplitude (at  $x = 0$ ) versus time for three different grid resolutions:  $\Delta x = 15/32$ ,  $\Delta x = 15/64$  and  $\Delta x = 15/128$ . The timestep for each case is  $\Delta t = 0.0008$ ,  $\Delta t = 0.0004$ , and  $\Delta t = 0.0002$ . In Table 5, we show the relative error between the 3 graphs ( $0 \leq t \leq 1$ ). In Fig. 14, we plot the amplitude for our sharp interface without liquid velocity extrapolation (Step 6 in Section 4). Without liquid velocity extrapolation (a.k.a. the “semi-implicit ghost-fluid” approach), the results do not converge nearly as rapidly as with velocity extrapolation. The “no extrapolation” results with  $\Delta x = 15/128$  are more poorly resolved than the  $\Delta x = 15/64$  results corresponding to our sharp-interface approach with liquid velocity extrapolation.

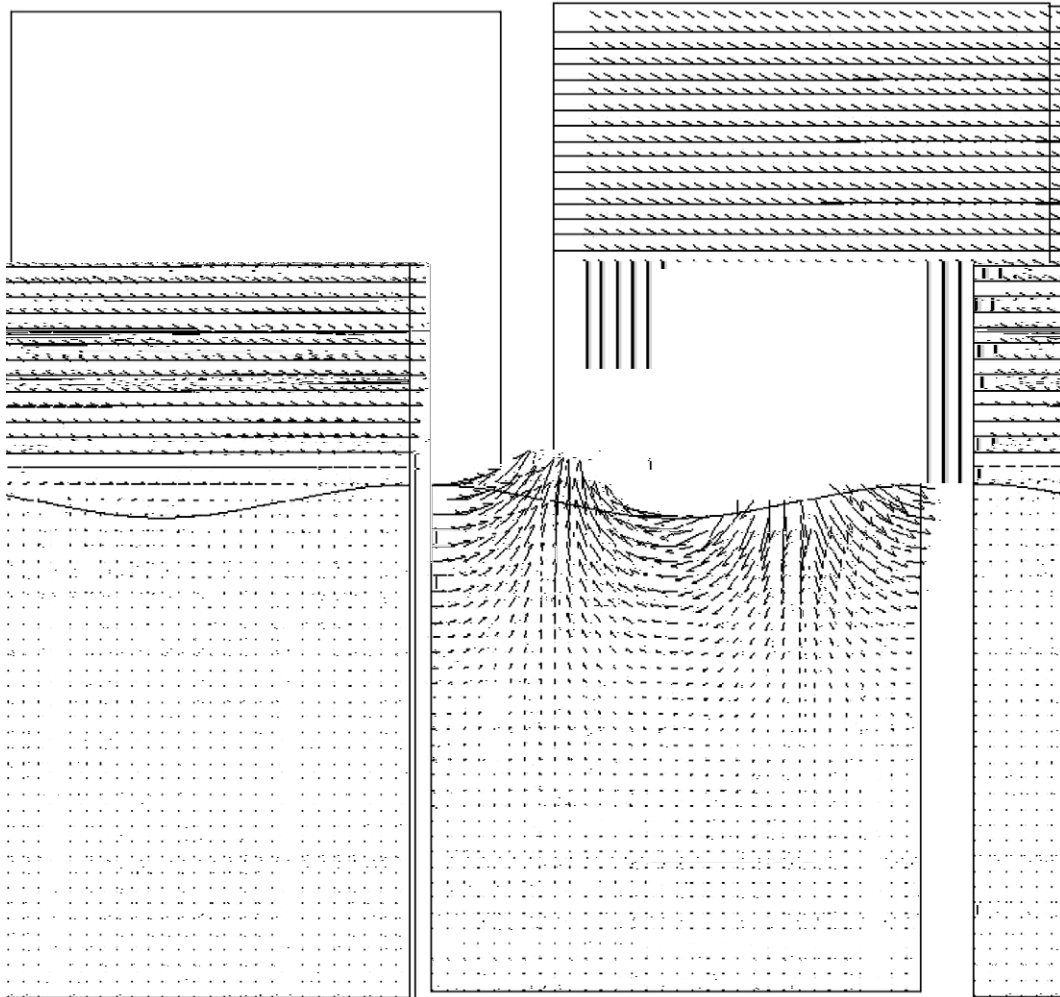


Fig. 12. Initial velocity field for wind driven wave problem. Left: initial liquid velocity  $U^L$  derived from stream function. Right: initial combined liquid/gas velocity  $U$ . “Wind” velocity in the gas has logarithmic profile. Grid resolution is  $128 \times 256$ .

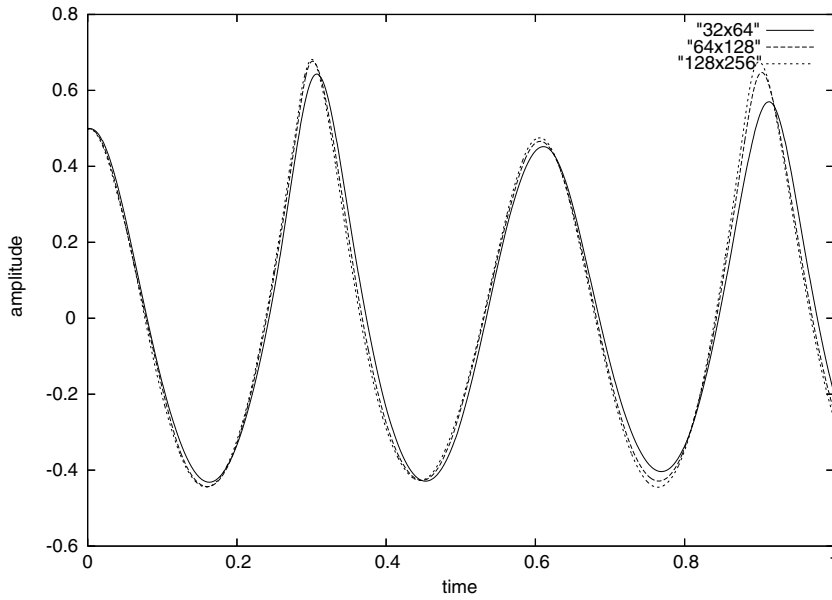


Fig. 13. Amplitude for traveling wave problem with wind. Density ratio 813:1 (two-phase sharp interface method).

Table 5

Convergence study: relative error between coarse grid computations with cell size  $\Delta x_{\text{coarse}}$  and fine grid computations with cell size  $\Delta x_{\text{fine}}$  for amplitude at  $x = 0$  for traveling wave problem with wind

$\Delta x_{\text{coarse}}$	$\Delta x_{\text{fine}}$	Maximum error	Average error
15/32	15/64	0.122	0.031
15/64	15/128	0.057	0.014

Relative error measured for the period  $0 \leq t \leq 1$ . The physical domain size is  $15 \times 30$ .  $\Delta x$  is the mesh spacing which is  $\frac{15}{n_x}$  where  $n_x$  is the number of cells in the  $x$  direction. For all our tests,  $\Delta x = \Delta y$ .

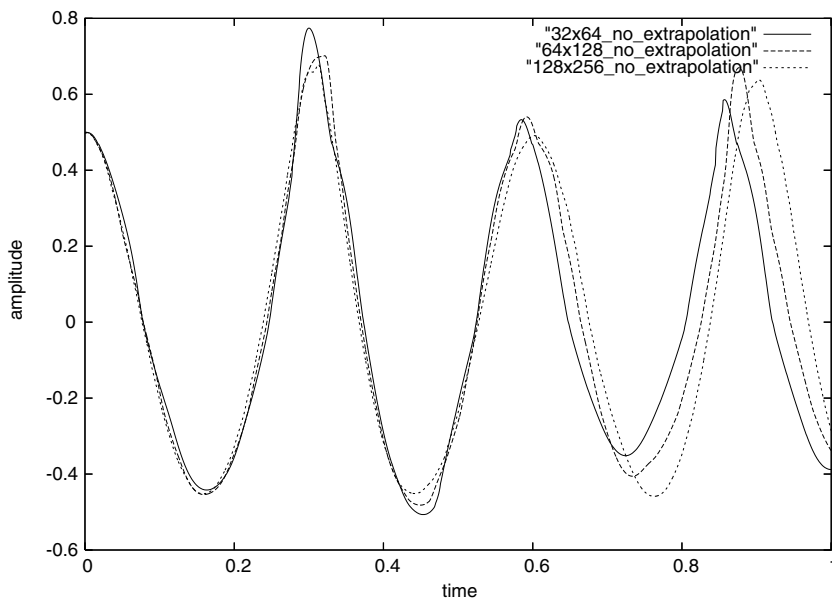


Fig. 14. Amplitude for traveling wave problem with wind. Density ratio 813:1. Velocity extrapolation is disabled (“semi-implicit ghost-fluid” method).



Remarks:

- We measured first-order accuracy using our sharp-interface method (with velocity extrapolation). We attribute this to how we obtained our initial velocity field in the liquid. The discretization to the right hand side of (33) is a low order approximation to the delta function; nonetheless, we see significant improvement in our calculations with velocity extrapolation, as opposed to without. Without velocity extrapolation, we do not see any convergence for the grid sizes used.
- The computations in [59] (using a “continuum approach”) were limited to a wave Reynolds number of around 150 and a density ratio of 100:1. Also, wind was not taken into account in their computations. In the results we present here, using the actual physical properties of air and water, the wave Reynolds number is  $Re = \frac{\rho^L C \lambda}{2\pi\mu^L} = 12875$  and the density ratio is 813:1.

6.5. Capillary instability

In this section, we test our sharp-interface approach on the classical Rayleigh capillary instability problem in which a slightly perturbed cylindrical column of liquid is driven to break up into droplets by surface tension (capillary) effects. In this test problem we use parameters that are comparable to those found in [50].

We consider an initially perturbed cylindrical column of water in air. The shape of the initial interface is

$$r(z) = r_0 + \epsilon \cos(2\pi z/\lambda) \tag{36}$$

We compute on a 3d-axisymmetric domain  $\Omega = \{(r,z)|0 \leq r \leq \lambda/4 \text{ and } 0 \leq z \leq \lambda/2\}$ . Symmetric boundary conditions are enforced at  $r = 0$ ,  $z = 0$  and  $z = \lambda/2$ . Outflow (pressure equals zero) boundary conditions are enforced at  $r = \lambda/4$ . The relevant dimensional parameters for this test problem are  $r_0 = 6.52 \mu\text{m}$ ,  $\epsilon = 1.3 \mu\text{m}$ ,  $\lambda = 60 \mu\text{m}$ ,  $\mu_L = 1.138 \times 10^{-2} \text{ g}/(\text{cm s})$ ,  $\mu_G = 1.77 \times 10^{-4} \text{ g}/(\text{cm s})$ ,  $\rho_L = 1.0 \text{ g}/\text{cm}^3$ ,  $\rho_G = 0.001225 \text{ g}/\text{cm}^3$ , and  $\sigma = 72.8 \text{ dynes}/\text{cm}$ . In our computations we use the following dimensionless parameters: the Reynolds number  $Re = \rho_L L U / \mu_L = 7.5$ , the Weber number  $We = \rho_L L U^2 / \sigma = 1.0$ ,  $L = 1 \mu\text{m}$ ,  $U = 8.53 \text{ m}/\text{s}$  and the density and viscosity ratios are 816 and 64, respectively.

In Fig. 15, we display the results of our computations for the capillary jet as it breaks up. In Table 6, we measure the relative errors for the interface and velocity field for grid resolutions ranging from  $16 \times 32$  to  $64 \times 128$ . The time step ranged from  $\Delta t = 0.04$  to  $\Delta t = 0.01$ .

As shown in Table 6, we obtain about first-order accuracy for the solution in the liquid. We attribute our low order accuracy to how we discretize the viscosity term,

$$\frac{1}{\rho} \nabla \cdot (2\mu D), \tag{37}$$

at the interface. Suppose  $\mu_G = 0$  and the zero level set crosses between cells  $(i,j)$  ( $\phi_{ij} < 0$ ) and  $(i+1,j)$  ( $\phi_{i+1,j} \geq 0$ ). In this case the values for  $\mu$  and  $\rho$  jump from  $\mu_G$  to  $\mu_L$  and from  $\rho_G$  to  $\rho_L$  abruptly where the level set function changes sign; i.e. our discretization of  $\mu$  and  $\rho$  in Eq. (37) does not incorporate specific information about the location of the zero level set in between cells  $(i,j)$  and  $(i+1,j)$ , except that the zero level set is somewhere between these two cells. We remark that, although we observe first-order accuracy using our sharp interface approach, our errors are considerably smaller than those presented in [50]. We also get comparable results when calculating the break-up of a liquid jet using our “single-phase” method (Section A.1, see Table 8 and Fig. 16).

If we reduce the viscosity further, i.e. set the Reynold’s number  $Re = 200$ , then we get much closer to second-order convergence using our sharp interface approach, as illustrated in Table 7.

6.6. Bubble dynamics

In this section, we compute the steady state shapes of a gas bubble rising in a viscous Newtonian liquid. For comparison, we use the experimental results found in [7,25] and computational results in [39].

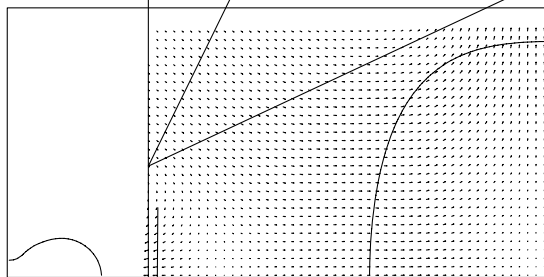


Table 7  
Convergence study for the Rayleigh capillary instability problem using the two-phase sharp interface method

Grid	$E_{\text{interface}}$	$E_{\text{Liquid}}^{\text{avg}}$	$E_{\text{Liquid}}^{\text{max}}$	$E_{\text{vapor}}^{\text{avg}}$
16 × 32	N/A	N/A	N/A	N/A
32 × 64	4.2	3.2	0.013	32.8
64 × 128	0.9	1.1	0.004	11.1

Elapsed time is  $t = 80$ . The viscosity and density ratios are  $\mu_L/\mu_G = 64$  and  $\rho_L/\rho_G = 816$ , respectively. The Reynolds number is 200.

Table 8  
Convergence study for the Rayleigh capillary instability problem using the single-phase sharp interface method

Grid	$E_{\text{interface}}$	$E_{\text{Liquid}}^{\text{avg}}$	$E_{\text{Liquid}}^{\text{max}}$
16 × 32	N/A	N/A	N/A
32 × 64	13.6	7.8	0.012
64 × 128	7.5	4.3	0.011

Elapsed time is  $t = 80$ . The Reynolds number is 7.5.

As in [7,25], we will present our computational results in terms of the following dimensionless groups. The Reynolds number  $Re$ , the Eötvös number  $EO$ , and the Morton number  $Mo$  are defined as follows:

$$Re = \frac{\rho LU}{\eta_L} \quad EO = \frac{gL^2U}{\sigma} \quad Mo = \frac{g\eta_L^4}{\rho\sigma^3} \tag{38}$$

where  $\rho$  is the liquid density,  $L$  is the bubble diameter,  $U$  is a characteristic velocity,  $\eta_L$  is the liquid viscosity,  $\sigma$  is the surface tension, and  $g$  is the acceleration of gravity.

Another set of useful dimensionless numbers, although not independent of those in (38), are the Weber number  $We$ , the Froude number  $Fr$ , and the drag coefficient  $C_D$ :

$$We = \frac{\rho LU^2}{\sigma} \quad Fr = \frac{U^2}{gL} \quad C_D = \frac{4\rho gL^2}{3\eta_L U}$$

In all of our bubble calculations, we use adaptive mesh refinement[46] with a base coarse grid of  $24 \times 72$  grid cells and three levels of adaptivity. The computational domain size was  $2.0 \times 6.0$ . Our computations use 3d-axisymmetric  $r-z$  coordinates. A comparison of computed terminal bubble rise velocity versus previous computational and experimental results are reported in Table 9. A comparison of computed terminal bubble shapes versus previous computational and experimental results are reported in Fig. 17. Our comparisons include oblate ellipsoidal cap bubbles studied by [7] ( $EO = 243$ ,  $Mo = 266$ , and  $Re = 7.77$  for bubble Fig. 2(d) and  $EO = 116$ ,  $Mo = 5.51$ , and  $Re = 13.3$  for bubble Fig. 3(d)), spherical cap bubbles studied by Hnat and Buckmaster [25] ( $Re = 19.4$ ,  $Mo = 0.065$ , and  $C = 4.95$ , where  $C = \frac{r}{(v^2/g)^{1/3}}$ ), and a disk-bubble studied by Ryskin and Leal [39] ( $R = 100$  and  $We = 10$ ).

Finally, we remark that for these bubble rise test problems, the “semi-implicit ghost-fluid” approach (our two-phase approach with velocity extrapolation disabled) produces results comparable with our two-phase approach. Results for the “semi-implicit ghost-fluid” approach are shown in Table 10 and Fig. 18.

### 6.6.1. Full 3d bubble dynamics

As a validation of our sharp interface method in 3 dimensions, we compute bubble motion in 3d-Cartesian coordinates ( $x$ ,  $y$ , and  $z$ ) and compare our results to the corresponding 3d-axisymmetric computations. The dimensions of the computational domain was  $4 \times 4 \times 6$ . We computed 3d bubble motion on an adaptive grid with base coarse grid size of  $16 \times 16 \times 24$  and 3 additional levels of adaptivity. In Fig. 19 we show the computed bubble shape in which we used the same physical properties as the  $D = 12.15$  case in Hnat and Buckmaster’s paper [25]. The experimental rise speed (in terms of the  $Re$  number) is 19.4 and our computed rise speed is 19.5. In Fig. 20 we show the computed bubble shape in which we used the same physical properties as in Fig. 3(d) of Bhaga and Weber’s paper [7]. The experimental rise speed (in terms of the  $Re$  number) is 13.3 and our computed rise speed is 13.6.

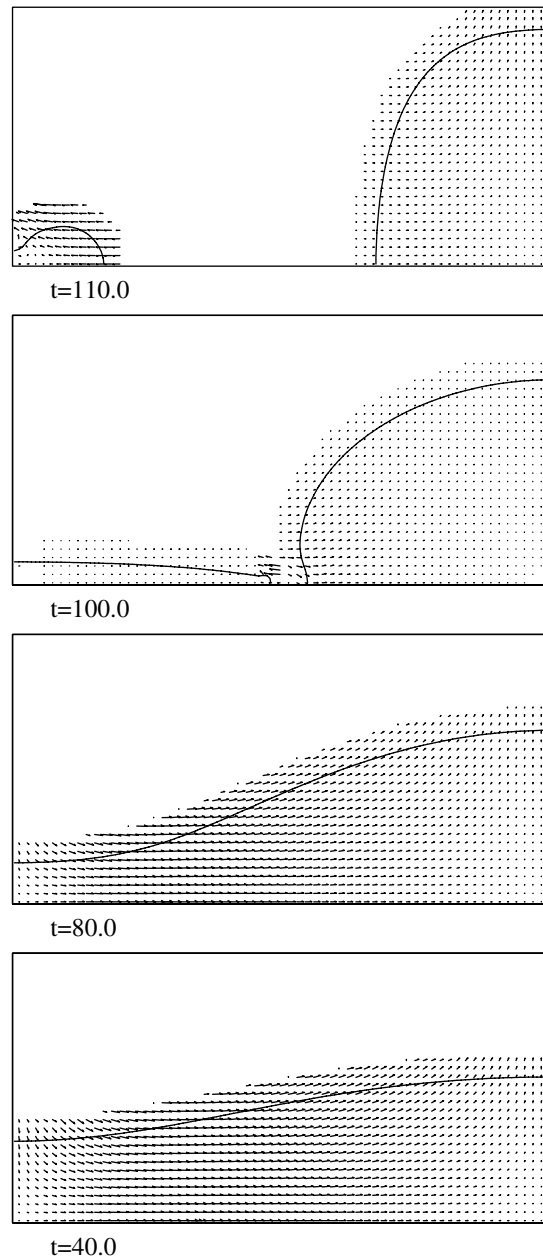


Fig. 16. Capillary instability. Single-phase sharp interface method.  $\rho_L/\rho_G = 816$ ,  $\mu_L/\mu_G = 64$ . Grid resolution is  $64 \times 128$ .

Table 9

Comparison of computed terminal bubble rise speed (in terms of the  $Re$  number) compared with experiments (Bhaga and Weber, Buckmaster) and compared with previous calculations (Ryskin and Leal)

Case	Sharp interface method	Experiment/previous result
Fig. 2d (Bhaga and Weber)	8.3	7.8
Fig. 3d (Bhaga and Weber)	14.1	13.3
Ryskin and Leal ( $Re = 100$ , $We = 10$ )	97.5	100
Buckmaster ( $D = 12.15$ )	19.8	19.4

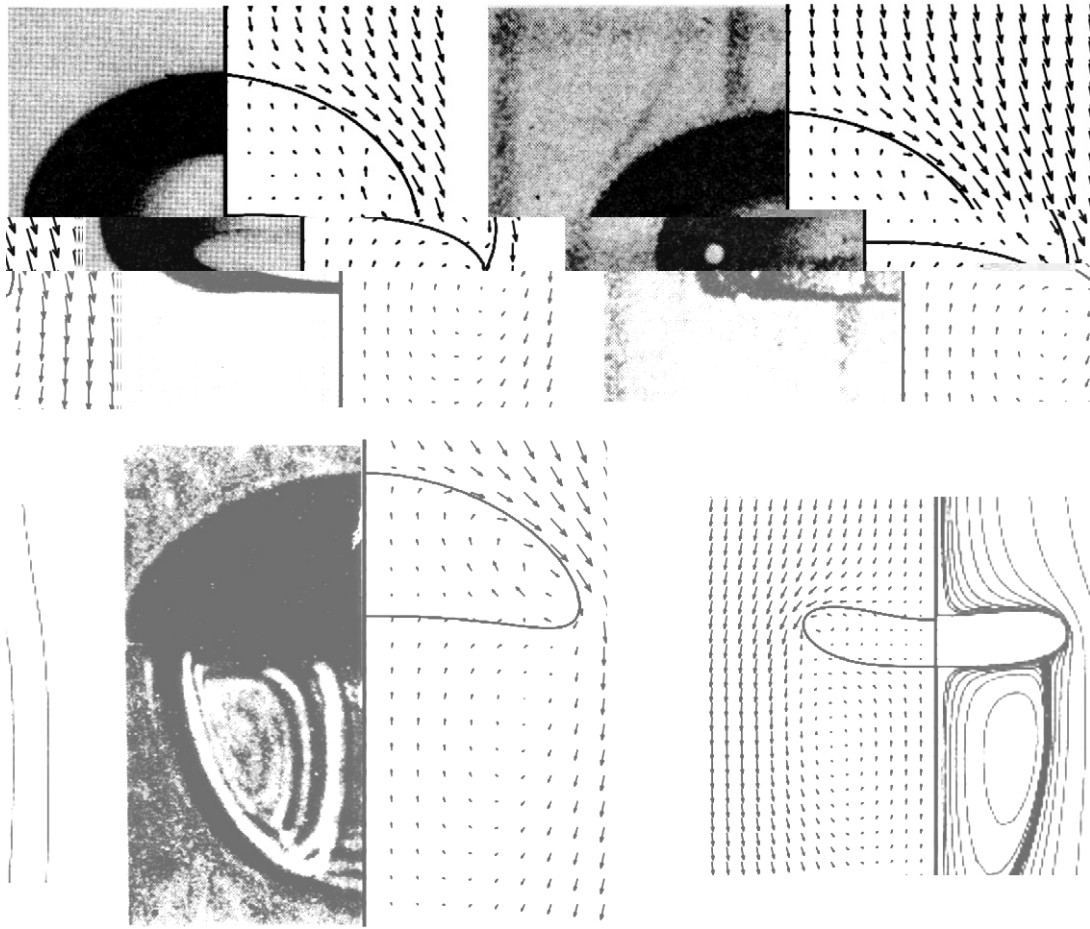


Fig. 17. Comparison of our numerical results (two-phase sharp interface method) with experimental/benchmark results. Upper left: Bhaga and Weber (Fig. 2, bubble (d)). Upper right: Bhaga and Weber (Fig. 3, bubble (d)). Lower left: Hnat and Buckmaster. Lower right: Ryskin and Leal.

Table 10

Comparison of computed terminal bubble rise speed (in terms of the  $Re$  number) using the “semi-implicit” ghost-fluid sharp interface method compared with experiments (Bhaga and Weber, Buckmaster) and compared with previous calculations (Ryskin and Leal)

Case	Semi-implicit ghost-fluid	Experiment/previous result
Fig. 2d (Bhaga and Weber)	8.1	7.8
Fig. 3d (Bhaga and Weber)	13.7	13.3
Ryskin and Leal ( $Re = 100$ , $We = 10$ )	97.6	100
Buckmaster ( $D = 12.15$ )	19.7	19.4

### 6.7. Bubble formation

In this section we compute the formation of bubbles caused by the injection of air into a container of liquid. Our computations use 3d-axisymmetric  $r$ - $z$  coordinates. We enforce inflow boundary conditions at the bottom of the domain ( $z = 0$ ),

$$\nabla p \cdot \mathbf{n} = 0$$

$$\mathbf{U} \cdot \mathbf{n} = \begin{cases} u_{\text{inflow}} & r < r_{\text{nozzle}} \\ 0 & \text{otherwise} \end{cases}$$

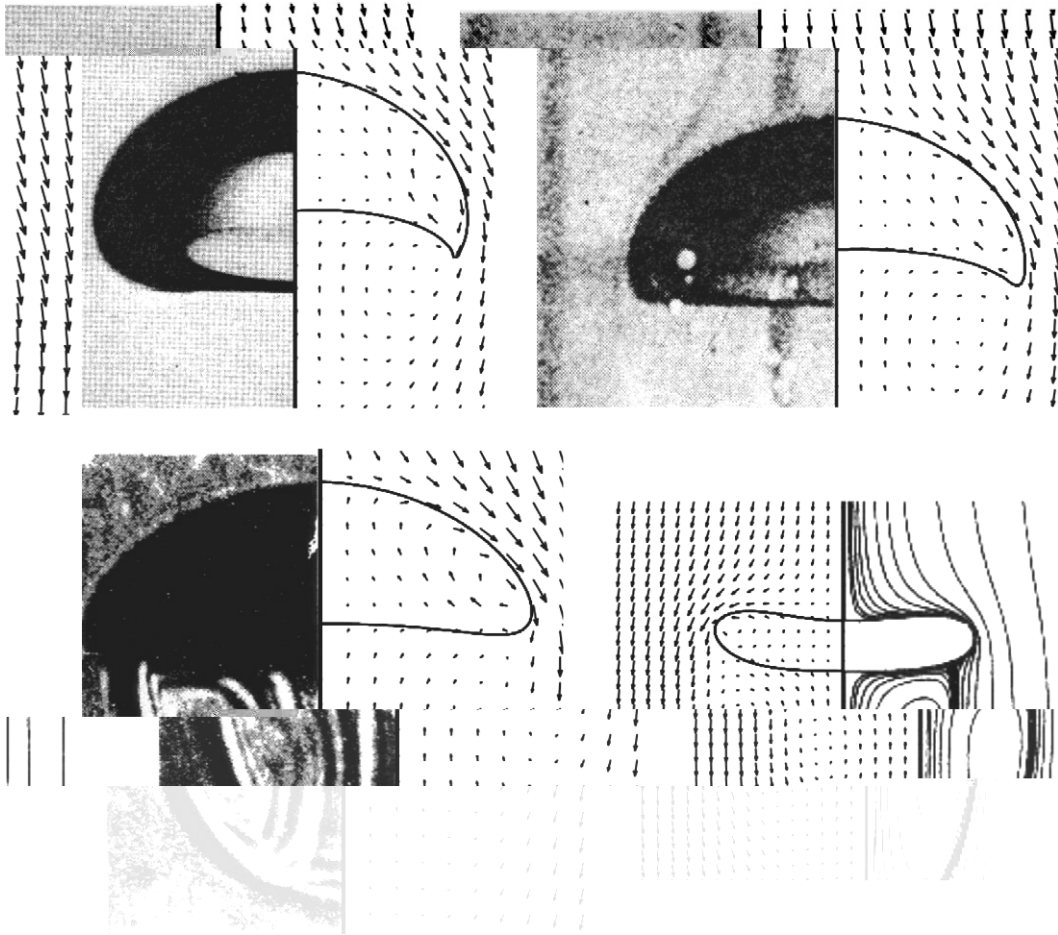


Fig. 18. Comparison of our numerical results (two-phase semi-implicit ghost-fluid method) with experimental/benchmark results. Upper left: Bhaga and Weber (Fig. 2, bubble (d)). Upper right: Bhaga and Weber (Fig. 3, bubble (d)). Lower left: Hnat and Buckmaster. Lower right: Ryskin and Leal.

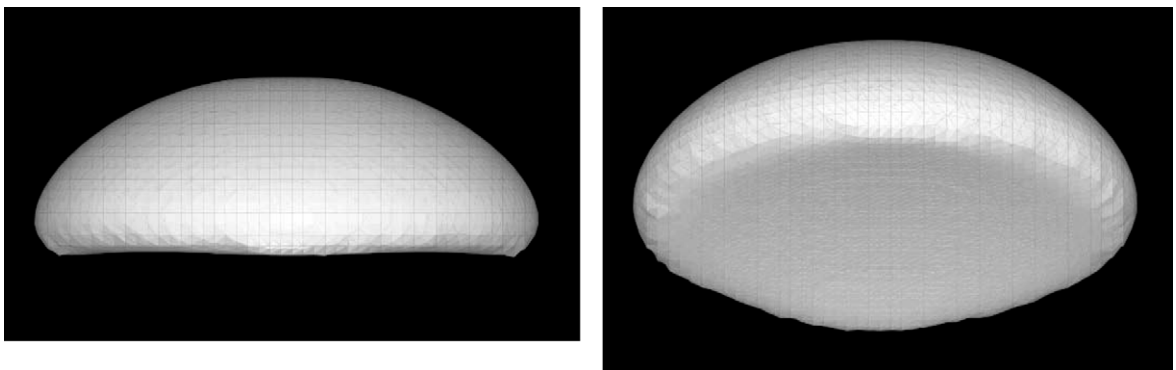


Fig. 19. Full 3d computations of a rising gas bubble in liquid. Physical properties correspond to the  $D = 12.15$  case in Hnat and Buckmaster. Left: side. Right: bottom.

Symmetry boundary conditions are given at  $r = 0$ , free-slip conditions at  $r = r_{\text{high}}$ , and outflow conditions at the top of the domain ( $z = z_{\text{high}}$ ):

$$p = 0$$

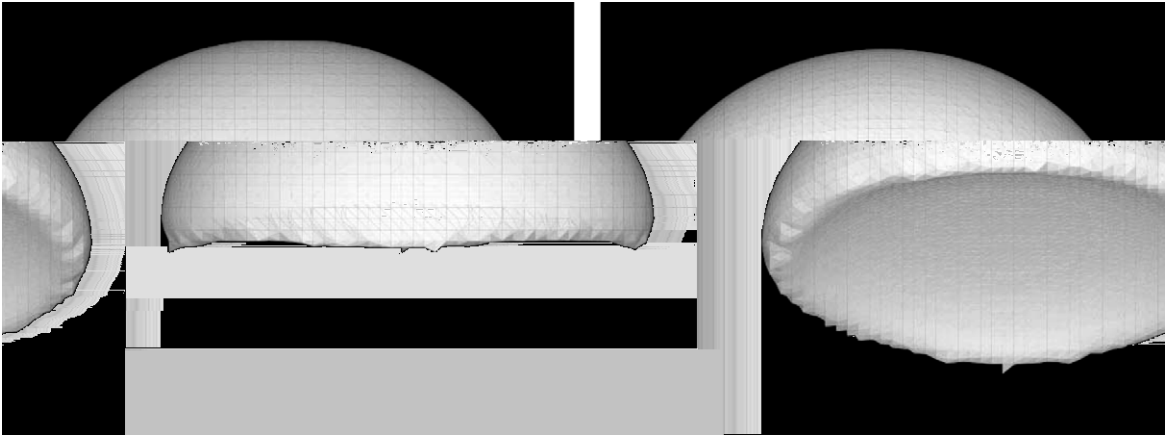


Fig. 20. Full 3d computations of a rising gas bubble in liquid. Physical properties correspond to Fig. 3(d) case in Bhaga and Weber. Left: side. Right: bottom.

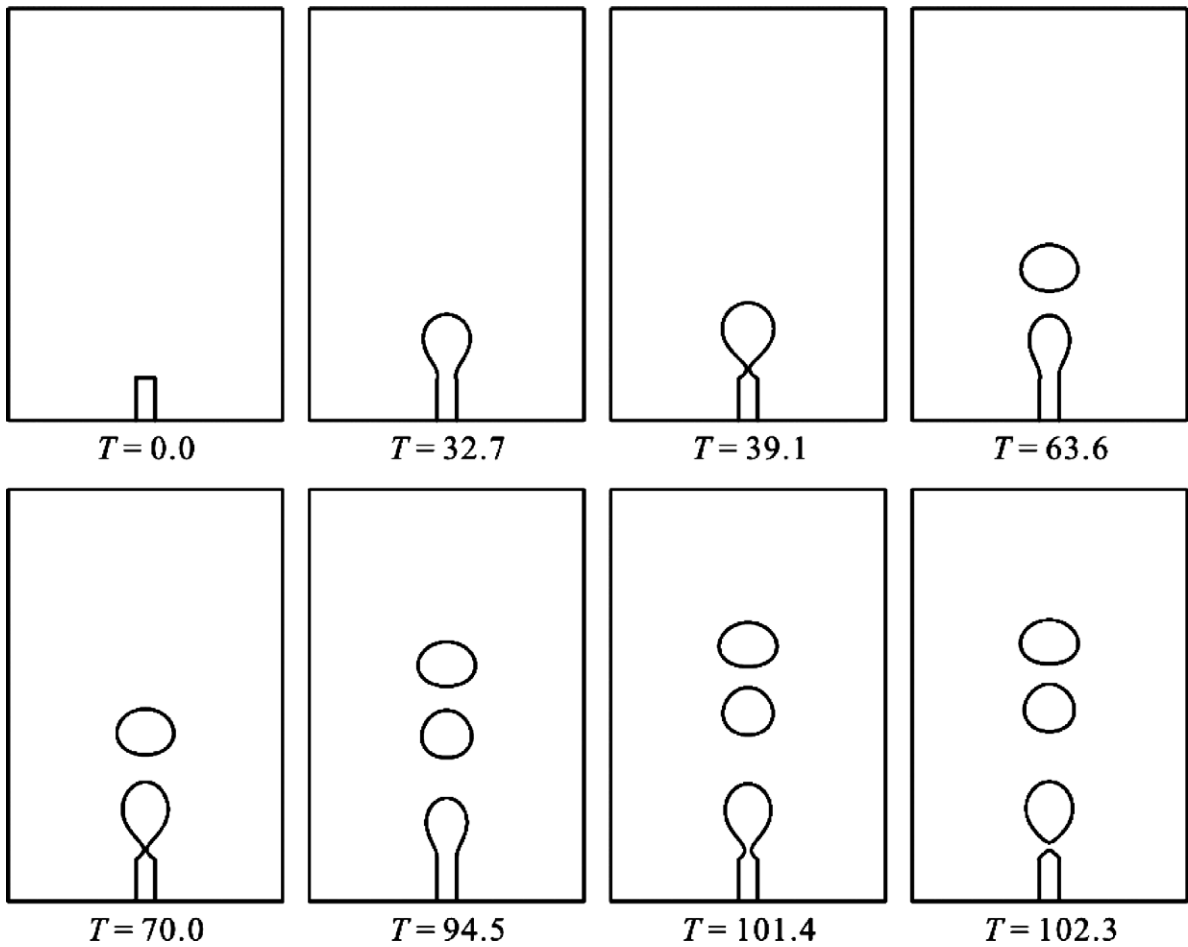


Fig. 21. Bubble formation computed using two-phase sharp interface method. Nozzle radius  $8.5E - 4m$ . Inflow velocity 0.44 m/s. Density ratio 1015:1, Viscosity ratio 6923:1.

We compare results of our two-phase sharp interface method with experimental results reported by Helsby and Tuson [23]. Our target is Fig. 1(e) in [23]. This corresponds with a nozzle radius of  $8.5E - 4m$  and an inflow velocity of 0.44 m/s. Based on the physical properties of the case-e system, one has the Reynolds number equal to 3.6, the Weber number equal to 3.06, the density ratio equal to 1015:1 and the viscosity ratio equal to 6923:1. We used Adaptive mesh refinement [46] to compute the solutions for the bubble formation problem with a base coarse grid of  $32 \times 96$  grid cells and three levels of adaptivity. There were 16 fine grid cells spanning the nozzle radius. In Fig. 21 we illustrate our computational results. The bubble diameters for the 2nd and 3rd bubbles were  $4.85E - 3m$  and  $4.90E - 3m$ , respectively, which is in good agreement with the experimental result  $4.99E - 3m$ .

## 7. Conclusions

A sharp interface method for two-phase flows has been developed. Our method has been designed to reduce to a “single-phase” approach in the limiting case of zero gas density and zero gas viscosity. Also, a new cell-centered semi-implicit treatment for the viscous terms has been developed which enables us to bypass the viscous time step constraint while treating the viscosity jump as “sharp.” For problems with a thin free-surface boundary layer, our results are superior to the “semi-implicit ghost-fluid” method. For problems in which the Reynolds number is large in the liquid, our results demonstrate second-order accuracy for the liquid solution of two-phase incompressible flows. For problems in which viscous effects are dominant, both our “two-phase” and “one-phase” sharp interface approaches become first-order accurate. In fact, the errors of all three approaches, (1) our proposed sharp interface method, (2) our “semi-implicit ghost-fluid” method, and (3) our “single-phase” method are all comparable to each other when viscous effects are sufficiently present. When viscous effects are weak, then our sharp interface approach gives higher accuracy than our “semi-implicit ghost-fluid” approach. This is expected, since it is for this class of problems that the solutions admitted from a ghost-fluid approach (which assumes continuity of the tangential velocity) diverge from our sharp-interface approach (and diverge from a “one-phase” approach). The improved accuracy over conventional first-order “continuum” approaches and “ghost-fluid” approaches allows us to resolve computations using a coarse mesh where otherwise a fine mesh is required. As demonstrated in our bubble formation test, our new method can reliably handle complex interfacial geometries.

## Acknowledgments

We thank D. Kikuchi and S. Yamaguchi for their help in preparing this manuscript.

## Appendix A

### A.1. One-phase algorithm

The one-phase algorithm addresses a two-phase flow problem in which the liquid is assumed to behave incompressibly, and the pressure in the gas is spatially constant [45,17,11].

In the liquid we have,

$$\rho \frac{DU}{Dt} = \nabla \cdot (-pI + 2\mu D) + \rho g \hat{z}$$

$$\nabla \cdot \mathbf{U} = 0$$

where  $\mathbf{U}$  is the velocity vector,  $\rho$  is the density,  $p$  is the pressure,  $\mu$  is the coefficient of viscosity,  $g$  is the gravity,  $I$  is the unit tensor,  $\hat{z}$  is the unit vector in the vertical direction, and  $D$  is the deformation tensor defined by

$$D = \frac{\nabla \mathbf{U} + (\nabla \mathbf{U})^T}{2}$$



In the vapor, we assume  $p(t)$  is constant in space. The vapor viscosity  $\mu_G$  and “density”  $\rho_G$  are assumed to be zero. The free surface boundary conditions are enforced by specifying the following pressure boundary condition at the free surface:

$$p(\mathbf{x}, t) = p_{\text{vapor}}(t) - \sigma\kappa + 2\mu_L(D_L \cdot \mathbf{n}) \cdot \mathbf{n} \tag{39}$$

where  $\kappa$  is the local mean curvature,  $\mu_L$  is the liquid viscosity, and  $D_L$  is the rate of deformation for the liquid.

If one defines the interface  $\Gamma$  as the zero level set of a smooth level set function,  $\phi$ , then the resulting equations are written as:

$$\rho \frac{DU}{Dt} = \nabla \cdot (-pI + 2\mu D) + \rho g \hat{z} - (\sigma\kappa - p_{\text{vapor}}(t))\nabla H \tag{40}$$

$$\nabla \cdot \mathbf{U} = 0$$

$$\frac{D\phi}{Dt} = 0 \quad \rho = \rho_L H(\phi) \quad \mu = \mu_L H(\phi) \tag{41}$$

where  $\kappa(\phi)$  and  $H(\phi)$  are defined by Eqs. (3) and (4), respectively.

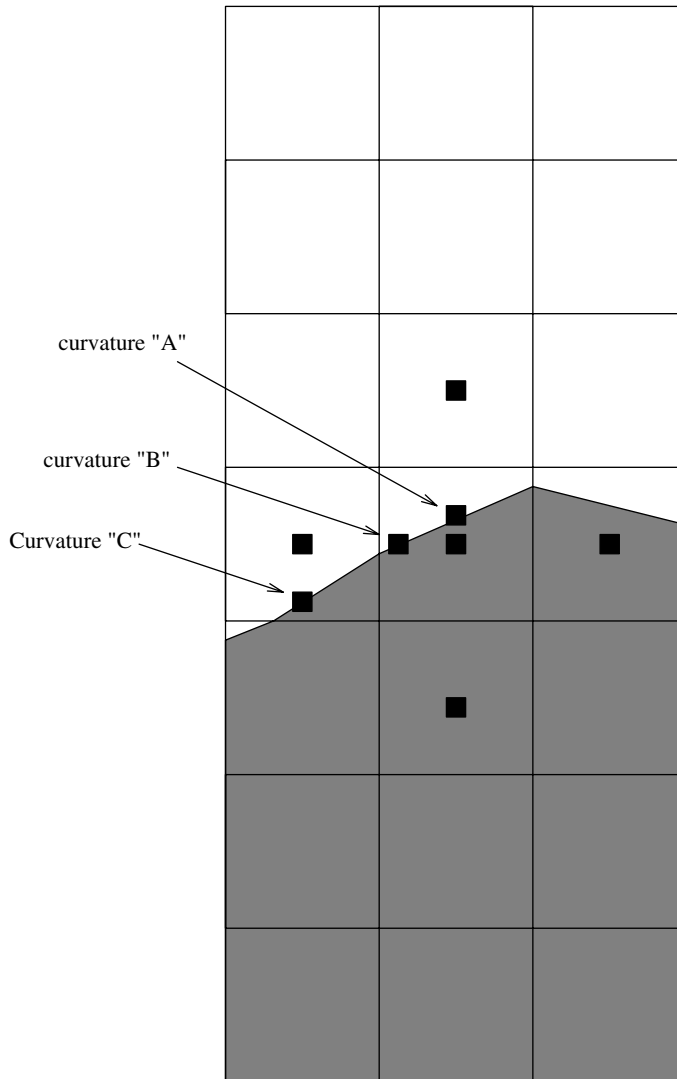


Fig. 22. The volume fractions in the following  $3 \times 7$  stencil are used to approximate curvature “A” to second-order accuracy. In order to compute curvature “B” to second-order accuracy, one must linearly interpolate between curvature “A” and curvature “C”.

Boundary conditions must be specified in the vapor ( $\phi < 0$ ). The boundary conditions are  $p = 0$  and  $U = U_{\text{extrapolate}}^{\text{liquid}}$ . In the pressure projection step, the density is expressed in terms of the height fraction (see Eq. (28) except replace  $\rho_G$  with zero). The discretization of the pressure projection step is second-order accurate (see [19]). As in [19,45], we prescribe a cutoff for the height fraction  $\theta_{i+1/2,j}$  (see Eq. (28)) which is 0.001. Further details for the discretization of (40) thru (41) are given in [45].

A.2. Curvature discretization

The curvature on the free surface is computed to second-order accuracy directly from the volume fractions [22]. Previous work in this area include that by Chorin [13], Poo and Ashgriz [34], Aleinov and Puckett [1], Williams et al. [55] and more recently, using “PROST”, Renardy et al. [37]. The method we use here is explicit, localized, and can be shown thru Taylor series expansion to be second-order accurate for  $r$ - $z$  or 3d coordinate systems. The method is based on reconstructing the “height” function directly from the volume fractions [22]. Without loss of generality, we assume that the free surface is oriented more horizontal than vertical. The orientation of the free surface is determined from the level set function since  $\mathbf{n} = \nabla\phi/|\nabla\phi|$ . A  $3 \times 7$  stencil of volume fractions is constructed about cell  $(i,j)$  (see Fig. 22). The 3 vertical sums,  $F_{i'}$ ,  $i' = i - 1, i, i + 1$  correspond to the integrals of the height function  $h(x)$  (see Fig. 23); i.e.  $F_i = \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} h(x) dx + C(j)$ . It can be shown that  $(F_{i+1} - F_{i-1})/(2\Delta x)$  is a second-order approximation to  $h'(x_i)$  and that  $(F_{i+1} - 2F_i + F_{i-1})/\Delta x^2$  is a second-order approximation to  $h''(x_i)$ . A slightly more complicated procedure is used in axisymmetric coordinate systems; the height function  $h(r)$  is assumed to have the form  $ar^2 + br + c$ . The integral of  $rh(r)$  is related with  $F_{i'}$ ,  $i' = i - 1, i, i + 1$  in order to solve for the 3 unknowns  $a$ ,  $b$  and  $c$ . For vertically oriented interfaces in axisymmetric coordinate systems, the  $F_{j'}$  represent the integrals of the square of the height function  $h(z)$  (up to a constant):  $F_{j'} = \frac{1}{\Delta z} \pi \int_{z_{j'-1/2}}^{z_{j'+1/2}} (h(z))^2 dz + C(i)$ . In other words,  $(F_{j+1} - F_{j-1})/(2\Delta z)$  is a second-order approximation to  $dh(z)^2/dz$  and  $(F_{j+1} - 2F_j + F_{j-1})/\Delta z^2$  is a second-order approximation to  $d^2(h(z)^2)/dz^2$ . The resulting curvature is obtained directly from the height function (whether it be  $h(r)$ ,  $h(z)$  or  $h(x,y)$ ).

This procedure for finding curvature will return a second-order approximation to the curvature on the interface passing thru cell  $(i,j)$  located at  $x = (i + 1/2)\Delta x$  (horizontal orientation) or  $y = (j + 1/2)\Delta y$  (vertical

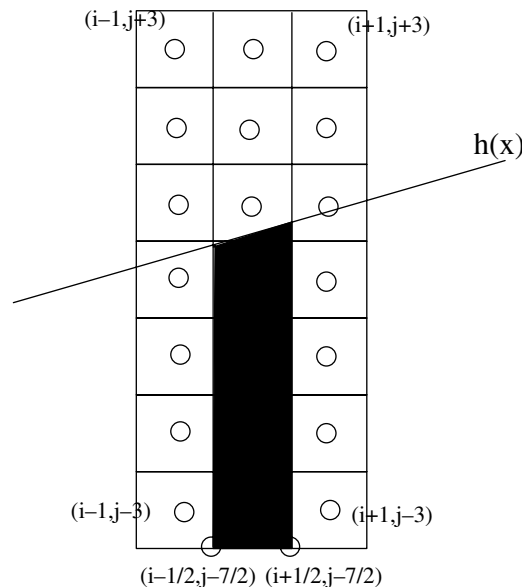


Fig. 23. Stencil for calculating the curvature in cell  $(i,j)$  when the level set function changes sign between cells  $(i,j)$  and  $(i,j + 1)$ . The shaded area corresponds to the vertical sum of the volume fractions,  $\Delta x \Delta y \sum_{j=j-3}^{j+3} F_{i,j}$ , and the shaded area also corresponds to the integral of the height function  $h(x)$ ,  $\int_{x_{i-1/2}}^{x_{i+1/2}} h(x) dx + C(j)$ .

Table 11

Convergence study for computing curvatures from volume fractions of a unit sphere in axisymmetric geometry

$\Delta x$	Maximum error	Average error
1/16	0.0104	0.0037
1/32	0.0024	0.0009
1/64	0.0006	0.0002

The physical domain size is  $2 \times 4$ .  $\Delta x$  is the mesh spacing which is  $2/n_x$ , where  $n_x$  is the number of cells in the  $x$  direction. For all of our tests,  $\Delta x = \Delta y$ .

Table 12

Convergence study for computing curvatures from volume fractions of a unit sphere in three-dimensional geometry

$\Delta x$	Maximum error	Average error
1/8	0.094	0.0125
1/16	0.050	0.0036
1/32	0.010	0.0009

The physical domain size is  $4 \times 4 \times 4$ .  $\Delta x$  is the mesh spacing which is  $4/n_x$ , where  $n_x$  is the number of cells in the  $x$  direction. For all of our tests,  $\Delta x = \Delta y = \Delta z$ .

orientation). In order to find  $\kappa_I(F)$  to second-order accuracy (17), we have two different cases when the level set function changes sign between cells  $(i,j)$  and  $(i+1,j)$ : (1) the interface is orientated vertically, in which case

$$\kappa_I = \begin{cases} \kappa_{ij} & \theta < 1/2 \\ \kappa_{i+1,j} & \text{otherwise} \end{cases}$$

or (2) the interface is orientated horizontally, in which case

$$\kappa_I = (1 - \theta)\kappa_{ij} + \theta\kappa_{i+1,j}$$

In Tables 11 and 12, we display the average error and maximum error for the case of a sphere in axisymmetric and three-dimensional coordinate systems, respectively.

## References

- [1] I. Aleinov, E.G. Puckett, Computing surface tension with high-order kernels, in: Proceedings of the 6th International Symposium on Computational Fluid Dynamics, Lake Tahoe, CA, 1995.
- [2] A.S. Almgren, J.B. Bell, P. Colella, L.H. Howell, M. Welcome, A conservative adaptive projection method for the variable density incompressible Navier–Stokes equations, *J. Comput. Phys.* 142 (1998) 1–46.
- [3] E. Aulisa, S. Manservigi, R. Scardovelli, A mixed markers and volume-of-fluid method for the reconstruction and advection of interfaces in two-phase and free-boundary flows, *J. Comp. Phys.* 188 (2003) 611.
- [4] E. Aulisa, S. Manservigi, R. Scardovelli, A surface marker algorithm coupled to an area-preserving marker redistribution method for three-dimensional interface tracking, *J. Comp. Phys.* 197 (2) (2004) 555–584.
- [5] J.B. Bell, P. Colella, H.M. Glaz, A second-order projection method for the incompressible Navier–Stokes equations, *J. Comput. Phys.* 85 (December) (1989) 257–283.
- [6] J.B. Bell, D.L. Marcus, A second-order projection method for variable-density flows, *J. Comput. Phys.* 101 (1992) 334–348.
- [7] D. Bhaga, M.E. Weber, Bubbles in viscous liquids: shapes, wakes and velocities, *J. Fluid Mech.* 105 (1981) 61–85.
- [8] A. Bourlioux, A coupled level-set volume-of-fluid algorithm for tracking material interfaces, in: Proceedings of the 6th International Symposium on Computational Fluid Dynamics, Lake Tahoe, CA, 1995.
- [9] J.U. Brackbill, D.B. Kothe, C. Zemach, A continuum method for modeling surface tension, *J. Comput. Phys.* 100 (1992) 335–353.
- [10] S. Bradford, N.D. Katopodes, Hydrodynamics of turbid underflows. Part II: aggradation, avulsion and channelization, *Journal of Hydraulic Engineering*, ASCE 125 (10) (1999) 1016–1028.
- [11] A. Caboussat, M. Picasso, J. Rappaz, Numerical simulation of free surface incompressible liquid flows surrounded by compressible gas, *J. Comput. Phys.* 203 (2) (2005) 626–649.
- [12] Y.C. Chang, T.Y. Hou, B. Merriman, S. Osher, Eulerian capturing methods based on a level set formulation for incompressible fluid interfaces, *J. Comput. Phys.* 124 (1996) 449–464.
- [13] A.J. Chorin, Curvature and solidification, *J. Comput. Phys.* 57 (1985) 472–490.
- [14] T.J. Craft, J.W. Kidger, B.E. Launder, Three dimensional modelling of turbulent free-surface jets, in: W. Rodi, D. Laurence (Eds.), *Engineering Turbulence Modelling and Measurements*, vol. 4, Elsevier, Amsterdam, 1999, pp. 73–82.

- [15] D.G. Dommermuth, M. Gharib, H. Huang, G.E. Innis, P. Maheo, E.A. Novikov, J.C. Talcott, D.C. Wyatt, Turbulent free-surface flows: a comparison between numerical simulations and experimental measurements, in: Proceedings of the 21st Symposium on Naval Hydro, Trondheim, 1996, pp. 249–265.
- [16] D. Enright, R. Fedkiw, J. Ferziger, I. Mitchell, A hybrid particle level set method for improved interface capturing, *J. Comp. Phys.* 183 (1) (2002) 83–116.
- [17] D. Enright, S. Marschner, R. Fedkiw, Animation and rendering of complex water surfaces, in: SIGGRAPH 2002, volume ACM TOG 21, 2002, pp. 736–744.
- [18] M. Francois, S. Cummins, E. Dendy, D. Kothe, J. Sicilian, M. Williams, A balanced-force algorithm for continuous and sharp interfacial surface tension models within a volume tracking framework, *J. Comput. Phys.* 213 (1) (2006) 141–173.
- [19] F. Gibou, R. Fedkiw, L.-T. Cheng, M. Kang, A second order accurate symmetric discretization of the poisson equation on irregular domains, *J. Comput. Phys.* 176 (2002) 205–227.
- [20] D. Gueyffier, J. Li, A. Nadim, R. Scardovelli, S. Zaleski, Volume of fluid interface tracking with smoothed surface stress methods for three-dimensional flows, *J. Comput. Phys.* 152 (1999) 423–456.
- [21] B. Helenbrook, L. Martinelli, C. Law, A numerical method for solving incompressible flow problems with a surface of discontinuity, *Journal of Computational Physics* 148 (1999) 366–396.
- [22] J. Helmsen, P. Colella, E.G. Puckett, Non-convex profile evolution in two dimensions using volume of fluids, LBNL Technical Report LBNL-40693, Lawrence Berkeley National Laboratory, 1997.
- [23] F.W. Hulsby, K.R. Tuson, Behaviour of air bubbles in aqueous solutions, *Research* 8 (1955) 270.
- [24] S. Hieber, P. Koumoutsakos, A Lagrangian particle level set method, *Journal of Computational Physics* 210 (1) (2005) 342–367.
- [25] J.G. Hnat, J.D. Buckmaster, Spherical cap bubbles and skirt formation, *Physics of Fluids* 19 (2) (1976) 182–194.
- [26] M. Kang, R. Fedkiw, X.-D. Liu, A boundary condition capturing method for multiphase incompressible flow, *J. Sci. Comput.* 15 (2000) 323–360.
- [27] H. Lamb, *Hydrodynamics*, Dover Publications, New York, 1932.
- [28] H. Liu, S. Krishnan, S. Marella, H.S. Udaykumar, Sharp interface cartesian grid method II: a technique for simulating droplet interactions with surfaces of arbitrary shape, *J. Computational Physics* 210 (1) (2005) 32–54.
- [29] Daniel Lorstad, Laszlo Fuchs, High-order surface tension VOF-model for 3D bubble flows with high density ratio, *Journal of Computational Physics* 200 (1) (2004) 153–176.
- [30] E. Marchandise, J.-F. Remacle, N. Chevaugnon, A quadrature-free discontinuous Galerkin method for the level set equation, *Journal of Computational Physics* 212 (1) (2006) 338–357.
- [31] M.L. Minion, On the stability of Godunov-projection methods for incompressible flow, *J. Comput. Phys.* 123 (2) (1996) 435–449.
- [32] D. Nguyen, R. Fedkiw, M. Kang, A boundary condition capturing method for incompressible flame discontinuities, *J. Comput. Phys.* 172 (2001) 71–98.
- [33] J.E. Pilliod, E.G. Puckett, Second-order accurate volume-of-fluid algorithms for tracking material interfaces, *Journal of Computational Physics* 199 (2) (2004) 465–502.
- [34] J.Y. Poo, N. Ashgriz, A computational method for determining curvatures, *J. Comput. Phys.* 84 (1989) 483–491.
- [35] S. Popinet, S. Zaleski, A front-tracking algorithm for accurate representation of surface tension, *International Journal for Numerical Methods in Fluids* 30 (6) (1999) 775–793.
- [36] H. Rasmussen, A. Bach, O. Hassager, 3D transient free surface viscoelastic flow, in: XIIth International Workshop on Numerical Methods for Non-Newtonian Flows, Abstracts, June, 2001, Monterey Bay, USA.
- [37] Y. Renardy, M. Renardy, Prost: A parabolic reconstruction of surface tension for the volume-of-fluid method, *J. Comput. Phys.* 183 (2) (2002) 400–421.
- [38] M. Rudman, A volume tracking method for interfacial flows with large density variations, *Int. J. Numer. Methods Fluids* 28 (1998) 357–378.
- [39] G. Ryskin, L.G. Leal, Numerical solution of free boundary problems in fluid mechanics. Part 2 buoyancy-driven motion of a gas bubble through a quiescent liquid, *J. Fluid Mech.* 148 (1984) 19–35.
- [40] R. Scardovelli, S. Zaleski, Interface reconstruction with least-square fit and split Eulerian–Lagrangian advection, *Int. J. Numer. Meth. Fluids* 41 (2003) 251–274.
- [41] Seungwon Shin, Damir Juric, Modeling three-dimensional multiphase flow using a level contour reconstruction method for front tracking without connectivity, *J. Comp. Phys.* 180 (2) (2002) 427–470.
- [42] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock capturing schemes, ii, *J. Comput. Phys.* 83 (1989) 32–78.
- [43] J. Strain, Tree methods for moving interfaces, *J. Comput. Phys.* 151 (2) (1999) 616–648.
- [44] G. Strang, On the construction and comparison of difference schemes, *SIAM J. Numer. Anal.* 5 (1968) 506–517.
- [45] M. Sussman, A second order coupled levelset and volume of fluid method for computing growth and collapse of vapor bubbles, *Journal of Computational Physics* 187 (2003) 110–136.
- [46] M. Sussman, A parallelized, adaptive algorithm for multiphase flows in general geometries, *Computers and Structures* 83 (2005) 435–444.
- [47] M. Sussman, A. Almgren, J. Bell, P. Colella, L. Howell, M. Welcome, An adaptive level set approach for incompressible two-phase flows, *J. Comput. Phys.* 148 (1999) 81–124.
- [48] M. Sussman, D. Dommermuth, The numerical simulation of ship waves using cartesian grid methods, in: Proceedings of the 23rd Symposium on Naval Hydrodynamics, September, 2000, Val-De-Reuil, France.

- [49] M. Sussman, M.Y. Hussaini, A discontinuous spectral element method for the level set equation, *J. Scientific Computing* 19 (2003) 479–500.
- [50] M. Sussman, E.G. Puckett, A coupled level set and volume of fluid method for computing 3D and axisymmetric incompressible two-phase flows, *J. Comp. Phys.* 162 (2000) 301–337.
- [51] M. Sussman, P. Smereka, S.J. Osher, A level set approach for computing solutions to incompressible two-phase flow, *J. Comput. Phys.* 114 (1994) 146–159.
- [52] O. Tatebe, The multigrid preconditioned conjugate gradient method, in: 6th Copper Mountain Conference on Multigrid Methods, Copper Mountain, CO, April 4–9, 1993.
- [53] S.O. Unverdi, G. Tryggvason, A front-tracking method for viscous, incompressible, multi-fluid flows, *J. Comput. Phys.* 100 (1992) 25–37.
- [54] B. van Leer, Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov’s method, *J. Comput. Phys.* 32 (1979) 101–136.
- [55] M. Williams, D. Kothe, E.G. Puckett, Convergence and accuracy of kernel-based continuum surface tension models, in: Proceedings of the 13th U.S. National Congress of Applied Mechanics, Gainesville, FL, June 16–21, 1998.
- [56] J. Wu, Wind-induced drift currents, *J. Fluid Mech.* 68 (1975) 49–70.
- [57] B. Yang, A. Prosperetti, A second-order boundary-fitted projection method for free-surface flow computations, *J. Comput. Phys.* 213 (2006) 574–590.
- [58] X. Yang, A. James, J. Lowengrub, X. Zheng, V. Cristini, An adaptive coupled level-set/volume-of-fluid interface capturing method for unstructured triangular grids, *J. Comput. Phys.* (in press), doi:10.1016/j.jcp.2006.01.007.
- [59] Y. Yang, G. Tryggvason, Dissipation of energy by finite amplitude surface waves, *Computers and Fluids* 27 (1998) 829–845.
- [60] T. Ye, W. Shyy, J.N. Chung, A fixed-grid, sharp interface method for bubble dynamics and phase change, *J. Comp. Phys.* 174 (2001) 781–815.