ICLASS 2009, 11th Triennial International Conference on Liquid Atomization and Spray Systems, Vail, Colorado, USA, July 2009

A Particle-Based Method Coupled with Finite Volume Solver for Interfacial Flows

Sourabh V. Apte*
School of Mechanical, Industrial, and Manufacturing Engineering, Oregon State University,
204 Rogers Hall, Corvallis, OR 97331

Abstract

A hybrid Lagrangian-Eulerian (hLE) scheme, combining a particle-based, mesh-free technique with a finite-volume flow solver, is developed for direct simulations of two-phase flows. The approach uses marker points around the interface and advects the signed distance to the interface in a Lagrangian frame. The kernel–based derivative calculations typical of particle methods are used to extract the interface normal and curvature from unordered marker points. Connectivity between the marker points is not necessary. The fluid flow equations are solved on a background, fixed mesh using a co-located grid finite volume solver together with balanced force algorithm (Francois *et al.* JCP, 2006, Herrmann JCP, 2007) for surface tension force. The numerical scheme is applied to standard test cases to show promising results: (i) parasitic currents in a stationary spherical drop, (ii) small amplitude damped surface waves, (iii) capillary waves on droplet surface, (iv) Rayleigh-Taylor instability, and (v) gravity-driven bubble/droplet in a stationary fluid.

Introduction

Particle-based Lagrangian, *mesh-free* algorithms such as moving particle-methods [1], vortex-in cell methods [2, 3], and smoothed-particle hydrodynamics [4] have been popular for large-scale free-surface flows. These pure Lagrangian methods are promising as they avoid enormous memory requirements for a three-dimensional mesh. These methods automatically provide adaptive resolution in the high-curvature region [2] and have been applied successfully to many two-phase flow problems [5, 6, 7]. However, they exhibit other difficulties such as high cost of finding nearest neighbors in the zone of influence of a Lagrangian point, true enforcement of continuity (or incompressibility) conditions, and problems associated with accurate one-sided interpolations near boundaries [2].

In the present work, we develop a hybrid approach, wherein the Lagrangian nature of the interface motion is captured by particle-based method, and the fluid flow is computed using a finite-volume solver using variable density, single-fluid model. The basic idea is to merge the locally 'adaptive' mesh-free particle-based methods with the relative 'ease' of Eulerian finite-volume formulation in order to inherit the advantages offered by individual approaches. The interface between two fluids is represented and tracked using Lagrangian points or fictitious particles [7]. Unlike particle level set method [8] or the semi-Lagrangian methods [9], in the present approach the interface is represented by Lagrangian points (LPs) (or particles[1]) that are advanced in a Lagrangian frame. The motion of the interface is determined by a velocity field (interpolated to the particle locations) obtained by solving the Navier-Stokes equations on a *fixed* background mesh in an Eulerian frame. The interface location, once determined, identifies the region of the mesh to apply *jump-conditions* in fluid properties. In this sense, it is in the realm of Arbitrary Lagrangian-Eulerian (ALE) schemes, wherein the computational grid deforms to conform to the shape of the dispersed phase. The potential advantage of the present hybrid method is that the background mesh could be of any kind: *structured, body-fitted, or arbitrary shaped unstructured* (hex, pyramids, tetrahedrons, prisms) and may be *stationary or changing in time (adaptive refinement)*. Here, we use a co-located grid, incompressible flow solver based on the energy conserving finite-volume algorithm [10].

The Lagrangian points (LPs) in our interface calculations, are particles distributed in a narrow band around the interface [11]. These LPs are initially uniformly spaced and carry information such as the signed distance to the interface (SDF) along the characteristic paths. Variations in flow velocities leads to an irregular distribution of the initially uniform LPs. Regularization of the particles are performed by mapping the particles on a uniformly spaced lattice [7]. Values for particle properties at new LP locations are obtained through kernel mollification as done in Smoothed Particle Hydrodynamics [4] and remeshed-SPH [6]. The novelty in our approach is that this mesh-free interface representation is integrated with a finite-volume solver where the governing equations for flow

*Corresponding Author: sva@engr.orst.edu

[1] In this paper, the term 'particles' means Lagrangian points (LPs) that are used to represent the interface.

evolution are solved. The Lagrangian points provide sub-grid resolution and in this respect the method is similar to the Refined Level Set Grid (RLSG) approach [12, 13]. However, here the LPs move in space with the flow velocity and different discretizations are necessary and we use high-order schemes based on mollification kernels.

Hybrid Lagrangian-Eulerian (hLE) Scheme

Following Hieber & Koumoutsakos [7], the interface between two fluids is represented using uniformly spaced Lagrangian points (LPs) or fictitious particles in a narrow band around the interface. Each LP is associated with position $\mathbf{x}_p$, velocity $\mathbf{u}_p$, volume $\mathcal{V}_p$ and a scalar function Φ_p which represents the signed distance to the interface. The average spacing (h between the uniformly spaced LPs is related to the volume $\mathcal{V}_p$. In this work, we use cubic elements ($h = \mathcal{V}_p^{1/3}$). As the LPs move, they carry the SDF value along the characteristic paths and implicitly represent the motion of the interface. The evolution of the interface is calculated by solving level set equations in the Lagrangian form:

$$\frac{D\Phi_p}{Dt} = 0; \;\; \frac{D\mathcal{V}_p}{Dt} = \langle \nabla \cdot \mathbf{u} \rangle_p \mathcal{V}_p; \;\; \frac{D\mathbf{x}_p}{Dt} = \mathbf{u}_p, \tag{1}$$

where p denotes the Lagrangian point or particle. For incompressible fluids, the velocity field is divergence free and theoretically, the change in volume of the LPs ($D\mathcal{V}_p/Dt$) is zero.

As is done in Smoothed Particle Hydrodynamics (SPH) and mesh-free methods, smoothed approximation of the level set function and its derivatives can be obtained by using a mollification operator with LPs as quadrature points. The localized mollification kernel ξ_ϵ generates a smooth continuous approximation of Φ around the particle at location $\mathbf{x}_p$ using SDF of other particles at locations $\mathbf{x}_q$: $\Phi_q = \sum_{p=1}^{N} \mathcal{V}_p \Phi_p \xi_\epsilon(\mathbf{x}_q - \mathbf{x}_p)$ where $\Phi_q = \Phi(\mathbf{x}_q)$ and $\sum_{p=1}^{N} \xi_\epsilon \mathcal{V}_p = 1$. Different mollification kernels have been proposed such as quartic spline and M_n splines [4]. In this paper, we use the quartic spline function. The surface normal and curvature calculations require derivatives of the scalar function Φ on the particles. These are approximated in a conservative form by using derivatives of the quartic spline kernels [14]. These mollification operations require information about the neighboring particles inside the region of influence, and a Verlet list and linked lists are used to optimize storage and location of the neighbors.

Once the location of the LPs and the associated Φ_p values are obtained, a color function $\Psi(\mathbf{x})$ can be constructed. Following the definition of color function, finding Ψ on the LPs is straightforward: $\Psi = 0$ when $\Phi \geq 0$ and $\Psi = 1$ for $\Phi < 0$. Then the color function field can also be obtained on the background computational mesh by interpolating Ψ from the LPs. In order to obtain a smooth function, the M_4' kernel interpolation in three-dimensions is used [7]. Once the color function Ψ is obtained at a control volume (cv), the density and viscosity are given as:

$$\rho_{cv} = \rho_1 + (\rho_2 - \rho_1)\Psi_{cv}; \qquad \mu_{cv} = \mu_1 + (\mu_2 - \mu_1)\Psi_{cv} \tag{2}$$

Then the flow field is computed on a background mesh (which could be structured or unstructured) by solving the Navier-Stokes equations for the two-fluid system:

$$\nabla \cdot \mathbf{u} = 0 \tag{3}$$

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{1}{\rho}\nabla p + \frac{1}{\rho}\nabla \cdot \left(\mu(\nabla \mathbf{u} + \nabla^T \mathbf{u})\right) + \mathbf{g} + \frac{1}{\rho}\mathbf{F}_\sigma \tag{4}$$

where $\mathbf{u}$ is velocity vector of fluid, p is pressure, ρ and μ are fluid density and viscosity (uniform inside each fluid), $\mathbf{g}$ body force, and $\mathbf{F}_\sigma$ is the surface tension force which is non-zero only at the interface location ($\Phi = 0$). Following Brackbill *et al.* [15], the surface tension force is modeled as a continuum surface force (CSF). A common issue with numerical simulations involving surface tension force, is the development of *spurious currents* (unphysical velocity field) [16] due to inaccuracies in the discrete approximations to the surface-tension forces (equation 4). In order to obtain a consistent coupling of the surface tension force with the pressure gradient forces in a finite-volume approach, Francois *et al.* [17] indicated that the surface tension force must be evaluated at the faces of the control volumes as: $\mathbf{F}_{\sigma,f}^{CSF} = \sigma \kappa_f \left(\nabla \Psi\right)_f$ where the subscript f stands for the face of the control volume, σ is the surface tension coefficient (assumed constant in the present work), κ is the curvature, $\hat{\mathbf{n}}$ the interface normal. The surface tension force at the cv-centers can be obtained through reconstruction from the faces of each cv.

To compute the surface tension force, accurate estimation of the curvature of the interface is necessary. Herrmann [12] developed a procedure to compute the curvature accurately in the level-set framework. Here we follow a similar procedure for curvature evaluations. First, the curvature and normals ($\kappa = \nabla \cdot \frac{\nabla \Phi}{|\nabla \Phi|}; \mathbf{n} = \frac{\nabla \Phi}{|\nabla \Phi|}$)

are evaluated at the LPs close to the interface ($|\Phi| \leq 2\Delta_{LP}$, where Δ_{LP} is the spacing between the LPs. The gradients in the curvature and surface normal computations are evaluated from derivatives of the mollification kernels. For each of these LPs (with $|\Phi| \leq 2\Delta_{LP}$) , a base point on the interface is obtained by projecting normals onto the interface. Curvature on the base point is evaluated by using curvature values on LPs in its neighborhood using the M_4'-kernel based interpolation. Once curvatures on all interface points are evaluated, these values are *assigned* to the corresponding LPs from which these interface points were obtained. Curvature at the background control volume *cv* is then computed by simply adding the curvatures of LPs that lie inside the control volume. Curvature at the faces of the control volume are evaluated by arithmetic average of the two control volumes associated with the face. Here, the average is taken only if the both *cv*s contain the interface, i.e. color function $0 < \Psi_{cv} < 1$, else κ_f is assigned the value of κ_{cv} containing the interface.

Numerical Algorithm

The governing equations are solved using a co-located grid finite-volume algorithm [10]. Accordingly, all variables are stored at the control volume (*cv*) centers with the exception of a face-normal velocity, located at the face centers, and used to enforce the divergence-free constraint. The variables are staggered in time so that they are located most conveniently for the time advancement scheme. Knowing the velocity field at the *cv* centers, it is interpolated to the particle locations using the quartic spline kernel. The LPs are then advanced by solving the equations 1 using a third-order Runge-Kutta scheme. The LPs are re-located and duly transferred to different processors.

Because of this advection, the original uniform map of LPs may get distorted, and they are reconfigured (if a threshold distortion is exceeded) to a uniform Cartesian lattice. This process of remeshing or reconfiguration was introduced in Remeshed-Smoothed Particle Hydrodynamics [2]. Remeshing removes any unphysical kinks in the interface and gives the ‘entropy-satisfying viscous solution’ [7]. It also eliminates unnecessary points away from the interface. For remeshing, we use the M_4' kernel to obtain the interpolated SDF values. Although the reconfiguration procedure provides the entropy solution, it does not guarantee that Φ remains a signed-distance to the interface. In this work, reinitialization is implemented according to the method suggested by Sussman et al. [18, 19] in which the reinitialization equation is solved on uniformly spaced LPs: $\frac{\partial\Phi}{\partial\tau} = \text{sign}(\Phi_0)(1 - |\nabla\Phi|)$ where $\Phi(x, 0) = \Phi_0$ and $\text{sign}(\Phi_0) \equiv 2(H_\epsilon(\Phi) - 1/2)$ and $H_\epsilon(\Phi)$ is the Heaviside function. We apply redistancing in a two-layer narrow band around the interface and using the procedure described in Gomez et al. [13].

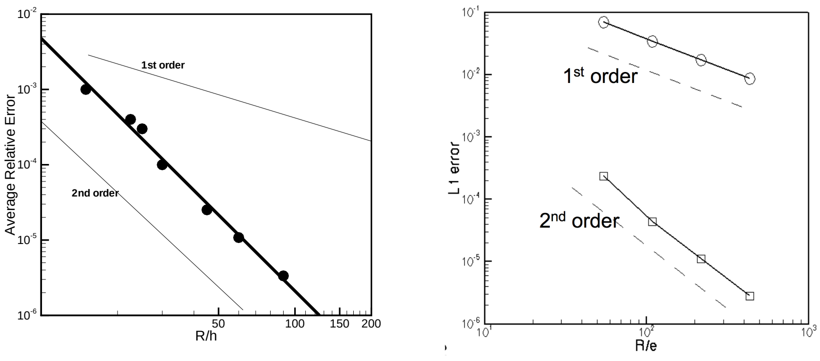
Once the new LP locations are obtained, interface properties such as curvature, surface normal are obtained by using conservative formulation based on the derivatives of the kernel functions. The curvatures at the cv-centers and a color function are obtained as described in the previous section. The two-phase flow properties are then evaluated at the *cv* centers, and the Navier-Stokes equations are solved using the balanced force algorithm [17, 12] that enforces discrete balance of surface tension force and pressure gradient in the absence of any flow and other external forces.

Results

In this section, some numerical examples of standard test cases using the hLE scheme are presented. First, the accuracy of the pure Lagrangian advection approach is evaluated by performing standard test cases such as the Zalesak disc rotation and the evolution of a circular interface in a deformation field [20]. These showed comparable results with published data [7] and are not shown here.

Estimation of Surface Normal and Curvature

The accuracy of the surface normal and curvature calculation by using the procedure described before is tested on a circular interface. The Lagrangian points (LPs) are uniformly distributed in a narrow band around the interface and initialized by *exact* signed distance function. The surface normals and curvatures are first calculated on the LPs. Only those LPs are considered where $|\Phi| \leq 2\Delta_{LP}$, where Δ_{LP} is the LP-spacing. The average relative error in surface normal calculation is shown in Figure 1 indicating second order convergence. For all these LPs, corresponding points on the interface are calculated using the normals and signed distance function (Φ). The interface-projected curvature values at the interface points are evaluated using the M_4'-kernel based interpolation from the neighboring LP values. These curvatures are then compared with the exact curvature $\kappa_{exact} = 1/R$ for a two-dimensional interface. The corresponding L_1-errors are plotted at different LP-resolutions in Figure 1b, showing second order convergence similar to Herrmann [12]. Error calculations based on the κ values at the LPs are also plotted, showing only a first-order convergence, indicating the importance of the interface projected curvature calculation.

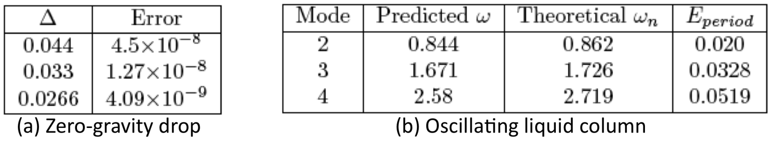


(a) Normal (b) Curvature

Figure 1: Error versus grid spacing for circular interface: (a) average relative error in surface normal, (b) absolute error in curvature–circles indicate error at LP location, squares indicate error at interface location.

Static Drop in Equilibrium

To validate the curvature and surface tension force calculation in hLE, we consider the test of static drop in a quiescent medium with zero gravity. The pressure gradient across the interface balances the surface tension force resulting a zero velocity and static drop at all times. Errors in representation of the surface tension and curvature at the interface, however, lead to non-zero velocity, or the so called 'spurious currents'. The exact solution for the pressure jump across the interface for a circular two-dimensional drop is: $\Delta P_{exact} = \sigma\kappa_{exact}$ where $\kappa_{exact} = 1/R$ and R is the radius of the drop. We consider a square domain having sides of eight units. A drop of radius $R = 2$ is placed at the center of the domain. The surface tension coefficient σ is taken to be 73, the drop density is 1 and the surrounding fluid density is 0.1. Accordingly, the pressure jump across the interface should be $\Delta P = 36.5$ units. All parameters are in SI units and correspond to the test case simulated by Francois *et al.* [17]. The background grid consists of uniform Cartesian elements with resolution of $R/\Delta = 10$. The resolution of the Lagrangian points is refined successively to have $R/\Delta_{LP} = 45, 60, 75$. The time step is fixed at $\Delta t = 10^{-3}$. The interface remains a perfect circle after $t = 0.5$ with low magnitudes of spurious currents. Table 2a shows the convergence of L_1-error in total kinetic energy at $t = 0.5$, indicating larger than second order convergence. Remeshing and reinitialization are *suppressed* in the above calculations.



Δ	Error
0.044	4.5×10^{-8}
0.033	1.27×10^{-8}
0.0266	4.09×10^{-9}

(a) Zero-gravity drop

Mode	Predicted ω	Theoretical ω_n	E_{period}
2	0.844	0.862	0.020
3	1.671	1.726	0.0328
4	2.58	2.719	0.0519

(b) Oscillating liquid column

Figure 2: (a) Absolute error in spurious current magnitude as function of grid spacing, (b) Error in the predicted period of an oscillating liquid column for different modes on a 32^2 grid with $\frac{\Delta}{\Delta_{LP}} = 4.5$

Droplet Oscillation

Simulation of oscillating droplet or liquid column due to perturbations on the surface under zero-gravity conditions are performed to analyze the accuracy of the solver for capillary waves. A cylindrical liquid column with the radius perturbed according to, $r = r_0 + \alpha\cos(n\theta)$, and has a frequency of oscillation given by $\omega_n^2 = \frac{(n^3-n)\sigma}{(\rho_d+\rho_e)r_0^3}$, where ρ_d and ρ_e are the density, interior and exterior to the liquid column, respectively [21]. The cases considered include $\sigma = 1$, $\alpha = 0.1$ (this is 10% larger than the perturbation considered in [21]), $\rho_d = 1$, $\rho_e = 0.01$, $r_0 = 2$ in a $[-10, 10]^2$ doubly connected computational domain. The grid resolution is 32^2 and the Lagrangian particle resolution is fixed at $\frac{\Delta}{\Delta_{LP}} = 4.5$. The second, third, and fourth modes are simulated. The relative error in period of oscillation $E_{period} = |T_{numerical}\omega_n/2\pi - 1|$ is given in Table 2b. The errors are comparable to those reported in [21], even for ten times larger perturbation used in the present study.

Damped Surface Waves

Small amplitude damped surface wave between two immiscible fluids is investigated by comparing the numerical solution to the theoretical solution of the initial value problem obtained by Prosperetti [22]. Initially the interface between the two fluids inside a box $[0\ 2\pi] \times [0\ 2\pi]$ is perturbed by a sinusoidal wave disturbance of wavelength $\lambda = 2\pi$ and amplitude $A_0 = 0.01\lambda$ [12]. Periodic boundary conditions are used in the x direction, and slip conditions are used in the y-direction. The case analyzed consists of two fluids of equal density $\rho_1 = \rho_2 = 1$, and equal kinematic viscosities $\nu = 0.006472$. For $\sigma = 2$ and $\Delta t = 0.02$, the time-evolution of the amplitude of the surface is plotted in figure 3. Two different grid resolutions 16^2 and 32^2 are used for this test case with the $\frac{\Delta}{\Delta_{LP}} = 4$. The coarse grid solution shows large errors in period and amplitude, however, with grid refinement, converging results are obtained.

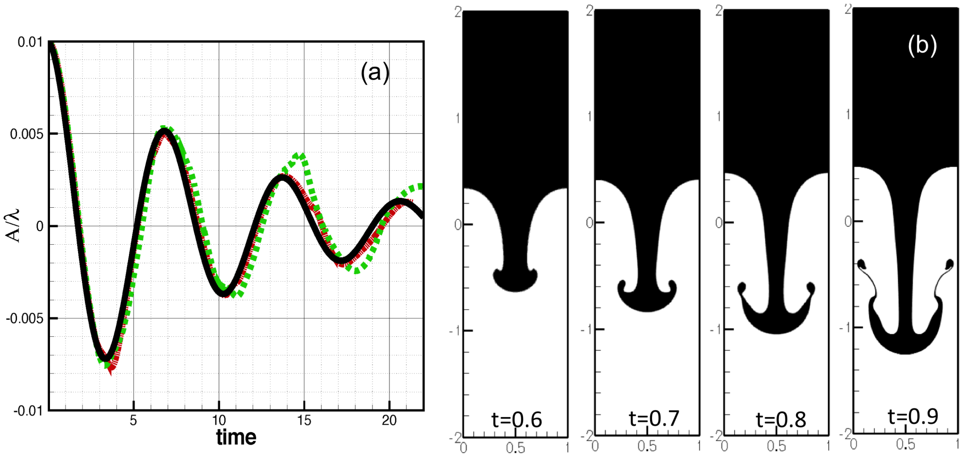


Figure 3: (a) Time evolution of a damped surface wave for two grid resolutions $\Delta = 16^2$ (dashed) and 32^2 (dash dot) compared with theoretical result (solid). The LP resolution is held fixed at $\frac{\Delta}{\Delta_{LP}} = 4.$; (b) Time sequence of Rayleigh Taylor instability of $\Delta = 1/288$ grid.

Rayleigh Taylor Instability

The common test of heavier fluid over a lighter fluid giving rise to Rayleigh-Taylor instability is performed to evaluate the accuracy of the scheme ([23, 13, 12]. The simulation parameters are: $\rho_1 = 1.225$, $\mu_1 = 0.00313$ (heavy fluid), $\rho_2/\rho_1 = 0.1383$, $\frac{\mu_2}{\mu_1} = 1$, $g = 9.81$, $\sigma = 0$ in a computational domain of size $[1, 4]$. Initially the interface between the two fluids (at the center of the domain) is perturbed by a cosine wave of amplitude 0.05. The heavier fluid falls due to gravity, giving rise to a Rayleigh-Taylor instability of the interface perturbation. The surface tension forces are neglected. The boundaries in the x direction are assumed periodic whereas slip conditions are employed in the y direction. The time step is fixed at 5×10^{-4} and the flow evolutions for different grid resolutions are compared at certain time instances. The grid resolutions used are $\Delta = 1/64$, $1/128$, and $1/288$. The LP resolution relative the grid is fixed at $\frac{\Delta}{\Delta_{LP}} = 4.5$. Figure (3b) shows the instantaneous snapshots obtained for the finest grid resolution, comparing well with the published data [12].

Summary

A new hybrid Lagrangian-Eulerian (hLE) scheme, combining a mesh-free Lagrangian technique with a finite-volume flow solver, has been developed for direct simulations of two-phase flows with fully resolved interfaces. This approach merges the naturally adaptive nature of particle-based schemes, for efficient representation of the interface between two media, with the relative flexibility offered by grid-based solvers for complex flows. In hLE, a mesh-free, particle-based scheme for interface tracking [7] is integrated with a co-located grid based finite volume solver. The potential advantage of the hLE method is that the background mesh could be of any kind: *structured, body-fitted, or arbitrary shaped unstructured* (hex, pyramids, tetrahedrons, prisms) and may be *stationary or changing in time (adaptive refinement)*. In this work, we used uniform Cartesian grids for the background mesh. A balanced force algorithm [17] for accurate representation of surface tension forces and considerably reduced magnitudes of spurious currents, is used to solve the two-phase flow equations. The predictive capability of the hLE scheme is verified by performing a series of standard test cases for interface tracking showing good predictive capability.

Acknowledgments

This work is supported by the Office of Naval Research (ONR) grant number N000140610697, under the supervision of Dr. Ki-Han Kim. SVA acknowledges Marcus Herrmann (Arizona State University) for several useful discussions on level set methods, curvature calculations and his RLSG approach.

References

[1] S. Koshizuka, A. Nobe, and Y. Oka. *Int. J. Numer. Meth. Fluids*, 26(7):751–769, 1998.

[2] P. Koumoutsakos. *Annu. Rev. Fluid Mech.*, 37(1):457–487, 2005.

[3] IF Sbalzarini, JH Walther, M. Bergdorf, SE Hieber, EM Kotsalis, and P. Koumoutsakos. *J. Comput. Phys.*, 215(2):566–588, 2006.

[4] JJ Monaghan. *Reports on Progress in Physics*, 68(8):1703–1759, 2005.

[5] HY Yoon, S. Koshizuka, and Y. Oka. *Int. J. Multiphase Flow*, 27(2):277–298, 2001.

[6] JH Walther, T. Werder, RL Jaffe, and P. Koumoutsakos. *Physical Review E*, 69(6):62201, 2004.

[7] S.E. Hieber and P. Koumoutsakos. *J. Comput. Phys.*, 210(1):342–367, 2005.

[8] D. Enright, R. Fedkiw, J. Ferziger, and I. Mitchell. *J. Comput. Phys.*, 183(1):83–116, 2002.

[9] J. Strain. *J. Comp. Phy.*, 170:373–394, 2001.

[10] K. Mahesh, G. Constantinescu, and P. Moin. *J. Comput. Phys.*, 197(1):215–240, 2004.

[11] D. Peng, B. Merriman, S. Osher, H. Zhao, and M. Kang. *J. Comput. Phys.*, 155(2):410–438, 1999.

[12] M. Herrmann. *J. Comp. Physics*, 227(4):2674–2706, 2008.

[13] P. Gómez, J. Hernández, and J. López. *Int. J. Numer. Meth. Eng.*, 2005.

[14] Jeff D. Eldredge, Anthony Leonard, and Tim Colonius. *J. Comput. Phys.*, 180(2):686–709, 2002.

[15] JU Brackbill, DB Kothe, and C. Zemach. *J. Comput. Phys.*, 100(2):335–354, 1992.

[16] G. Tryggvason, B. Bunner, A. Esmaeeli, N. Al-Rawahi, W. Tauber, J. Han, YJ Jan, D. Juric, and S. Nas. *J. Comput. Phys.*, 169(2):708–759, 2001.

[17] M.M. Francois, S.J. Cummins, E.D. Dendy, D.B. Kothe, J.M. Sicilian, and M.W. Williams. *J. Comput. Phys.*, 213(1):141–173, 2006.

[18] E. Fatemi and M. Sussman. *SIAM J. Sci. Comput.*, 158(1):36–58, 1995.

[19] M. Sussman, E. Fatemi, P. Smereka, and S. Osher. *Computers & Fluids*, 27(5):663–680, 1998.

[20] E. Shams and Apte S.V. *ILASS America's* 20*th* *Annual Conference on Liquid Atomization and Spray Systems, Chicago, IL, May 2007*, 30, 2007.

[21] D.J. Torres and J.U. Brackbill. *J. Comp. Physics*, 165(2):620–644, 2000.

[22] A. Prosperetti. *Phy. Fluids*, 24(7):1217–1223, 1981.

[23] S. Popinet and S. Zaleski. *Int. J. Numer. Methods Fluids*, 30:775–793, 1999.