A Numerical Method for Interface Tracking

Meizhong Dai, Haoshu Wang, J. Blair Perot, and David P. Schmidt*
Department of Mechanical and Industrial Engineering
University of Massachusetts, Amherst
Amherst, MA 01003

Abstract

Direct interface tracking computes spray behavior based only on first principles. It is an advanced form of direct numerical simulation, but with the emphasis shifted from resolving details of turbulence to details of multiphase flow. The moving interface requires special treatment and advanced numerical methods. A code which is capable of accurate resolution of three-dimensional free-surface deformation has been constructed. The Navier-Stokes equations for the liquid phase are solved on a deforming unstructured mesh. This technique tracks the boundary precisely, similar to marker-and-cell methods. However the adaptive mesh follows the interface. Furthermore, this new method avoids the surface reconstruction required in volume of fluid methods. A numerical method for calculating free surface distortion has been described, examined, and the surface tension force is shown to be second-order accurate in space. By locally fitting the free surface to a parabola when evaluating curvature, problems with numerical noise in the solution are avoided. A new time step criterion is introduced based on free surface numerical stability. The results for a deforming drop and collapsing ligament are presented. The code is validated by comparing to the theoretical period for drop deformation.

Introduction

The complexities of spray behavior are often very difficult to observe directly. Sprays usually evolve over small time and space scales. Furthermore, high number densities of droplets can impede optical access. For understanding basic spray physics, simulation based on first principles may be helpful. In a model that relies only on the Navier-Stokes equations, the simulation can generate trustworthy results that would provide complete detail about droplet behavior.

There are significant difficulties inherent in multiphase flow calculations. Interface tracking codes require several hundred percent more CPU time than single-phase codes, making such simulations expensive. Also, it has proven difficult to attain reasonable accuracy in interface tracking [1]. These issues have hindered the development of high-performance three-dimensional interface tracking codes. Helenbrook has developed a very accurate spectral method, but this code is currently limited to two-dimensional, axisymmetric droplet simulations [2]. Cristini et al. have a three-dimensional multiphase flow code that is based on the boundary element method [3]. There have been only a few major attempts to apply interface tracking to primary atomization [4,5]. The current work attempts to advance the numerical methods so that a wide range of drop sizes can be resolved.

* Correspondence to: David P. Schmidt, Department of Mechanical & Industrial Engineering, University of Massachusetts, Amherst.
Email: dschmidt@cobia.ecs.umass.edu.

Governing Equations and Numerical Scheme

The numerical method used in this work is based on a stream-function formulation of the Navier-Stokes equations. The complete Navier-Stokes equations were solved in two and three dimensions on a deforming unstructured mesh. The basic equations were solved for a deforming, moving, control volume, avoiding the interpolation errors inherent in global remeshing. The method has some common features with Arbitrary Lagrangian-Eulerian methods. However the current approach is not a fractional step method.

Nallapati and Perot[6] have provided some information about the numerical scheme that will be used. The method uses an unstructured mesh, which allows maximum flexibility in matching mesh cells with the boundary surfaces. For a two-dimensional mesh, each triangle will be a control volume. In three dimensions, the basic control volume is a tetrahedra. The basic parameter and real unknown is the stream function, which is a scalar in 2 dimensions and a vector in 3 dimensions.

Momentum equations can be developed after the velocities are obtained. The procedure is complex since the mesh has to be moved in order to track the free surface. Beginning with the Reynolds transport theorem for incompressible flow, we have:

$$\frac{d}{dt}\int_{CV}\rho\mathbf{u}dv+\int_{CV}\nabla\cdot\rho\mathbf{u}(\mathbf{u}-\mathbf{v})dv=\int_{CV}\rho\mathbf{f}dv-\int_{CV}\nabla pdv+\int_{CV}\mu\nabla^2\mathbf{u}dv \quad (1)$$

where **f** is the body force, **v** is the mesh velocity, and CV is the control volume. Assuming the control volume is small enough and the velocity **u** is linear within each of them, the integral form can be changed to:

$$\frac{1}{\Delta t}\left[(\rho \mathbf{u}\Delta v)^{n+1} - (\rho \mathbf{u}\Delta v)^{n}\right] + (\nabla \cdot \rho \mathbf{u}(\mathbf{u}-\mathbf{v})\Delta v)^{n} = (\rho \mathbf{f}\nabla v)^{n} - (\nabla p \Delta v)^{n+1} + (\mu \nabla^2 \mathbf{u}\Delta v)^{n+1/2} \quad (2)$$

$$\frac{1}{\Delta t}\left[(\rho \mathbf{u})^{n+1} - (\rho \mathbf{u})^{n}\frac{\Delta v^{n}}{\Delta v^{n+1}}\right] + (\nabla \cdot \rho \mathbf{u}(\mathbf{u}-\mathbf{v}))^{n}\frac{\Delta v^{n}}{\Delta v^{n+1}} = (\rho \mathbf{f})^{n}\frac{\Delta v^{n}}{\Delta v^{n+1}} - \nabla p^{n+1} + (\mu \nabla^2 \mathbf{u})^{n+1/2}\frac{\Delta v^{n}}{\Delta v^{n+1}} \quad (3)$$

where the parameters **u**, p, and ρ are defined at the center of each control volume, and n and n+1 represent different time steps. The value of the cell volume at the n+1 time step Δv^{n+1} is predicted from velocity at the n time step; therefore this is still a linear equation. Note that an implicit method, Crank-Nicolson differencing, is applied to develop implicit schemes for the diffusion terms. The divergence operators in the convection and diffusion terms can be evaluated using Gauss' Divergence Theorem:

$$[\nabla \cdot \rho \mathbf{u}(\mathbf{u}-\mathbf{v})] = \frac{1}{\Delta v}\sum_{\text{cell faces}} \rho \mathbf{u}(\mathbf{u}-\mathbf{v})\cdot \hat{\mathbf{n}} A_f \quad (4)$$

$$[\mu \nabla \cdot (\nabla \mathbf{u})] = \frac{1}{\Delta v}\sum_{\text{cell faces}} \mu \nabla \mathbf{u}\cdot \hat{\mathbf{n}} A_f \quad (5)$$

In the above equations the gradient term in each cell can be calculated by least squares data fitting, if the center of gravity is used as the cell center where the velocity is positioned. As a final step, the velocity u is expressed in terms of the stream function component $\mathbf{s}\cdot\mathbf{t}$ defined on each cell edge. Then Eqn.3 is integrated on a closed path around the edges, producing the discrete equivalent of a curl operation. These equations are solved for the moving, deforming cells.

Surface Tension Evaluation

A constant pressure boundary condition is used for the free surface, and surface tension is treated as an additional term in the boundary cells' pressure.

$$p = p_0 + \sigma\left(\frac{1}{R_1} + \frac{1}{R_2}\right) \quad (6)$$

Here the problem is how to evaluate the boundary curvature. The values of R_1 and R_2 are the radii of curvature in two orthogonal directions along the interface. The sum of the reciprocals of R_1 and R_2 will be denoted as $1/R_{effective}$, so that the equations may be generally represented for two or three dimensions. In this work curvature in three dimensions is calculated on each boundary face by a new surface fitting method. Though Zinchenko et al. [7] first used a parabolic fit for surface tension, they calculated curvature at nodes rather than faces. Zinchenko et al.'s algorithm was necessarily very complicated. The current implementation is much simpler, because the force is defined at the face center. In three dimensions, the local surface of a cell face is fitted by:

$$z(x,y) = f(x,y) = a_6 x^2 + a_5 y^2 + a_4 xy + a_3 x + a_2 y + a_1 \quad (7)$$

and coefficients a_1 through a_6 are obtained from coordinates of six nodes. Three nodes are from the triangular face itself and three nodes are from the three neighboring faces, producing a six by six linear system. This dependence is ideal, because it produces a template that is centered around the face of interest. Since the curvature is independent of the coordinate system, one can also apply transformations to set up a local coordinate system whose z axis coincides with the face normal vector. This step is necessary to avoid an ill conditioned matrix. Then the local curvature is:

$$\frac{1}{R_{effective}} = -\frac{1}{2}\left[(1+f_y^2)f_{xx} - 2f_x f_y f_{xy} + (1+f_x^2)f_{yy}\right] / \left[1+f_x^2+f_y^2\right]^{3/2} \quad (8)$$

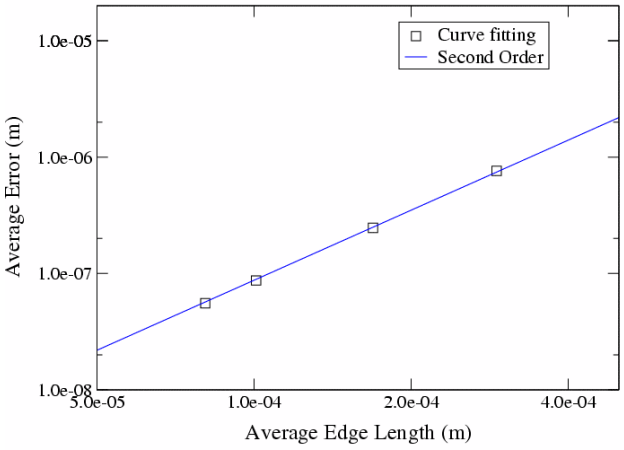

Figure 1. Numerical Curvature Scheme: The surface tension evaluation method is checked for convergence and accuracy.

Fig. 1 shows the numerical results of surface tension calculation for a perfect sphere of diameter 2*mm*, using the parabolic fitting scheme. It shows that the curve fitting method has roughly second order

precision, which is expected from the fact that a quadratic function is used to fit the surface.

Mesh Moving Methods

This work uses the adaptive mesh approach to track the free surface. The mesh nodes on the free surface are moved in a Lagrangian manner, and when the mesh evolution produces some poorly shaped or distorted cells, mesh improvement methods are used to adjust the interior nodes to achieve high mesh quality. Here Delaunay triangulation is used as the quality measure. A Delaunay triangulation may be characterized in several ways. One way is by the easily shown property that the circumcircle of any triangle (circumsphere of any tetrahedron in three dimensions) contains no nodes other than the ones on its boundary. Two kinds of methods are used to transform the non-Delaunay cells: mesh smoothing and mesh flipping.

In a two-dimensional mesh, mesh smoothing simply treats each cell face as if it were under tension proportional to its length. This is analogous to a spring, except that this "spring" has zero equilibrium length. The resultant forces at each node are computed from the tensions to adjust its position. Mesh flipping turns out to be more complicated than smoothing. The two-dimensional case is relatively simple. When two neighboring cells lose their Delaunay property, one can just flip their common face to the other two nodes as shown in Fig. 2:

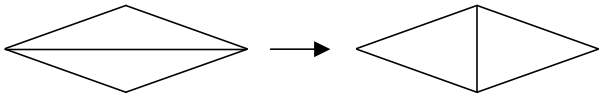

Figure 2. Mesh flipping in two dimensions.

This technique has proved effective in two dimensions, but it is not straightforward in three dimensions. Unfortunately, in three dimensions, the Delaunay criteria is not always sufficient to guarantee good cells. Flipping occasionally produces cells that are very flat, with their four vertices nearly coplanar. These cells, called “sliver cells” are a known defect of using the Delaunay criteria [8,9]. For three dimensional flipping the minimum dihedral angle is the criterion in the current work. Flipping proceeds only where it increases the minimum dihedral angle present in a set of cells.

Stability

In addition to the CFL limit, a criterion of free surface numerical stability is required. The mixed character of the flow at the surface makes this difficult. To derive a stability relation, the momentum normal to the interface was considered. For simplicity, the viscous and convective acceleration terms were dropped. Viscosity is treated implicitly in this work, and the convective terms would give rise to the CFL limit. The simplified situation is potential flow driven by surface tension. This is the well-known case of the propagation of small waves in deep water with negligible gravity, also known as capillary waves. Sinusoidal capillary waves travel at a speed, c, that depends on the wavelength of the disturbance λ [10].

$$c = \sqrt{\frac{2\pi\sigma}{\rho\lambda}} \tag{9}$$

The value of λ would be approximately the same as the mesh resolution, assuming that the instability manifests itself at the highest resolvable frequency. This assumption results in propagation with a wave speed of $\sqrt{\frac{2\pi\sigma}{\rho\Delta x}}$. Thus by the CFL criterion, the surface will be stable for an appropriate explicit scheme when:

$$\Delta t < C\sqrt{\frac{\rho\Delta x^3}{2\pi\sigma}} \tag{10}$$

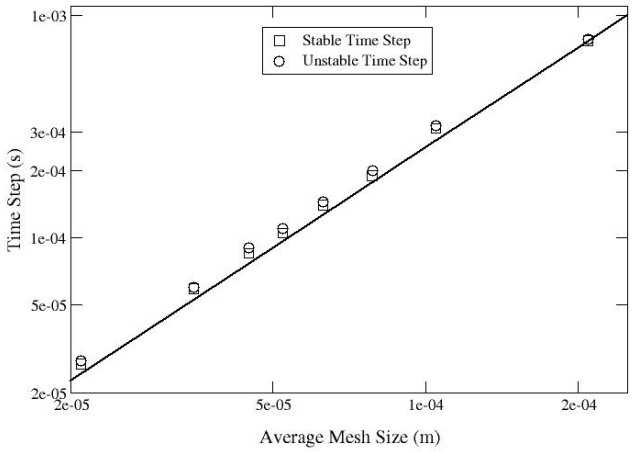

Figure 3. Evaluation of the largest stable time step.The analytical result is Eqn. (10) with the constant set to the theoretical value of 2.0. The liquid properties are varied and the mesh resolution is held constant.

The constant *C* in Eqn. (10) is of order unity and depends on the numerical method. Equation (10) provides a way of predicting a constraint on the time step in free surface calculations. A test of this stability criterion with a third-order Runge-Kutta scheme is shown in Fig. 3. For purely hyperbolic problems, this discretization method has a theoretical value of 2.0 for *C*.

σ / ρ (m^3/s^2)	10-8	10-7	10-6	10-5	7.56x10-5
T (s): Theoretical value	0.702	0.222	0.0702	0.0222	0.00808
T (s): Test value	0.715	0.228	0.0705	0.0223	0.00800
Percentage Error	1.8519%	2.7027%	0.4274%	0.4505%	-0.9901%

Table 1. Liquid drop oscillation period value. Initial parameters are: r_0=0.001m, ν =1.781x10^{-6} m^2/s. A fine mesh of 4,500 cells was used.

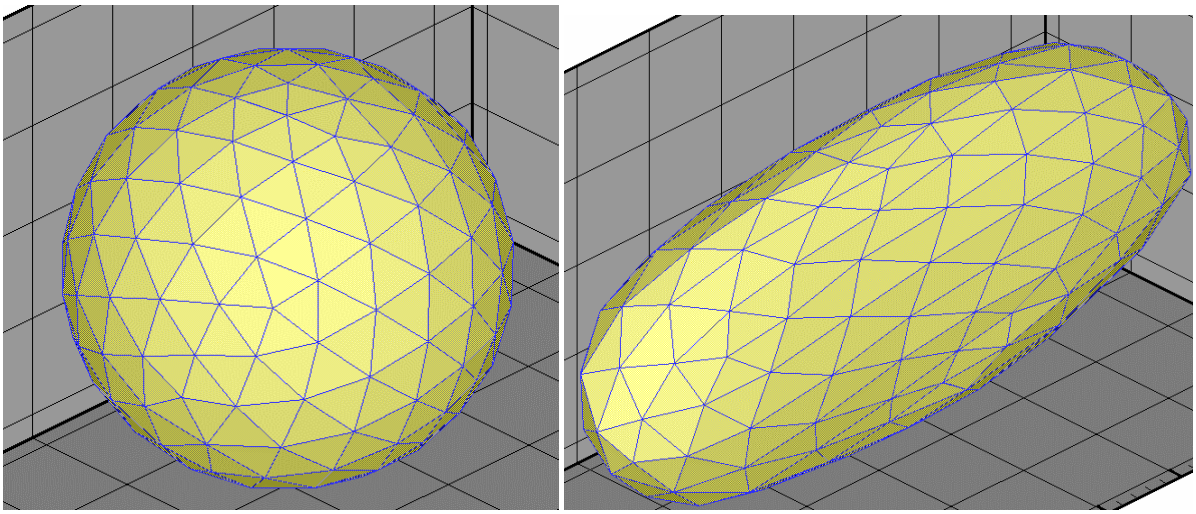

Figure 4. Liquid drop oscillation subject to a large disturbance. A coarse mesh of 1,300 cells was used.

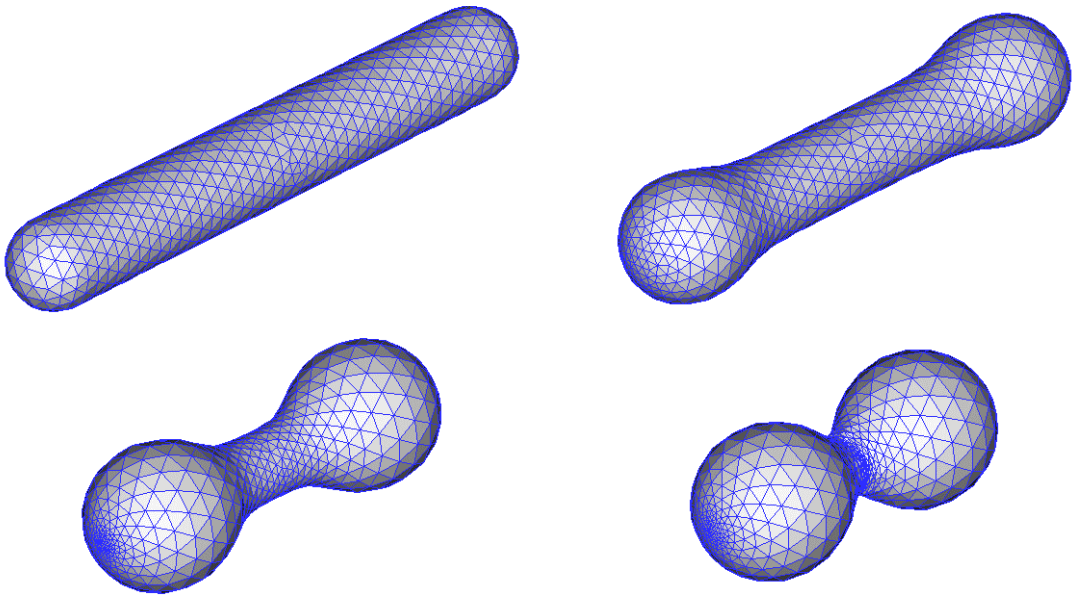

Figure 5. A ligament collapsing under the effect of surface tension.

Results

As a three-dimensional test case, an oscillating droplet was calculated using 4,500 cells. The initial stream function is:

$$\boldsymbol{\psi} = 1.5yz\mathbf{i} - 0.5xz\mathbf{j} + 0.5yx\mathbf{k} \qquad (11)$$

And the initial velocity field is:

$$v_x = x \qquad (12)$$

$$v_y = y \qquad (13)$$

$$v_z = -2z \qquad (14)$$

According to Lamb [11] the drop oscillation period for a small, mode 2 perturbation is:

$$T = 2\pi \frac{1}{\sqrt{n(n-1)(n+2)\frac{\sigma}{\rho r_0^3}}} = 2\pi \frac{1}{\sqrt{8\frac{\sigma}{\rho r_0^3}}} \qquad (15)$$

Table 1 shows the numerical results of the liquid drop oscillation calculation. They show good accuracy and help to confirm the validity of the current approach. For larger deformation, mesh smoothing and flipping are critical. The results for an oscillating droplet with a large perturbation are shown in Fig. 4. The droplet on the left is in its initial state with its initial mesh. The droplet is given an initial velocity field that is similar to the previous accuracy test, except that the magnitude is greater. The right side of Fig. 4 shows the deformed drop and the corresponding shape of the mesh.

This method is not unique to droplets. The value of this code is that it can be applied to any shape of liquid interface. Thus one can study droplets, ligaments, and films. The results shown in Fig. 5 are for a collapsing ligament with a small initial velocity. For this case, a surface-mesh redistribution algorithm was used to maintain adequate mesh resolution as the ligament began to pinch.

The equilibrium state of a ligament is various numbers of droplets, depending on the initial state. In this case, the middle of the ligament pinches and the ends swell. The ligament is on its way to forming two droplets. However, the code is not yet capable of handling the separation of the liquid into multiple parts. This capability is planned for the near future.

References

1. Rider, W.J., Kothe, D.B., Mosso, S.J., and Cerutti, J.H., "Accurate Solution Algorithms for Incompressible Multiphase Flows," *AIAA-95-0699*.

2. Helenbrook, B.T., "Numerical Studies of Droplet Deformation and Break-up," *ILASS Americas 14th Annual Conference on Liquid Atomization and Spray System*, Dearborn, MI, May 2001, pp.(123-128).

3. Cristini, V., Blawzdziewicz, J., and Loewenberg, M., "An Adaptive Mesh Algorithm for Evolving Surfaces: Simulations of Drop Breakup and Coalescence," *J Comp. Phys* 168(2): 445-463 APR 10 2001.

4. Lafaurie, B., Mantel, T., and Zaleski, S., "Direct Numerical Simulation of Liquid Jet Atomization," *Third International Conference on Multiphase Flow*, ICMF'98, Lyon, France, June 8-12,1998.

5. Tauber, W., and Trygvasson, G., "Primary Atomization of a Jet," *Proc. of ASME FED Summer Meeting*, Boston, 2000.

6. Nallapati, R., and Perot, J.B., "Numerical Simulation of Free-Surface Flows Using a Moving Unstructured Mesh," *Proceedings of ASME FEDSM'00*.

7. Zinchenko, A.Z., Rother, M.A., and Davis, R.H., "A Novel Boundary-Integral Algorithm for Viscous Interaction of Deformable Drops," *Phys. Fluids*, 9 (6), 1997.

8. Bern, M. and Eppstein, D., "Mesh Generation and Optimal Triangulation," Tech. Rep. CSL-92-1, Xerox PARC, 1992.

9. Du, D.Z., Hwang, F.K., eds., *Computing in Euclidean Geometry*, World Scientific, 1992, pp. 23-90.

10. Currie, I.G., *Fundamental Mechanics of Fluids*, 1974, McGraw-Hill, New York, pp. 186-188.

11. Lamb, H., *Hydrodynamics*, 1945, 6th edition, Dover Publications, Inc.