Journal of Computational Physics 228 (2009) 5819–5837

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

An improved particle correction procedure for the particle level set method

Zhaoyuan Wang, Jianming Yang *, Frederick Stern

IIHR – Hydroscience and Engineering, University of Iowa, Iowa City, IA 52242, USA

ARTICLE INFO

Article history:
Received 22 April 2008
Received in revised form 13 April 2009
Accepted 30 April 2009
Available online 18 May 2009

PACS:
02.70.−c
47.55.−t

Keywords:
Particle level set method
Level set correction
Particle reseeding
Two-phase interfacial flows

ABSTRACT

The particle level set method [D. Enright, R. Fedkiw, J. Ferziger, I. Mitchell, A hybrid particle level set method for improved interface capturing, J. Comput. Phys. 183 (2002) 83–116.] can substantially improve the mass conservation property of the level set method by using Lagrangian marker particles to correct the level set function in the under-resolved regions. In this study, the limitations of the particle level set method due to the errors introduced in the particle correction process are analyzed, and an improved particle correction procedure is developed based on a new interface reconstruction scheme. Moreover, the zero level set is "anchored" as the level set functions are reinitialized; hence the additional particle correction after the level set reinitialization is avoided. With this new scheme, a well-defined zero level set can be obtained and the disturbances to the interface are significantly reduced. Consequently, the particle reseeding operation will barely result in the loss of interface characteristics and can be applied as frequently as necessary. To demonstrate the accuracy and robustness of the proposed method, two extreme particle reseeding strategies, one without reseeding and the other with reseeding every time step, are applied in several benchmark advection tests and the results are compared with each other. Three interfacial flow cases, a 2D surface tension driven oscillating droplet, a 2D gas bubble rising in a quiescent liquid, and a 3D drop impact onto a liquid pool are simulated to illustrate the advantages of the current method over the level set and the original particle level set methods with regard to the smoothness of geometric properties and mass conservation in real physical applications.

1. Introduction

The level set method, pioneered by Osher and Sethian [13], has been widely used for capturing complicated evolving interfaces in many different applications [17,14]. Especially, it has been shown significant success to handle complex flow problems involving severe topological changes, such as breakup and coalescence with its ease and simplicity of implementation [19]. The major issue with the level set method is its susceptibility to numerical dissipation which results in a poor mass conservation property. When the level set function is advected, the sharp edges are usually smoothed out and hence mass loss occurs, which is even worse when a coarse grid is used. This problem has been addressed in numerous studies and some attempts have been made to remedy it. Among them is the particle level set method by Enright et al. [2], in which the massless marker particles seeded near the interface are used to correct the level set function in the under-resolved regions. This particle correction procedure will take place when a particle on one side is advected to the opposite side of the interface (as defined by the grid-based level set function) by a distance of its radius (defined as "escaped" in [2]). The particle level set

* Corresponding author. Tel.: +1 319 335 5749; fax: +1 319 335 5238.
E-mail address: jianming-yang@uiowa.edu (J. Yang).

0021-9991/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2009.04.045

method has been shown remarkable improvement in conserving mass and retaining the fine features of the interface through a number of benchmark tests for both two- and three-dimensional (2D and 3D) problems. Its implementation is relatively easy as compared to the coupled level set and volume-of-fluid (CLSVOF) method [20,26] where complicated geometric procedures are involved. Due to its desirable visual effects at simulating realistic fluids, the particle level set method has been adopted in computer graphics and physics-based animations [4]. Its applications in free surface flows and two-phase interfacial flows can be found in [5,7], respectively. An application of the particle level set method in the simulation of the material response to impact and shocks/detonations has also been reported in [24].

However, one of the limitations of the particle level set method is that the accuracy of the simulations is sensitive to the particle reseeding strategies which vary for different problems [2]. Particle reseeding is an operation in which particles are dynamically added and deleted in the areas with deformed interfaces. In the advection benchmark tests conducted by Enright et al. [3], very good results are obtained without the reseeding operations. In [5], it was recommended to reseed particles every 20 time steps for the free surface flow problems. From a user's perspective, particle reseeding strategies relying on ad hoc and problem-dependent adjustment will affect the versatility and robustness of the particle level set method. Another limitation of the particle level set method is that for some complex interfacial flow problems, the interface represented by the corrected level set function is not smooth and numerical fluctuations in the calculations of the geometric properties (i.e., normal and curvature) may also occur. In [5], an extensive particle deletion strategy (removing particles escaped more than 1.5 times their radii from the interface) was suggested in order to maintain a smooth interface in the simulations of free surface flows. But this strategy may lead to mass loss, since the escaped particles contain important characteristic information lost in the level set.

The two limitations of the particle level set method mentioned above are primarily caused by the errors introduced in the particle correction process, which will be analyzed in detail in Section 2.3.1 and a brief summary is given here. According to the particle correction procedure in [2], an escaped particle is usually used to correct the level set on one side of the interface, whereas the level set on the other side can hardly be corrected. Also, the corrected level set function cannot be guaranteed to be a distance function when an escaped particle is far off the normal direction of the interface at the grid point. After this particle correction procedure, the zero level set in the under-resolved regions becomes ill-defined since the position of the interface is determined by the level set values at the two sets of closest grid points across the interface. In [2,3], the particle correction procedure was applied both after the level set advection and reinitialization steps. The particle correction followed the level set reinitialization step shifts the zero level set locally, which will cause the failure of the level set function across the interface to be the distance to the corrected interface. As a result, the accuracy of the geometric properties, such as normal and curvature, computed near the interface will be impaired. These inaccurate geometric properties together with the disturbed zero level set will contaminate the simulations of complex interfacial flows. Due to the ill-defined zero level set, particles added in the middle of the computations will no longer contain the correct characteristic information as those originally seeded, and they cannot be used to accurately fix the grid-based level set function. This explains why the particles seeded in the very beginning of the numerical tests play a substantial role in the original method and frequent particle reseeding was not suggested [2,12].

In this study, an improved particle correction procedure is developed to eliminate the above-mentioned two limitations and provide a well-defined zero level set. In contrast to the original procedure, the level set at both sides of the interface will be corrected by the same escaped particle; and the normal orientation of the interface will be used to determine whether the correction should be performed or not. The zero level set is anchored when solving the reinitialization equation; consequently, the second particle correction step used in the original method is not required anymore and a significant portion of CPU time can be saved. Furthermore, the particle reseeding operation will barely result in the loss of interface characteristic information and can be applied as frequently as necessary, since accurate interface information from the grid-based level set function can be retained with the proposed procedure. The performance of the current particle level set method will be evaluated through a series of benchmark advection tests with prescribed velocity fields. Two extreme particle reseeding strategies, one without reseeding and the other with reseeding every time step, will be applied to show the accuracy and robustness of the current method and to examine the effects of reseeding operations. Then a 2D surface tension driven oscillating droplet, a 2D gas bubble rising in a quiescent liquid, and a 3D drop impact onto a liquid pool will be simulated in conjunction with CFDShip-Iowa version 6, a sharp interface Cartesian grid solver for two-phase incompressible flows recently developed at IIHR [27,28]. These cases will be used to further demonstrate the applicability of the new method to complicated two-phase interfacial flows with improved mass conservation, enhanced smoothness properties, and desirable interface features over the level set and the original particle level set methods.

2. Numerical method

2.1. Level set method

The interface Γ is defined as the zero level set of a signed distance function, ϕ, or the level set function. The position of the interface can be tracked by solving the level set evolution equation [13]

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = 0. \tag{1}$$

To keep ϕ as a signed distance function in the course of evolution, the level set function is reinitialized by iterating the following equation [19]:

$$\frac{\partial \phi}{\partial \tau} + S(\phi_o)(|\nabla \phi| - 1) = 0, \tag{2}$$

where τ is the pseudo time for the iteration and $S(\phi_o)$ is the numerically smeared-out sign function

$$S(\phi_o) = \frac{\phi_o}{\sqrt{\phi_o^2 + (\Delta h)^2}}, \tag{3}$$

with ϕ_o the initial values of ϕ and Δh a small amount, e.g., the grid cell size, to smear out the sign function. The reinitialization equation can be rewritten as

$$\frac{\partial \phi}{\partial \tau} + \mathbf{u}_{\text{ext}} \cdot \nabla \phi - S(\phi_o) = 0, \tag{4}$$

where the extension velocity is

$$\mathbf{u}_{\text{ext}} = S(\phi_o)\frac{\nabla \phi}{|\nabla \phi|}. \tag{5}$$

Then the advection term in Eq. (4) can be evaluated the same as that in the evolution equation.

Since the level set function is a signed distance function, the unit normal vector, $\mathbf{n}$, and the local curvature, κ, of the interface can be readily calculated by applying standard finite difference schemes as discussed in [2] to the level set function

$$\mathbf{n} = \frac{\nabla \phi}{|\nabla \phi|} \tag{6}$$

and

$$\kappa = \nabla \cdot \mathbf{n} = \nabla \cdot \frac{\nabla \phi}{|\nabla \phi|}, \tag{7}$$

respectively.

The volume (area for a 2D case) enclosed by the zero level set isosurface can be integrated as follows:

$$V = \int H(\phi) d\Omega, \tag{8}$$

where Ω is the domain and H is the Heaviside function defined by

$$H(\phi) = \begin{cases} 0, & \text{if } \phi > 0, \\ 1, & \text{if } \phi \leqslant 0. \end{cases} \tag{9}$$

A second-order piecewise linear interface calculation (PLIC) scheme [16] is used to reconstruct the interface and calculate the partial volume $\Delta V = F\Delta x\Delta y\Delta z$ in each cell with F the volume fraction function. Then the total volume can be readily obtained from $V = \sum \Delta V$.

The accuracy of the interface location is evaluated similarly to [2] by using the L_1 norm defined by

$$\frac{1}{V}\int |H(\phi_{\text{expected}}) - H(\phi_{\text{computed}})| d\Omega = \frac{1}{V}\sum |F_{\text{expected}} - F_{\text{computed}}|\Delta x\Delta y\Delta z. \tag{10}$$

Notice that instead of area (volume for a 3D case), the length of the expected interface was used to calculate the L_1 norm in [2].

A third-order TVD Runge–Kutta scheme [18] for time advancement and a fifth-order Hamilton–Jacobi WENO scheme [9] for spatial discretization are used to solve the level set evolution and reinitialization equations within a narrow band [15].

2.2. Particle level set method

In the particle level set method [2,3], massless marker particles are utilized to correct the level set function in the under-resolved areas of the interface in order to preserve mass. Two sets (positive and negative) of particles are randomly placed within a band across the interface. A given number (N_p) of particles of each sign are placed in each cell, for instance, 16 (64) particles in 2D (3D) were used in [2] and this study. An "attraction" technique is used to relocate the particles to the corresponding side (of the same sign) of the interface. The above process is the so-called particle seeding operation. Each particle stores its position and radius, which is used to perform error correction to the level set function. The radius of the particles is given by

$$r_p = \begin{cases} r_{\max}, & \text{if } s_p\phi(\mathbf{x}_p) > r_{\max}, \\ s_p\phi(\mathbf{x}_p), & \text{if } r_{\min} \leqslant s_\phi(\mathbf{x}_p) \leqslant r_{\max}, \\ r_{\min}, & \text{if } s_p\phi(\mathbf{x}_p) < r_{\min}, \end{cases} \tag{11}$$

where s_p is the sign of the particle, set to +1 if $\phi(\mathbf{x}_p) > 0$ and -1 if $\phi(\mathbf{x}_p) < 0, r_{\min} = 0.1\Delta h$ and $r_{\max} = 0.5\Delta h$ with $\Delta h = \min(\Delta x, \Delta y, \Delta z)$.

In each time step, the level set function is advected for a whole Runge–Kutta cycle first, then the positions of the particles are updated using the third-order TVD Runge–Kutta scheme. In the well-resolved regions, the level set solution is adequately accurate and the particles just follow the motion of the interface as defined by the grid-based level set function. But in the under-resolved regions, particles may drift across the interface due to excessive regularization of the level set solution. When a particle escapes the interface by more than its radius, it will be used to perform error correction on the interface. The level set value given by an escaped particle with a radius r_p and a position vector $\mathbf{x}_p$ is defined as:

$$\phi_p(\mathbf{x}) = s_p(r_p - |\mathbf{x} - \mathbf{x}_p|). \tag{12}$$

Error correction is performed using the grid-based ϕ^+ and ϕ^-, which represent the positive and negative level set regions (with reduced errors) generated by the escaped positive and negative particles, respectively. For each positive escaped particle, ϕ_p is found for each corner of the cell that contains the particle. The value for each corner is then set to

$$\phi^+ = \max(\phi_p, \phi^+). \tag{13}$$

For each negative escaped particle, ϕ_p is similarly defined for each corner of the cell that contains the particle. The value for each corner is then set to

$$\phi^- = \min(\phi_p, \phi^-). \tag{14}$$

Note that both ϕ^+ and ϕ^- in the above two equations are initialized with ϕ.

The level set function is then reconstructed using ϕ^+ and ϕ^- by choosing the value with minimum magnitude at each grid node

$$\phi = \begin{cases} \phi^+, & \text{if } |\phi^+| \leqslant |\phi^-|, \\ \phi^-, & \text{if } |\phi^+| > |\phi^-|. \end{cases} \tag{15}$$

Error corrections are performed both after the level set advection and reinitialization.

In summary, the procedure of the operations in the original method is: advect the level set function and update the positions of the particles, correct level set function using the escaped particles, reinitialize the level set function, correct the level set function using the particles again, and finally adjust the particle radii. When an interface involves severe deformations, some regions may lack a sufficient number of particles whereas a large amount of particles may pile up in some other regions. It is necessary to perform particle reseeding operations, i.e., periodically readapt particle distributions near the interface. The interested reader is referred to [2,3] for more detailed description of the particle level set method.

2.3. Particle correction procedure

2.3.1. Original method

As shown in Fig. 1, a positive particle is drifted (escaped) from the positive side to the negative side of the interface. The level set values stored at the four (eight for a 3D case) corners of the computational cell containing the escaped particle should be corrected. At points $(i, j+1)$ and $(i+1, j)$ in Fig. 1(a), according to Eq. (12),

$$\phi_p(i, j+1) = -|\vec{CD}| \tag{16}$$

and

$$\phi_p(i+1, j) = -|\vec{AB}|. \tag{17}$$

Each ϕ_p is compared against the current value of ϕ^+ and the maximum is assigned to ϕ^+, i.e., Eq. (13). Hence the level set values corrected by the escaped positive particle at points $(i, j+1)$ and $(i+1, j)$ are:

$$\phi^+(i, j+1) = -|\vec{CD}| \tag{18}$$

and

$$\phi^+(i+1, j) = \phi(i+1, j). \tag{19}$$

The above two equations indicate that the level set value at point $(i, j+1)$ has been corrected, whereas point $(i+1, j)$ still keeps its old level set value.

For an escaped particle, the distance from the cell corner to the particle center, i.e., $|\mathbf{x} - \mathbf{x}_p|$, is usually greater than its radius r_p, therefore, the level set value predicted by an escaped positive particle usually takes a minus sign. As a result, the level

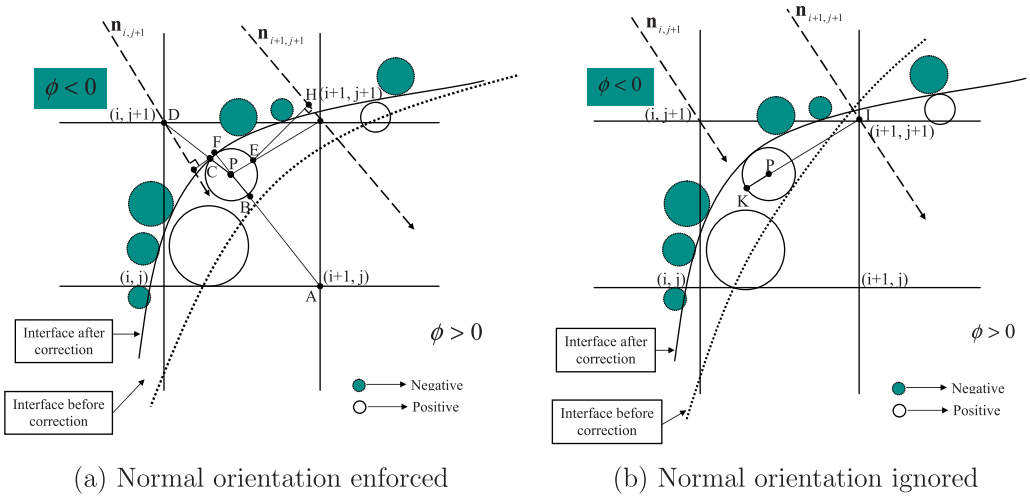


Fig. 1. Particle correction for an escaped positive particle.

set values in the positive side can hardly be corrected by the escaped positive particles, and vice versa. Since the interface represented by the zero level set contour is determined by the level set values at both sides of the interface, the interface can not be accurately located based on the above particle correction procedure. This inaccuracy in the interface representation will be transferred to the particles newly added in the reseeding process. Therefore, frequent reseeding does not improve but decrease the accuracy of the interface representation in the under-resolved regions. Some other problems as addressed in the introduction section also occur due to the inaccurate particle correction.

2.3.2. Improved particle correction procedure

Herein, an improved particle correction procedure based on the escaped particles is used to re-distance the level set function across the interface. The initialization and update of the particles will be the same as outlined in the previous section. To accurately correct the level set function using the escaped particles, the level set value defined by an escaped particle is modified, e.g., for an escaped positive particle P,

$$\phi_p(\mathbf{x}) = \begin{cases} s_p\ (r_p - |\mathbf{x} - \mathbf{x}_p|), & \text{if } \phi(\mathbf{x}) \leqslant 0, \\ s_p\ (r_p + |\mathbf{x} - \mathbf{x}_p|), & \text{if } \phi(\mathbf{x}) > 0. \end{cases} \tag{20}$$

Similarly, for an escaped negative particle P,

$$\phi_p(\mathbf{x}) = \begin{cases} s_p\ (r_p - |\mathbf{x} - \mathbf{x}_p|), & \text{if } \phi(\mathbf{x}) > 0, \\ s_p\ (r_p + |\mathbf{x} - \mathbf{x}_p|), & \text{if } \phi(\mathbf{x}) \leqslant 0. \end{cases} \tag{21}$$

This modification guarantees that the level set values on both sides of the interface can be corrected. As the example shown in Fig. 1(a), at points $(i, j+1)$ and $(i+1, j)$, the predicted level set values by the escaped particle P are

$$\phi_p(i, j+1) = -|\vec{CD}| \tag{22}$$

and

$$\phi_p(i+1, j) = |\vec{AF}|, \tag{23}$$

respectively. Therefore, in contrast to Eqs. (18), (19), the level set values corrected by the escaped positive particle according to Eq. (13) at points $(i, j+1)$ and $(i+1, j)$ will be

$$\phi^+(i, j+1) = -|\vec{CD}| \tag{24}$$

and

$$\phi^+(i+1, j) = |\vec{AF}|, \tag{25}$$

respectively.

Note that the level set values calculated via the two equations given above are the distance from the cell corner to the particle surface, which does not guarantee the corrected level set function to be a distance function. For example, as shown in Fig. 1(b), the corrected level set value at point $(i+1, j+1)$ is

$$\phi^+(i+1, j+1) = |\vec{IK}|. \tag{26}$$

However, this corrected level set value is larger than the distance to the interface which is "virtually" reconstructed by the escaped particles. Therefore, the level set value stored at this point should not be corrected by the escaped particle shown in the figure. Otherwise, interface disturbance can be induced and the smoothness of the interface will be damaged. This is especially serious for the problems with complex flow patterns. The level set values stored in the four corners should be selectively corrected by using the local geometrical information of the interface. If an escaped particle deviates too far away from the normal direction of a corner, this escaped particle will not be used to correct the level set value stored at that corner. In practice, this can be determined by the angle between the normal direction and the line from the corner to the center of the escaped particle. As shown in Fig. 1(a), the level set values stored at points $(i,j+1)$ and $(i+1,j)$ are corrected by the escaped particle, whereas points (i,j) and $(i+1,j+1)$ will keep their old level set values. In the present study, an alternative method is used to determine whether particle correction should be performed or not at each cell corner. The point on the particle surface intercepted by the line from the cell corner to the particle center is determined first, and then is projected to the line through the corner in the normal direction. The coordinate of the projection point on the normal is given by

$$\mathbf{x}' = \mathbf{x} + \gamma \,\mathrm{proj}_{\mathbf{n}}(\mathbf{x}_p - \mathbf{x}), \tag{27}$$

where $\mathbf{x}$ is the coordinate of the cell corner, $\mathbf{n}$ is the normal defined by the grid-based level set function in Eq. (6), and the coefficient γ is defined as

$$\gamma = 1 - \frac{r_p}{|\mathbf{x}_p - \mathbf{x}|}. \tag{28}$$

If the projection point $\mathbf{x}'$ falls inside the cell, the level set value at the corresponding corner will be corrected; otherwise, correction is not applied. For example, as shown in Fig. 1(a), the projection of interception point C on the normal is inside the cell, the level set value at point $(i,j+1)$ will be corrected by the escaped particle. As for point $(i+1,j+1)$, the projection of point E on the normal is outside of the cell, the level set value will not be corrected. Although the method discussed above is simple, very good results have been obtained as shown in the later sections. Using the angle (between the normal and the line from the corner to the escaped particle center) as the criterion for the particle correction is also possible, but not pursued in this study.

With this new procedure, the disturbances due to the particle correction can be reduced and more accurate interface representation can be obtained. The choice of particle reseeding strategies becomes less critical to the performance of the whole method. In this study, particles are initially placed within a narrower band of $\pm 1.5\Delta h$ near the interface rather than a band of $\pm 3\Delta h$ in the original method. The extreme strategy of reseeding every time step is used for all cases unless otherwise declared. In each time step, particles drifting too far away from the interface (e.g., $1.5\Delta h$) are deleted. Then, particles are randomly added or deleted to satisfy the criterion of maximum/minimum (e.g., $1.5/0.5\, N_p$) number of particles in a single cell. In addition, the interface is "anchored" when solving the reinitialization equation. The idea of anchoring the zero level set was proposed in [21] and a slightly different implementation is used herein. Basically, to anchor the interface, the level set value in a cell with a differently signed neighboring cell will be fixed during the reinitialization iterations. This eliminates the "drift" of the zero level set. Therefore, the level set functions are only corrected after the advection process and additional correction after the reinitialization is not needed. This not only reduces the computational cost but also ensures the level set function to be a distance function, avoiding the disturbance otherwise caused by the second correction in the original method.

3. Advection tests

The performance of the current particle level set method is evaluated through several benchmark advection tests with prescribed velocity fields. The parameters chosen in these tests are the same as those used by Enright et al. [3]. As mentioned previously, two extreme particle reseeding strategies, one without reseeding and the other with reseeding every time step, will be applied to examine the effects of reseeding operations.

3.1. Zalesak's disk

A slotted disk with a radius of 0.15 and slot width of 0.05 is initially located at $(0.50, 0.75)$ on a 1.00×1.00 computational domain. This problem, referred to as the Zalesak's problem [29], is often used for the interface tracking/capturing scheme test. The prescribed velocity field is given as

$$\begin{aligned} u &= (\pi/3.14)(0.50 - y), \\ v &= (\pi/3.14)(x - 0.50), \end{aligned} \tag{29}$$

with the axis of rotation centered at $(0.50, 0.50)$.

The results for the cases with and without reseeding procedures are compared with the initial shape after two rotations in Fig. 2. The corresponding particle distributions around the interface of the slotted disk after two rotations are shown in Fig. 3.

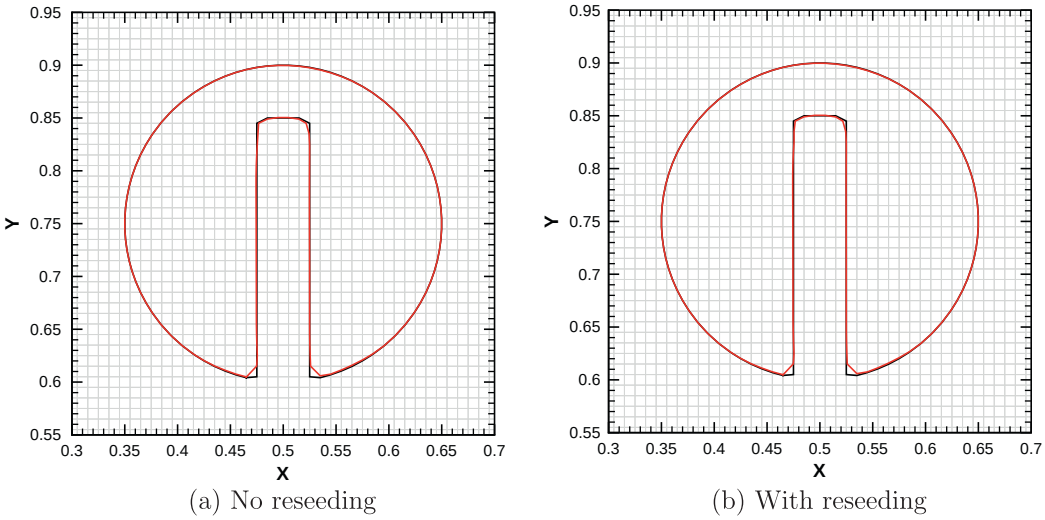


Fig. 2. Comparison of the theory (black) and particle level set solutions (red) on a 100×100 grid for the rotating slotted disk after two rotations. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

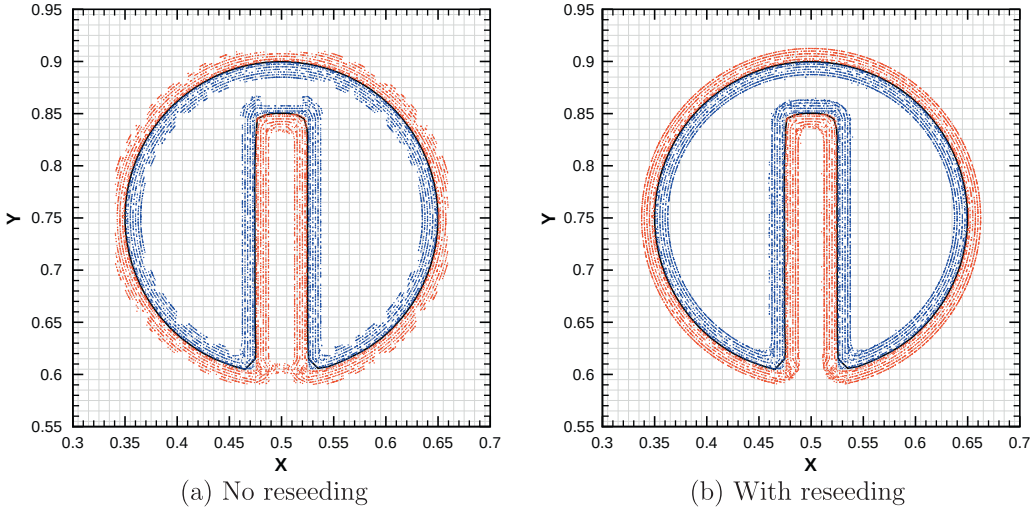


Fig. 3. Particle positions on a 100×100 grid for the rotating slotted disk after two rotations.

It can be seen from the figures that the computed results for both cases match the initial configuration very well. The sharp edges of the slot are well preserved with negligible mass loss. The L_1 errors and corresponding orders of the accuracy for both cases on different grids are given in Table 1. As shown in the table, the results from both cases are very close, and those obtained without the reseeding operations are slightly better. It should be noted that for the rigid body rotation, as in this test, there is no relative movement among the particles. All the initially prescribed particles can always return back to their original locations after every rotation except a small numerical error accumulation of order $O(\Delta t^3)$ from the third-order TVD Runge–Kutta scheme. Therefore, it is not surprising that better results can be obtained without reseeding operations in this test.

In order to investigate the effects of the anchored reinitialization on the accuracy of the proposed method, three different combinations are examined, namely, one correction followed by the anchored reinitialization (C-AR), one correction followed by the non-anchored reinitialization (C-NAR), and one correction followed by the non-anchored reinitialization and then the second correction (C-NAR-C). The non-anchored scheme refers to the classic reinitialization scheme developed in [19]. For the last case, one pass of particle correction is performed after solving the level set advection equation and the second pass is applied following the level set reinitialization as used in the original particle level set method. For all three cases, the new particle correction procedure presented in Section 2.3.2 is used and no reseeding operation is applied in order to simplify the analysis of the results. As shown in Table 2, in the cases with the non-anchored reinitialization, the second

Table 1
Slotted disk: L_1 error.

Grid	No reseeding				With reseeding			
	One rotation		Two rotations		One rotation		Two rotations	
	L_1 error	Order	L_1 error	Order	L_1 error	Order	L_1 error	Order
50	0.030	N/A	0.033	N/A	0.032	N/A	0.036	N/A
100	0.005	2.35	0.007	2.16	0.008	2.00	0.009	2.00
200	0.002	1.70	0.003	1.62	0.003	1.70	0.004	1.58

Table 2
Slotted disk: area loss (grid: 100×100).

	One rotation		Two rotations	
Method	Area	Area loss %	Area	Area loss %
Initial value	582.81		582.81	
C-AR	582.32	0.08	582.23	0.10
C-NAR	581.76	0.18	581.43	0.24
C-NAR-C	582.18	0.11	581.85	0.16

particle correction does improve the mass conservation property of the particle level set method. However, the anchored reinitialization scheme can give further improvements even without using the second particle correction. The same test case was performed in [2] and area losses of the slotted disk are 0.31% and 0.72% for one and two rotations, respectively. It is evident that the mass loss after two rotations has more than doubled in their simulation. Although these percentages of mass loss cannot be directly compared with those in Table 2 due to the different method used here for the area loss calculation, all three cases in this study show very small increases of mass loss after the second rotation even with the non-anchored reinitialization. An additional test using the original particle correction procedure and anchored reinitialization (OC-AR) is also performed. The areas (area loss percentages) at one and two rotations are 581.77 (0.18%) and 581.57 (0.21%), respectively. The results are slightly better than those from C-NAR, but not as good as those from C-NAR-C or C-AR.

This test demonstrates that the new particle correction procedure is essential to the success of the whole method. The anchoring process reduces the computational cost and further improves the accuracy.

3.2. Single vortex flow

A circle evolving in a shearing flow [1] is another challenging test for advection schemes, which involves severe topological changes. The circle is stretched and torn in this vortex flow where very thin filaments on the scale of the mesh size can be produced. A prescribed shearing flow is given by

$$\begin{aligned} u &= -\sin^2(\pi x)\sin(2\pi y)\cos(\pi t/T), \\ v &= \sin^2(\pi y)\sin(2\pi x)\cos(\pi t/T), \end{aligned} \tag{30}$$

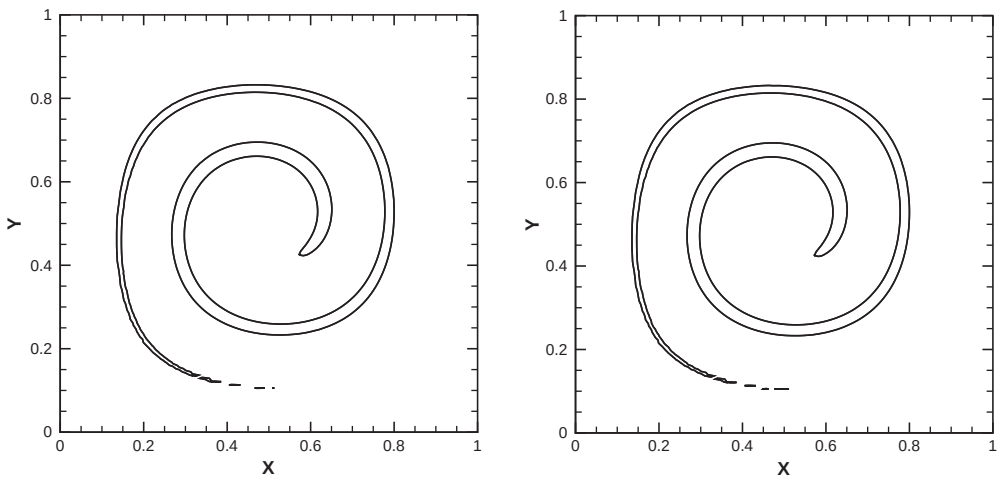
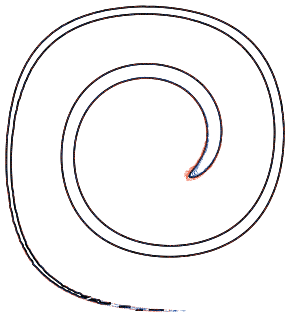
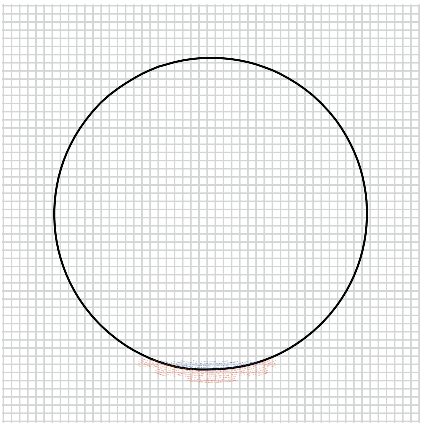


Fig. 4. The particle level set solutions on a 128×128 grid for the single vortex flow at $t = 4$.

where t is time, T is the time at which the flow returns back to its initial shape. In this case $T = 8$ is used. A circle with a radius of 0.15 is prescribed at $(0.5, 0.75)$ on a computational domain of 1.0×1.0. As in the previous test, two cases without and with particle reseeding procedures are considered in this test.

The results at $t = 4$ when a maximal stretching is reached are shown in Fig. 4. A thin and elongated filament is well produced on the scale of the grid spacing for both cases. Fig. 5 shows particle positions along the interface of the filament at $t = 4$. For the case without particle reseeding operations (see Fig. 5(a)), many particles are concentrated in the regions at the head and tail of the filament, whereas sporadic particles are scattered at the interface of the main body of the filament. The recovered shapes at $t = 8$ are compared with the initial circle in Fig. 6. Fig. 7 shows the particle distributions when the flow field returns back to its initial state. From these figures, it is evident that the recovered circles are very comparable to the initial shape for both cases. The area losses, the errors and the orders of the accuracy with three different grid sizes are summarized in Table 3. The results computed without and with the particle reseeding procedures are very comparable. The former case shows a little more accurate results. For this particular case, the most possible areas where mass loss can occur during the stretching process are the head and tail of the filament. As shown in Fig. 5(a), during the entire stretching process, sufficient particles always exist in these two areas even without the particle reseeding procedures. After the flow field is reversed, all the initially prescribed particles can return back to practically their initial positions except the small numerical error accumulation from the time-advancement scheme. However, in the latter case, many initially prescribed particles





are deleted during the particle reseeding process and the newly added particles contain more numerical errors. It is conceivable that the accuracy of the particle level set method will be affected by the reseeding operations and better results are presumptively expected when the reseeding process is not involved.

In order to demonstrate the enhanced smoothness properties of the new technique, the relative errors of the calculated curvature along the circumference (clockwise from the top of the circle) are estimated. The results computed using both the current and original particle level set methods without particle reseeding procedures are shown in Fig. 8. In most regions along the circle, the curvature errors calculated from the current particle level set method are less than those obtained from the original particle level set method. Notice that in the region with a circumferential angle in the range of 330°–360° (top of the recovered circle), the curvature errors of the current particle level set method are larger than those from the original method. This region is corresponding to the tail of the filament during the stretching process, which experienced the maximum amount of tear and completely relies on the correction of the particles to recover the initial shape. Although the same

set of particles are present for both cases as no reseeding process is applied, much less particles can be used for correction with the new method due to the normal orientation constraint. The lack of enough correction explains the deficient recovery in this region. Even though, the overall performance of the current method is better than the original one.

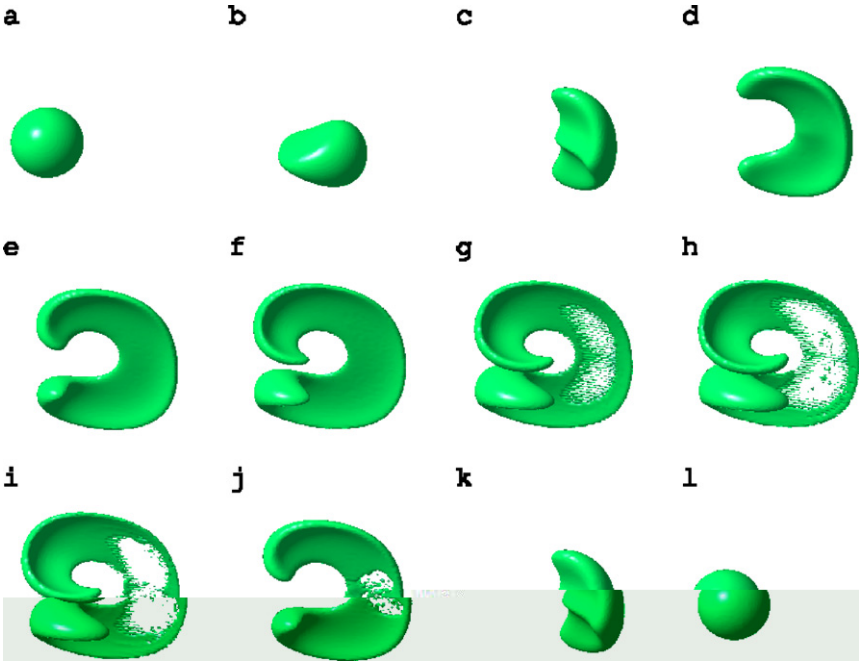


Fig. 9. 3D deformation field: No reseeding. (a) $t = 0.00$; (b) $t = 0.2$; (c) $t = 0.4$; (d) $t = 0.6$; (e) $t = 0.8$; (f) $t = 1.0$; (g) $t = 1.2$; (h) $t = 1.4$; (i) $t = 1.8$; (j) $t = 2.2$; (k) $t = 2.6$; (l) $t = 3.0$.

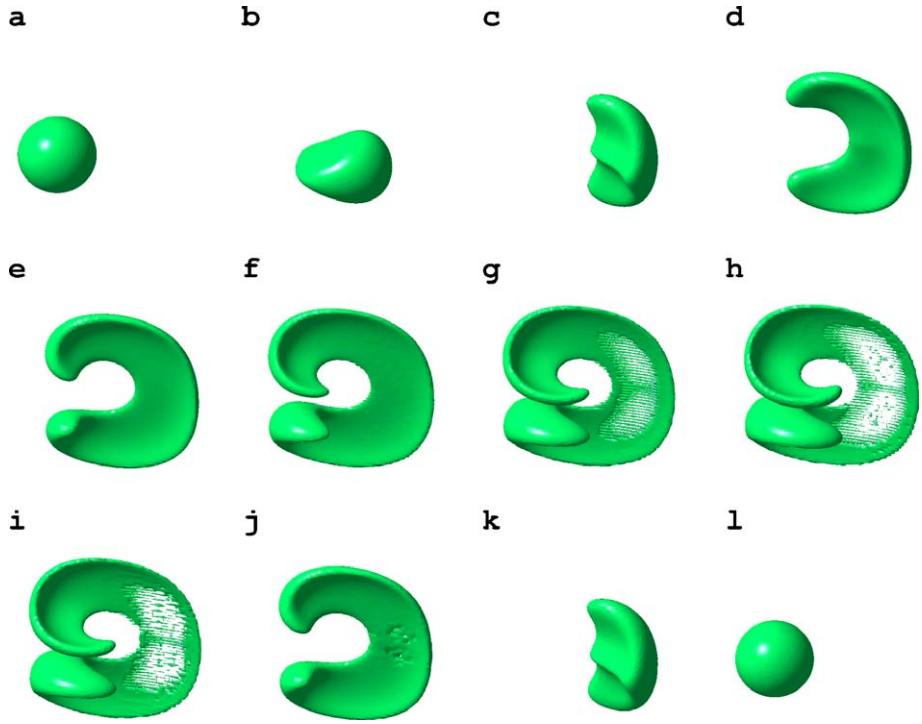


Fig. 10. 3D deformation field: with reseeding. (a) $t = 0.00$; (b) $t = 0.2$; (c) $t = 0.4$; (d) $t = 0.6$; (e) $t = 0.8$; (f) $t = 1.0$; (g) $t = 1.2$; (h) $t = 1.4$; (i) $t = 1.8$; (j) $t = 2.2$; (k) $t = 2.6$; (l) $t = 3.0$.

Table 4
3D deformation field.

	No reseeding			With reseeding		
	Volume	Volume loss %	L_1 error	Volume	Volume loss %	L_1 error
Exact	0.0142			0.0142		
Half cycle	0.0136	2.253	N/A	0.0139	2.063	N/A
Full cycle	0.0139	1.603	0.0217	0.0140	1.008	0.009

3.3. 3D deformation field

To demonstrate the ability of the method to capture 3D deformations, a 3D incompressible velocity field given in [2] which combines deformations both in the $x-y$ and $x-z$ planes is considered. The velocity field is given by

$$\begin{aligned} u &= 2\ \sin^2(\pi x)\sin(2\pi y)\sin(2\pi z)\cos(\pi t/T),\\ v &= -\sin(2\pi x)\sin^2(\pi y)\sin(2\pi z)\cos(\pi t/T),\\ w &= -\sin(2\pi x)\sin(2\pi y)\sin^2(\pi z)\cos(\pi t/T) \end{aligned} \tag{31}$$

with $T=3$. A sphere of radius 0.15 is placed within a unit computational domain at $(0.35, 0.35, 0.35)$. A uniform grid of $100\times100\times100$ is used. The sphere is stretched by two rotating vortices which initially scoop out opposite sides of the sphere and then reverse them back to the initial shape.

The time evolutions of the deformed shapes are shown in Figs. 9 and 10 for the cases without and with reseeding procedures, respectively. As shown in the figures, deformed shapes in both cases are generally similar. It is noticeable that mass loss occurs in both cases during the stretching, especially, when the interface at $t=2.2$ is compared with that of $t=0.8$, which represents the same shape as $t=2.2$ before the velocity field is reversed. In contrast to the previous two benchmark tests, particle reseeding operations in this test improve the mass conservation of the results, as evident in Fig. 10 where the mid portion of the deformed shape is better resolved than that in Fig. 9. This is also demonstrated by the data presented in Table 4. In this test, mass loss mainly occurs at the mid portion of the deformed shape where a very thin film is produced. Without particle reseeding operations, the number of particles at the mid portion of the deformed shape decreases during the stretching. Fig. 11 shows a clear illustration of the particle distributions along the slices in different coordinate directions. The number of particles is insufficient to keep the characteristic information in this region at the maximum stretching of the interface; as a result, mass loss inevitably occurs in this area. When the reseeding operations are applied, more particles are added in this area as shown in Fig. 12, and better mass conservation property is achieved. Although the reseeding procedure improves the mass conservation property, the thin film still can not be effectively resolved by the grid at maximum stretching as shown in Fig. 10. This is because the thin film is less than one grid cell wide at maximum stretching, so any attempt to accurately reconstruct this thin film with the underlying grid will be impossible. In [8], a resolution study was performed using a Lagrangian particle level set method and it was found that the disappearance of the interface at its maximum stretching can be avoided by a grid of $200\times200\times200$, which means a halved grid spacing of the current one.

In general, the particle level set method performs quite well without the particle reseeding operations in the three benchmark tests discussed above, especially for the Zalesak's disk and single vortex flow tests in which even better results have been produced. Without the particle reseeding operations, particles are advanced independently and their motions/positions are not affected by the level set function. Since the velocity fields in all these tests are prescribed and not affected by the interface, the behavior of the particles is predictable. The level set function can be very accurately corrected by these particles carrying the accurate characteristic information of the interface. However, particle reseeding is a necessity when an interface undergoes drastic topologic changes, e.g., in many complex interfacial flow problems. Therefore, the performance of the particle level set method with particle reseeding operations is more meaningful for the real physical flow problems. As demonstrated in the above tests, the results obtained with particle reseeding operations are comparable to those without particle reseeding procedures. This is because with the improved particle correction scheme, accurate interface information from the grid-based level set function can be retained. The errors induced by particle reseeding are thus greatly restricted, and frequent particle reseeding does not significantly affect overall accuracy of the particle level set method. This guarantees the accuracy of the method in real interfacial flow problems where frequent particle reseeding may be needed. However, it should be noted that it is possible to adjust the reseeding strategies in a case by case manner in order to reduce the use of the computationally expensive particle reseeding steps.

4. Two-phase interfacial flows

In this section, the particle level set method in conjunction with a flow solver is applied for two-phase interfacial flow problems. CFDShip-Iowa version 6, a sharp interface Cartesian grid solver for two-phase incompressible flows recently

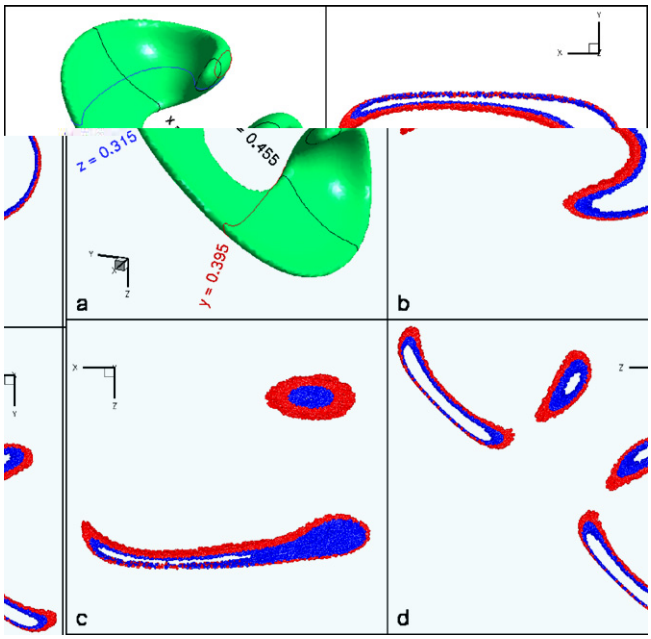


Fig. 11. Particle distribution along slices of the deformed shape at $t = 1.0$: no reseeding. (a) Deformed shape and postions of slices; (b) $z = 0.315$; (c) $y = 0.395$; (d) $x = 0.455$.

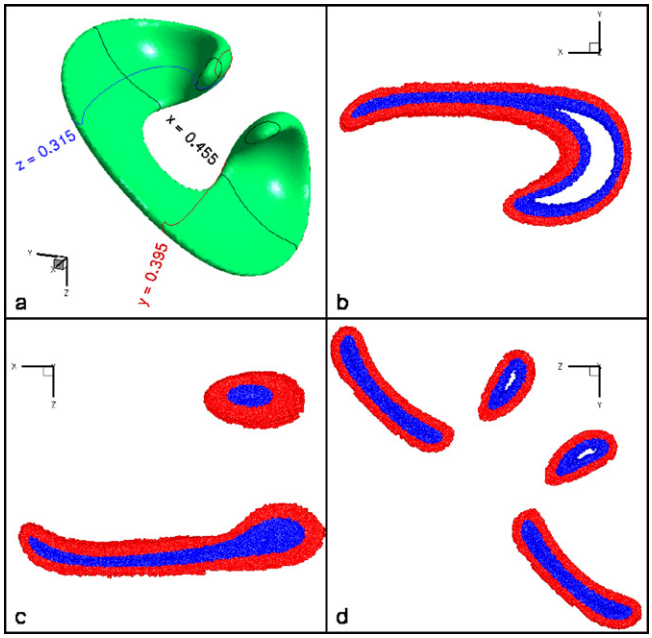


Fig. 12. Particle distribution along slices of the deformed shape at $t = 1.0$: with reseeding. (a) Deformed shape and postions of slices; (b) $z = 0.315$; (c) $y = 0.395$; (d) $x = 0.455$.

developed at IIHR, is used for the simulations. In this solver, the governing equations are discretized on a staggered Cartesian grid using the standard second-order central difference schemes and optional high order convective schemes. A semi-implicit time-advancement scheme is adopted to integrate the momentum equations with the second-order Crank–Nicolson scheme for the diagonal viscous terms and the second-order Adams–Bashforth scheme for the convective terms and other terms. A four-step fractional-step method is employed for velocity–pressure coupling. The interface is tracked by the level set method. The ghost fluid method [10,11] is adopted to handle the jump conditions across the interface, in which the

density and surface tension are treated in a sharp manner, and the viscosity is smeared by a smoothed Heaviside function [19]. Details of the numerical methods can be found in the studies by Yang and Stern [27,28].

4.1. Oscillating drop

As an example of surface tension driven flow problem, an initially non-spherical viscous drop with subsequent oscillating decay to an equilibrium static shape was simulated in [23]. This is a demanding and challenging test for the surface tension models and interface tracking methods, in which an accurate and smooth curvature field is critical. The Lagrangian-based methods usually exhibit better performance because more precise interfacial geometry can be provided when the topologic changes are small and smooth (no breakup and/or merging) as in this test [6]. The parameters of the problem are chosen the same as those in [23,6,22]. An elliptical liquid drop, initially specified by equation $x^2/9 + y^2/4 = 1$, is used for the calculations without considering gravity. The physical properties are as follows: densities $\rho_L = 1.0, \rho_G = 0.01$, viscosities $\mu_L = 0.01, \mu_G = 5 \times 10^{-5}$, and surface tension $\sigma = 1.0$. The level set, original, and current particle level set methods are employed for this test to evaluate their performances. The particle reseeding operation is applied every time step for both the original and current particle level set methods, and the second particle correction step is applied in the orginal method.

Due to the surface tension force, the elliptical drop starts to oscillate after release and finally reaches an equilibrium static state (spherical shape) by the damping effect of viscosity. The kinetic energy, $\frac{1}{2}\int \rho \mathbf{u} \cdot \mathbf{u} dV$, versus time plots are shown in Fig. 13 using the level set, original particle level set and current particle level set methods. With the current particle level set method, it is clear that the kinetic energy decays gradually and a relatively static state is reached finally. The results of the current particle level set method agree very well with the point-set method by Torres and Brackbill [23] (not shown here) in terms of oscillation amplitudes and frequency. The original particle level set method shows strong fluctuations after $t = 50$ in the kinetic energy decay and the drop fails to reach a static state. As for the level set method, very slow energy decays can be observed in the figure. The slow kinetic energy decay of the level set method is probably caused by the mass loss that occurs during the computation. Fig. 14 shows the mass versus time plots for the three different methods. Mass is lost by nearly 15% in the level set method, whereas both the original and current particle level set methods well maintain the mass during the computations. The current method gives smaller mass fluctuations than the original method in the course of simulation.

4.2. Rising bubble in a quiescent liquid

A 2D gas bubble rising in a quiescent liquid was used in [7] for studying the mass conservation properties of the particle level set method. In this test, a 2D gas bubble with a diameter of 1.0 is initially placed at $(1.25, 1.25)$ of a computational domain of 2.5×5.0. The physical parameters are chosen the same as in [7]. A relatively coarse grid of 40×80 is used here which is corresponding to the case shown in Fig. 2(a) of [7]. The computation is continued until $t = 5.0$ when the bubble rises to the height of approximately half of the domain. It is another challenging test for interface tracking methods since complex interface changes are involved.

Time sequences of the deformed bubble shape are shown in Fig. 15. The bubble, exerted by buoyancy force, rises rapidly after release. The bubble deforms severely from the initial spherical shape to the final oblate-ellipsoidal cap shape as shown in the figure. During this ascension process, very thin bubble skirt is formed behind the bubble and finally is pinched off into two small bubbles. The bubble shapes predicted using the current particle level set method are in very good agreement with

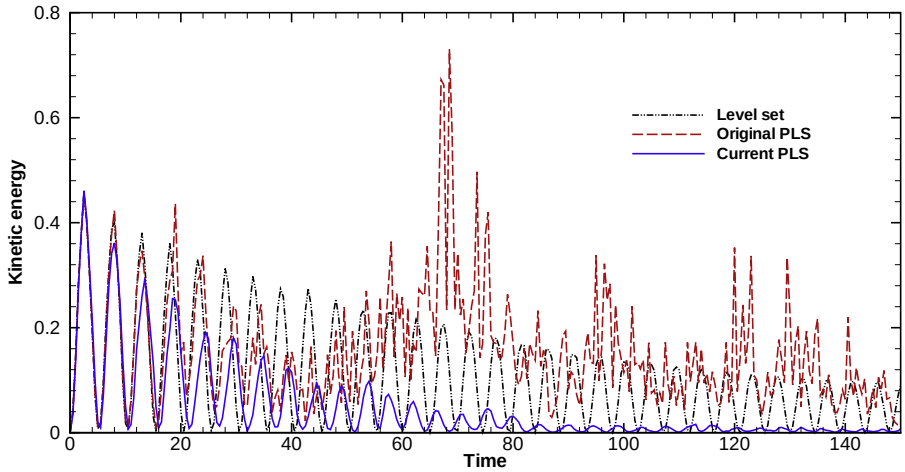


Fig. 13. Kinetic energy versus time plot for a 2D oscillating drop.

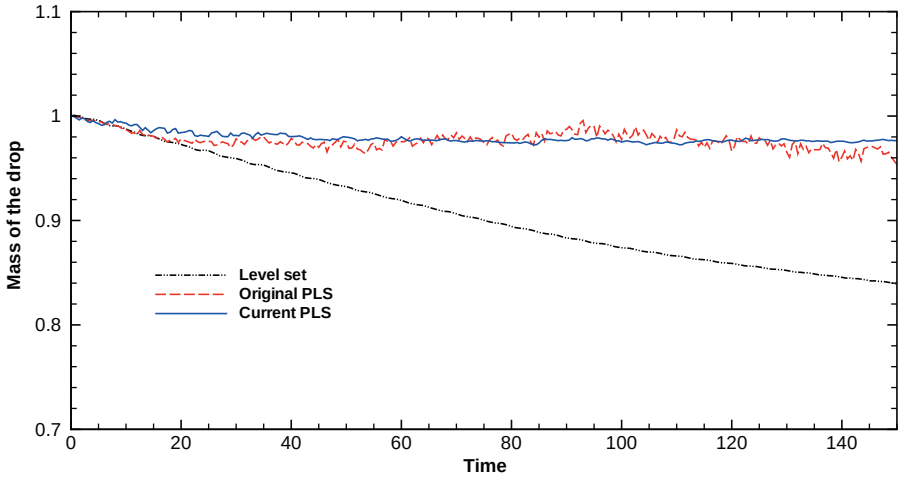


Fig. 14. Mass versus time of a 2D oscillating drop.

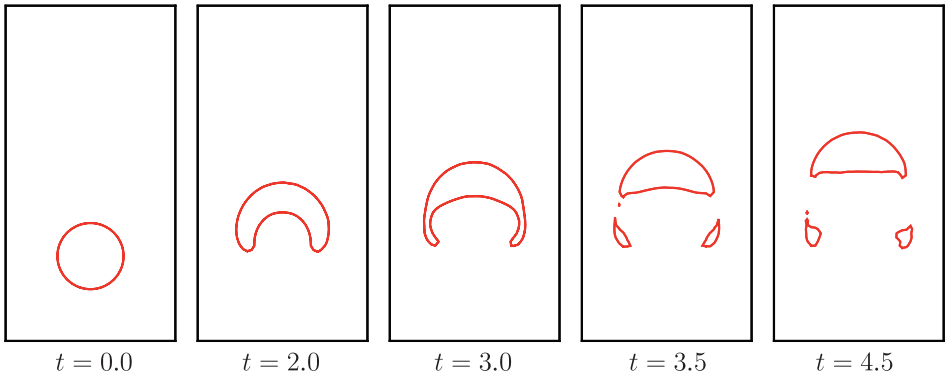


Fig. 15. Computed shapes of a 2D gas bubble rising in a quiescent liquid.

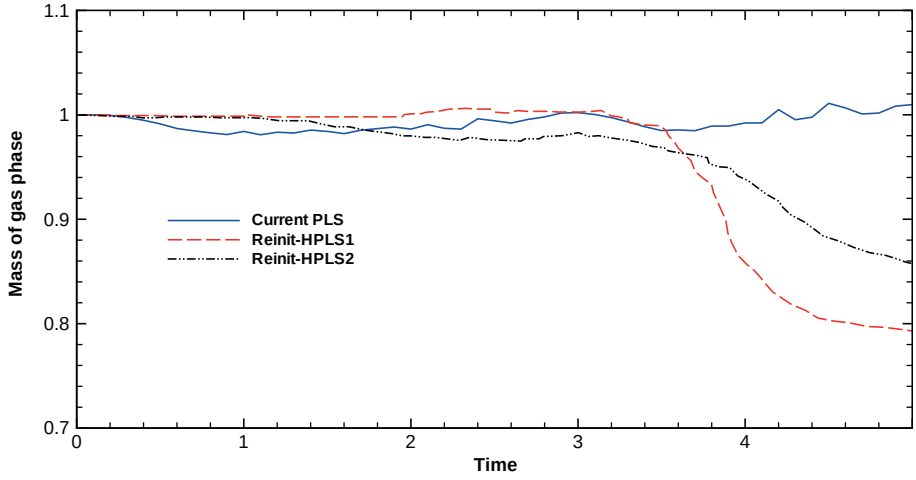


Fig. 16. Mass versus time of a 2D gas bubble rising in a liquid. Reinit-HPLS1 and Reinit-HPLS2 are results from [7].

those reported in [7] where different level set and particle level set schemes were used. Fig. 16 shows the time series of the bubble mass, in which the Reinit-HPLS1 scheme refers to the one presented in [2] and the Reinit-HPLS2 scheme refers to the same scheme without the second particle correction step [7]. With the current method, the overall mass errors during

the entire computational process are less than 2%, which are better than those results reported in [7] using either Reinit-HPLS1 or Reinit-HPLS2 scheme, especially, after the two small bubbles are pinched off.

4.3. 3D drop impact

In [5], a liquid drop impact onto the surface of a liquid pool was simulated using the particle level set method. It is a good example for interface modeling methods since it involves complicated flow deformations such as splashing, coalescence and bouncing. A similar case is adopted here to demonstrate the applicability of the new method to complicated two-phase interfacial flows and the improved mass conservation over the level set and the original particle level set methods.

In this test, a water droplet of radius $D = 1/3$ m initially positioned at $(3D, 3D, 4.5D)$ impinges upon a pool of water of $3D$ in depth. The initial velocity used here is -1 m s^{-1} which is much less than that (-5 m s^{-1}) used in [5]. It is computational expensive to use a large initial velocity, since it not only requires a large computational domain to capture the rebounding liquid jet but also needs a very small time step due to the time step constraints. Constant physical properties of air and water are chosen as: $\rho_L = 1000$ kg m^{-3}, $\rho_G = 1.226$ kg m^{-3}, $\mu_L = 1.137 \times 10^{-3}$ kg m^{-1} s^{-1}, $\mu_G = 1.78 \times 10^{-5}$ kg m^{-1} s^{-1}, $\sigma = 0.0728$ kg s^{-2}, and the gravity $g = -9.8$ m s^{-2}. The computations are carried out on a domain of $6D \times 6D \times 9.6D$. The grid used is $60 \times 60 \times 96$ and no-slip solid wall boundary conditions are specified at all the domain boundaries.

Time sequences of the drop impingement process for the level set, the original particle level set, and the present particle level set methods are given in Figs. 17–19, respectively. Note that for the case using the original particle level set method, the level set functions are corrected using the same procedure as given in Eqs. (12)–(15), and particles are reseeded every 20 time steps both after the advection and reinitialization processes [5]. As shown in the figures, after the impact of the droplet onto the water surface of the pool, a big dent on the water surface is produced first, followed by the formation of a water jet. The jet then falls back to the water surface. Generally, the major events of the impingement process are demonstrated in all the three methods.

At the early stage of the impingement, the results obtained from all three methods agree well with one another. Interface disturbances are evident in the case using the original particle level set method as shown at $t = 0.625$ s in Fig. 18, where the interface is not as smooth as those obtained from the level set and the present particle level set methods. Obvious mass loss occurs in the case using the level set method after the formation of the liquid jet, where the jet shrinks gradually and then vanishes. Better results are obtained with the original particle level set method in which the jet still exists at $t = 1.25$ s and $t = 1.375$ s as shown in Fig. 18. However, the water jet surface is not smooth and it also disappears when time reaches $t = 1.775$ s with the original method. In the present particle level set method, the jet is properly captured as it falls down to the water surface. As shown in Fig. 19, the water jet reaches its maximum height between $t = 1.25$ s and $t = 1.375$ s. The maximum jet height is about $3.6D$ which is comparable to the maximum jet height of approximately $3.75D$ in [5], even

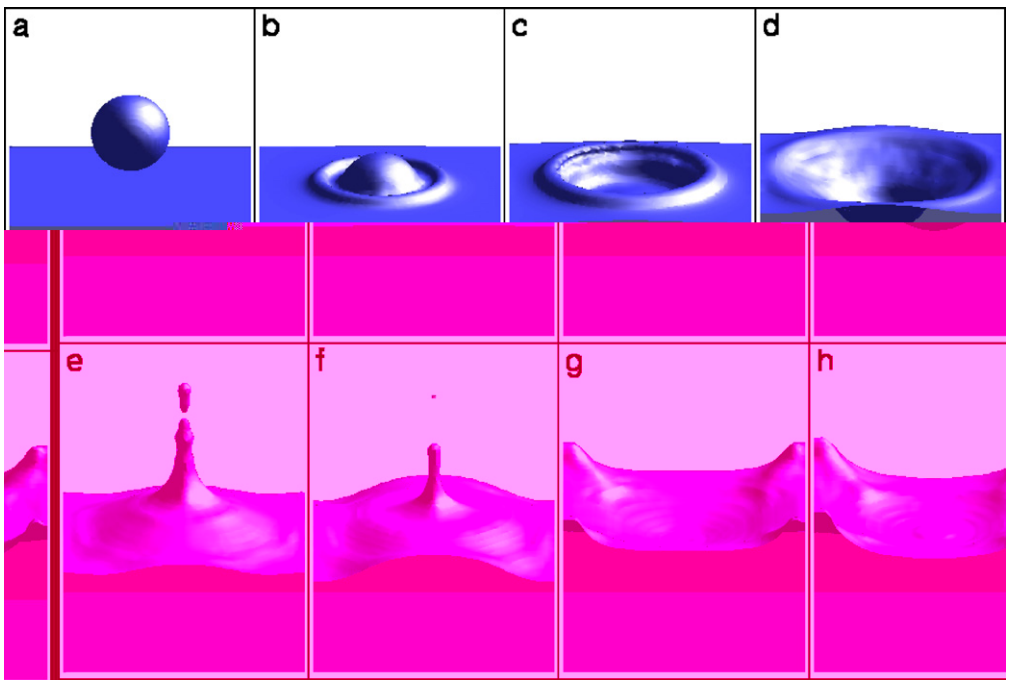


Fig. 17. 3D drop impact: level set method. (a) $t = 0.000$ s; (b) $t = 0.250$ s; (c) $t = 0.375$ s; (d) $t = 0.625$ s; (e) $t = 1.250$ s; (f) $t = 1.375$ s; (g) $t = 1.775$ s; (h) $t = 1.875$ s.

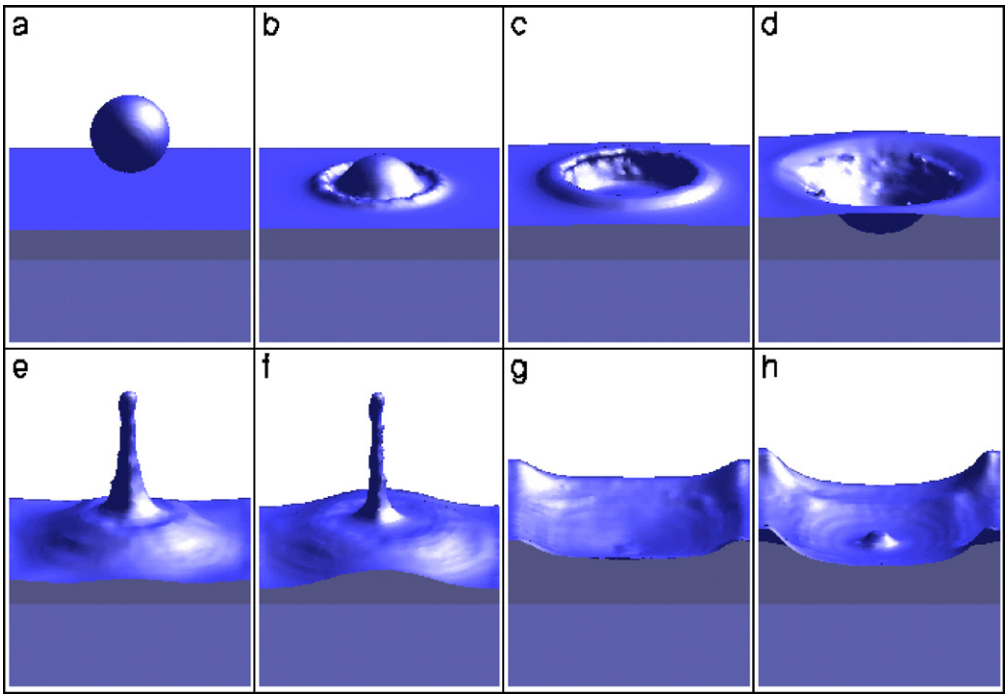


Fig. 18. 3D drop impact: original particle level set method. (a) $t = 0.000$ s; (b) $t = 0.250$ s; (c) $t = 0.375$ s; (d) $t = 0.625$ s; (e) $t = 1.250$ s; (f) $t = 1.375$ s; (g) $t = 1.775$ s; (h) $t = 1.875$ s.

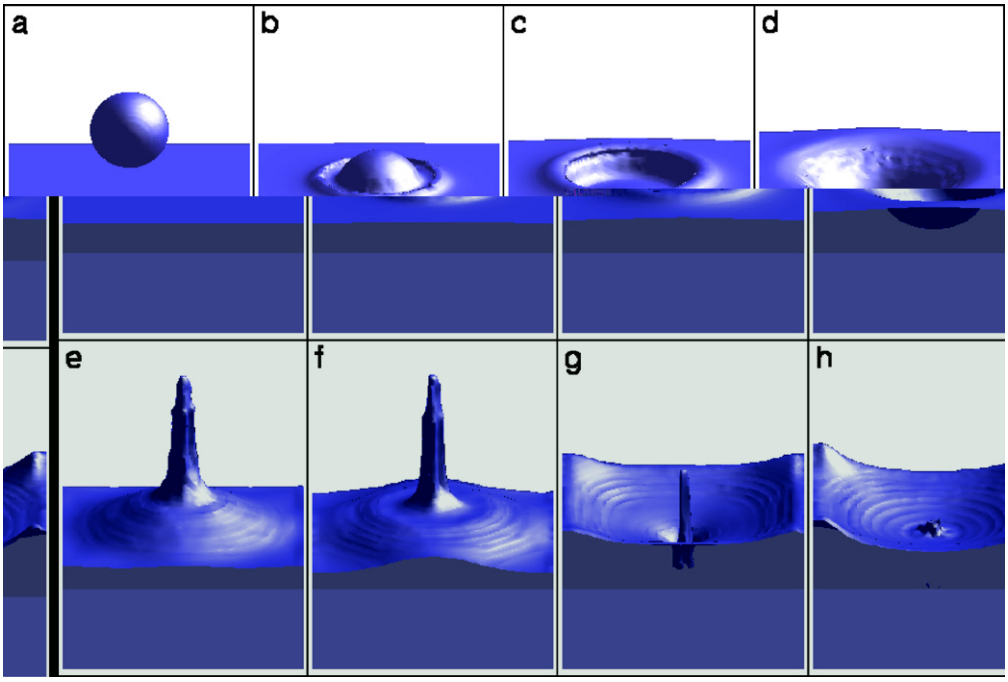


Fig. 19. 3D drop impact: current particle level set method. (a) $t = 0.000$ s; (b) $t = 0.250$ s; (c) $t = 0.375$ s; (d) $t = 0.625$ s; (e) $t = 1.250$ s; (f) $t = 1.375$ s; (g) $t = 1.775$ s; (h) $t = 1.875$ s.

though, where a much larger initial velocity was used. Moreover, water ripples are generated on the liquid surface due to the drop impact, which are more clearly indicated with the present particle level set method. As the jet falls down, some air is entrained around the jet ($t = 1.775$ s in Fig. 19). The same case was conducted using the CLSVOF method in [25], where air

bubbles below the water surface were more clearly demonstrated after the jet falls down. The computed mass errors of the liquid phase at $t = 1.875$ s for the three different methods are 0.29%, 0.14%, and 0.11%, respectively. Although mass losses are noticeable in Figs. 17 and 18, the total mass errors of the level set and original particle level set methods are not that large since the mass gains (mass losses of the gas phase) occurred in some regions offset the mass losses in other regions during the computations.

It is clear that the present particle level set method performs much better than both the level set method and the original particle level set method in mass conservation. More detailed events of the drop impact process, such as water ripples, air entrainment, are better captured with the present particle level set method. In contrast to the original particle level set method, the present particle level set method gives smoother interface with less unphysical disturbances. However, when compared with the CLSVOF method [25], the particle level set method still gives undesirable amount of mass losses, especially, for problems involving a very long computational time. But the particle level set method is still a good alternative to the CLSVOF method due to its easy and simple implementation to improve the mass conservation properties of the level set method.

5. Conclusions

The limitations of the particle level set method due to the errors introduced in the particle correction process are analyzed, and an improved particle correction procedure is developed based on a new interface reconstruction scheme. With this new scheme, the level set at both sides of the interface can be corrected by the same escaped particle; and the local geometric information of the interface is used to determine whether the correction should be performed or not according to the relative positioning of grid point, escaped particle, and "virtually" reconstructed interface. A well-defined zero level set can be established after the particle correction process. With an "anchored" zero level set in the reinitialization step, the reinitialized level set will be a distance function to a well-defined interface with recovered characteristic information from the particles. Consequently, the second particle correction step used in the original method is not required and a significant portion of CPU time can be saved. Furthermore, the particle reseeding operation will barely result in the loss of interface characteristic information and can be applied as frequently as necessary.

The performance of the current particle level set method is evaluated through a series of benchmark advection tests with prescribed velocity fields. Two extreme particle reseeding strategies, one without reseeding and the other with reseeding every time step, are applied in several benchmark advection tests and the results are compared with each other to show the accuracy and robustness of the current method and to illustrate the functionality of the particle reseeding operations. Comparison of the results shows that with the current particle correction procedure, particle reseeding can be conducted as frequently as needed (every time step in this study) without leading to unacceptable penalties in accuracy. It also shows that particle reseeding is inevitable for improving the conservation property of the level set method in some problems.

A 2D surface tension driven oscillating droplet is simulated in which an accurate and smooth curvature field is critical. The computed results are compared with those reported in literature and good agreement is obtained. A 2D gas bubble rising in a quiescent liquid is also modeled and better results are obtained than those presented in [7]. Finally, a 3D drop impact onto a water pool are calculated using the level set, original, and current particle level set methods, respectively. The current method gives remarkably fine features of the interface which are missing in level set simulation; and in contrast to the original particle level set method, it exhibits unnoticeable disturbances to the interface. These cases further demonstrate the applicability of the new method to complicated two-phase interfacial flows with improved mass conservation, enhanced smoothness properties, and desirable interface features over the level set method and the original particle level set method.

Acknowledgment

This work was sponsored by the US Office of Naval Research through research grants N00014-01-1-0073 and N00014-06-1-0420 under the administration of Dr. Patrick Purtell. The computations were performed using DoD HPC resources. The authors are grateful to the anonymous reviewers for their valuable comments and constructive suggestions.

References

[1] J.B. Bell, P. Colella, H.M. Glaz, A second-order projection method for the incompressible Navier–Stokes equations, J. Comput. Phys. 85 (1989) 257–283.
[2] D. Enright, R. Fedkiw, J. Ferziger, I. Mitchell, A hybrid particle level set method for improved interface capturing, J. Comput. Phys. 183 (2002) 83–116.
[3] D. Enright, F. Losasso, R. Fedkiw, A fast and accurate semi-Lagrangian particle level set method, Comput. Struct. 83 (2005) 479–490.
[4] D. Enright, S. Marschner, R. Fedkiw, Animation and rendering of complex water surfaces, Proc. ACM SIGGRAPH 2002 21 (2002) 736–744.
[5] D. Enright, D. Nguyen, F. Gibou, R. Fedkiw, Using the particle level set method and a second order accurate pressure boundary condition for free surface flows, in: M. Kawahashi, A. Ogut, Y. Tsuji (Eds.), Proceedings of the Fourth ASME-JSME Joint Fluids Engineering Conference FEDSM2003-45144, 2003.
[6] M.M. Francois, S.J. Cummins, E.D. Dendy, D.B. Kothe, J.M. Sicilian, M.W. Williams, A balanced-force algorithm for continuous and sharp interfacial surface tension models within a volume tracking framework, J. Comput. Phys. 213 (2006) 141–173.
[7] D. Gaudlitz, N.A. Adams, On improving mass-conservation properties of the hybrid particle-level-set method, Comput. Fluids 37 (2008) 1320–1331.
[8] S.E. Hieber, P. Koumoutsakos, A Lagrangian particle level set method, J. Comput. Phys. 210 (2005) 342–367.
[9] G.-S. Jiang, D. Peng, Weighted ENO schemes for Hamilton–Jacobi equations, SIAM J. Sci. Comput. 21 (2000) 2126–2143.
[10] M. Kang, R.P. Fedkiw, X.-D. Liu, A boundary condition capturing method for multiphase incompressible flow, J. Sci. Comput. 15 (2000) 323–360.
[11] X.D. Liu, R. Fedkiw, M. Kang, A boundary condition capturing method for Poissons equation on irregular domains, J. Comput. Phys. 154 (2000) 151–178.
[12] E. Mokberi, P. Faloutsos, A particle level set library. <http://www.magix.ucla.edu/software.html>.

[13] S. Osher, J. Sethian, Fronts propagating with curvature-dependent speed: algorithms based on Hamilton–Jacobi formulations, J. Comput. Phys. 79 (1988) 12–49.
[14] S. Osher, R. Fedkiw, Level Set Methods and Dynamic Implicit Surfaces, Springer Verlag, New York, NY, 2002.
[15] D. Peng, B. Merriman, S. Osher, H. Zhao, M. Kang, A PDE-based fast local level set method, J. Comput. Phys. 155 (1999) 410–438.
[16] R. Scardovelli, S. Zaleski, Analytical relations connecting linear interfaces and volume fractions in rectangular grids, J. Comput. Phys. 164 (2000) 228–237.
[17] J.A. Sethian, Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science, Cambridge University Press, Cambridge, 1999.
[18] C.W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, J. Comput. Phys. 77 (1988) 439–471.
[19] M. Sussman, P. Smereka, S. Osher, A level set approach for computing solutions to incompressible two-phase flow, J. Comput. Phys. 114 (1994) 146–159.
[20] M. Sussman, E.G. Puckett, A coupled level set and volume-of-fluid method for computing 3D axisymmetric incompressible two-phase flows, J. Comput. Phys. 162 (2000) 301–337.
[21] M. Sussman, E. Fatemi, An efficient interface preserving level set re-distancing algorithm and its application to interfacial incompressible fluid flow, SIAM J. Sci. Comput. 20 (1999) 1165–1191.
[22] A.Y. Tong, Z. Wang, A numerical method for capillarity-dominant free surface flows, J. Comput. Phys. 221 (2007) 506–523.
[23] D.J. Torres, J.U. Brackbill, The point-set method: front-tracking without connectivity, J. Comput. Phys. 165 (2000) 620–644.
[24] L.B. Tran, H.S. Udaykumar, A particle-level set-based sharp interface Cartesian grid method for impact, penetration, and void collapse, J. Comput. Phys. 193 (2004) 469–510.
[25] Z. Wang, J. Yang, F. Stern, Comparison of particle level set and CLSVOF methods for interfacial flows, in: 46th AIAA Aerospace Sciences Meeting and Exhibit, January 2008, Reno, Nevada, AIAA-2008-530.
[26] Z. Wang, J. Yang, B. Koo, F. Stern, A coupled level set and volume-of-fluid method for sharp interface simulation of plunging breaking waves, Inter. J. Multiphase Flow 35 (2009) 227–246.
[27] J. Yang, F. Stern, A sharp interface method for two-phase flows interacting with moving bodies, in: 18th AIAA Computational Fluid Dynamics Conference, June 2007, Miami, Florida, AIAA-2007-4578.
[28] J. Yang, F. Stern, Large-eddy simulation of breaking waves using embedded-boundary/level-set method, in: 45th AIAA Aerospace Sciences Meeting and Exhibit, January 2007, Reno, Nevada, AIAA-2007-1455.
[29] S.T. Zalesak, Fully multidimensional flux-corrected transport algorithms for fluids, J. Comput. Phys. 31 (1979) 335–362.