INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2009; **60**:753–778
Published online 24 September 2008 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1912

A hybrid interface method for three-dimensional multiphase flows based on front tracking and level set techniques

Seungwon Shin[1,*,†] and Damir Juric[2]

[1] *Department of Mechanical and System Design Engineering, Hongik University, Seoul 121-791, Korea*
[2] *Laboratoire d'Informatique pour la Mécanique et les Sciences de l'Ingénieur (LIMSI), CNRS-UPR 3251, BP133, 91403 Orsay, France*

SUMMARY

Successful interface methods for multiphase flows need to be designed to operate well in the opposite extremes of strongly surface tension-dominant flows on the one hand and strongly deforming flows on the other. To this end, recent advances in direct numerical simulation of multiphase flows have involved the hybridization of popular methods. One hybrid approach developed by the authors is the level contour reconstruction method (LCRM), which combines the characteristics of both the front tracking and the level set method. It was designed specifically for general 3D multiphase flow problems where very dynamic and deformable interfaces interact and where accuracy, reliability, and simplicity are essential features. In this paper, we carry the hybridization of the LCRM with the level set technique to a further level in that the explicit calculation of a distance function is introduced and plays a crucial role in the interface reconstruction procedure as well as in the calculation of the surface tension force. An accurate method of computing the distance function directly from the tracked interface is presented whereby a vector distance function is found, i.e. the minimum distance to the interface as well as the corresponding minimum distance point location on the interface itself. This information allows us to calculate a compact curvature field for the computation of the surface tension force, which decreases the level of parasitic currents to a negligible level. Various benchmark test cases to demonstrate the accuracy of the new schemes compared with other existing methods are provided. Copyright © 2008 John Wiley & Sons, Ltd.

Received 19 February 2008; Revised 24 July 2008; Accepted 29 July 2008

KEY WORDS: numerical simulation; front tracking; level set; multiphase flow; curvature; parasitic current

[*] Correspondence to: Seungwon Shin, Department of Mechanical and System Design Engineering, Hongik University, 72-1, Sangsu-dong, Mapo-gu, Seoul 121-791, Korea.

[†] E-mail: sshin@hongik.ac.kr

Contract/grant sponsor: Korean Government; contract/grant number: KRF-2007-331-D00058
Contract/grant sponsor: Institut du Développement et des Ressources en Informatique Scientifique (IDRIS)

1. INTRODUCTION

Accurate characterization of the interfacial front in multiphase flow simulations has been a crucial issue for the computational modeling of numerous important engineering applications. Highly dynamic and deforming interfaces, together with complex interfacial physics from geometric curvature associated with surface tension as well as phase transformation make it a still more formidable task to come up with an accurate, reliable, and yet simple method. Successful interface methods for multiphase flows need to be designed to operate well in the opposite extremes of strongly surface tension-dominant flows on the one hand and strongly deforming flows on the other. In strongly surface tension-dominant flows, numerical inaccuracies in the calculation of interface geometry result in so-called parasitic or spurious currents. In strongly deforming flows, the difficulty lies in maintaining a high-quality interface while conserving mass and gracefully allowing arbitrary and repeated topology changes.

Front capturing type methods such as volume-of-fluid (VOF) [1–3], level set [4], and phase field [5], which represent the interface implicitly are relatively simpler in their formulation than front-tracking type methods [6–8], which explicitly track the interface. On the other hand, front capturing methods usually incorporate special types of convection schemes to retain interfacial sharpness during the simulation, whereas front-tracking schemes are well known to maintain sharp interface structures and mass conservation as they preserve material characteristics. When fluid filaments become too thin, VOF methods tend to form irregular interfaces, which are not smooth and even discontinuous in the attempt to maintain precise local mass conservation. Level set methods lose/gain a significant amount of mass, lowering the accuracy and ultimately compromising the entire solution. Several researchers have worked to improve the accuracy of the geometrical information in the VOF method [1–3, 9] and numerous attempts have been made to improve mass conservation in level set methods [4] using a variety of reinitialization techniques and higher order ENO/WENO convection schemes. On the other hand, front-tracking schemes pose many difficulties in reconstruction of the interface especially in three dimensions for the essential operations of addition/deletion, breaking, and merging of the computational interface elements.

Each of the above methods has some advantage over the others and varying degrees of success in modeling general 2D multiphase problems over wide parameter ranges. On the contrary, it is still difficult to conduct full direct simulations for 3D multiphase flows, even with current computer resources. Thus, accuracy under limited memory and computational time is an essential feature in developing a numerical algorithm. Recently, there have been efforts to construct hybrids among the above mentioned methodologies with the intention of facilitating simulations of general 3D problems.

Sussman and Puckett [10] proposed a coupled level set/VOF method, which combines the accuracy in mass conservation of the VOF method and the facility of handling complex topology change of the level set method. The resulting scheme still remains Eulerian and alleviates some of the geometrical connectivity problems associated with the VOF method, but it still has some problems of accuracy in under-resolved regions by blindly applying the VOF local mass constraint.

Enright *et al.* [11] added Lagrangian marker particles in Eulerian level set methods to precisely rebuild the zero distance function near the interface. Combining front-tracking type characteristics allows the level set method to obtain subgrid scale accuracy near the interface, as in pure front tracking methods, and better mass conservation properties in under-resolved regions. Thus, the unsatisfactory description of interfacial geometry in the VOF method can be overcome by using the level set function, which maintains fine geometric properties. Their ‘particle level set’ method

DOI: 10.1002/fld

compares favorably with VOF and front tracking in mass conservation as well as in interface resolution. The lack of connectivity between marker particles makes the implementation much easier than front tracking but to maintain accuracy the interface marker particles, positioned randomly and passively advected by the flow, have to be distributed as uniformly as possible. Therefore, a special algorithm is necessary for adding/deleting particles in over or under populated regions.

Aulisa *et al.* [12, 13] present a new hybrid method, which combines markers and VOF methods. Two distinct markers of grid intersection and mass conservation have been used to describe the interface. Both the markers are advected numerically to update the volume fraction. The conservation markers inside each cell keep the local volume fraction at the reference value, whereas the intersection markers, which locate the interface on the grid lines, eliminate the necessity of remeshing the system. Thus, they obtained both smooth motion of the interface as well as good mass conservation as in the standard VOF method.

As an alternative hybrid method, we have developed the level contour reconstruction method (LCRM) [14–17] for general 3D multiphase flows including phase change. The main idea was focused on simplicity and a robust algorithm especially for the 3D case. It combines the characteristics of both front tracking and level set methods. It is still, however, primarily a front-tracking method as it uses Lagrangian elements (line segment elements in 2D and triangles in 3D) to describe the interface and its motion.

During the course of a simulation the interface is periodically regenerated from contour field values (e.g. indicator function or distance function as in the level set method) to avoid undue irregularity of interfacial elements, which can become too large or too small as a result of interface advection. The LCRM can be seen as adopting some ideas from level set techniques, whereas the particle level set method [11] adopts some ideas from the front-tracking techniques. The LCRM uses 2D linear or 3D triangular elements for the interface rather than point particles dispersed in the vicinity of the interface as in the particle level set method. The LCRM is basically a front-tracking type method, which tracks the implicitly connected individual interface elements. At the same time, we take advantage of the fact that the interface can also be represented by a Eulerian function field. Reconstruction of the interface at a certain level of this contour field enables us to naturally, automatically, and robustly model the merging and pinch off of interfaces as in the level set method. It also retains the subgrid accuracy of interfacial motion of front tracking, while eliminating the burden of bookkeeping of the logistic information of neighboring elements as is necessary in the classic front-tracking approach and which incurs a large memory and calculation cost.

Recently, the surface tension force has been formulated in a hybrid Eulerian/Lagrangian form [15], which gives a more accurate representation of curvature. In addition, the LCRM was further improved by the introduction of a high-order reconstruction procedure [16]. By introducing a higher-order interpolation kernel for the interface reconstruction, we were able to increase the accuracy and smoothness of the interface considerably, thus reducing the parasitic currents to a minimal level.

Although the hybrid curvature formulation and high-order reconstruction can maintain the parasitic currents at very small values during the simulation, maintaining an acceptable level is always more difficult in three dimensions than in two dimensions as the interfacial elements are generally more irregular in three dimensions. As we pointed out in Shin [17], the curvature field calculated using the hybrid formulation exhibits sharp disturbances away from the interface due to the nature of the numerical delta function distribution. Using curvature values not at the grid

DOI: 10.1002/fld

cell at hand but at the interface element of interest is preferred in order to further reduce spurious currents.

To differentiate the material field of each phase, we used an indicator function field found by solving a Poisson equation whose right side contains geometric information from the interfacial elements. In most of our previous work we used the FISHPAK [18] routine to compute the indicator function field. Solving a Poisson equation has some drawbacks in defining a continuous boundary condition in cases where the interface is in the vicinity of one of the domain boundaries.

In this paper, we carry the hybridization of the LCRM with the level set technique to a further level in that the explicit calculation of a distance function is introduced and plays a crucial role in the interface reconstruction procedure as well as in the calculation of the surface tension force. An accurate method of computing the distance function directly from the tracked interface is presented, whereby a vector distance function is found, i.e. the minimum distance to the interface as well as the corresponding minimum distance point location on the interface itself. This information then allows us to calculate a compact curvature field based on the curvature at this minimum distance point. The curvature field calculated in this way is compact in the sense that its value is restricted to the value of interface curvature calculated at the corresponding minimum distance point on the interface.

The remainder of this paper is organized as follows. We describe the mathematical formulation in Section 2. Section 3 then describes the direct computation of the distance function from the interface, a brief description of the high-order LCRM, and a description of the compact curvature field calculation. We then present test cases to demonstrate the accuracy of this new hybrid interface method compared with other existing methods.

2. NUMERICAL FORMULATION

2.1. Governing equations

The governing equations for isothermal, incompressible multifluid motion can be expressed in a single field formulation as:

$$\nabla\cdot\mathbf{u}=0 \tag{1}$$

$$\rho\left(\frac{\partial\mathbf{u}}{\partial t}+\mathbf{u}\cdot\nabla\mathbf{u}\right)=-\nabla P+\rho\mathbf{g}+\nabla\cdot\mu(\nabla\mathbf{u}+\nabla\mathbf{u}^{\mathrm{T}})+\mathbf{F} \tag{2}$$

here, $\mathbf{u}$ is the velocity, P the pressure, $\mathbf{g}$ the gravitational acceleration, and $\mathbf{F}$ is the local surface tension force at the interface, which can be described by the hybrid formulation [15] as:

$$\mathbf{F}=\sigma\kappa_H\nabla I \tag{3}$$

where σ is the surface tension coefficient (assumed constant here), I is the indicator function, a smoothed characteristic function for the interface, and κ_H is twice the mean interface curvature field calculated on the Eulerian grid.

The expression for the curvature, κ_H, is given as:

$$\kappa_H=\frac{\mathbf{F}_{\mathrm{L}}\cdot\mathbf{G}}{\sigma\mathbf{G}\cdot\mathbf{G}} \tag{4}$$

DOI: 10.1002/fld

where

$$\mathbf{F}_{\mathrm{L}}=\int_{\Gamma(t)}\sigma\kappa_f\mathbf{n}_f\delta_f(\mathbf{x}-\mathbf{x}_f)\,\mathrm{d}s \tag{5}$$

here, $\mathbf{n}_f$ is the unit normal to the interface, $\mathbf{x}_f=\mathbf{x}(s,t)$ is a parameterization of the interface $\Gamma(t)$, and $\delta_f(\mathbf{x}-\mathbf{x}_f)$ is a 3D Dirac distribution that is non-zero only when $\mathbf{x}=\mathbf{x}_f$. $\mathrm{d}s$ is the length (2D) or area (3D) of the element, κ_f is again twice the mean interface curvature but computed in a Lagrangian fashion. The detailed procedure for calculating the force in Equation (5) can be found in [8, 14, 15]. $\mathbf{G}$ and I will be described in the next section.

Material property fields can be described using the indicator function, $I(\mathbf{x},t)$. For example, the density is calculated as

$$\rho(\mathbf{x},t)=\rho_1+(\rho_2-\rho_1)I(\mathbf{x},t) \tag{6}$$

where the subscripts 1 and 2 refer to the respective fluids. A similar equation is used to define the viscosity, μ.

The interface is advected in a Lagrangian fashion by integrating

$$\frac{\mathrm{d}\mathbf{x}_f}{\mathrm{d}t}=\mathbf{V} \tag{7}$$

where $\mathbf{V}$ is the interface velocity vector interpolated at $\mathbf{x}_f$.

The method used to solve the fluid velocity and pressure is the projection method of Chorin [19]. We use a first-order, forward Euler integration in time. For the spatial discretization, we use the well-known staggered mesh (MAC method) of Harlow and Welch [20]. The pressure is located at the cell centers, whereas the x, y, and z components of velocity are located at the respective faces. All spatial derivatives except the convective term are approximated by standard second-order centered differences. The convective term is discretized using a second-order ENO procedure [21, 22]. On the staggered grid, quantities needed at cell centers are linearly interpolated from cell faces and vice versa. The detailed solution procedure and discretization of the governing equations can be found in [14, 15, 23].

2.2. *Construction of the distance function from an existing front*

The indicator function, I, has essentially the same characteristics as the Heaviside function and can be found by solving the following Poisson's equation with a standard FFT package such as FISHPAK [18] on a uniform Cartesian grid:

$$\nabla^2 I=\nabla\cdot\mathbf{G} \tag{8}$$

where the distribution, $\mathbf{G}$, is the geometric information computed directly on the interface and then distributed onto a Eulerian grid:

$$\mathbf{G}=\int_{\Gamma(t)}\mathbf{n}_f\delta_f(\mathbf{x}-\mathbf{x}_f)\,\mathrm{d}s \tag{9}$$

To compute the indicator function field by solving Equation (8), we need to prescribe boundary conditions for all faces. In the case of evolution of the interface without contact with the boundary,

DOI: 10.1002/fld

it is straightforward to describe the boundary values for I. Since the interface is not in direct contact with the boundary, explicit Dirichlet boundary conditions, i.e. either one or zero for the indicator function value, can be assigned for Equation (8). However, if the interface is in contact with the boundary, it becomes more complex to implement the correct boundary conditions.

An indicator function field, which has the characteristics of a Heaviside side function, can also be found using the distance function as in the level set method. Here, we will describe a new concept for computing a vector distance function from the existing interface (Figure 1). For a given interfacial element, we can identify a sufficiently large local neighborhood of grid nodes and calculate the minimum distance to that element and where on that element the minimum distance point lies. This minimum distance vector can originate either from a point inside the line segment of the element or from the edge of the element. After sweeping through all of the elements, we can construct the final distance function value at each grid node as well as the minimal distance point on the interface element. In this way, the distance function field can be calculated for cells neighboring the interface, usually six cells wide in each direction, and, to save resources, we can denote the distance function away from this strip as an arbitrary sufficiently large number.

We also need the sign function field to provide the correct signed distance function for the indicator function calculation. Because the interface element has a specific orientation given by its normal $\mathbf{n}$, determining the sign function is straightforward when the minimal distance point is located at a point on the line segment of the element (point A in Figure 1). In case the minimal distance point is located at the endpoint of the element (point B in Figure 1 or point Q in Figure 2), more than one element normal might affect the sign function value (point Q in Figure 2). By summing the dot product of element normals ($\mathbf{n}_1$ and $\mathbf{n}_2$ in Figure 2) for the grid point of interest and the direction vector from the minimum distance point on the interface to the cell center location (**PQ** in Figure 2), we can calculate the correct sign function value at Q in Figure 2 as

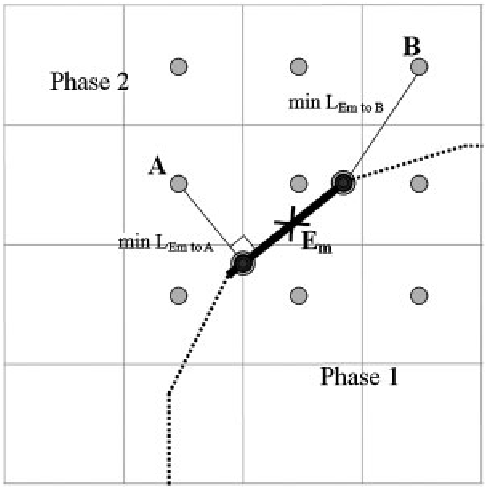

Figure 1. Computing distance function from the existing 1D linear interfacial elements for two-dimensional simulation. The minimum distance point can locate to either inside the line segment or to the edge of the element.

DOI: 10.1002/fld

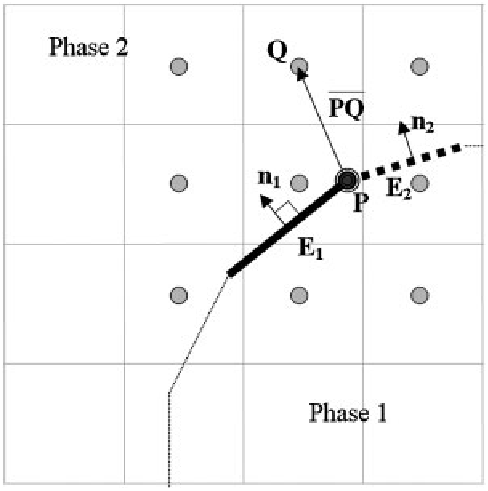

Figure 2. Sign function computation in the case where the minimum distance lies at the edge of the element.

follows:

$$S_0(Q) = \text{sign}\left(\sum_{i=1}^{m} \mathbf{PQ}\cdot\mathbf{n}_i\right) \quad (i = 1, \ldots, m) \tag{10}$$

here, m is the total number of elements and the sign function returns $+1$ or -1 depending on the input. After sweeping the entire interface and assigning a value of either $+1$ or -1 to the nearby grid cells, we can generate the sign function field, S_0, for a few cells near the interface as in Figure 3.

We need to expand this initial sign function, S_0, to the entire domain. We follow an approach similar to that taken by Torres and Brackbill [24] who iteratively solve:

$$\nabla^2 S(\mathbf{x}) = 0 \tag{11}$$

starting from the S_0 initial sign function field near the interface as described in Figure 3. It is not necessary to converge to the complete solution of Equation (11) as the desired quantity is simply the sign field and a few iterations, depending on the grid resolution but usually less than 10 for 2D problems, suffice to calculate it for the entire domain. Figure 4(b) shows a distance function field calculated from a starfish-shaped interface given as:

$$\begin{aligned} x &= \cos(\theta) + 0.4\sin(\theta)\cos(\theta) \\ y &= \sin(\theta) + 0.4\sin(\theta)\sin(\theta) \end{aligned} \tag{12}$$

where θ has values between 0 and 2π. Figure 4(a) shows the initial interface in a 4×4 box with a grid resolution of 50×50. The indicator function computed using the distance function in Figure 4(b) is shown in Figure 4(c). Even with this relatively low resolution, the distance and indicator functions have been calculated correctly.

DOI: 10.1002/fld

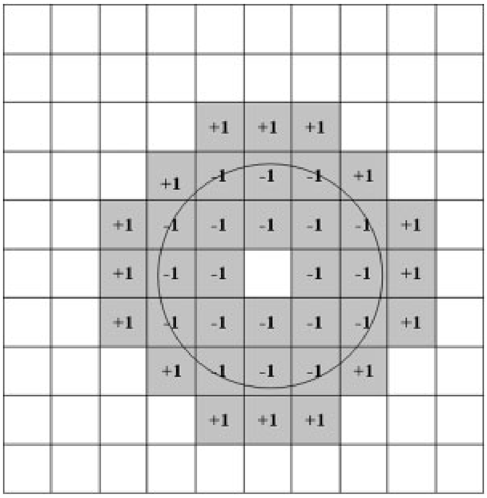

Figure 3. Sign function distribution near the interface before expansion.

The concept for generating the indicator function field for 3D cases will be exactly the same as in 2D cases except that the minimal distance point may lie inside the 2D triangular element or on the edge of the element or at a vertex. Here, we describe the details of the procedure in the 3D case.

In the plane of the triangular element, the location of the minimal distance point from that element to nearby grid cells can lie either inside the triangle or outside (Figure 5). Details of the procedure are as follows:

1. Given a grid cell center location C, draw a normal line to the plane formed by the vertices of the triangle ①②③. The intersection of this normal and the plane is at point B, which can lie either inside the triangular element (Figure 5(a)) or outside the element (Figure 5 (b)).
2. The ordering of the vertices upon creation of the triangular element provides the element with an orientation given by its normal, $\mathbf{n}$, and likewise its edge normals, $\mathbf{n}_{12}, \mathbf{n}_{23}$, and $\mathbf{n}_{31}$, can easily be defined. Points A_1, A_2, A_3 can be found by drawing normal vectors from base point B to each line formed by the legs ①②, ②③, and ③①. In the case $\mathbf{n}_{12}\cdot\mathbf{A_1B}>0$, $\mathbf{n}_{23}\cdot\mathbf{A_2B}>0$, and $\mathbf{n}_{31}\cdot\mathbf{A_3B}>0$, the base point B is located inside the element of interest as shown in Figure 5(a). Then the minimal distance vector to the interface will be $\mathbf{BC}$.
3. If the base point B is outside of the element of interest, at least one of the values $\mathbf{n}_{12}\cdot\mathbf{A_1B}$, $\mathbf{n}_{23}\cdot\mathbf{A_2B}$, $\mathbf{n}_{31}\cdot\mathbf{A_3B}$ will be of negative sign. In the case where two of the values are negative then the minimal distance point will be one of the vertices ① or ② or ③. For example, in Figure 5(b) we can determine that base point B is outside the element from the fact that $\mathbf{n}_{12}\cdot\mathbf{A_1B}>0$, $\mathbf{n}_{23}\cdot\mathbf{A_2B}>0$, and $\mathbf{n}_{31}\cdot\mathbf{A_3B}<0$. Thus, the minimal distance point will lie on the line formed by ①③. If A_3 lies between ① and ③, then the minimum distance point will be A_3. If A_3 is located outside of the segment ①③ then either vertex ① or ③ will be the minimum distance point.

As in the 2D case, we perform a sweep of the entire interface with this procedure. Thus, the distance function field is determined as the minimum distance from grid cells to triangular interface

DOI: 10.1002/fld

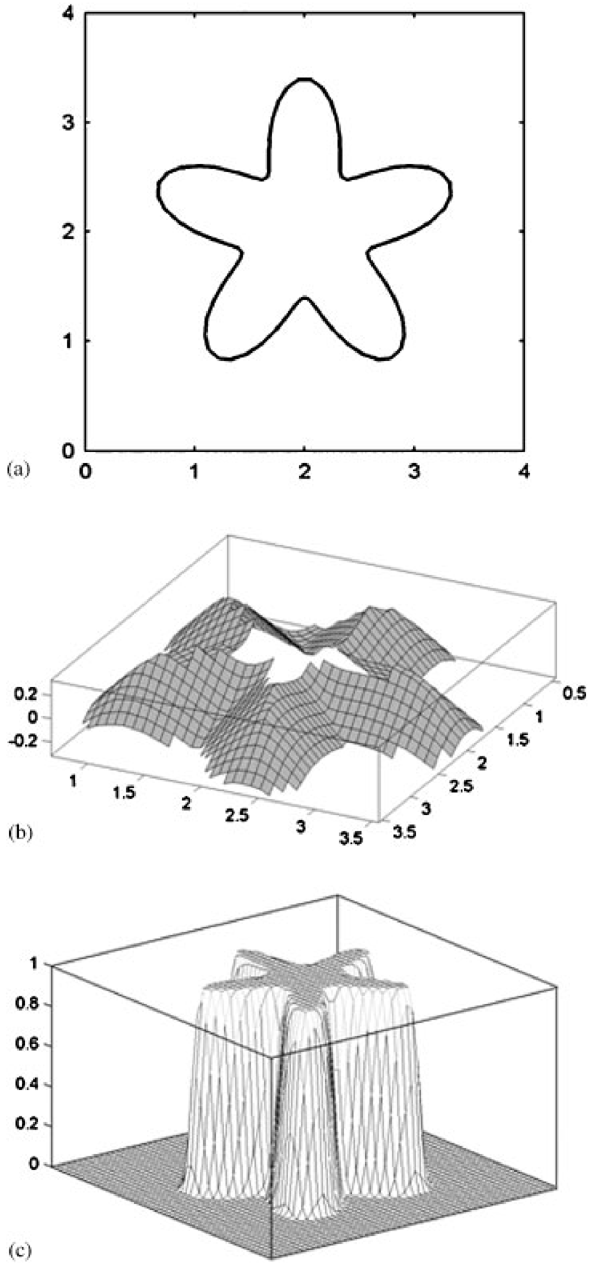

Figure 4. Calculating the distance function of a starfish-shaped interface: (a) original interface; (b) distance function computed directly from interface; and (c) Heaviside function using distance function (b).

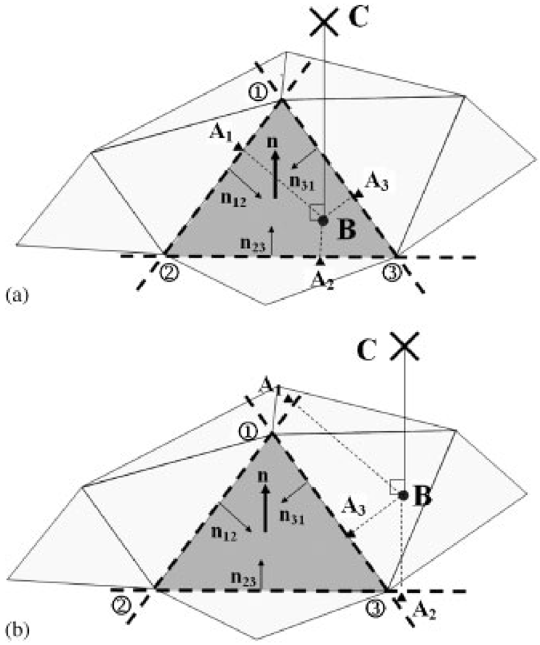

Figure 5. Computing distance function from the existing 2D triangular interfacial elements for three-dimensional simulation: (a) intersection of normal where the interface lies inside the element and (b) intersection of normal where the interface lies outside the element.

elements (whether to a point inside the triangle or on its periphery). Determination of the sign function for the 3D case is exactly the same as in the 2D procedure.

2.3. High-order reconstruction method

During reconstruction using the original LCRM [14], the interface elements were created and located using a linear interpolation of the given indicator function field. We found that reconstruction using this linear interpolation introduces a small disturbance, which eventually dies out very quickly after a few time steps. This indicates that the reconstructed interface is continuous but not smooth with linear interpolation. The effect is usually negligible as reconstruction is not performed at every time step. However, this slight perturbation may cause instability of the solution for simulations requiring frequent reconstruction of the interface, especially with low resolution. Moreover, slight discrepancies in the interface points can induce undesirable results where the exact location of the interface is extremely important. Thus, high-order reconstruction was introduced in [16]. The method is simpler and has an accuracy comparable to other methods using complicated procedures for smoothing the interface.

Here, we briefly introduce the basic concept contained in the high-order reconstruction procedure. The distance function $\phi(i,j,k)$ generated using the process described in Section 2.2 can be modified by adding a trial function, $\psi(i,j,k)$ as follows:

$$\phi^H(\mathbf{x})=\sum_{\mathrm{g}}[\phi(i,j,k)+\psi(i,j,k)]S(\mathbf{x}-\mathbf{x}_{\mathrm{g}}) \tag{13}$$

here, $\mathbf{x}$ is the evaluation point, $\mathbf{x}_\mathrm{g}$ is the grid cell center, and the summation is performed across a small multiple of the mesh, usually two grid cells wide, in each direction. The tensor product of 1D B-splines [25], M, given as:

$$S(\mathbf{x}-\mathbf{x}_\mathrm{g}) = M(x-x_\mathrm{g}, \Delta x)M(y-y_\mathrm{g}, \Delta y)M(z-z_\mathrm{g}, \Delta z) \tag{14}$$

has been used for $S(\mathbf{x}-\mathbf{x}_\mathrm{g})$ to interpolate the local distance function values smoothly. We used cubic B-splines $M_3(x,h)$ suggested by Torres and Brackbill [24]:

$$M_3(x,h) = \begin{cases} \frac{2}{3} - \left(\frac{|x|}{h}\right)^2 + \frac{1}{2}\left(\frac{|x|}{h}\right)^3, & 0 \leqslant \frac{|x|}{h} \leqslant 1 \\ \frac{1}{6}\left(2-\frac{|x|}{h}\right)^3, & 1 \leqslant \frac{|x|}{h} \leqslant 2 \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

here, h represents the grid cell size in each direction. The trial function has been added since the original distance function itself cannot exactly describe the zero contour level of the original interface location. There are several ways of approximating the trial function, $\psi(i, j, k)$ and we used the following function:

$$\psi(i, j, k) = \sum_{N_\mathrm{p}} \delta I_\mathrm{p} S(\mathbf{x}_\mathrm{g} - \mathbf{x}_\mathrm{p}) \tag{16}$$

here, $\mathbf{x}_\mathrm{p}$ is the location of the original interface points before reconstruction, δI_p is the increment needed at the original interface points, and the integral has been performed over all of the elements. The basic idea is similar to the point set method of Torres and Brackbill [24] except that their indicator function has been computed directly from the interface element. The detailed procedure for high-order reconstruction can be found in [16].

2.4. Compact curvature field

As we pointed out in Shin [17], the curvature field from the hybrid formulation shows that higher errors in curvature exist away from the interface (i.e. toward the edge of the interfacial zone) due to the finite support of the numerical Dirac delta distribution in Equation (9). Both the $\mathbf{F}_\mathrm{L}$ and $\mathbf{G}$ distributions are only non-zero in the zone around the interface. They will have values near zero at the edges of this zone, thus any errors become more sensitive to the data away from the interface. In Shin [17] a procedure for calculating curvature locally at the interfacial points and then redistributing this result back to the Eulerian grid was recommended. However, here we make a modification, which we call the compact curvature calculation: when distributing the curvature from the interface element to the Eulerian grid, it is physically more appealing to take a value at the minimum vector distance from the interfacial element of interest. As we have already calculated a minimal distance point to the interface in Section 2.2, the curvature field can be constructed in cells near the interface by taking the curvature value of the minimal distance point on the interface as follows:

$$\kappa_C(i, j, k) = \kappa_H(x_{bt}, y_{bt}, z_{bt}) \tag{17}$$

here, κ_C stands for the compact curvature formulation and each i, j, and k represents the grid cell center location (Figure 6). κ_H is the field distribution obtained using Equation (4) and (x_{bt}, y_{bt}, z_{bt})

DOI: 10.1002/fld

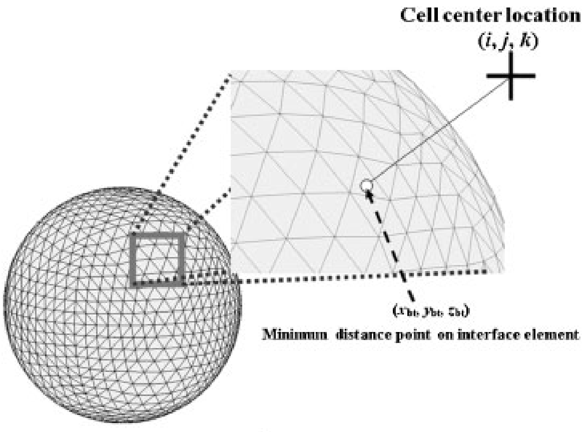

Figure 6. Minimum distance points on the interface from cell center location in three-dimensional simulation. The curvature value at location (i, j, k) can be interpolated at point $(x_{\mathrm{bt}}, y_{\mathrm{bt}}, z_{\mathrm{bt}})$ for compact curvature support.

is the minimum distance point on the interface as in Figure 6. The value of κ_H is interpolated to (x_{bt}, y_{bt}, z_{bt}) using the high-order kernel of Equation (14).

3. RESULTS AND DISCUSSION

3.1. Geometry tracking test

We first demonstrate the accuracy of the high-order reconstruction method on the rigid body rotation of Zalesak's disk [26] in a constant vorticity field. This classic problem has been used extensively to test the advection ability of interface tracking methods. A 0.15 radius slotted disk centered at (0.5, 0.75) will rotate in a unit domain following a velocity field given as:

$$\mathbf{v}(\mathbf{x}) = \frac{\pi}{314}\begin{bmatrix} 50-y \\ x-50 \end{bmatrix} \tag{18}$$

The disk with a slot width of 0.05 and a slot length of 0.175 completes one revolution in 628 time units. Figure 7 shows a comparison between the high-order reconstruction method with a 100×100 grid resolution and a highly accurate front-tracking solution. About 60 interface reconstructions have been performed with high-order reconstruction. After one revolution, high-order reconstruction shows fairly good results considering that only five grid cells capture the slot width. We also plot a detailed image near the sharp corners inside the slot. With this given resolution, the high-order reconstruction method shows a smoothing of the corners as the reconstruction depends on the resolution of the underlying Eulerian grid.

We also tested 3D rigid body rotation using a 3D version of the Zalesak's disk problem [11]. The sphere placed at (0.5, 0.75, 0.5) has a radius of 0.15. The slot has a width of 0.05 and length of 0.175 in a unit domain. It rotates one cycle in the $z=0.5$ plane around the point (0.5, 0.5, 0.5)

DOI: 10.1002/fld

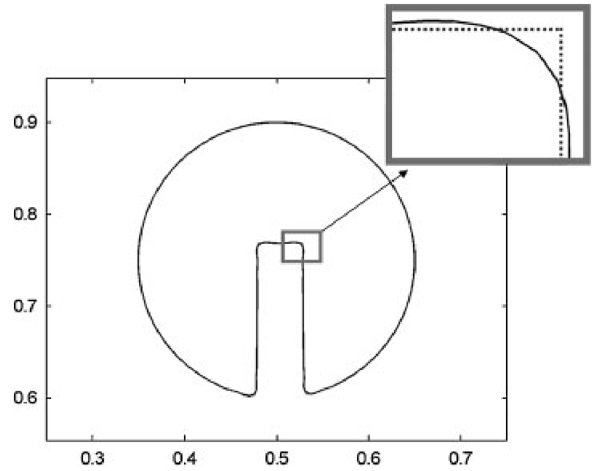

Figure 7. Rigid body rotation of Zalesak's two-dimensional disk. A 0.15 radius slotted disk centered at $(0.5, 0.75)$ will rotate in a unit domain. Figure represents the interface after one revolution at 628 time units.

over 628 time units. The velocity field is defined as follows:

$$\mathbf{v}(\mathbf{x}) = \frac{\pi}{314}\begin{bmatrix} 50-y \\ x-50 \\ 0 \end{bmatrix} \tag{19}$$

A $100\times 100\times 100$ grid resolution has been used and again only about five grid cells capture the slot width and seven the slot length. The same number of reconstructions have been performed as in the 2D simulations. We can see that the high-order reconstruction method shows very accurate results, comparable to those from the hybrid particle level set method [11], with negligible numerical diffusion of the interface (Figure 8). As expected, at $t=628$, the interface is nearly identical to the initial surface at $t=0$. We also checked mass conservation after one revolution for both tests. Mass loss was 10^{-3} and 10^{-1} % for the 2D and 3D cases, respectively.

A much more difficult test is the interface stretching and tearing in the 'vortex in a box' problem introduced by Bell *et al.* [27]. The test demonstrates the method's ability to accurately resolve thin filaments on the scale of a grid cell. A unit computational domain is used with a circle of radius 0.15 placed at $(0.5, 0.75)$ using a 32×32 and 128×128 grid cell domain. Figure 9 shows the interface at $t=5$ with velocity field defined as:

$$\mathbf{v}(\mathbf{x}) = 2\begin{bmatrix} -\sin^2(\pi x)\sin(\pi y)\cos(\pi y) \\ \sin^2(\pi y)\sin(\pi x)\cos(\pi x) \end{bmatrix} \tag{20}$$

The resulting velocity field stretches out the circle into a very long, thin spiral filament, which progressively wraps itself toward the center of the box. As can be seen in Figure 9, the fine structure of the interface has been well captured with sufficiently high resolution. In Figure 9(b), we plot both the high-order LCRM and a high-resolution front-tracked solution with a grid resolution of 128×128 and it is difficult to differentiate one from the other. In this case, the interface

DOI: 10.1002/fld

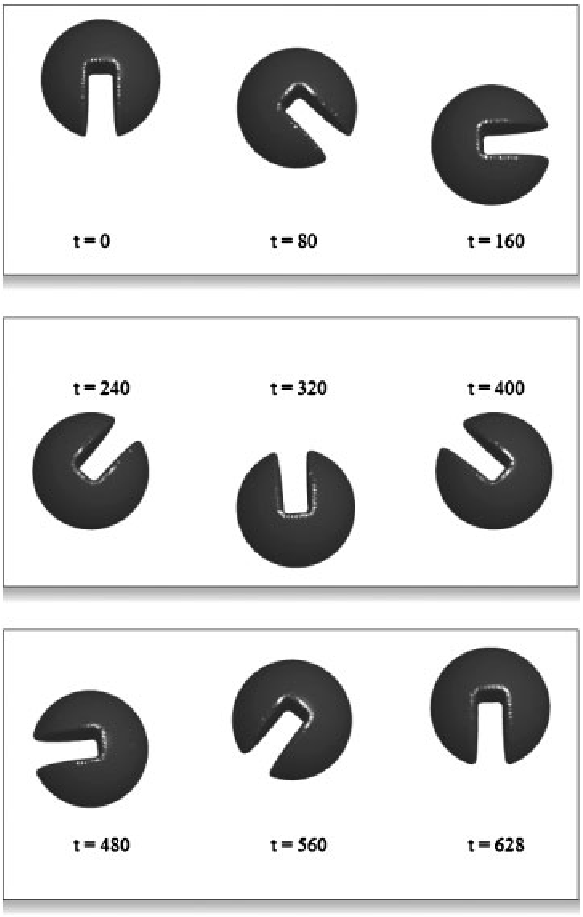

Figure 8. Three-dimensional version of Zalesak's disk. The sphere placed at $(0.5, 0.75, 0.5)$ has a radius of 0.15. The slot has a width of 0.05 and length of 0.175 in a unit domain. It rotates one cycle in the $z = 0.5$ plane around the point $(0.5, 0.5, 0.5)$ over 628 time units.

elements get elongated by the flow and the reconstruction procedure becomes essential for an accurate simulation. Reconstruction has been performed at every 100th time step and a total of 50 reconstructions have been used for both the simulations.

For the purpose of error analysis, the velocity field is time reversed by multiplying by $\cos(\pi t/T)$ where T is the time at which the flow returns to its initial state. The reversal period used in the error analysis of the vortex problem is $T = 8$ producing a maximal stretching similar to the interface in Figure 9. We checked the percentage area loss after one period of motion of the interface. As can be seen from Figure 10, the ability of the high-order LCRM to model interfaces undergoing substantial stretching is about two orders of magnitude better when compared with the hybrid particle level set method and shows second-order convergence with increasing resolution. We plotted the interface shape at $t = T/4, T/2, 3T/4$, and T in Figure 11 with a 64×64 grid resolution. We also plotted the high-resolution front-tracking result at corresponding times for comparison.

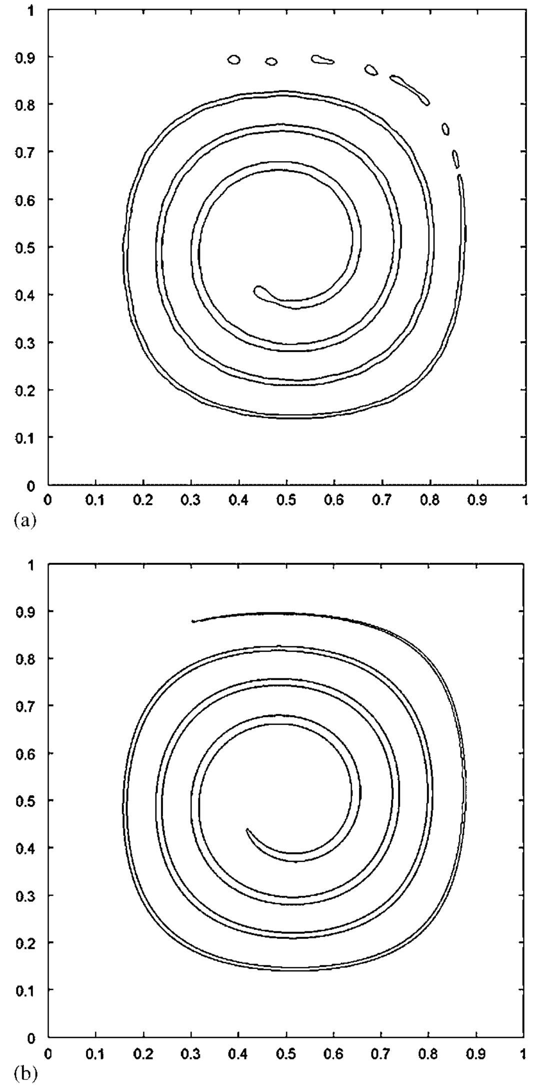

Figure 9. Interface plot at $t=5$ for the test of stretching and tearing of the interface in the 'vortex in a box' problem. A unit computational domain is used with a circle of radius 0.15 placed at $(0.5, 0.75)$ using: (a) a 32×32 grid cell domain and (b) a 128×128 grid cell domain. High-resolution front-tracking solutions have been plotted at the same time for (b) and the two are indistinguishable.

Even with this low resolution, the high-order reconstruction method can capture the thin filament well except near the very tail of the interface at $t=T/2$. The filament is only about one grid cell wide near the tail of the interface and yet the high-order LCRM nearly produces as accurate a result as the high-resolution front-tracked solution.

An even more difficult test case is the entrainment of a circular body in a deformation field defined by 16 vortices as introduced by Smolarkiewicz [28]. A circle centered at $(0.5, 0.5)$ of

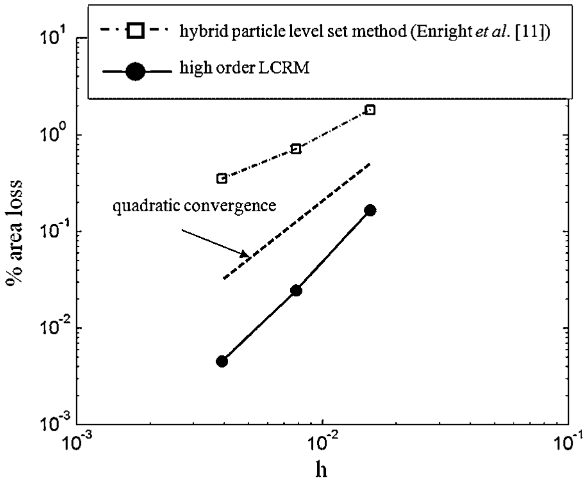

Figure 10. Percentage area loss after one revolution of the interface for the 'vortex in a box' problem in Figure 9 with time reversing flow field. High-order LCRM is about two orders of magnitude better when compared with the hybrid particle level set method and shows second-order convergence with increasing resolution.

radius 0.15 is placed in a unit domain and 200 total interface reconstructions have been performed throughout the simulation. The periodic velocity field is given as

$$\mathbf{v}(\mathbf{x}) = \begin{bmatrix} -\sin[4\pi(x+0.5)]\sin[4\pi(y+0.5)] \\ -\cos[4\pi(x+0.5)]\cos[4\pi(y+0.5)] \end{bmatrix} \tag{21}$$

We multiply the above equation by $\cos(\pi t/T)$ where again T is the period for the interface to reach maximum deformation and return to its initial position. A period of $T=2$ is used with three different grid resolutions for the error analysis. Figure 12 shows percentage area loss vs grid resolution. The error is somewhat high at low resolution using high-order reconstruction but converges quickly showing second-order accuracy. In Figure 13, we generate the interface at $t=T/2$ where maximum stretching of the interface occurs and $t=T$ where the interface should be identical to its initial position with three different resolutions of 256×256, 128×128, and 64×64, respectively. The very thin filament has not been correctly captured with lower resolution because the interface is reconstructed from the Eulerian distance function field. Thus, the high-order reconstruction method still depends on the resolution of the underlying grid. However, with sufficiently high resolution, the high-order reconstruction method shows very accurate results comparable to both the front tracking and the hybrid particle level set method.

To check the accuracy of the high-order reconstruction method with large interface deformation in three dimensions [11], we stretch the interface using the velocity field given as

$$\mathbf{v}(\mathbf{x}) = \begin{bmatrix} 2\sin^2(\pi x)\sin(2\pi y)\cos(2\pi z) \\ -\sin(2\pi x)\sin^2(\pi y)\sin(2\pi z) \\ -\sin(2\pi x)\sin(2\pi y)\sin^2(\pi z) \end{bmatrix} \tag{22}$$

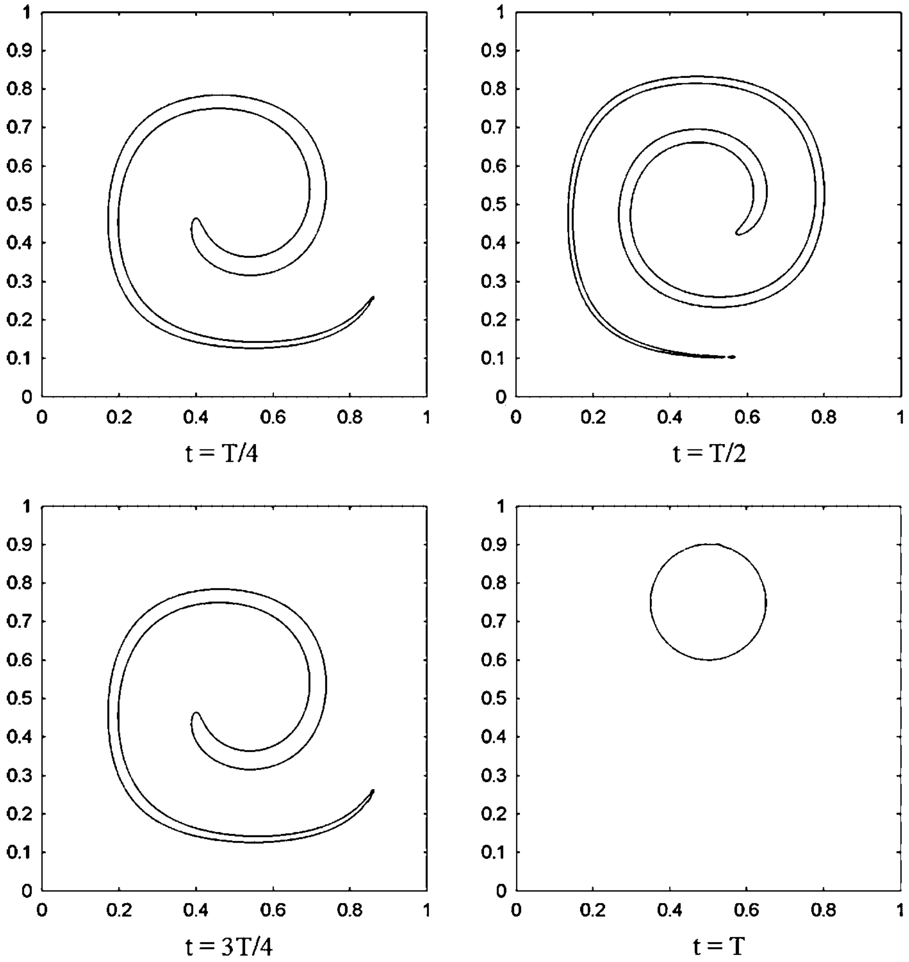

Figure 11. Interface shape at $t=T/4, T/2, 3T/4$, and T for the 'vortex in a box' problem in Figure 9 with a 64×64 grid resolution. High-resolution front-tracking solutions have been plotted at the same time and the two are indistinguishable.

which induces a deformation of the interface in the x–y plane as well as in the x–z plane. The flow field is modulated in time with a period of $T=3$. We simulate a unit computational domain containing a radius 0.15 sphere at $(0.35, 0.35, 0.35)$ with a $100\times 100\times 100$ grid resolution. The sphere becomes highly deformed by the vortical flow field. Figure 14 shows the interface shape at $t=0$ (top), $0.5, 1, 1.5, 2, 2.5$, and 3. Note that the final frame at $t=T=3$ should ideally be identical to the initial sphere at $t=0$. In addition, the frames at $t=1$ and $t=2$ should be identical and the frames at $t=2.5$ and $t=0.5$ should be identical. Maximum stretching occurs at $t=1.5$ where the volume loss is 0.37%. After completing a full cycle, at $t=3$, the high-order LCRM has lost only 0.22% of initial volume. Compared with the hybrid particle level set method, which lost 2.6% after one period, the high-order reconstruction method shows remarkable accuracy even in three dimensions. A total of 247 894 elements are used to capture the interface at $t=1.5$ and

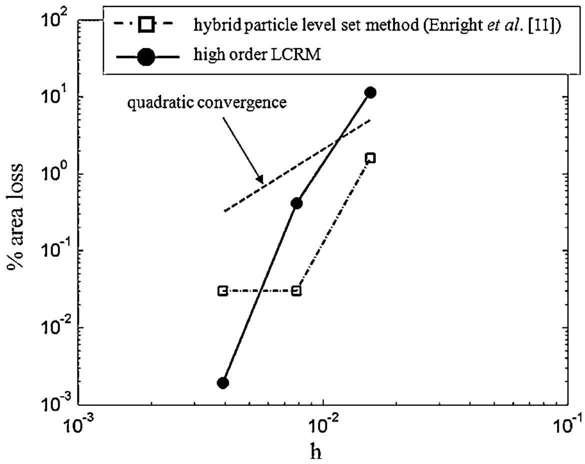

Figure 12. Percentage area loss vs grid resolution for the entrainment of a circular body in a deformation field defined by 16 vortices. A circle centered at (0.5, 0.5) of radius 0.15 is placed in a unit domain.

58 064 elements are used for the initial and final interface in Figure 14. A total of 300 interface reconstructions have been performed during the simulation.

3.2. Parasitic currents

Spurious or parasitic currents become an important problem in two-phase modeling as they impose limitations on the application of interface methods to two-phase flows. Parasitic currents plague all interface methods and numerous attempts at remediation have been implemented in order to reduce or eliminate these currents. By using the hybrid formulation of the surface tension force [15] combined with the high-order reconstruction method [16], we can suppress spurious currents to a minimal level compared with the classic front-tracking formulation. We will check the effect of the compact curvature formulation on parasitic currents and will focus only on the 3D problem as the interface after reconstruction is usually more irregular in three dimensions than in two dimensions and thus parasitic currents will have a greater effect on the solution.

Renardy and Renardy [29] proposed an algorithm for improving the surface tension computation and thus reducing the parasitic currents in VOF methods. They calculated the interface curvature from an optimal fit for a quadratic approximation to the color function over groups of cells near the interface. Using this, parasitic currents are reduced by two orders of magnitude compared with the conventional VOF-CSF [1] or VOF-CSS [2, 3] methods. We repeated the simulation in Table I of Renardy and Renardy [29]. We placed a spherical drop centered at (0.5, 0.5, 0.5), with a radius of 0.125 and surface tension of 0.357 in a $1\times1\times1$ box with $10\times10\times10$, $20\times20\times20$, and $40\times40\times40$ resolutions. The boundary condition is no slip at the top and bottom walls, and periodic in the x and y directions. Both fluids have an equal density of 4 and a viscosity of 1. The initial velocity field is zero. The exact solution is zero velocity for all time. Figure 15 shows the L_∞, L_1, and L_2 norms of the velocity field for the spurious currents. As we discussed in [15], if the initial interface is a circle or sphere with uniformly distributed elements along the interface where exact curvature has been employed or an exact distance function has been prescribed, we can maintain the parasitic currents at machine zero level precision. This being given, all of the subsequent tests

DOI: 10.1002/fld

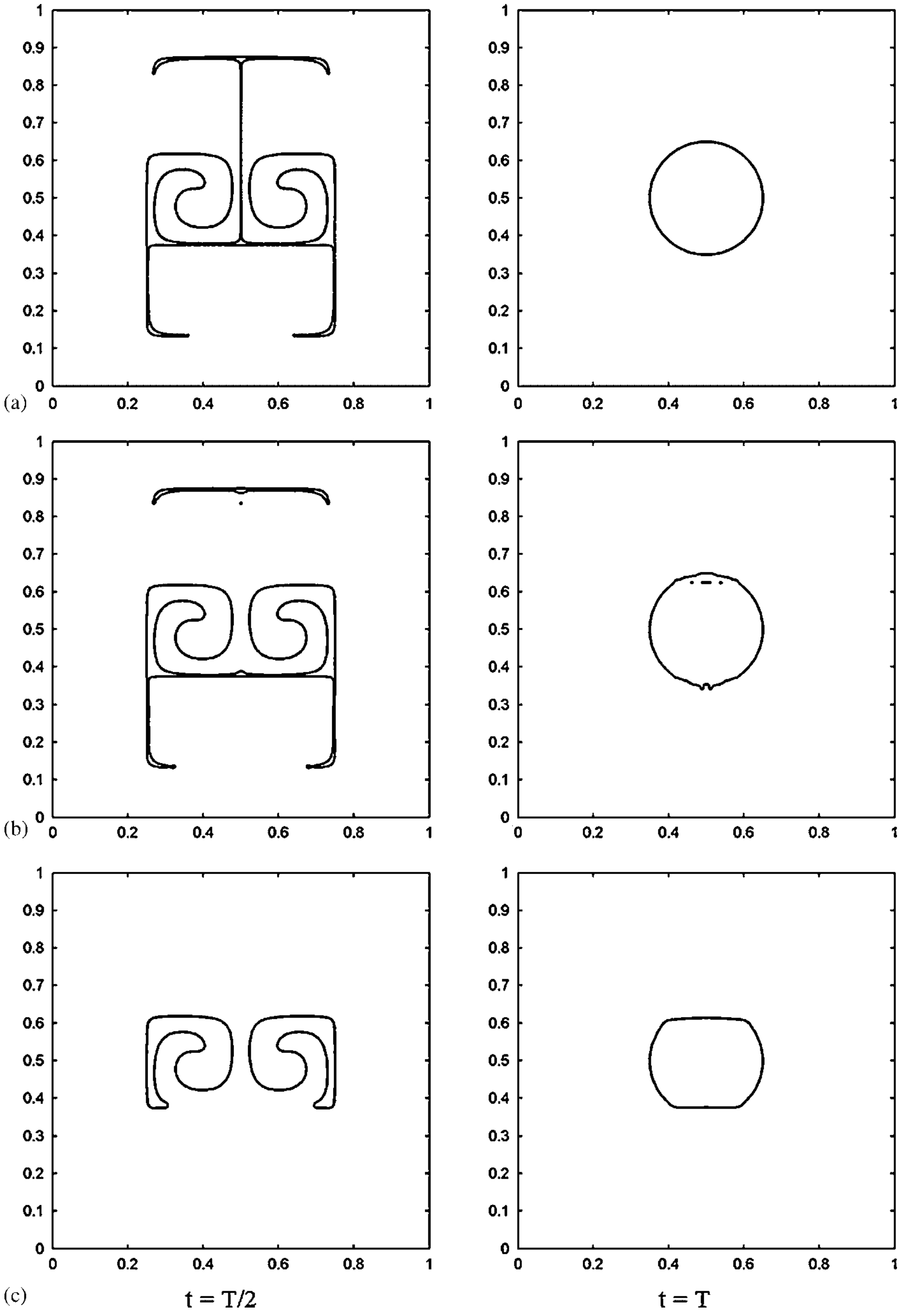

Figure 13. Interface plot at $t = T/2$ where maximum stretching of the interface occurs and at $t = T$ where the interface should be identical to its initial position with three different resolutions of: (a) 256×256; (b) 128×128; and (c) 64×64.

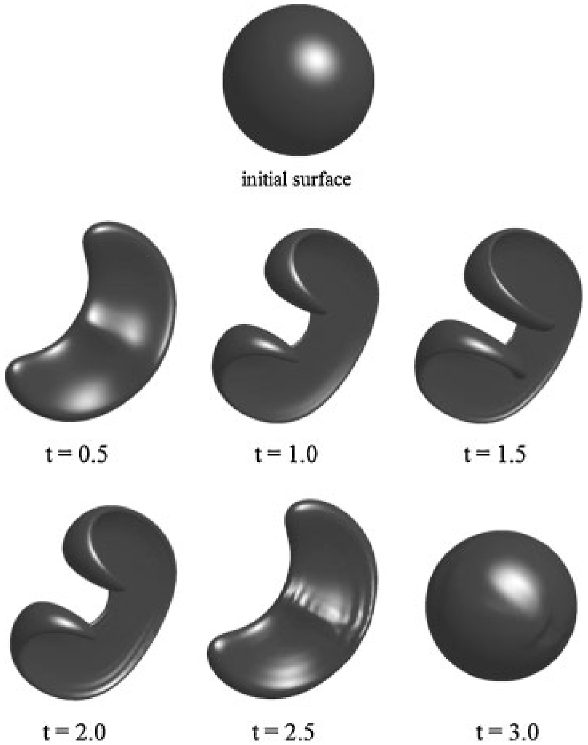

Figure 14. Large interface deformation in three dimensions. A sphere of radius 0.15 is placed within a unit computational domain at $(0.35, 0.35, 0.35)$. A $100\times 100\times 100$ grid resolution was used.

are for interfaces with non-uniformly distributed elements at the start of the simulation, which can be generated with a single reconstruction of the interface from the given distance function for a circle or sphere. As can be seen from Figure 15, the compact curvature formulation shows remarkable improvement over the hybrid surface tension formulation. Each norm of the compact curvature formulation is slightly better compared with the results of Renardy and Renardy [29] and the convergence rate is the same as theirs.

We also tested a more severe case with a high density and viscosity ratio. A drop of radius 0.25 is placed in a $1\times 1\times 1$ box resolved by a $20\times 20\times 20$ mesh. The material properties have been chosen similar to those of a water droplet in air ($\rho_1/\rho_2 = 1000$, $\mu_1/\mu_2 = 100$). The Laplace number ($La = \sigma\rho D/\mu^2$) based on the water properties is 5×10^5 and air properties is 5×10^6. As we can see in Figure 16, the maximum value of the spurious currents with the compact curvature calculation is almost one order of magnitude lower than that of the hybrid surface tension formulation alone and the high-order reconstruction method generates no spikes in spurious currents after reconstruction. Interface reconstruction has been performed at every 10th time step. We conclude that parasitic currents have been shown to be small in the standard front-tracking method [6–8] but not negligible. By using the compact curvature formulation with the high-order reconstruction method, we reduce the parasitic currents to a minimal level with no additional disturbance during reconstruction, which is especially beneficial in 3D simulations where grid resolution is generally lower and where the

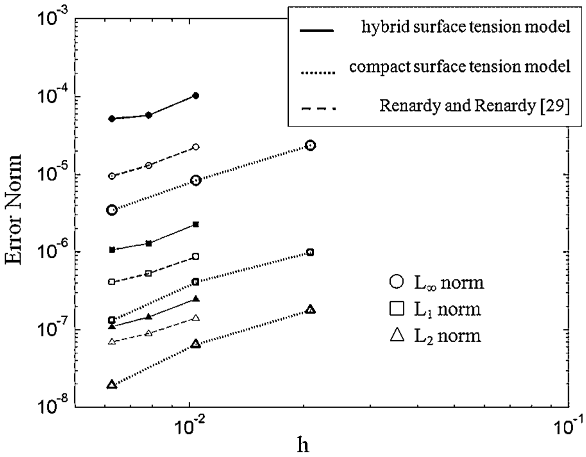

Figure 15. Velocity norms of spurious current vs grid resolution compared with the results from Renardy and Renardy [29]. Spherical drop centered at $(0.5, 0.5, 0.5)$, with a radius of 0.125 and a surface tension of 0.357 in a $1\times1\times1$ box with 10^3, 20^3, and 40^3 resolution. The boundary condition is no slip at the top and bottom walls, and periodic in the x and y directions. Both fluids have an equal density of 4 and a viscosity of 1.

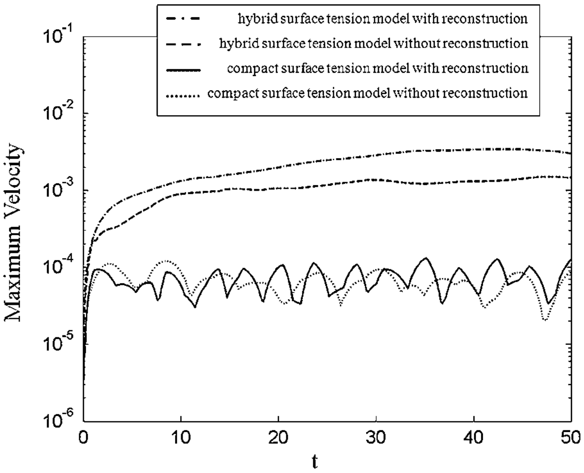

Figure 16. Maximum spurious velocity with a high density and viscosity ratio. A drop of radius 0.25 is placed in a $1\times1\times1$ box resolved by a $20\times20\times20$ mesh. The material properties have been chosen similar to those of a water droplet in air $(\rho_1/\rho_2=1000, \mu_1/\mu_2=100)$. The Laplace number $(La=\sigma\rho D/\mu^2)$ based on the water properties is 5×10^5 and air properties is 5×10^6.

DOI: 10.1002/fld

triangular interface elements generally can have a more irregular shape compared with the line elements for 2D cases.

3.3. 3D bubble rise

We calculated the rise of a single bubble to validate the accuracy of the current method. We used a domain of size 4R × 4R × 12R with a density ratio of 100, viscosity ratio of 10, a Reynolds number of 29.23, and a Weber number of 1.95 as used by Esmaeeli and Tryggvason [30]. They obtained a terminal bubble rise velocity of 20.5 using the front-tracking method. In our current simulation, a 32 × 32 × 96 grid has been used with a time step of 10^{-4}. As can be seen from Figure 17(a), the rise velocity of the bubble reaches a steady-state value of 20.5 and is everywhere within 1% of the results calculated by Esmaeeli and Tryggvason [30]. The rise velocity calculated with both

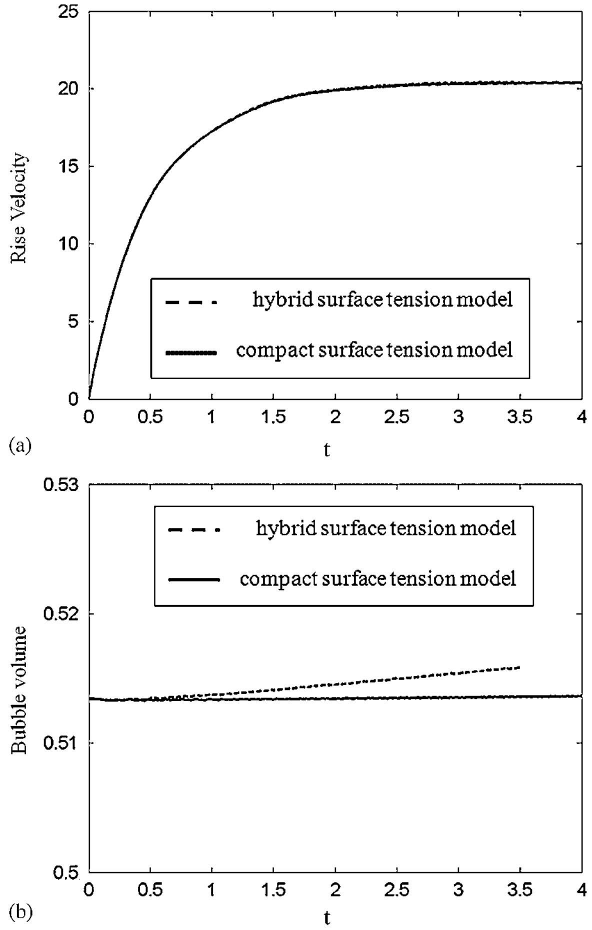

Figure 17. Rise of a single bubble. Domain of size 4R × 4R × 12R with a density ratio of 100, viscosity ratio of 10, a Reynolds number of 29.23, and a Weber number of 1.95. A 32 × 32 × 96 grid has been used with a time step of 10^{-4}: (a) rise velocity and (b) mass conservation during the simulation.

the compact curvature formulation and the hybrid formulation has been plotted in Figure 17(a) and shows identical results. Total mass change during the simulation was plotted in Figure 17(b) and is almost negligible for the compact curvature formulation. The mass change is somewhat higher with the hybrid surface tension formulation alone. Using the same time step for both the compact curvature and hybrid surface tension formulation, we see that for the hybrid formulation the simulation eventually breaks down near $t = 3.5$.

The curvature field for a cross section of the interface at early time is shown in Figure 18 for both the compact curvature formulation and the hybrid surface tension formulation. The compact curvature formulation shows a nearly flat curvature field but the hybrid surface tension formulation has spikes at the edge of the interfacial zone and the curvature field has slight peaks and bumps throughout the interfacial region. Owing to this perturbation of the curvature field with the hybrid formulation, the simulation requires a relatively smaller time step compared with the compact

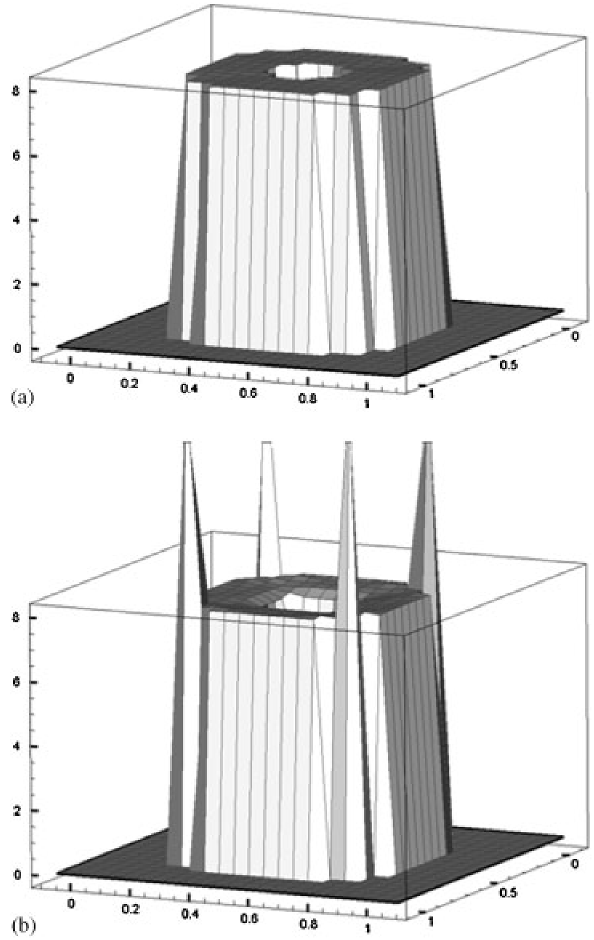

Figure 18. Curvature field for a cross section of the interface at early time in the bubble rise simulation: (a) compact curvature formulation and (b) hybrid surface tension formulation alone.

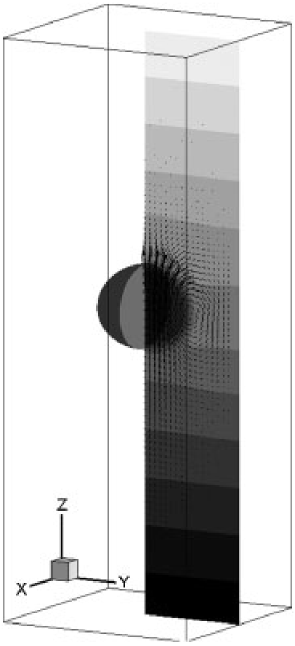

Figure 19. Velocity and pressure distribution through a vertical center plane along with the interface for the bubble rise simulation.

curvature formulation. The velocity and pressure distribution through a vertical center plane along with the interface is plotted in Figure 19.

4. CONCLUSION

We have proposed a hybrid interface method for multiphase flows, which combines some essential features of front tracking and level set techniques. While retaining an explicitly tracked interface using interfacial elements, the calculation of a vector distance function plays a crucial role in the periodic reconstruction of the interface elements in the LCRM to maintain excellent mass conservation and interface fidelity. In addition, the vector distance function is now incorporated as an essential ingredient in the calculation of the surface tension force thereby reducing parasitic currents to a negligible level.

Despite the fact that the linear reconstruction used with the original LCRM shows good overall performance, it experiences a relatively large error in the exact interface location and local mass conservation after reconstruction. This usually leads to instability of the solution where repetitive reconstruction is necessary. By using high-order reconstruction [16], we could reduce the mass redistribution between two distinct interfaces or widely differing curvature regions to a negligible level and relocate the interface much more accurately and smoothly. However, the accuracy of the reconstruction procedure is essentially dictated by the underlying Eulerian grid resolution for the distance function and thus reconstruction frequency may become an important issue with low-resolution simulations. With sufficiently high resolution, we found that the frequency of

DOI: 10.1002/fld

reconstruction has a very minor impact on the final solution. Several tests including surface convolution, interface evolution in a vortex field, and large surface deformation in three dimensions have been performed and the high-order reconstruction procedure showed comparable or even better results compared with other existing methods.

A new procedure for calculating the indicator function field has been presented, which uses a vector distance function computed directly from the tracked interface. The procedure is simple, easily applied for interfaces in both 2D and 3D simulations and can handle complicated boundary conditions in a straightforward manner. This is especially important where the interface contacts the wall.

The improvement using the compact curvature formulation along with high-order reconstruction is evident in tests of the magnitude of spurious currents. Focus has been placed on the 3D problem and there the parasitic currents are about one order of magnitude smaller than with the hybrid surface tension formulation [15] alone. As there is virtually no perturbation caused by the high-order reconstruction with the compact curvature scheme, we can achieve an accurate surface shape even with exceptionally low grid resolution. This can be very important in 3D simulations where the use of sufficient grid resolution is restricted due to the available computational resources.

ACKNOWLEDGEMENTS

This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD, Basic Research Promotion Fund) (KRF-2007-331-D00058) and by computer resources of the Institut du Développement et des Ressources en Informatique Scientifique (IDRIS) of the CNRS, France.

REFERENCES

1. Brackbill JU, Kothe DB, Zemach C. A continuum method for modeling surface tension. *Journal of Computational Physics* 1992; **100**:335–354.
2. Lafaurie B, Nardone C, Scardovelli R, Zaleski S, Zanetti G. Modeling merging and fragmentation in multiphase flows with SURFER. *Journal of Computational Physics* 1994; **13**:134–147.
3. Scardovelli R, Zaleski S. Direct numerical simulation of free-surface and interfacial flow. *Annual Review of Fluid Mechanics* 1999; **31**:567–603.
4. Osher S, Fedkiw RP. Level set methods: an overview and some recent results. *Journal of Computational Physics* 2001; **169**:463–502.
5. Jamet D, Lebaigue O, Coutris N, Delhaye JM. The second gradient method for the direct numerical simulation of liquid–vapor flows with phase-change. *Journal of Computational Physics* 2001; **169**:624–651.
6. Unverdi SO, Tryggvason G. A front-tracking method for viscous, incompressible, multi-fluid flows. *Journal of Computational Physics* 1992; **100**:25–37.
7. Unverdi SO, Tryggvason G. Computations of multi-fluid flows. *Physica D* 1992; **60**:70–83.
8. Tryggvason G, Bunner B, Esmaeeli A, Juric D, Al-Rawahi N, Tauber W, Han J, Nas S, Jan Y-J. A front tracking method for the computations of multiphase flow. *Journal of Computational Physics* 2001; **169**:708–759.
9. Dyadechko V, Shashkov M. Reconstruction of multi-material interfaces from moment data. *Journal of Computational Physics* 2008; DOI: 10.1016/j.jcp.2007.12.029.
10. Sussman M, Puckett EG. A coupled level set and volume-of-fluid method for computing 3D and axisymmetric incompressible two-phase flows. *Journal of Computational Physics* 2000; **162**:301–337.
11. Enright D, Fedkiw R, Ferziger J, Mitchell I. A hybrid particle level set method for improved interface capturing. *Journal of Computational Physics* 2002; **183**:83–116.
12. Aulisa E, Manservisi S, Scardovelli R. A mixed markers and volume-of-fluid method for the reconstruction and advection of interfaces in two-phase and free-boundary flows. *Journal of Computational Physics* 2003; **188**:611–639.

DOI: 10.1002/fld

13. Aulisa E, Manservisi S, Scardovelli R. A surface marker algorithm coupled to an area-preserving marker redistribution method for three-dimensional interface tracking. *Journal of Computational Physics* 2004; **197**: 555–584.
14. Shin S, Juric D. Modeling three-dimensional multiphase flow using a level contour reconstruction method for front tracking without connectivity. *Journal of Computational Physics* 2002; **180**:427–470.
15. Shin S, Abdel-Khalik SI, Daru V, Juric D. Accurate representation of surface tension using the level contour reconstruction method. *Journal of Computational Physics* 2005; **203**:493–516.
16. Shin S, Juric D. High order level contour reconstruction method. *Journal of Mechanical Science and Technology* 2007; **21**:311–326.
17. Shin S. Computation of the curvature field in numerical simulation of multiphase flow. *Journal of Computational Physics* 2007; **222**:872–878.
18. Swarztrauber PN, Sweet RA. Efficient Fortran subprograms for the solution of separable elliptic partial differential equations. *ACM Transactions on Mathematical Software* 1979; **5**:352–364.
19. Chorin J. Numerical solution of the Navier–Stokes equations. *Mathematics of Computation* 1968; **22**:745–762.
20. Harlow FH, Welch JE. Numerical calculation of time dependent viscous incompressible flow of fluid with free surface. *Physics of Fluids* 1965; **8**:2182–2189.
21. Shu C-W, Osher S. Efficient implementation of essentially non-oscillatory shock capturing schemes II. *Journal of Computational Physics* 1989; **83**:32–78.
22. Sussman M, Fatemi E, Smereka P, Osher S. An improved level set method for incompressible two-phase flows. *Computers and Fluids* 1998; **27**:663–680.
23. Juric D, Tryggvason G. Computations of boiling flows. *International Journal of Multiphase Flow* 1998; **24**: 387–410.
24. Torres DJ, Brackbill JU. The point-set method: front-tracking without connectivity. *Journal of Computational Physics* 2000; **165**:620–644.
25. Monaghan J. Particle methods for hydrodynamics. *Computer Physics Report* 1985; **3**:71–124.
26. Zalesak S. Fully multidimensional flux-corrected transport algorithms for fluids. *Journal of Computational Physics* 1979; **31**:335–362.
27. Bell JB, Colella P, Glaz HM. A second-order projection method for two-phase flow consisting of separate compressible and incompressible regions. *Journal of Computational Physics* 1989; **85**:257–283.
28. Smolarkiewicz PK. The multi-dimensional crowley advection scheme. *Monthly Weather Review* 1982; **110**: 1968–1983.
29. Renardy Y, Renardy M. PROST: a parabolic reconstruction of surface tension for the volume-of-fluid method. *Journal of Computational Physics* 2002; **183**:400–421.
30. Esmaeeli A, Tryggvason G. Direct numerical simulation of bubbly flow. Part 1. Low Reynolds number array. *Journal of Fluid Mechanics* 1998; **377**:313–345.

DOI: 10.1002/fld