

Drop Impact Simulation with a Velocity-Dependent Contact Angle

S. Afkhami* and M. Bussmann
Department of Mechanical and Industrial Engineering
University of Toronto, Toronto, ON M5S 3G8

Abstract

The simulation of axisymmetric drop impact onto a solid surface is now commonplace, but precise agreement between simulation and experiment is elusive, particularly when capillary effects are important. The reason is related to the complex relationship between the contact angle and the contact line velocity. In fact, most results are generated by assuming a constant value of the contact angle, or perhaps two values, for advancing and receding contact lines. Here, a methodology similar to [1, 2, 3] was implemented for the accurate calculation of surface tension forces in a volume-of-fluid based model, and the methodology was extended to phenomena with moving contact lines. Different models of contact angle variation with contact angle velocity were then implemented, to consider the effect of different choices of the relationship between the contact angle and the contact line speed on predictions of drop impact.

*Corresponding Author

Introduction

The numerical simulation of interfacial flows has a wide range of engineering applications. In many of these applications surface tension matters. What complicates the simulation of such flows is the presence of moving contact lines, and the surface tension associated with these. In numerical treatment of moving contact lines, contact angles are required and applied as a boundary condition at the contact line. The equilibrium contact angle θ_{eq} , which is a static contact angle, can be measured experimentally. However, the dynamic contact angle (at a moving contact line) is not in general equal to the static contact angle, rather, it is well known that the contact angle of the advancing and receding contact lines depends on the velocity of the moving contact line. However, determining this relationship is a difficult problem.

Renardy et al. [4] used a volume-of-fluid method to impose a fixed contact angle condition in two dimensions. Liu et al. [5] modeled moving contact lines assuming constant advancing θ_a and receding θ_r contact angles. Baer et al. [6] developed a finite element moving mesh model with a simplified linear relationship between the contact angle and the contact line velocity. Šikalo et al. [7] studied the drop impact using a volume-of-fluid based method and a moving contact line model with dependence on the instantaneous advancing/receding contact line velocity.

Here, a recent methodology for the accurate estimation of surface tension forces in a VOF framework [1, 2, 3] is implemented and extended to interface reconstructions and moving contact lines [8]. We have developed a model for evaluating surface tension forces to high order accuracy and applied it to moving contact lines. Our method is based on the volume-of-fluid reconstruction of interfaces and accounts for the dynamics of contact lines. Our code is an extension of an early version of the "Gerris" code of Popinet [9] for the solution of the incompressible Euler equations [10]. The code did not include a surface tension implementation nor an implementation to model viscous stresses. Over the past few years, we have implemented surface tension and variable-viscosity models in the code, and recently, we have developed a model for moving contact lines and incorporated it into the code, for the simulation of moving contact lines.

In the following sections, we present a brief description of the model, an overview of our numerical methodology, and a description of the technique for curvature evaluation and the implementation of the contact angle boundary condition. We also present two examples where contact lines drive

motion, to demonstrate the accuracy of the model. Finally, we present computational results that compare three dynamic contact angle models applied to inclined droplet impacts, where the contact line velocity varies along the moving contact line.

Formulation of the problem

The Navier-Stokes equations govern an incompressible two-phase flow:

$$U_t + \nabla \cdot (UU) = -\frac{1}{\rho}(\nabla p + \nabla \cdot (\mu(\nabla U + \nabla U^T)) + F_b) \quad (1)$$

where $U(X, t)$ is the velocity field, $\rho(X, t)$ is the density, $\mu(X, t)$ is the viscosity, and F_b represents any body forces acting on the fluid. Density and viscosity may vary from phase to phase, but are assumed constant in a particular phase. Each fluid is considered to be incompressible; thus the continuity equation

$$\nabla \cdot U = 0 \quad (2)$$

is valid for the whole domain. For a two fluid system, a characteristic function f ($= 0$ in fluid 1, and $= 1$ in fluid 2) is used to track the evolution of the interface. The advection equation for f is expressed as

$$\partial_t f + U \cdot \nabla f = 0 \quad (3)$$

Solving this equation for f leads to volume-weighted formulae for the density and viscosity:

$$\rho = \rho_1 + (\rho_2 - \rho_1)f \quad (4)$$

$$\mu = \mu_1 + (\mu_2 - \mu_1)f \quad (5)$$

where subscripts 1 and 2 refer to the two fluid phases respectively.

An adaptive refinement projection method based on a variable-density fractional-step scheme is utilized to discretise equations (1)-(5) in space and time. Variables are collocated at cell centers and advection terms are discretised using a second-order upwind scheme. A multilevel Poisson solver is used to calculate the pressure. A face-centered velocity field is exactly projected, and the cell-centered velocity field is approximately projected, onto a divergence-free velocity field, with the pressure field obtained as the solution of a Poisson equation. Since face-centered velocities are divergence-free, volume fractions are advected using these velocities. The "Continuum Surface Force" (CSF) approach [11] is used to discretise the surface tension which is then

included in the momentum equation (1) as a component of the body force.

The domain is discretised using adaptive grids. The grid is locally refined by recursively dividing it into subgrids. In this algorithm, cells that satisfy a given criterion are seeded for refinement/coarsening. Different criteria can be used to decide if a newly refined grid is to be created. One attractive approach, for example, is to refine the region around the interface of a two-phase flow. This can be used to generate a high resolution adaptive remeshing around the interface which results in more accurate calculations.

The volume-of-fluid model

In the VOF methodology, the discretised form of the characteristic function f represents the fraction of a cell filled with one of the fluid phases. Away from an interface, $f = 0$ or 1 ; in cells cut by an interface f is between zero and one. From one timestep to the next, this interface representation is advected in order to obtain new values of volume fractions in each cell. A VOF method consists of two steps: an interface reconstruction, followed by the advection of the reconstructed pieces. A "piecewise linear interface calculation" (PLIC) as in [12] is used, along with a Lagrangian advection algorithm. In the PLIC technique, given the volume fraction in an interface cell and an approximate normal vector to the interface, a linear/planar interface is constructed within each cell, corresponding exactly to the normal and the volume fraction. Methods for calculating the interface normal and curvature, and for implementing the contact angle boundary condition, are described in the following sections.

Normals, curvatures, and a surface tension model

In a CSF/VOF methodology, surface tension $F_{ST} = \sigma\kappa\delta$ is evaluated as a volume force [11], where the delta function δ is usually discretised as $\nabla\tilde{f}$, and the curvature κ is evaluated as $-\nabla\cdot(\nabla\tilde{f}/|\nabla\tilde{f}|)$, where \tilde{f} is a smoothed volume fraction field. Unfortunately, this approach can be very inaccurate, and this becomes most apparent when surface tension forces dominate a phenomenon.

It has been recently shown [13, 14] that for a CSF/VOF-based methodology, there are two ingredients for the accurate representation of surface tension forces. The first is a proper discretisation of the delta function δ ; as in [13] and [14], we approximate δ as ∇f .

The second ingredient is the accurate estimation of interface curvature κ [1, 2, 3, 13]. The method we have implemented is that of Sussman [1] and Francois et al. [3], in which a height function (HF) is

reconstructed from volume fractions and used to estimate the interfacial curvature to second order accuracy.

Here, we extend this methodology to problems with moving contact lines. We have implemented height functions to evaluate κ and F_{ST} at contact lines, that yield accurate estimates of κ and F_{ST} and, that converge with mesh refinement. In our implementation, height functions are also used for computing normals to an interface, which leads to a more accurate estimation of normals and, that also converge with spatial refinement. This guarantees that the normal vector $\vec{\mathbf{n}}(X)$ to an interface is discretised in a way that is consistent with the discretisation of the curvature κ .

The height function technique

The HF methodology is a VOF based technique; a local height function is defined as the summation of volume fractions in a direction most normal to an interface, which we evaluate via a simple finite difference discretisation of ∇f . In two dimensions, a 7×3 stencil is constructed around a cell (i, j) , as illustrated in figure 1. In this case, $|n_y| > |n_x|$, therefore, height functions are constructed by integrating volume fractions in the vertical direction as

$$h_{i,j} = \sum_{j-3}^{j+3} f_{i,j} \Delta y_j \quad (6)$$

where Δy_j denotes the mesh size in the y -direction. These height functions can then be used to compute the curvature and normals at the center of the cell (i, j) :

$$\kappa = \frac{h_{xx}}{(1 + h_x^2)^{3/2}} \quad (7)$$

$$\vec{\mathbf{n}}(X) = \begin{bmatrix} h_x \\ -1 \end{bmatrix} \quad (8)$$

where h_{xx} and h_x are discretised using second-order central-differences. In 3D, consider the case of $|n_y| > |n_z| \& |n_x|$. Nine height functions are defined as

$$h_{i,j,k} = \sum_{j-3}^{j+3} f_{i,j,k} \Delta y_j \quad (9)$$

and the curvature and normals at the center of the cell (i, j, k) can be written as

$$\kappa = \frac{h_{xx} + h_{yy} + h_{xx}h_y^2 + h_{yy}h_x^2 - 2h_{xy}h_xh_y}{(1 + h_x^2 + h_y^2)^{3/2}} \quad (10)$$

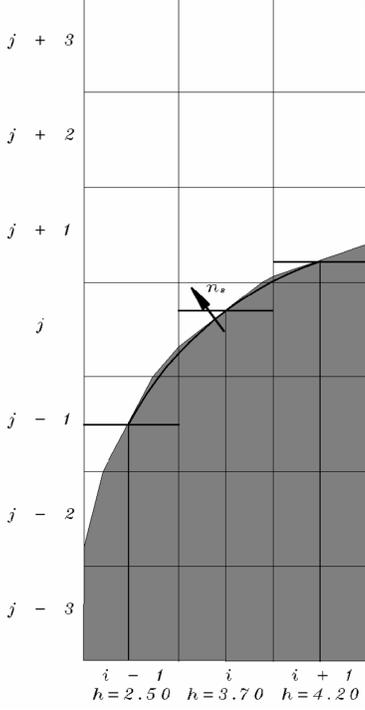


Figure 1. 2D example of a 7×3 stencil used to construct height functions.

$$\vec{\mathbf{n}}(X) = \begin{bmatrix} h_x \\ h_y \\ -1 \end{bmatrix} \quad (11)$$

Contact angle boundary condition

The orientation of the interface near the contact line reflects the contact angle, which is the angle between the normal to the interface and the normal to the solid surface at the contact line. The implementation of the contact angle boundary condition both for the curvature and normal is first explained for two dimensions, and then for three.

In 2D, the contact line is a point where the in-

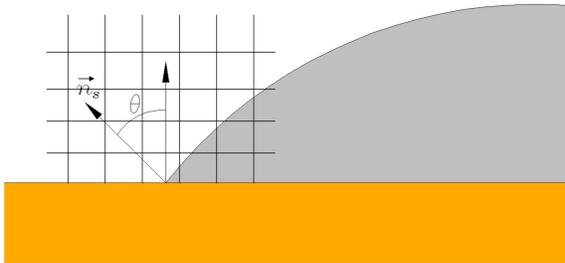


Figure 2. The contact angle θ defines the normal $\vec{\mathbf{n}}_s$ to the interface.

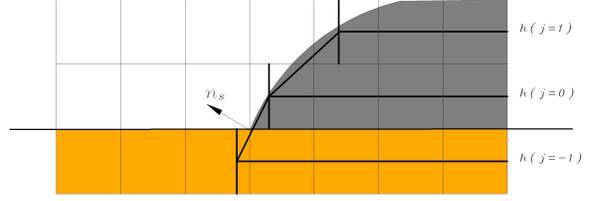


Figure 3. An example of extrapolating a height function at the contact line to calculate a height function in the ghost cells.

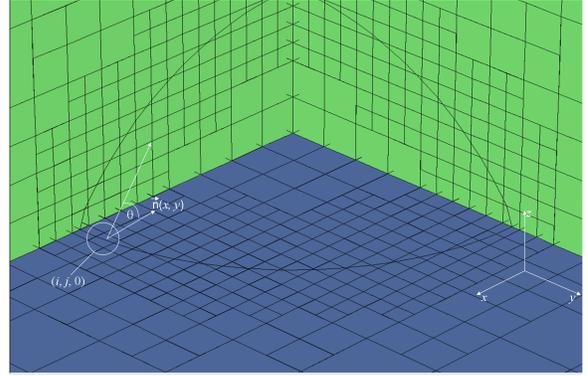


Figure 4. 3D illustration of the contact line.

terface meets a solid surface. Figure 2 shows a 2D contact line and the contact angle θ . At the center of contact line cells, the normal $\vec{\mathbf{n}}(X)$ is re-oriented to reflect the contact angle θ .

To calculate the curvature at the contact line, we limit this discussion to $45 \leq \theta \leq 135$, so that the height functions are constructed horizontally, as illustrated in figure 3. An extrapolated height is needed in the so-called "ghost cells" to compute the curvature at the contact line. This is obtained by requiring that the line through $h_{j=-1}$ and $h_{j=0}$ reflect the contact angle, *i.e.* $\vec{\mathbf{n}}_s$ is perpendicular to the line that passes through $h_{j=-1}$ and $h_{j=0}$ (figure 3).

The approach used in two dimensions cannot be easily extended to 3D contact lines. Consider a layer of cells in the $x-y$ plane just above a solid surface (figure 4). For cell $(i, j, 0)$, for example, the component of a cell centered normal in the $x-y$ plane is defined as

$$\vec{\mathbf{n}}(x, y) = \begin{bmatrix} \partial h / \partial y \\ -1 \end{bmatrix} \quad (12)$$

where in this case, x is the direction most normal to the contact line. The z -component of the normal is then defined to reflect the contact angle.

To evaluate the curvature, the local height function for the contact line cell is constructed either in

the x or y direction depending on the largest component of the normal in the x - y plane. At each contact line cell, for a given contact angle, ghost cell height functions are required to estimate the curvature: using the normal vector $\vec{n}(x, y)$ which was computed for cell $(i, j, 0)$ (figure 4), and the prescribed contact angle, the height functions in the domain are extrapolated to the ghost cells to reflect the contact angle.

Numerical results on surface tension driven flows

In this section, we present the results of our numerical algorithm applied to contact line-driven phenomena. First, consider a sessile 2D semicircle initially at rest, when suddenly a non-equilibrium contact angle is imposed. Due to the difference between the equilibrium contact angle and the initial contact angle ($\theta = 90^\circ$), the free surface will move toward a circular shape defined by the prescribed contact angle. Figure 5 illustrates results of a circle of radius $R = 0.25$ positioned at $(0.0, -0.5)$ in a 1×1 domain with outflow boundary conditions everywhere and no-slip boundary condition at $y = -0.5$. The Ohnesorge number is $Oh = 1.2 \times 10^{-2}$. The density ratio is 10^3 and the viscosity ratio is 10^2 . The prescribed contact angle is set to either $\theta = 60^\circ$ or $\theta = 120^\circ$. The results are computed on an adaptive mesh with the highest resolution equal to $1/256$. Figure 5 shows sequences of configurations for $\theta = 60^\circ$ and $\theta = 120^\circ$. The two frames at the bottom of figure 5 show the simulations at steady state; the VOF reconstructions are virtually identical to the circular analytic profiles.

The above simulations were repeated for a hemisphere in three dimensions. Figure 6 demonstrates sequences of configurations for either $\theta = 60^\circ$ or $\theta = 120^\circ$. Again, the two frames at the bottom of figure 6 show the results at steady state. A test of our 3D code is its ability to maintain symmetry during an axisymmetric process. Figure 7 illustrates the steady state contact lines for $\theta = 60^\circ$ and $\theta = 120^\circ$, computed using the HF technique. Figure 8 presents results of the same tests computed using the technique of [12, 11]. As can be seen in figure 8, results are not symmetrical. As a result of our numerical implementation, simulations have drastically improved and remained axisymmetric throughout such simulations.

Numerical results on three-dimensional inclined impact

To demonstrate the moving contact line model, we present results of inclined impacts where the contact angle varies with contact line velocity. A wa-

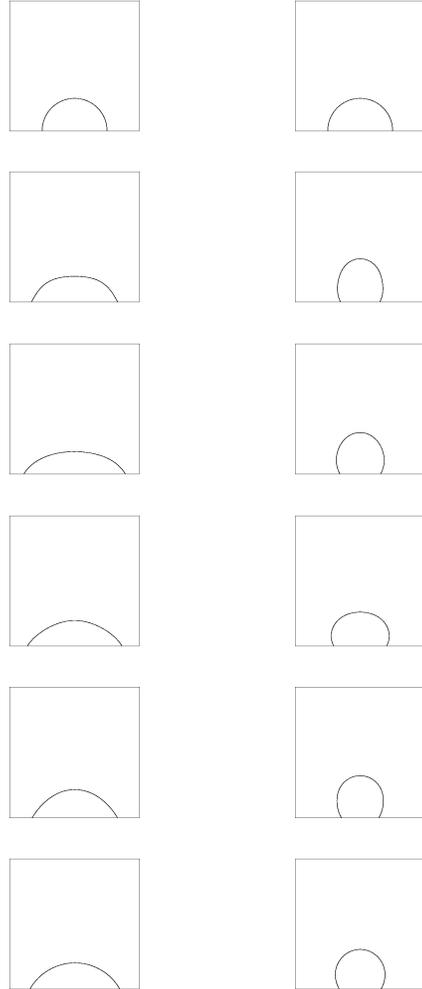


Figure 5. Snapshots of 2D droplet shapes, for $\theta = 60^\circ$ (left) and $\theta = 120^\circ$ (right) applied to an initial semi-circle, at non-dimensional times $\tau = t\sigma/(\mu D)$ of 0, 10, 20, 30, 40 and 50 (from top to bottom).

ter droplet of 2 mm diameter falls downward at 1 m/sec onto an inclined surface at an angle $\alpha = 45^\circ$. The corresponding Reynolds and Weber numbers are: $Re = 2000$ and $We = 27$ respectively.

A general correlation for contact angle as a function of contact line velocity is not available in the literature. For these results, we implemented three simple contact angle models (more advanced models can be easily implemented once established). These simple models rely on values of the contact angle at rapidly advancing (θ_a) and receding (θ_r) contact lines; the three models differ in the way in which these two values are bridged at small contact line velocities; see figure 9. Figures 10 and 11 depict a comparison of the simulation results obtained with

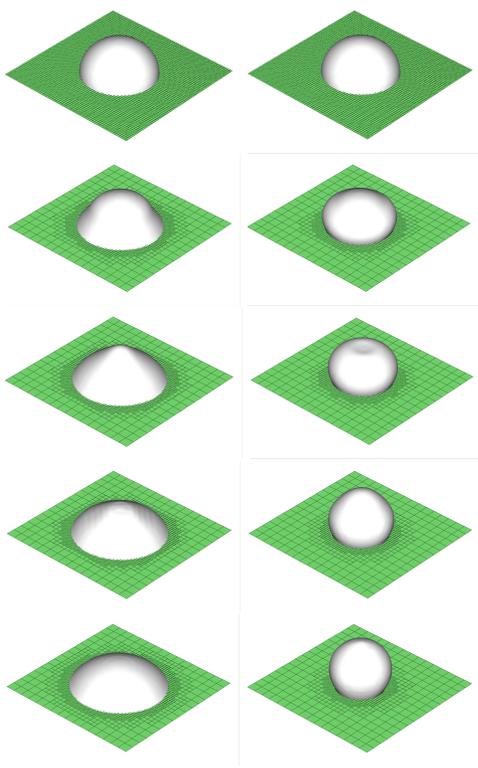


Figure 6. Snapshots of droplet shapes, for $\theta = 60^\circ$ (left) and $\theta = 120^\circ$ (right) applied to an initial hemisphere, at non-dimensional times τ of 0, 2.5, 5, 7.5, and 15 (from top to bottom).



Figure 7. Contact lines at steady state for $\theta = 60^\circ$ (left) and $\theta = 120^\circ$ (right), corresponding to the results of figure 6. Normals and curvatures are calculated using HF methodology.



Figure 8. Contact lines at steady state for $\theta = 60^\circ$ (left) and $\theta = 120^\circ$ (right). Normals and curvatures are calculated using older algorithms [12, 11].

these three different models. The difference is obviously most significant at later stages of impact.

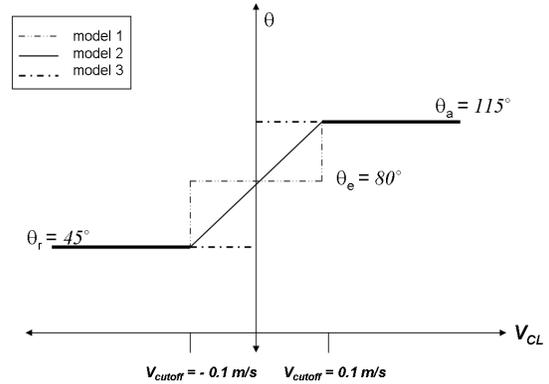


Figure 9. Models of dynamic contact angle versus the contact line velocity V_{CL} .

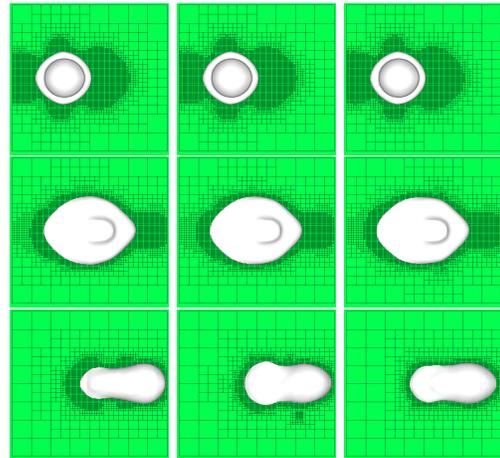


Figure 10. Comparison of normal views of the impact of a 2 mm diameter water droplet at 1 m/sec onto an 45° incline, based on three different models of dynamic contact angle, at times 1, 5, and 10 msec (from top to bottom). From left to right, models 1, 2, and 3 (figure 9).

Conclusion

An adaptive 3D model of droplet impact has been developed for the accurate calculation of surface tension forces in a VOF-based model. The model accounts for the dynamics of contact lines in a way that is consistent with the HF approach to modelling surface tension. The model was used to simulate inclined impacts for three simple models of dynamic contact angle versus contact line velocity and the simulation results were presented. The results demonstrated that a small difference in the

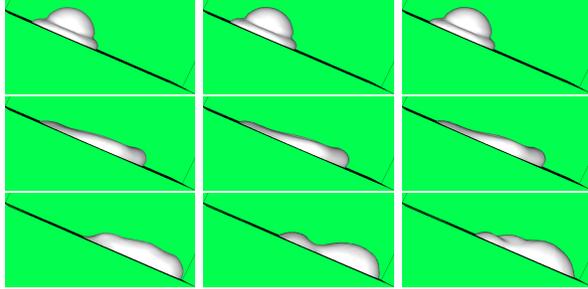


Figure 11. Comparison of side views of the impact of a 2 mm diameter water droplet at 1 m/sec onto an 45° incline, based on three different models of dynamic contact angle, at times 1, 5, and 10 msec (from top to bottom). From left to right, models 1, 2, and 3 (figure 9).

contact angle model can lead to different fluid deformations during droplet impact.

References

- [1] Mark Sussman. *Journal of Computational Physics*, 187:110–136, 2003.
- [2] M. Francois, S.J. Cummins, E.D. Dendy, D.B. Kothe, J.M. Sicilian, and M.W. Williams. A balanced-force algorithm for continuous and sharp interfacial surface tension models within a volume tracking framework. Technical Report LA-UR-05-0674, Los Alamos Scientific Laboratory, 2005.
- [3] S.J. Cummins M. Francois and D.B. Kothe. *Computers and Structures*, 83:425–434, 2005.
- [4] M. Renardy, Y. Renardy, and J. Li. *Journal of Computational Physics*, 171:243–263, 2001.
- [5] H. Liu, S. Krishnan, S. Marells, and H.S. Udaykumar. *Journal of Computational Physics*, 210:32–54, 2005.
- [6] T.A. Baer, R.A. Cairncross, P.R. Schunk, R.R. Rao, and P.A. Sackinger. *International Journal for Numerical Methods in Fluids*, 33:405–427, 2000.
- [7] Š. Šikalo, H.D. Wilhelm, I.V. Roisman, S. Jakirli, and C. Tropea. *Physics of Fluids*, 17:062103, 2005.
- [8] S. Afkhami and M. Bussmann. *International Journal of Numerical Analysis and Modeling*, submitted.
- [9] S. Popinet. *The Gerris Flow Solver*. <http://gfs.sourceforge.net>.
- [10] S. Popinet. *Journal of Computational Physics*, 190:572–600, 2003.
- [11] J.U. Brackbill, D.B. Kothe, and C. Zemach. *Journal of Computational Physics*, 100:335–354, 1992.
- [12] D. Gueyffier, A. Nadim, J. Li, R. Scardovelli, and S. Zaleski. *Journal of Computational Physics*, 152:423–456, 1998.
- [13] Y. Renardy and M. Renardy. *Journal of Computational Physics*, 183:400–421, 2002.
- [14] M.M. Francois, D.B. Kothe, E.D. Dendy, J.M. Sicilian, and M.W. Williams. Balanced force implementation of the continuum surface tension force method into a pressure correction algorithm. Technical Report LA-UR-05-0674, Los Alamos Scientific Laboratory, 2005.