INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2008; **56**:2271–2300
Published online 25 October 2007 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1621

A hybrid Lagrangian–Eulerian particle-level set method for numerical simulations of two-fluid turbulent flows

Zhaorui Li[1], Farhad A. Jaberi[1,*,†] and Tom I-P. Shih[2]

[1]*Department of Mechanical Engineering, Michigan State University, East Lansing, MI 48823, U.S.A.*
[2]*Department of Aerospace Engineering, Iowa State University, Ames, IA 50011, U.S.A.*

SUMMARY

A coupled Lagrangian interface-tracking and Eulerian level set (LS) method is developed and implemented for numerical simulations of two-fluid flows. In this method, the interface is identified based on the locations of notional particles and the geometrical information concerning the interface and fluid properties, such as density and viscosity, are obtained from the LS function. The LS function maintains a signed distance function without an auxiliary equation *via* the particle-based Lagrangian re-initialization technique. To assess the new hybrid method, numerical simulations of several 'standard interface-moving' problems and two-fluid laminar and turbulent flows are conducted. The numerical results are evaluated by monitoring the mass conservation, the turbulence energy spectral density function and the consistency between Eulerian and Lagrangian components. The results of our analysis indicate that the hybrid particle-level set method can handle interfaces with complex shape change, and can accurately predict the interface values without any significant (unphysical) mass loss or gain, even in a turbulent flow. The results obtained for isotropic turbulence by the new particle-level set method are validated by comparison with those obtained by the 'zero Mach number', variable-density method. For the cases with small thermal/mass diffusivity, both methods are found to generate similar results. Analysis of the vorticity and energy equations indicates that the destabilization effect of turbulence and the stability effect of surface tension on the interface motion are strongly dependent on the density and viscosity ratios of the fluids. Copyright © 2007 John Wiley & Sons, Ltd.

Received 26 March 2007; Revised 3 August 2007; Accepted 8 August 2007

KEY WORDS: two-fluid turbulent flows; particle-level set method; interface tracking

1. INTRODUCTION

Two-fluid flows have been widely studied due to their importance in various of engineering and scientific problems such as the spray atomization, oil–water separation, ink jet printing and coating

*Correspondence to: Farhad A. Jaberi, Department of Mechanical Engineering, Michigan State University, East Lansing, MI 48823, U.S.A.
†E-mail: jaberi@egr.msu.edu

Contract/grant sponsor: National Science Foundation; contract/grant number: 0325760

[1–3]. One of the important issues in the study of two-fluid flows is the stability and evolution of the interface between the fluids [4–6]. Experimentally, it is often difficult to visualize and quantify the interface motion in transitional and turbulent flows. This is due to intrinsic features of the interface, such as the near-zero thickness and complex topological changes. Numerical methods offer three advantages over the experimental methods: (1) numerical simulations can provide wealth of data on almost everything, (2) various physical processes can be turned on and off at will in numerical simulations [7], (3) it is relatively convenient to numerically study the effects of various fluid/flow parameters. Despite these advantages and extensive efforts that have been devoted to the development of reliable methods, the numerical study of two-fluid turbulent flows has been somewhat limited. A challenge is that in most two-fluid systems the physical properties (e.g. density and viscosity) change abruptly across the interface, creating a hard-to-capture 'discontinuity' in the system. Another challenge is associated with the breakup and coalescence of interfaces; processes that are very much dependent on the fluid properties and kinematics of the flow.

Generally, the interface motion is computed by (i) 'tracking' and (ii) 'capturing' methods. Examples of the interface-tracking methods are the boundary integral [8, 9] and front-tracking or surface-marker [4, 10] methods. In the boundary integral method, only the points on interface are discretized, making it relatively efficient in locating the interface. However, once the interface experiences merging, breaking and other significant topological changes, the boundary integral method often becomes hard to implement [11]. The advantage of front-tracking or surface-marker method is that the interface can be tracked explicitly and accurately *via* marker particles, whose number can be increased to improve the accuracy. In comparison with the interface-capturing method, the computational time needed to resolve the interface accurately is less in the front-tracking method since the interface is represented by Lagrangian particles. However, the front-tracking methods need special efforts like particle reseeding/eliminating to deal with stretching, disconnection and reconnection of the interface. In addition, it is difficult to calculate the surface tension and curvature from the Lagrangian particles [12].

In contrast to interface-tracking methods, interface-capturing methods implicitly locate the interface based on an indicator function. There have been two popular interface-capturing methods, one is the volume-of-fluid (VOF) method, and the other is the level set (LS) method [13]. In the VOF method, the indicator function is the volume fraction that is governed by a convection equation. This method can handle relatively complex topological changes and is implemented in many commercially available codes [14–16]. The VOF method is usually implemented in two steps, advection and interface reconstruction. The former updates the volume fraction and the latter constructs an approximated interface. The piecewise linear interface reconstruction algorithms that are used in most VOF schemes have limited accuracy and suffer from discontinuity problem [16–18]. Another problem with the VOF method is that it is usually difficult to calculate the interface curvature from the volume fraction. To address some of these problems, higher order interface reconstruction algorithms are suggested [7, 19].

In contrast to VOF methods, LS methods do not need interface reconstruction. In the LS method, the interface is modeled as a smooth interface function. Also in this method, the interface always has a fixed thickness to alleviate the numerical oscillations caused by discontinuities in fluid properties across the interface. Therefore, the interface is described as a signed distance function at any time to keep a uniform and constant front thickness. Like VOF method, no additional procedure is needed in the LS method to model the complex topological changes of the interface. Also, the interface-related curvature and surface tension can be directly obtained from the LS function. Owing to these advantages, the LS method has been widely adopted in numerical simulation

DOI: 10.1002/fld

of multi-fluid flows [20–24]. However, there are some problems with the LS method, the most important one being the unphysical mass loss/gain. To alleviate this problem, a 're-initialization' procedure is proposed in which the signed distance function is adjusted during the simulation [20, 24–26]. The re-initialization procedure cannot completely remove the mass loss/gain problem and as shown below generates erroneous results in two-fluid turbulent flows. Hybrid schemes have been suggested to improve the accuracy of VOF and LS methods by combining them [15, 27]. Most numerical methods developed for VOF and LS methods so far are based on the low-order finite volume and/or finite difference schemes. Examples are the variable density project method of Bell *et al*. [7, 24, 28, 29] for structured grids and the artificial compressibility matrix-free implicit dual-time scheme of Zhao *et al*. [30] for unstructured grids. These methods can usually handle large-density and viscosity ratios but are not fully applicable to two-fluid turbulent flows due to low-order numerical accuracy.

Motivated by the fact that the LS function can provide smooth geometrical information about the interface and the Lagrangian particle method can accurately capture the interface location without any mass loss/gain, we have developed and tested a coupled (Lagrangian) interfacial particle and (Eulerian) LS method that takes the advantages of both methods. The new method, termed the interfacial particle level set (IPLS), is different from the other hybrid particle-level set and particle-VOF methods [31–34] and is successfully applied here to various laminar and turbulent flows. The effect of turbulence on the interface and the effect of interface on the turbulence are studied by the IPLS method for various flow/interface parameters. The paper is organized as follows: in Section 2, the two-fluid 'incompressible' particle-level set equations and the zero Mach number (ZMA) form of the Navier–Stokes equations are presented and compared. The numerical solution procedure for IPLS and ZMA along with the re-initialization technique for IPLS is described in Sections 3 and 4. Section 5 discusses the IPLS and ZMA results for some representative two-fluid problems. The main conclusions are presented in Section 6.

2. MATHEMATICAL FORMULATION

Two different sets of equations are presented in this section. The first one corresponds to the IPLS method and the second one describes the variable-density ZMA method. The ZMA method is considered here for the assessment of the IPLS method. The IPLS and ZMA equations and the numerical solution procedures employed for solving them are different. However, these two methods should yield similar results under special limiting conditions, at least in theory. It is shown below that the results generated by them for a two-fluid isotropic turbulent flow are indeed similar. This indicates the consistency and the accuracy of both methods.

2.1. Interfacial particle-level set equations

In the IPLS method, instead of treating the two-fluid regions separately, a coupled solution is sought *via* one set of equations and 'appropriate' representation of the interface. The non-dimensional equations describing a two-fluid incompressible and immiscible system include the continuity and momentum equations [35]:

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

$$\frac{\partial \mathbf{U}}{\partial t} = -\frac{\nabla p}{\rho(\phi)} - (\mathbf{U}\cdot\nabla)\mathbf{U} + \frac{1}{Re}\frac{\nabla\cdot(2\mu(\phi)D)}{\rho(\phi)} - \frac{1}{We}\frac{k(\phi)\nabla H(\phi)}{\rho(\phi)} - \frac{1}{Fr}\frac{\mathbf{y}}{|\mathbf{y}|} \tag{2}$$

DOI: 10.1002/fld

where $\mathbf{U}$ and p represent the velocity and pressure fields, respectively. Equations (1) and (2) are non-dimensionalized with the dimensionless parameters being the Reynolds number $Re=\rho_1 L U_0/\mu_1$, the Weber number $We=\rho_1 L U_0^2/\sigma$, and the Froude number $Fr=U_0^2/gL$, where ρ_i and μ_i are the density and dynamic viscosity for fluid i, and L and U_0 are the reference length and velocity, respectively. In Equation (2), the $\mathbf{y}$ vector denotes the gravity direction and $D=(\nabla\mathbf{U}+(\nabla\mathbf{U})^{\mathrm{T}})/2$ is the rate of deformation tensor. Gravity and surface tension parameters are represented by g and σ, respectively. Here, the fluid density ρ, the viscosity μ, and the curvature coefficient k are all functions of the level set function ϕ,

$$\begin{aligned}\rho(\phi) &= \lambda+(1-\lambda)H_\delta(\phi)\\ \mu(\phi) &= \eta+(1-\eta)H_\delta(\phi)\\ k(\phi) &= \nabla\cdot(\nabla\phi/|\nabla\phi|)\end{aligned} \tag{3}$$

where $\lambda=\rho_2/\rho_1$ is the density ratio and $\eta=\mu_2/\mu_1$ is the viscosity ratio. The LS function itself is defined as

$$\phi(\mathbf{x},t)=\begin{cases}>0, & \mathbf{x}\in\text{fluid 1}\\ =0, & \mathbf{x}\in\text{interface}\\ <0, & \mathbf{x}\in\text{fluid 2}\end{cases} \tag{4}$$

In the IPLS method, the evolution of the LS function is described by the standard LS equation

$$\frac{\partial\phi}{\partial t}=-(\mathbf{U}\cdot\nabla)\phi \tag{5}$$

and the interface movement is obtained by the Lagrangian particle equation

$$\frac{\mathrm{d}\mathbf{x}_p}{\mathrm{d}t}=\mathbf{V}^*(\mathbf{x}_p) \tag{6}$$

where $\mathbf{x}_p$ is the location of the particle p and $\mathbf{V}^*(\mathbf{x}_p)$ is the fluid velocity at the corresponding particle position. In theory, the Heaviside function is defined as

$$H(\phi)=\begin{cases}0, & \phi<0\\ \frac{1}{2}, & \phi=0\\ 1, & \phi>0\end{cases} \tag{7}$$

However, to alleviate the numerical difficulty due to discontinuity of the fluid properties across the interface, the interface is given a nominal thickness ε comparable to the grid size and Equation (7) is modified as

$$H_\delta(\phi)=\begin{cases}0, & \phi<-\varepsilon/2\\ \frac{1}{2}, & |\phi|\leqslant\varepsilon/2\\ 1, & \phi>\varepsilon/2\end{cases} \tag{8}$$

DOI: 10.1002/fld

2.2. Zero Mach number variable density equations

For numerical simulations of low-speed variable density flows, the compressible equations may be solved in the zero Mach number limit. For this, the acoustic field that imposes stringent Courant–Friedrichs–Lewy restriction on the time step is removed from the compressible equations [36]. In this limiting condition, the non-dimensional zero Mach number equations take the following form:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{U}) = 0 \tag{9}$$

$$\frac{\partial (\rho \mathbf{U})}{\partial t} + \nabla \cdot (\rho \mathbf{U}\mathbf{U}) = -\nabla p + \frac{1}{Re} \nabla \cdot \underline{\underline{\tau}} \tag{10}$$

$$\frac{\partial T}{\partial t} + (\mathbf{U} \cdot \nabla) T = \frac{1}{Re\, Pr} \frac{1}{\rho} \nabla \cdot (\mu \nabla T) \tag{11}$$

$$\rho T = p^0 \tag{12}$$

where the viscous stress tensor is defined as $\underline{\underline{\tau}} = \mu[2D - \frac{1}{3}(\nabla \cdot \mathbf{U})\underline{\underline{I}}]$ and $\underline{\underline{I}}$ is the unit second-order tensor.

In Equations (9)–(12), p^0, T are the thermodynamic pressure and temperature, respectively. Note that, p^0 is uniform in space and its non-dimensional value is unity. The Prandtl number in temperature equation is defined as $Pr = C_p \mu / K$, where K is the thermal conductivity, C_p is the specific heat at constant pressure, and $\mu = \mu(T) = T^n$ is the dynamic viscosity.

It should be emphasized here that ZMA and IPLS equations describe the flows in different fluid systems; while ZMA is for an ideal gas, IPLS represents the flow in two incompressible (liquid or gas) fluids, separated by the interface. However, ZMA and IPLS equations become similar for some limiting conditions. When $Pr \to \infty$, the energy (temperature) equation in the ZMA formulation (Equation (11)) becomes the same as the interface equation in the IPLS formulation (Equation (5)). Also in this limit, Equations (1) and (2) can be derived from Equations (9) and (10) by ignoring the surface tension and gravity terms. Thus, under the above limiting conditions, ZMA and IPLS equations are the same and should yield similar results for velocity and density, even though the temperature in the ZMA equations might not have any physical meaning. This suggests that in theory the ZMA method is applicable to any two-fluid flow problem. However, in practice this method is limited to 'low'-density ratios and is not suitable for liquid–liquid or liquid–gas systems. Nevertheless, we are using the ZMA method here for partial validation of the IPLS method. In the numerical solution of ZMA equations presented below, the density and dynamic viscosity are not constant in each fluid region as they are in the IPLS method. But the variations are found to be very small and negligible in the tested conditions.

3. NUMERICAL SOLUTION OF IPLS EQUATIONS

The numerical scheme for the velocity and pressure calculations in the IPLS method is based on a variable density projection algorithm [35, 37]. In this algorithm, the momentum equation (Equation (2)) is expressed as

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\nabla p}{\rho(\phi)} = \mathbf{W} \tag{13}$$

DOI: 10.1002/fld

where

$$\mathbf{W} = -(\mathbf{U}\cdot\nabla)\mathbf{U} + \frac{1}{Re}\frac{\nabla\cdot(2\mu(\phi)D)}{\rho(\phi)} - \frac{1}{We}\frac{k(\phi)\nabla H(\phi)}{\rho(\phi)} - \frac{1}{Fr}\frac{\mathbf{y}}{|\mathbf{y}|}$$

By taking the divergence of Equation (13) and using the continuity equation, one can readily derive the Poisson equation for pressure p,

$$\nabla\cdot\left(\frac{\nabla p}{\rho(\phi)}\right) = \nabla\cdot\mathbf{W} \tag{14}$$

Once the pressure field p is determined using Equation (14), $\mathbf{U}$ is updated from the following equation:

$$\frac{\partial\mathbf{U}}{\partial t} = -\frac{\nabla p}{\rho(\phi)} + \mathbf{W} \tag{15}$$

3.1. Temporal discretization

The time differencing in the IPLS equations is based on the second-order Adams–Bashforth scheme [38] that yields the following difference equations:

$$\phi^{n+1} = \phi^n - \mathrm{d}t\times(\tfrac{3}{2}(\mathbf{U}^n\cdot\nabla)\phi^n - \tfrac{1}{2}(\mathbf{U}^{n-1}\cdot\nabla)\phi^{n-1}) \tag{16}$$

$$\mathbf{x}_p^{n+1} = \mathbf{x}_p^n + \mathrm{d}t\times(\tfrac{3}{2}\mathbf{V}^*(\mathbf{x}_p^n) - \tfrac{1}{2}\mathbf{V}^*(\mathbf{x}_p^{n-1})) \tag{17}$$

$$\mathbf{U}^* = \mathbf{U}^n + \mathrm{d}t\times(\tfrac{3}{2}\mathbf{W}^n - \tfrac{1}{2}\mathbf{W}^{n-1}) \tag{18}$$

$$\frac{\mathbf{U}^{n+1}-\mathbf{U}^*}{\mathrm{d}t} = -\frac{\nabla P^{n+1}}{\rho^n} \tag{19}$$

With the known intermedial velocity, $\mathbf{U}^*$ and the time step, $\mathrm{d}t$, Equation (19) is solved together with the following Poisson equation for $\mathbf{U}^{n+1}$ and P^{n+1} at time step $n+1$ with an iterative method proposed by Yu [39]:

$$\nabla\cdot\left(\frac{\nabla P^{n+1}}{\rho^n}\right) = \left(\frac{1}{\mathrm{d}t}\right)\nabla\cdot\mathbf{U}^* \tag{20}$$

3.2. Spatial discretization

To minimize the numerical oscillation near the interface, which is caused by the discontinuity of the density in the Poisson equation, a second-order finite volume method is used to discretize the left-hand side (LHS) of Equation (20). The final form of the discretization scheme can be summarized in the following (two-dimensional) compact form:

$$\left[\nabla\cdot\left(\frac{\nabla P}{\rho^n}\right)\right]_{i,j} = A\times(P)_{i+1,j} + B\times(P)_{i-1,j} + C\times(P)_{i,j+1} + E\times(P)_{i,j-1} - F\times(P)_{i,j} \tag{21}$$

DOI: 10.1002/fld

where

$$\begin{aligned} A &= \frac{1}{\Delta x^2 \rho^n_{i+1/2,j}} = \frac{2}{\Delta x^2(\rho^n_{i+1,j}+\rho^n_{i,j})} \\ B &= \frac{1}{\Delta x^2 \rho^n_{i-1/2,j}} = \frac{2}{\Delta x^2(\rho^n_{i,j}+\rho^n_{i-1,j})} \\ C &= \frac{1}{\Delta y^2 \rho^n_{i,j+1/2}} = \frac{2}{\Delta y^2(\rho^n_{i,j+1}+\rho^n_{i,j})} \\ E &= \frac{1}{\Delta y^2 \rho^n_{i,j-1/2}} = \frac{2}{\Delta y^2(\rho^n_{i,j}+\rho^n_{i,j-1})} \\ F &= (A+B+C+E) \end{aligned} \tag{22}$$

For low to moderate density ratios, both standard multigrid (MG) method [40] and multigrid preconditioned conjugate gradient (MGCG) method [41] yield quick convergence when they are applied to the above 5-stencil linear system (Equations (21) and (22)). However, for larger density ratios, the convergence rate of the MG method is relatively slow [42]. Thus, in this work we have used the MGCG method to solve the Poisson equation.

For the convection term in the interface equation, the following second-order upwind scheme is applied:

$$\begin{aligned} [(\mathbf{U}\cdot\nabla)\phi]_{i,j} = {} & [(u_{i,j}+|u_{i,j}|)(3\phi_{i,j}-4\phi_{i-1,j}+\phi_{i-2,j})+(u_{i,j}-|u_{i,j}|) \\ & \times(-3\phi_{i,j}+4\phi_{i+1,j}-\phi_{i+2,j})]/(4\Delta x) \\ & +[(v_{i,j}+|v_{i,j}|)(3\phi_{i,j}-4\phi_{i,j-1}+\phi_{i,j-2})+(v_{i,j}-|v_{i,j}|) \\ & \times(-3\phi_{i,j}+4\phi_{i,j+1}-\phi_{i,j+2})]/(4\Delta y) \end{aligned} \tag{23}$$

The convection term in the momentum equation is solved by a sixth-order central compact scheme proposed by Gamet *et al*. [43]. Here, the continuity equation (Equation (1)) is employed to rewrite the convection term, $(\mathbf{U}\cdot\nabla)\mathbf{U}$, as

$$(\mathbf{U}\cdot\nabla)\mathbf{U} = f_x + g_y \tag{24}$$

where

$$f = \begin{pmatrix} u^2 \\ uv \end{pmatrix}, \quad g = \begin{pmatrix} uv \\ v^2 \end{pmatrix}$$

and the subscripts x and y imply partial differentiation in x and y directions. Taking f_x as an example to show the details of compact scheme, f'_i is the sixth-order approximation of $(f_x)_i$ and satisfies

$$\frac{1}{3}f'_{i-1}+f'_i+\frac{1}{3}f'_{i+1} = \frac{14}{9}\frac{f_{i+1}-f_{i-1}}{2\Delta x}+\frac{1}{9}\frac{f_{i+2}-f_{i-2}}{4\Delta x} \tag{25}$$

DOI: 10.1002/fld

The viscous term is rewritten as

$$\frac{1}{Re}\frac{\nabla\cdot(2\mu D)}{\rho(\phi)}=\frac{1}{Re\rho(\phi)}\left\{\begin{array}{l}2D_x(\mu D_x u)+D_y(\mu(D_y u+D_x v))\\2D_y(\mu D_y v)+D_x(\mu(D_y u+D_x v))\end{array}\right\} \tag{26}$$

where D_x and D_y derivatives are again approximated by a sixth-order central compact scheme. The divergence term in Equation (20) and the pressure gradient term in Equation (19) are also calculated by similar compact schemes.

The curvature in the surface tension term is rewritten in a new form suggested by Osher and Fedkiw [44],

$$k(\phi)=\nabla\cdot(\nabla\phi/|\nabla\phi|)=(\phi_x^2\phi_{yy}-2\phi_x\phi_y\phi_{xy}+\phi_y^2\phi_{xx})/|\nabla\phi|^3 \tag{27}$$

and then approximated by a fourth-order central difference scheme.

To stabilize the high-order scheme, the unphysical high wavenumber numerical oscillations in velocity and level set function are removed by a sixth-order accurate filter function [45]

$$\begin{gathered}\alpha\hat{f}_{i-1}+\hat{f}_i+\alpha\hat{f}_{i+1}=af_i+\frac{d}{2}(f_{i+3}+f_{i-3})+\frac{c}{2}(f_{i+2}+f_{i-2})+\frac{b}{2}(f_{i+1}+f_{i-1})\\ a=\tfrac{1}{16}(11+10\alpha),\quad b=\tfrac{1}{32}(15+34\alpha),\quad c=\tfrac{1}{16}(-3+6\alpha),\quad d=\tfrac{1}{32}(1-2\alpha)\end{gathered} \tag{28}$$

where f_i is an unfiltered variable at grid point x_i and $\hat{f}_i$ denotes the filtered value. In this work, the filtering effect is very small and is confined to the smallest flow scales with $\alpha=0.475$.

3.3. The interface algorithm

In order to keep a uniform and constant interface thickness in the LS method [24], the LS function, ϕ, is usually set as a signed distance function close to the interface. However, it is often difficult to properly maintain this distance function when the interface undergoes significant changes. To resolve this problem, Sussman *et al.* [24] proposed a 're-initialization' procedure for maintaining ϕ as a proper signed distance function. In this procedure, an auxiliary equation

$$\frac{\partial\phi}{\partial\tau}=\mathrm{Sign}(\phi_0)(1-|\nabla\phi|) \tag{29}$$

is solved for the 'corrected steady-state' value of the interface function. In Equation (29), $\mathrm{Sign}(\phi_0)=\phi_0/\sqrt{\phi_0^2+\theta^2}$, where θ is a small number comparable to grid size and ϕ_0, ϕ are the LS functions before and after re-initialization. As mentioned in Reference [46], Equation (29) may introduce significant error into the calculations as the interface is not normally located on the grid points. Even with the re-initialization procedure, the LS method still generates considerable unphysical mass loss/gain [24]. Despite various corrective procedures and constraining techniques proposed to alleviate this problem [20, 25, 26], there still remains considerable numerical error in the LS method. In addition, it is often difficult to obtain a pseudo-steady-state solution for Equation (29) in turbulent flows, where the interface geometries are complex and undergoes significant topological changes.

In contrast to the particle-level set method described in Reference [32], in our IPLS method, the position of interface is explicitly and accurately represented by a set of Lagrangian particles. The LS function provides smooth geometrical information concerning the interface, while the particles

DOI: 10.1002/fld

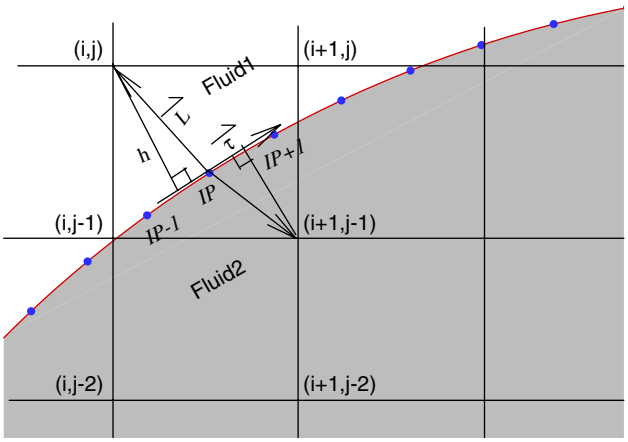

Figure 1. A schematic view of the interface, Eulerian grid and Lagrangian particles, illustrating the re-initialization technique in the particle-level set method. (i, j) are the coordinates of the Eulerian grid point and IP represents the nearest particle number.

are used for keeping the LS function as a signed distance function in the interface region without using any constraining techniques. The interface algorithm and the basic features of the IPLS method can be described in three main steps:

(i) Calculate the terms in the right-hand side (RHS) of Equations (5), (6) and (13) by using the variables at present and previous time steps (i.e. $\mathbf{U}^n$, ϕ^n, $\mathbf{x}_p^n$, $\mathbf{U}^{n-1}$, ϕ^{n-1}, $\mathbf{x}_p^{n-1}$). Note that the particle velocity $\mathbf{V}^*(\mathbf{x}_p^n)$ is obtained from the surrounding Eulerian velocity $\mathbf{U}^n$ by a fourth-order interpolation scheme.
(ii) The primary variables at next time step (i.e. $\mathbf{U}^{n+1}$, ϕ^{n+1} and $\mathbf{x}_p^{n+1}$) are obtained from Equations (16)–(20).
(iii) After certain number of iterations, the values of LS function ϕ^{n+1} are corrected by using a new re-initialization technique based on the Lagrangian particles' locations $\mathbf{x}_p^{n+1}$ to maintain the LS function as a signed distance function. The re-initialization procedure is described in detail below with the help of Figure 1.

For correcting the LS function ϕ at an Eulerian grid point, like grid (i, j) in the Fluid 1 side in Figure 1, the particle that has the minimum distance to this point is found first. This particle is represented with index IP in Figure 1. The indexes of Lagrangian particles resided on the interface are monotonically increased in an initially predetermined direction (anti-clockwise in Figure 1) along the interface. Hence, for the particle with index IP, the tangential direction vector $\boldsymbol{\tau}$ of interface at the particle location can be approximated as $\boldsymbol{\tau} \approx (\mathbf{x}_{\mathrm{IP}+1} - \mathbf{x}_{\mathrm{IP}-1})/|\mathbf{x}_{\mathrm{IP}+1} - \mathbf{x}_{\mathrm{IP}-1}|$. The vector starting from particle IP and heading to the node (i, j) is given by $\mathbf{L} = \mathbf{x}(i, j) - \mathbf{x}_{\mathrm{IP}+1}$. In the two-dimensional space, the corrected $\phi_{i,j}$ is the cross product of $\boldsymbol{\tau}$ and $\mathbf{L}$, $\phi_{i,j} = \boldsymbol{\tau} \times \mathbf{L}$, where the magnitude of $\phi_{i,j}$ (i.e. h in Figure 1) is the approximate distance of node (i, j) to the interface. The positive sign of $\phi_{i,j}$indicates that the node (i, j) is in Fluid 1 side and the negative sign of $\phi_{i+1,j-1}$ means node $(i+1, j-1)$ is in Fluid 2 as shown in Figure 1. The advantage of this technique is that the corrected $\phi_{i,j}$ automatically maintains the right signed distance function and less number of particles is needed. Note that, the computational cost of particles' transportation

DOI: 10.1002/fld

and re-initialization are linearly dependent on the number of particles. The IPLS method described here has some similarities with the hybrid Lagrangian–Eulerian method proposed by Ceniceros and Roma [47]. In the latter, the interface represented by piecewise linear functions and polygons containing a series of nearby points are used to find the minimum distance. Hybrid particle-LS methods are also proposed by Enright *et al*. [32] and Hieber and Koumoutsakosu [33]. However, in these works the particles are uniformly distributed in a narrow band around the interface and are only used for adjusting the zero LS function.

For the IPLS method, the particle addition and removal procedures are necessary as the particle number density may become too low or two high in the 'stretching' and 'compressing' regions of the interface. Inadequate number of particles make the re-initialization erroneous and cause numerical oscillations as discussed in Section 5.4. Too many particles cause unnecessary computational cost. The particle addition and removal algorithms implemented in the IPLS method are straightforward and are based on the distance between neighboring particles. Taking particles in Figure 1 as example, once the distance between IP-1 and IP particles becomes larger than an upper threshold, the indexes larger than IP-1 are increased by 1 and a new particle with index IP is added in the middle location between particles IP-1 and IP$+1$. On the contrary, if the distance between particles IP-1 and IP$+1$ becomes smaller than a lower threshold, the particle IP is removed (or is given a zero weighting) and those indexes larger than IP is reduced by 1. In practice, it is not necessary to implement the particle addition/removal procedure at every time step.

4. NUMERICAL SOLUTION OF ZMA EQUATIONS

In this paper, the predictor–corrector integration algorithm of Najm *et al*. [48] is used for solving the ZMA equations (Equations (9)–(12)). The algorithm is briefly described here. First, the equation of state (Equation (12)) is used to rewrite the time rate of change of density as

$$\frac{\partial \rho}{\partial t} = -\frac{\rho}{\mathrm{T}} \frac{\partial \mathrm{T}}{\partial t} \tag{30}$$

Subsequently, Equation (30) is used together with Equation (11) to calculate the local time derivatives of $(\partial T/\partial t)^n$ and $(\partial \rho/\partial t)^n$ in the predictor step. The second-order Adams–Bashforth scheme is implemented to evaluate the predicted density $\rho^*$and the intermediate velocity $\hat{u}_i$, while predicted temperature T^* is obtained from Equation (12),

$$\frac{\rho^* - \rho^n}{\mathrm{d}t} = \frac{3}{2}\left(\frac{\partial \rho}{\partial t}\right)^n - \frac{1}{2}\left(\frac{\partial \rho}{\partial t}\right)^{n-1} \tag{31}$$

$$\frac{\rho^* \hat{u}_i - \rho^n u_i^n}{\mathrm{d}t} = \frac{3}{2}\left(-\frac{\partial(\rho u_i u_j)}{\partial x_j} + \frac{1}{Re}\frac{\partial \tau_{ij}}{\partial x_j}\right)^n - \frac{1}{2}\left(-\frac{\partial(\rho u_i u_j)}{\partial x_j} + \frac{1}{Re}\frac{\partial \tau_{ij}}{\partial x_j}\right)^{n-1} \tag{32}$$

$$T^* = \frac{P^0}{\rho^*} \tag{33}$$

DOI: 10.1002/fld

The intermediate (dynamic) pressure is determined by inverting the Poisson equation,

$$\nabla^2 P^* = \frac{1}{\mathrm{d}t}\left(\frac{\partial(\rho^*\hat{u}_i)}{\partial x_i} + \frac{1}{2\mathrm{d}t}(3\rho^* - 4\rho^n + \rho^{n-1})\right) \tag{34}$$

Finally, in the projection step, the predicted velocity u_i^* is determined as

$$\frac{\rho^* u_i^* - \rho^* \hat{u}_i}{\mathrm{d}t} = -\frac{\partial P^*}{\partial x_i} \tag{35}$$

In the corrector step, the new corrected values for density are calculated using a second-order quasi Crank–Nicolson integration scheme to stabilize the oscillation caused by the variable density effects,

$$\frac{\rho^{n+1} - \rho^n}{\mathrm{d}t} = \frac{1}{2}\left(\left(\frac{\partial\rho}{\partial t}\right)^n + \left(\frac{\partial\rho}{\partial t}\right)^{**}\right) \tag{36}$$

In Equation (36), $(\partial\rho/\partial t)^{**}$ is calculated from Equations (11) and (30) based on the 'predicted' values. By replacing ρ^*, T^*, P^* and u_i^* with ρ^{n+1}, T^{n+1}, P^{n+1} and u_i^{n+1}, Equations (32)–(35) are employed again for the corrected values of T^{n+1}, P^{n+1} and u_i^{n+1}.

The Poisson equation is solved by the spectral method and fast Fourier transform (FFT). Note that, there is no discontinuity in the Poisson equation which is solved in the predictor and corrector steps. The above predictor–corrector integration scheme allows the maximum computable density ratio to increase [36]. In this work, all spatial derivatives are calculated by high-order spectral/FFT schemes.

5. RESULTS AND DISCUSSIONS

Before applying the IPLS method to a two-fluid turbulent problem, its reliability and accuracy are first established for simpler 'standard' test problems. In Section 5.1, a single-fluid isotropic turbulent flow is simulated *via* IPLS and ZMA methods by setting the fluid properties the same and by ignoring the surface tension term. The ability of IPLS method in capturing the interface evolution in various moving-interface problems is demonstrated in Section 5.2. In Section 5.3, the rising of a low-density bubble in a high-density fluid is simulated by the IPLS. The results obtained by the IPLS and ZMA methods for a two-fluid isotropic turbulent flow are presented in Section 5.4.

All turbulent simulations are conducted in a square domain with uniform grids and grid resolutions of 256×256 and 512×512. The turbulent kinetic energy spectrums in Figures 2(b) and 16(b) and all the other results not shown in this paper indicate that for both single- and two-fluid flows the turbulence is well resolved with 256×256 grid points. The results shown below are obtained with lower (256×256) grid points to establish the performance of the IPLS method and its interface-tracking algorithm in the limiting conditions.

5.1. Single-fluid isotropic turbulence

In this section, simulations of an isotropic decaying turbulent flow with periodic boundary conditions are conducted to establish the accuracy of the numerical models for single-fluid flows before

DOI: 10.1002/fld

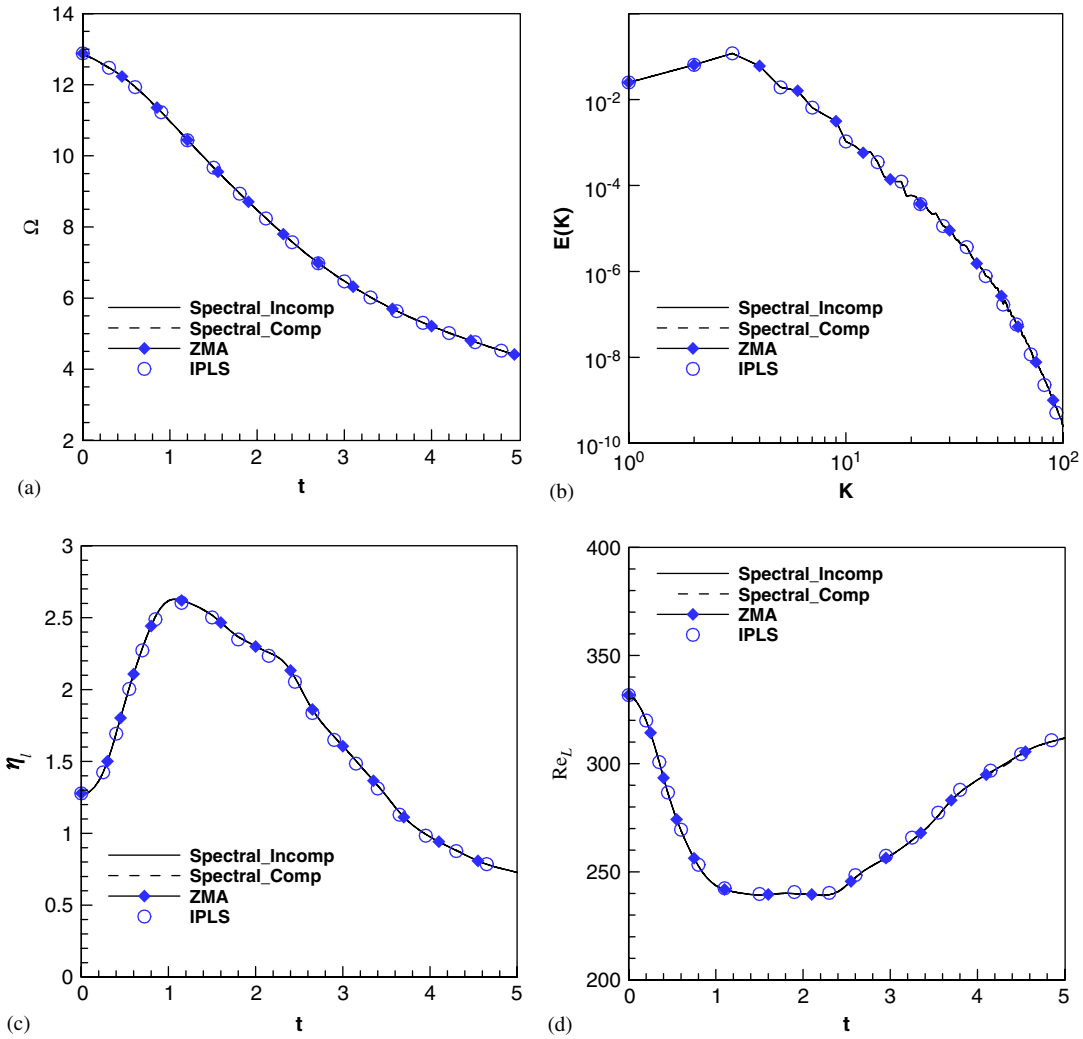

Figure 2. Turbulent statistics calculated by different methods: (a) enstrophy; (b) turbulent energy spectral density at $t=4.0$; (c) enstrophy dissipation rate; and (d) Reynolds number based on integral scale and turbulent energy.

applying them to two-fluid problems. The initial turbulence in all of our isotropic turbulent flow simulations is the long-time solution of an initially random solenoidal velocity field with Gaussian spectrum [49]. In addition to ZMA method, spectral-based incompressible and compressible methods are also employed for further assessment of the IPLS method. The compressible simulation is conducted at low Mach number ($Ma=0.05$). In the following, several important turbulent statistics as obtained by different methods are compared. This includes the mean turbulent energy $k_t=\frac{1}{2}\langle u_i u_i\rangle$, the mean enstrophy, $\Omega=\frac{1}{2}\langle\omega_i\omega_i\rangle$, and the enstrophy dissipation rate, $\eta_l=(\mu/\rho)\langle|\nabla\times\boldsymbol{\omega}|^2\rangle$,

DOI: 10.1002/fld

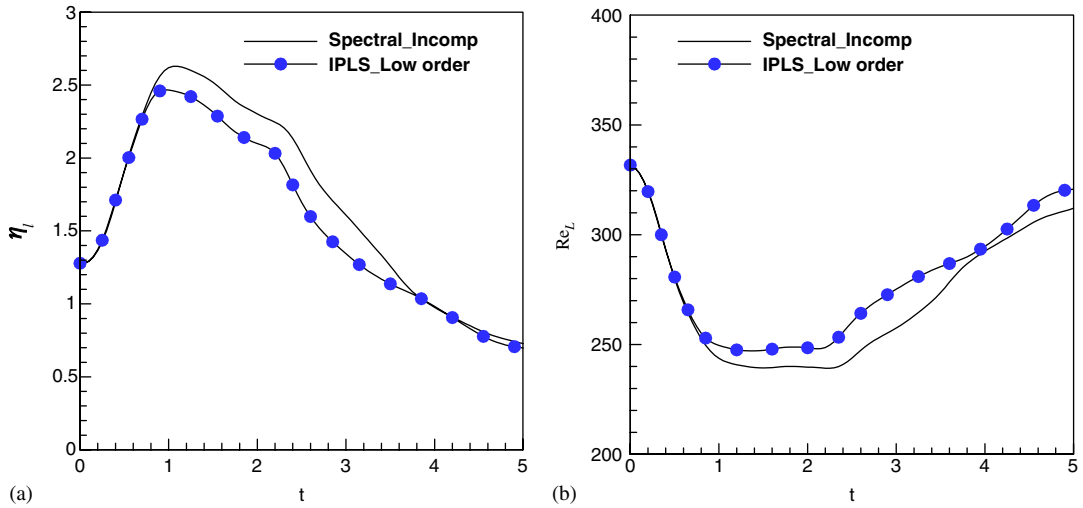

Figure 3. Turbulent statistics obtained by incompressible pseudo-spectral and modified IPLS method: (a) enstrophy dissipation rate and (b) Reynolds number based on integral scale and turbulent energy. In the modified IPLS method, all the derivatives are calculated by second-order numerical methods.

where $L = k_t^{1/2}/\eta_l^{1/3}$ and Re_L are turbulent macro length scale and the corresponding Reynolds number.

Figure 2 shows that the predicted long-time statistics by the IPLS, ZMA and spectral methods (compressible and incompressible) are virtually identical. This indicates the reliability and the accuracy of both ZMA and IPLS methods in single-fluid turbulent flows. Note that in the IPLS method, the pressure (Poisson) equation is solved with a second-order finite volume scheme to avoid the numerical instability due to discontinuity in density in two-fluid calculations. Nevertheless, the IPLS results remain accurate as long as the other spatial derivatives are approximated with high-order difference schemes. This is confirmed in Figure 3, where it is shown that the IPLS method loses its accuracy when the spatial derivatives are calculated by second-order schemes instead of sixth- or fourth-order schemes. The IPLS results discussed below are all based on high-order spatial differencing.

5.2. *Interface tracking via IPLS method*

In this section, the IPLS method is used for prediction of the interface movement in two types of problems. In the first one, the velocity field, $\mathbf{V} = \mathbf{V}(\mathbf{x})$, is independent of the interface and is externally imposed. The so-called Zalesak and vortex stretching problems belong to this type. The other type of problem considered here is the one in which the velocity field is self-generated and is directly dependent on the interface. An example is a closed interface that moves inward with a local velocity which is proportional to the local curvature of the interface, i.e. $\mathbf{V} = -C_1\kappa\mathbf{N}$ (C_1 is a constant), where $\mathbf{N} = \nabla\phi/|\nabla\phi|$ is the unit normal direction vector, and $\kappa = \nabla\cdot(\mathbf{N})$ is the curvature. The so-called 'collapsing' circle, C-shape and star-shape problems are among the second type of problems which are considered here. These problems are more difficult to model or simulate than the first type of problems in which the velocity field is not affected by the interface.

DOI: 10.1002/fld

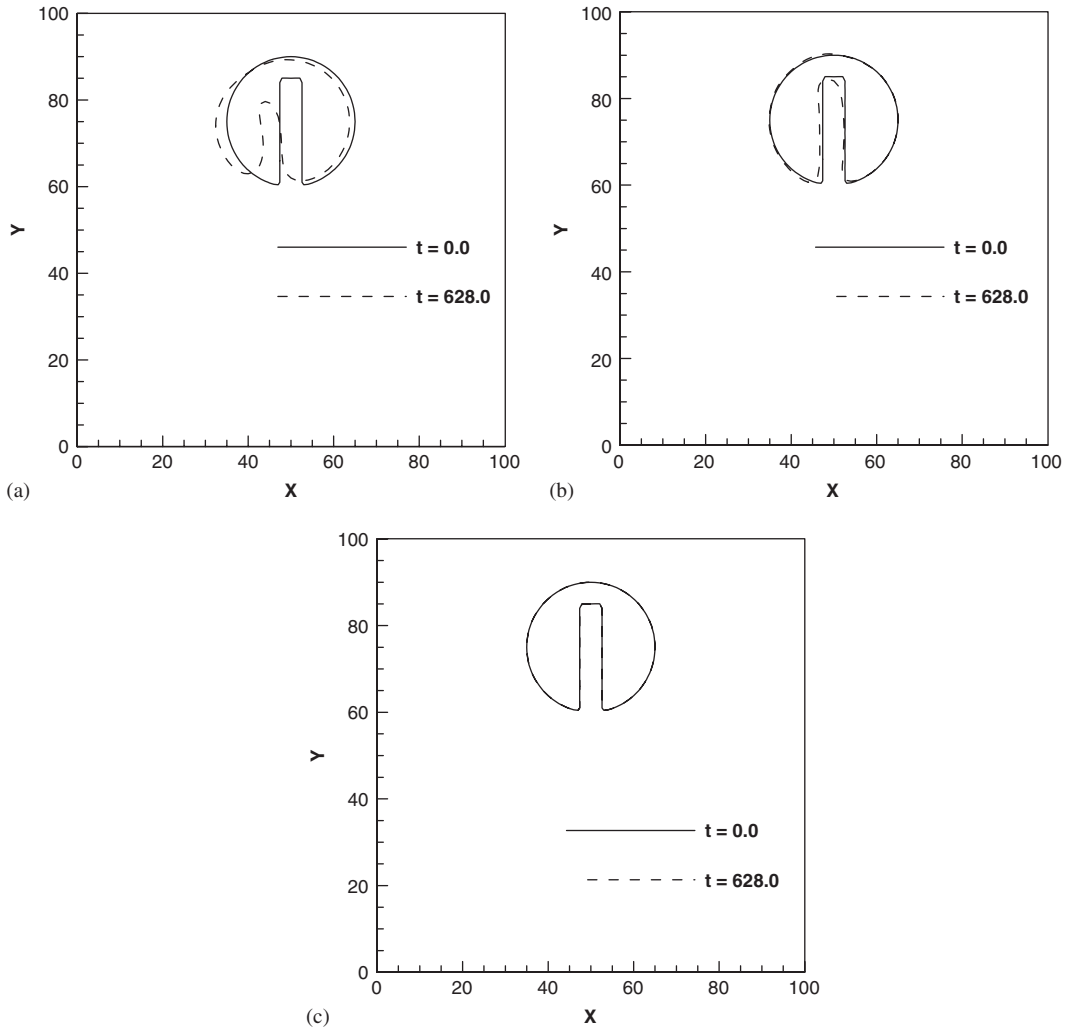

Figure 4. The interface movement as predicted by various methods for the Zalesak problem: (a) level set method without reinitialization; (b) level set method with reinitialization; and (c) IPLS method.

5.2.1. Zalesak problem. In the so-called Zalesak problem [50], the interface is a two-dimensional rotating slotted disk that is moved with a specified velocity field. This problem has been considered by many investigators for testing of the interface-tracking models and is considered here for preliminary assessment of the Lagrangian interface-tracking algorithm in the IPLS method. Figure 4 shows the IPLS results for the Zalesak problem. The computational domain size is 100×100 units and the slotted circle which has the radius of 15 units is initially centered at $(x = 50, y = 75)$ point. The width and length of the slot are 5 and 25, respectively. The interface revolves by a pre-specified velocity field defined as $u = (\pi/314) \times (50 - y)$, $v = (\pi/314) \times (x - 50)$, where u and v are the x and y components of the velocity vector, respectively. The LS function, is set as

DOI: 10.1002/fld

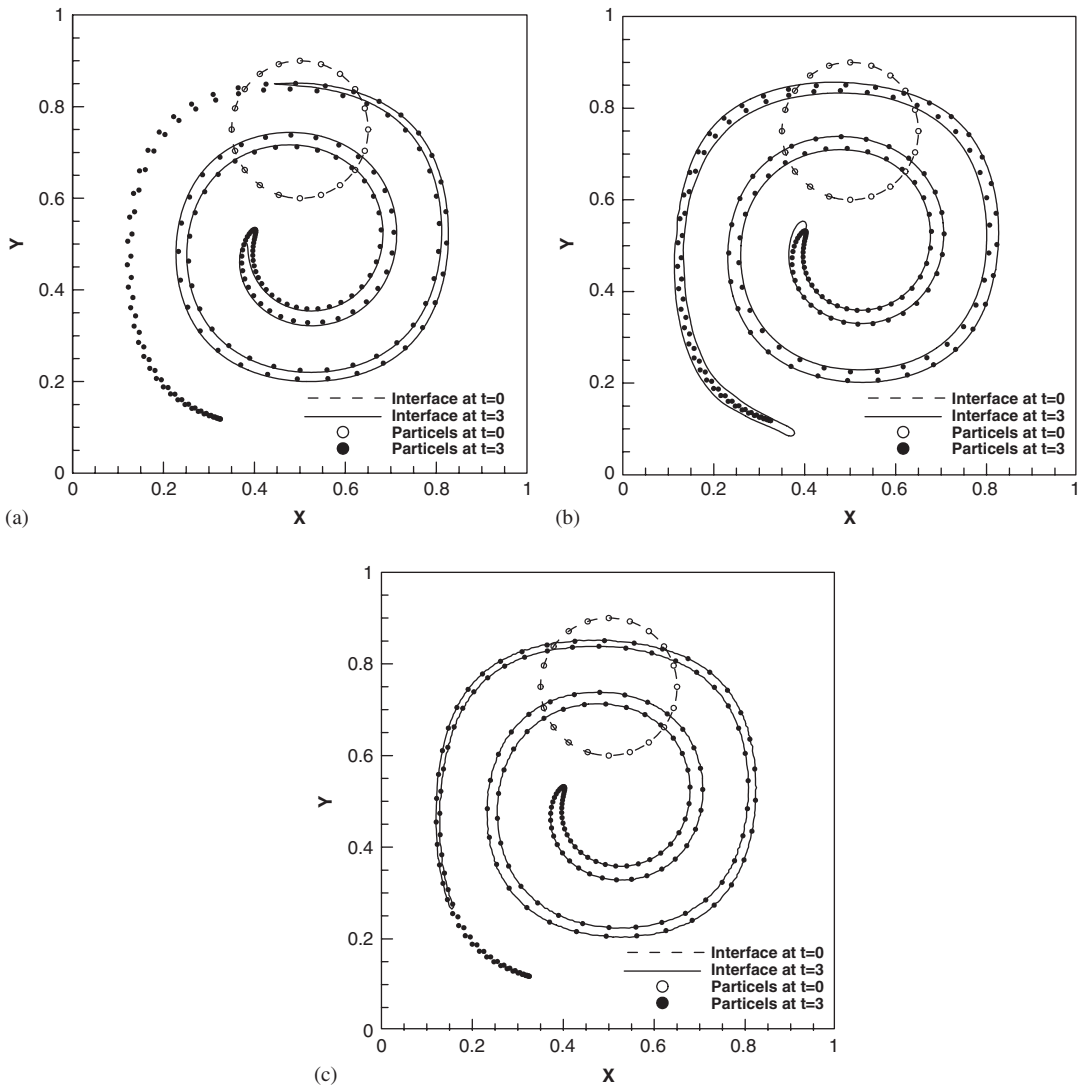

Figure 5. The interface and particles' distribution in a vortex flow (vortex stretching problem) as obtained by different methods at $t=0.0, 3.0$; (a) level set method without reinitialization; (b) level set method with reinitialization; and (c) IPLS method. In figures (a) and (b), $N_X=N_Y=200$ and in figure (c) $N_X=N_Y=128$. A total of $n_p=2000$ particles is used in the IPLS method.

$\phi<0$ inside the disk, and $\phi>0$ outside the disk. The interface moves from time $t=0$ to 628 to its initial position after a complete revolution of the disk. Figure 4 shows the interface ($\phi=0$) contours at initial ($t=0$) and final ($t=628$) times as obtained by the IPLS and standard LS methods. Evidently, the IPLS predictions in Figure 4(b) are very accurate as the final and initial interfaces are virtually identical and there is no measurable numerical error even though the grid resolution is moderate in these calculations. Unlike IPLS method that perfectly maintains the

DOI: 10.1002/fld

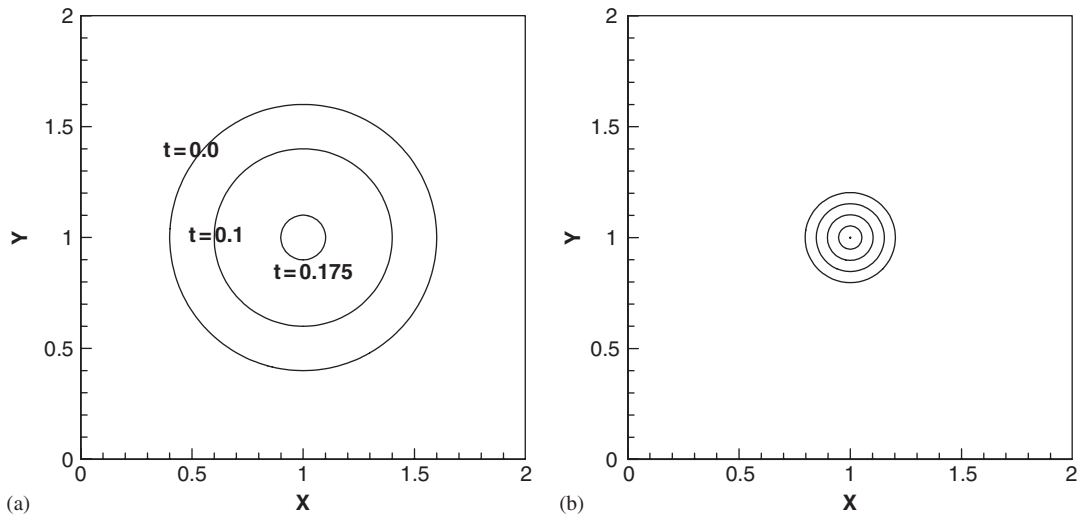

Figure 6. A collapsing circle interface with $N_X = N_Y = 100$, $n_p = 1000$ as predicted by the IPLS method: (a) the interface at different times and (b) the contours of level set function at $t = 0.175$, the contour levels from outer to interior are $\phi = 0.1,\ 0.05,\ 0.0,\ -0.05,\ -0.1$.

interface shape at all locations, the standard LS method is not so accurate and causes significant interface shape change (Figure 4). This indicates that the IPLS method is superior to standard LS methods as it can handle complex interface topology with a high level of accuracy. It is to be noted that the LS simulations are conducted with no 'reinitialization' technique. The LS results obtained with sophisticated reinitialization techniques [26, 46] (not shown here) are better than those shown in Figure 4(a) but are still considerably less accurate than those predicted by the IPLS method.

5.2.2. Vortex stretching. Turbulent flows are usually characterized by complex vortical motions with various lengths and time scales. Consequently, the interface is expected to experience significant changes in a turbulent environment. In some cases, the distance between two adjacent interfaces or the characteristic length scale of a fluid filament moving in another fluid could become comparable or smaller than the grid size. This could create some difficulties for the standard grid-based methods. The vortex stretching problem involves the stretching and folding of a fluid filament in an isolated vortex and was first introduced by Bell *et al.* [28] to assess the performance of numerical methods. Here, we consider this problem to access the accuracy and reliability of the IPLS method. For this, an initially circular interface is moved according to a pre-specified velocity field: $u = -\sin^2(\pi x) \times \sin(2\pi y),\ \ v = \sin^2(\pi y) \times \sin(2\pi x)$. The computational domain size is chosen to be 1×1 unit and the initial circle is centered at $(0.50, 0.75)$ with a radius of 0.15. Theoretically, the circle should be stretched into a long, thin fluid filament which progressively wraps itself around the center [32]. Figure 5 shows the predicted interface contours ($\phi = 0$) and the Lagrangian particles by the IPLS method. Evidently, the interface evolution is more accurately predicted by the IPLS (Figure 5(c)) than by the standard LS method with and without reinitialization (Figures 5(a) and (b)). In fact, the IPLS is able to capture the smallest scales in the interface which are

DOI: 10.1002/fld

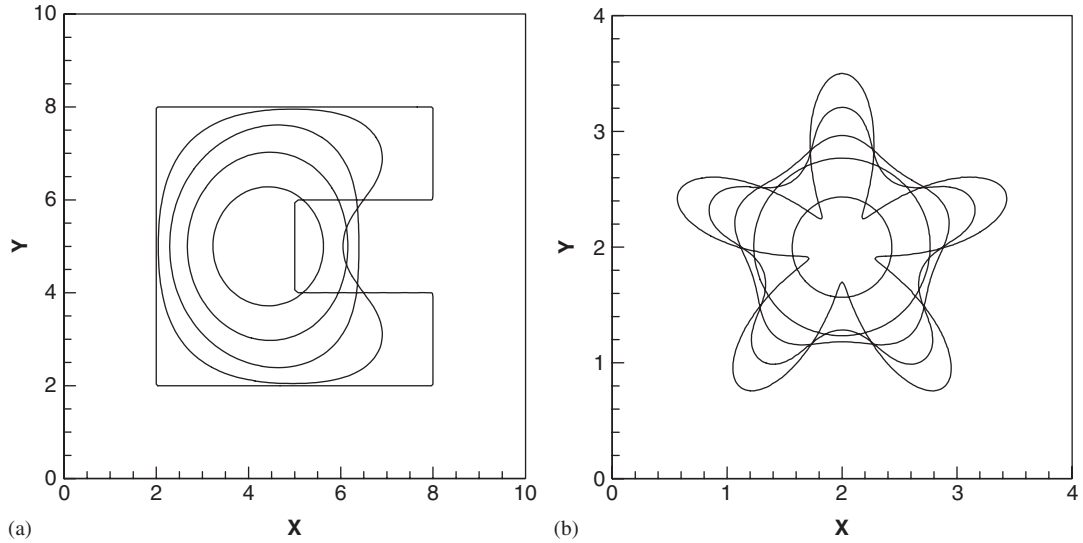

Figure 7. Temporal variations of the interface for: (a) collapsing C-shape and (b) collapsing star. In figure (a), $t=0.0$, 1.0, 2.0, 3.0, 4.0 from outer to interior, respectively. In figure (b), $t=0.0$, 0.05, 0.1, 0.2, 0.4 from outer to interior, respectively. In both figures, $N_X=N_Y=200$, $n_p=2200$.

comparable to or even smaller than the mesh size without the necessity of refining the grids near the interface [35]. With the Lagrangian scheme, the filament size is not limited by the grid size in the IPLS method and can be much smaller than the grid size. Accurate results are expected as long as the velocity field is well resolved and the interpolation scheme is sufficiently accurate.

5.2.3. Collapsing circle. The collapsing circle is an interesting interface moving problem that has an analytical solution. Here, the tangential velocity is zero on the interface and the normal velocity is a simple function of the interface curvature, κ,

$$V_n = \mathrm{d}r/\mathrm{d}t = -\kappa = -1/r$$

By integrating the above equation, we get the interface evolution equation, $r^2 = r_0^2 - 2t$, where r is the radius of the circular interface that has the initial radius of $r_0=0.6$. Figure 6(a) shows that the predicted interface *via* IPLS method match with the analytical solution very well at all times. Theoretically, the LS function ϕ cannot stay as a perfect signed distance function because the magnitude of interface advection velocity increases with $1/r$ and the distance between any two ϕ contours increases in time in a collapsing circle. However, Figure 6(b) shows that the LS function, ϕ, remains a proper signed distance function in the IPLS method.

5.2.4. Collapsing C-shape and star-shape. To further assess the IPLS method, the collapsing of a C-shape and a star-shaped interface under the influence of its curvature is also studied. The initial C-shape is made out of a 6×6 square and is shown in Figure 7(a). The initial star-shaped interface is shown in Figure 7(b) and is described by the following five-pointed star function:

$$\gamma(s) = 2.0 + 0.6\sin(5\times 2\pi s)(\cos(2\pi s), \sin(2\pi s)) + 0.9(\cos(2\pi s), \sin(2\pi s))$$

DOI: 10.1002/fld

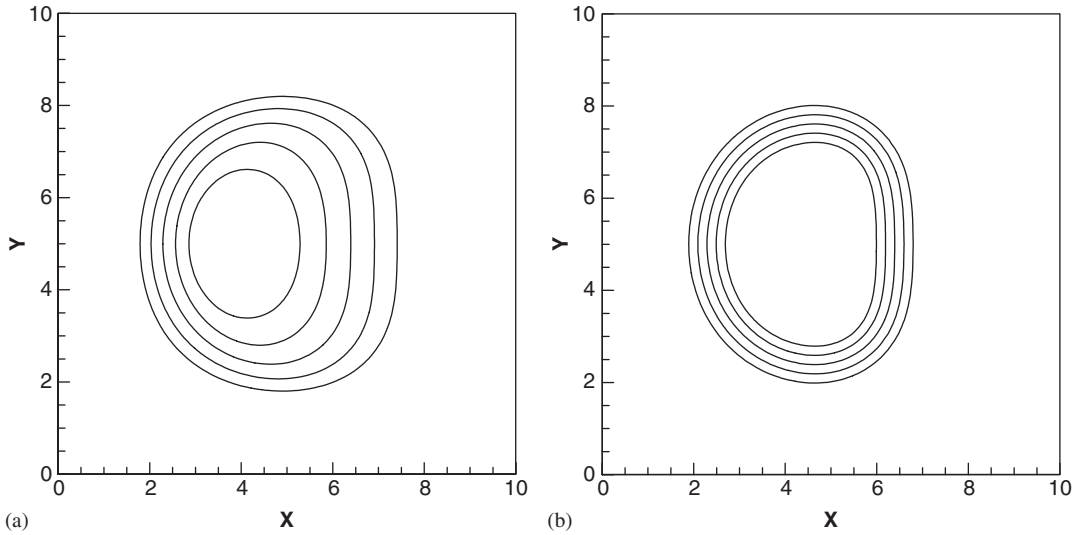

Figure 8. Signed distance function for the collapsing C-shape interface at $t=2.0$ as obtained by: (a) level set method without reinitialization and (b) IPLS method.

where s is a number between 0 and 1. The surface advection velocities for both problems are $\mathbf{V}=-\kappa\mathbf{N}$, where κ is the curvature and $\mathbf{N}$ is the unit normal vector as defined before. Figure 7 shows that for both problems the interface collapses into a circle due to curvature-dependent velocities. Also, in both problems, the interface experiences a relatively complex evolution that is not easy to capture with the standard interface-tracking algorithms [51]. Nevertheless, the IPLS method is able to accurately predict the interface movement and its evolution from a C-shape or a star-shaped initial form to a circle. The results in Figure 8(b) for the LS function are consistent with those in Figure 7 and indicate that the IPLS method maintains the LS function ϕ as a signed distance function very well. In contrast, without re-initialization the standard level set method is not accurate and yields erroneous results for the LS function (Figure 8(a)).

5.3. Rising bubble

In this section, the temporal evolutions of a two-dimensional bubble [30] in a fully filled container for various density ratios are studied *via* IPLS method. Initially, the bubble is a circle with radius R and the computational domain size is $4R\times 8R$. The density ratio, λ, and the viscosity ratio, η, are defined in Section 2.1, where fluid 2 is air inside the bubble and fluid 1 is the liquid in the container. All the other parameters are set to be the same as those considered in the numerical simulations of Zhao *et al*. [30], e.g. $Fr=1.0$, $Re=100$ and $We=200$.

Figures 9 and 10 show the temporal evolution of the rising bubbles with the density ratios of $\lambda=1:5$ and 1:100, respectively. In both figures, the viscosity ratio is $\eta=1:200$ and the grid resolution is 256×512. Expectedly, the bubble is driven upward by the buoyancy force and the larger the density ratio, the faster the rising speed would be. The larger pressure gradient on the lower surface of the bubble induces a jet of liquid that pushes the lower surface upward and causes the formation of vortex sheet around the lower surface. Owing to the presence of surface tension, the lower surface cannot fully penetrate and break the upper surface and the bubble takes a mushroom

DOI: 10.1002/fld

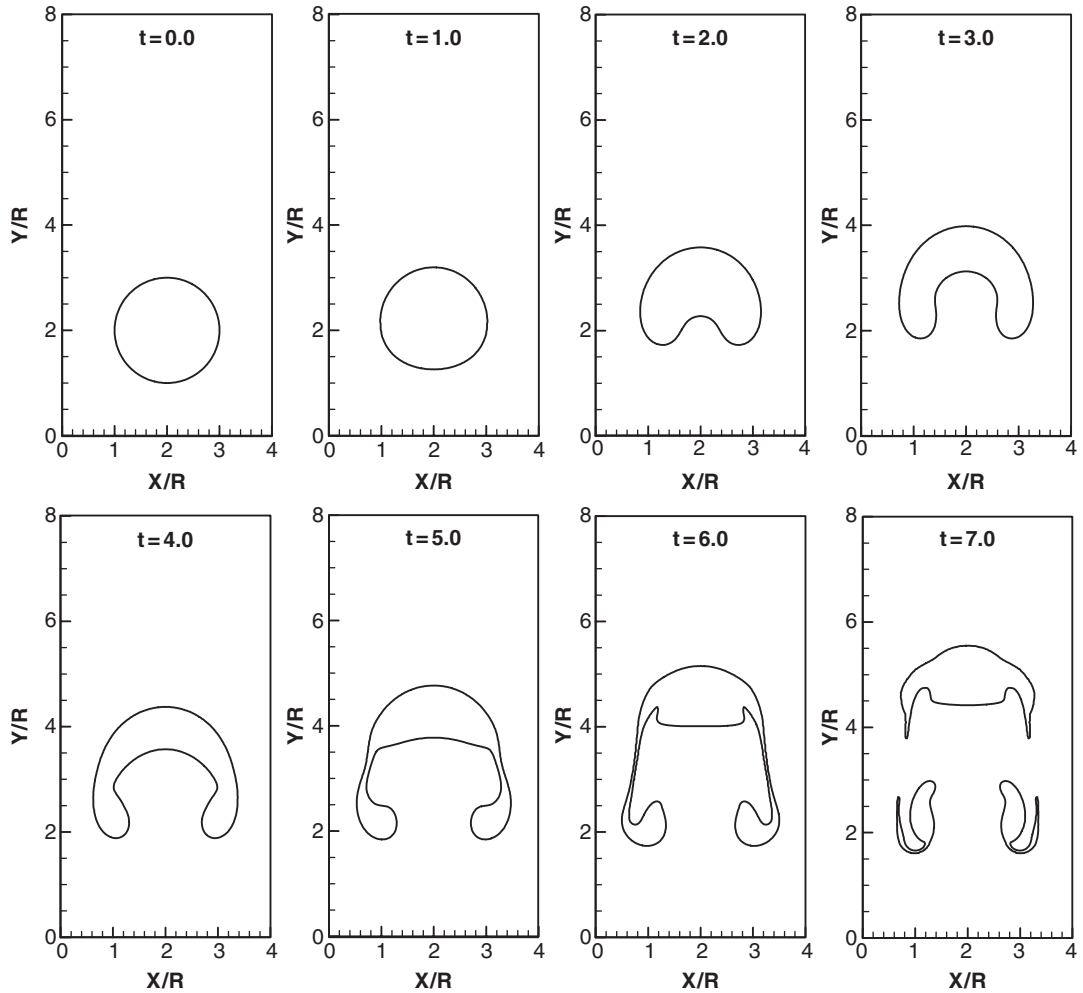

Figure 9. Temporal variations of the rising bubble as predicted by the IPLS method for the case with $Re=100$, $We=200$, $\lambda=1:5$ and $\eta=1:200$.

shape. At later times, the vortex sheet around the lower surface generates two symmetrical bubble skirts that fold inward. Eventually, the two bubble skirts detach from the bubble under the intensive stretching caused by the vortex sheet. Note that, although our results for density ratio of $\lambda=1:5$ are similar to those in Zhao *et al*. [30], there is a noticeable difference between the results obtained with smaller and larger density ratios. Figure 10 shows that the shedding bubbles can continue to break up into two separate smaller bubbles at later times, which is a reasonable and physically possible phenomenon that may not be well captured by low-order numerical methods. Also, unlike some of the other numerical methods which are not able to predict shedding bubbles and sometimes generate (unphysical) asymmetric results, the IPLS predictions are accurate and symmetric. These results indicate that the IPLS method is able to accurately predict the two-fluid flows. In Figure 11,

DOI: 10.1002/fld

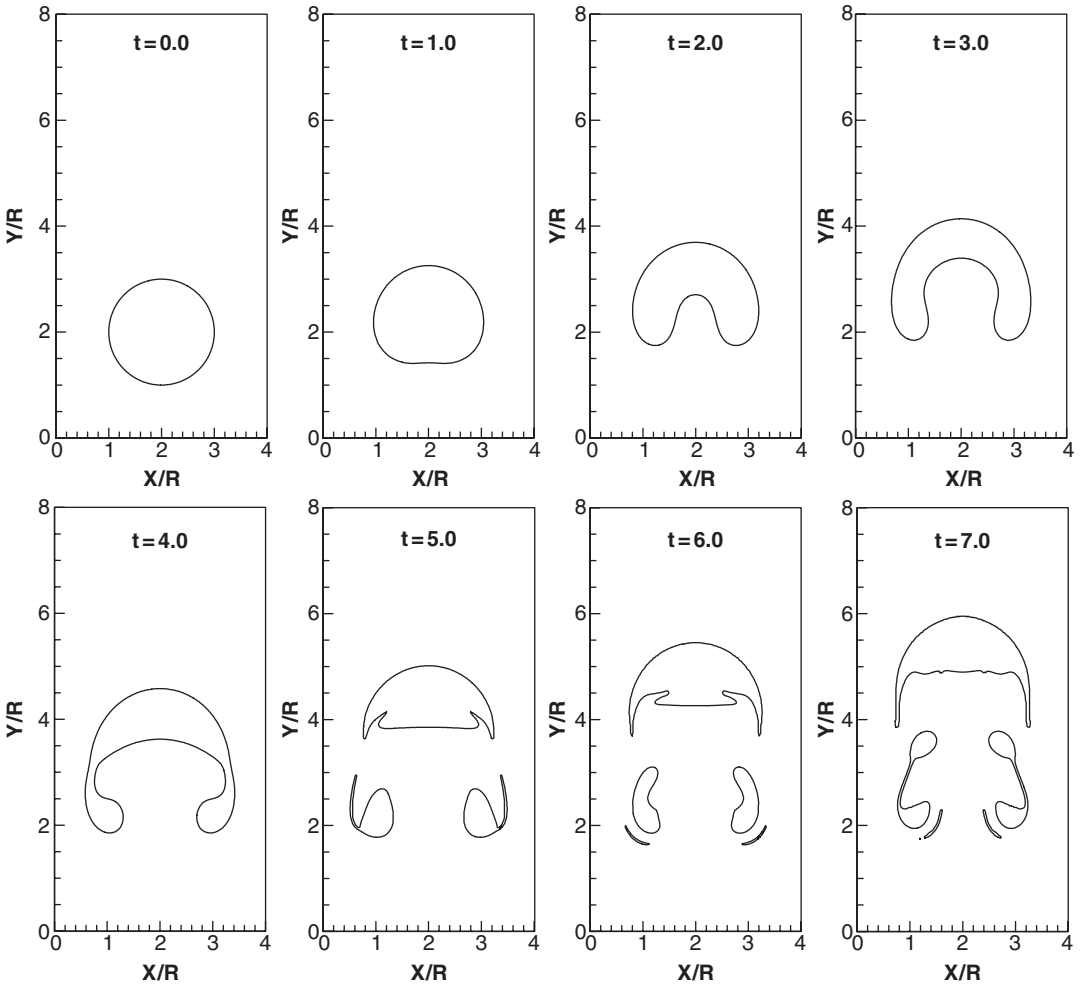

Figure 10. Temporal variations of the rising bubble as predicted by the IPLS method for the case with $Re=100$, $We=200$, $\lambda=1:100$ and $\eta=1:200$.

the rising bubbles as predicted by the IPLS method with different grid resolutions are compared at different times. At earlier time ($t=3.0$), there is virtually no difference between the interfaces predicted with different resolutions. At later time ($t=6.0$), only the smallest scales of the broken bubble are not adequately resolved in the lower resolution case and most of the interface including the two shedding bubbles are similarly predicted in both cases. This indicates that the employed high-resolution grid system in Figures 9 and 10 is adequate.

5.4. Two-fluid isotropic turbulence

The two-fluid turbulent flow simulations considered in this paper are two dimensional. We are well aware of the differences (and similarities) of material-element/interface transport in

DOI: 10.1002/fld

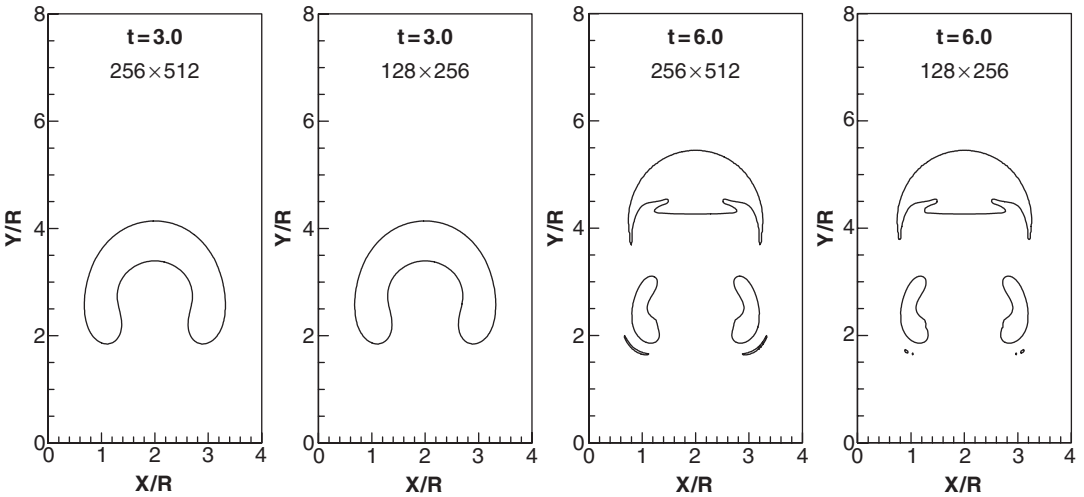

Figure 11. Rising bubble as predicted by the IPLS method with different grid resolutions at different times for the case with $Re=100$, $We=200$, $\lambda=1:100$ and $\eta=1:200$.

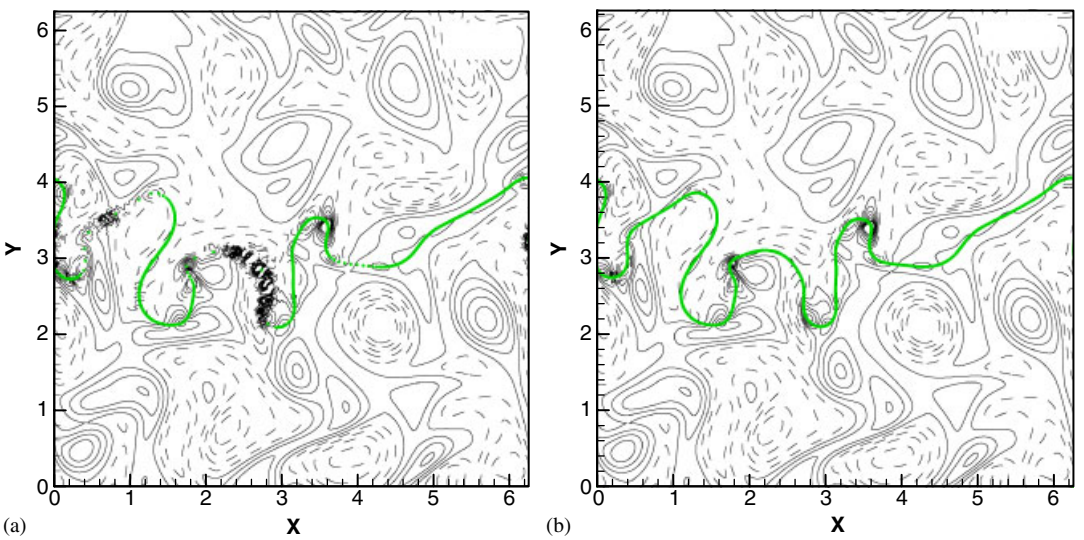

Figure 12. The predicted vorticity contours and interface in a two-fluid isotropic turbulent flow by the IPLS method: (a) without particle addition algorithm and (b) with particle addition algorithm.

two-dimensional and three-dimensional turbulent flows. However, even a two-dimensional two-fluid isotropic turbulent flow is a very challenging flow to model and simulate. We consider this flow as a relatively simple two-fluid turbulent flow that has some important features of more realistic three-dimensional inhomogeneous flows. Here, not only the interface experiences complicated changes (e.g. stretching, folding and breaking) due to turbulence, but the turbulence itself is affected by the interface in a two-way coupling manner.

DOI: 10.1002/fld

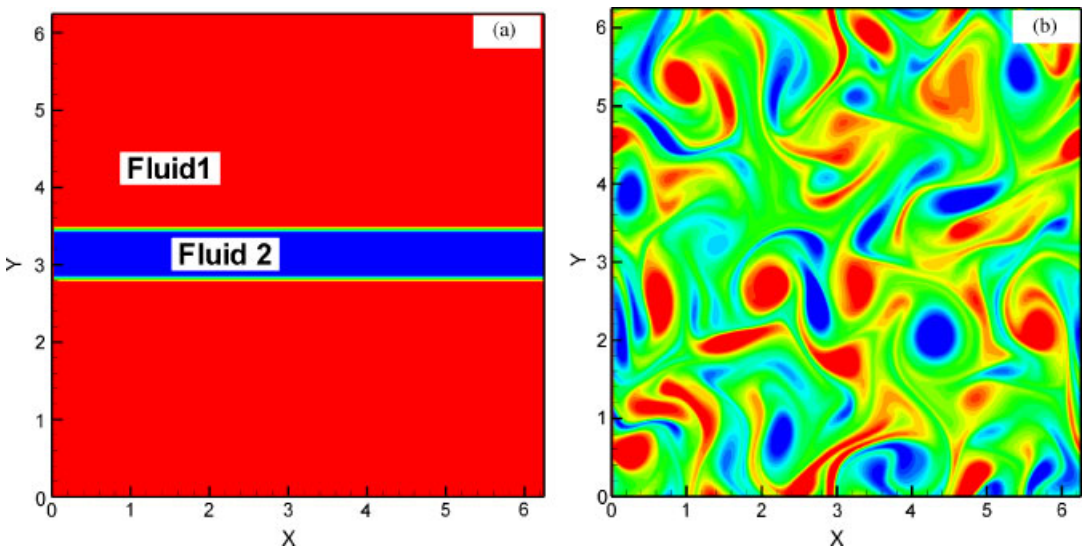

Figure 13. Initial contours of (a) density and (b) vorticity for an oil–water two-fluid isotropic turbulent system.

To correctly capture the interface movement in turbulent flows and the subsequent effects of the interface on turbulence, a particle addition/removal algorithm is implemented in the IPLS method. This is an inexpensive algorithm that is based on the distance between neighboring particles. When the distance becomes larger than an upper threshold, new particles are added and particles' indexes are adjusted accordingly. Alternatively, when the distance between particles becomes smaller than a lower threshold value, particles are removed and again their indexes are adjusted to maintain a proper order.

The importance of particle addition algorithm is illustrated in Figure 12, where the contours of the vorticity magnitude and the interface, obtained by the IPLS method, in two-fluid isotropic turbulent flow are shown. Evidently, without the particle addition algorithm, the number of particles is inadequate in the 'stretched' regions of interface which may cause some numerical oscillations as described in Section 3.3. However, the IPLS method with particle addition algorithm is able to accurately predict the interface evolution in a two-fluid turbulent system.

As shown in Figure 13(a), the two-fluid system considered in this paper is an oil layer that is surrounded by water. The oil thickness is changed by changing L_0/L_y, where L_0 and L_y are the initial thicknesses of oil layer and domain size in the normal (y) direction, respectively. A fully developed isotropic turbulence, illustrated by the vorticity contours in Figure 13(b), is implemented as the initial velocity field and periodic boundary conditions are applied in all directions. This two-fluid flow is simulated by the IPLS and ZMA methods. In the IPLS method, the density and viscosity ratios are set to fixed values of $\lambda = 0.8:1$ and $\eta = 200:1$, while the other parameters are varied. In the ZMA method, the desired viscosity ratio is obtained from $\mu = \mu(T) = T^n$ and Equation (12).

As indicated before, in the IPLS method, the interface can be identified by either particles' locations or the zero LS ($\phi = 0$) contours. In contrast, in the ZMA method, the interface is identified by monitoring the density itself. This is illustrated in Figures 14(a)–(d), where the results obtained

DOI: 10.1002/fld

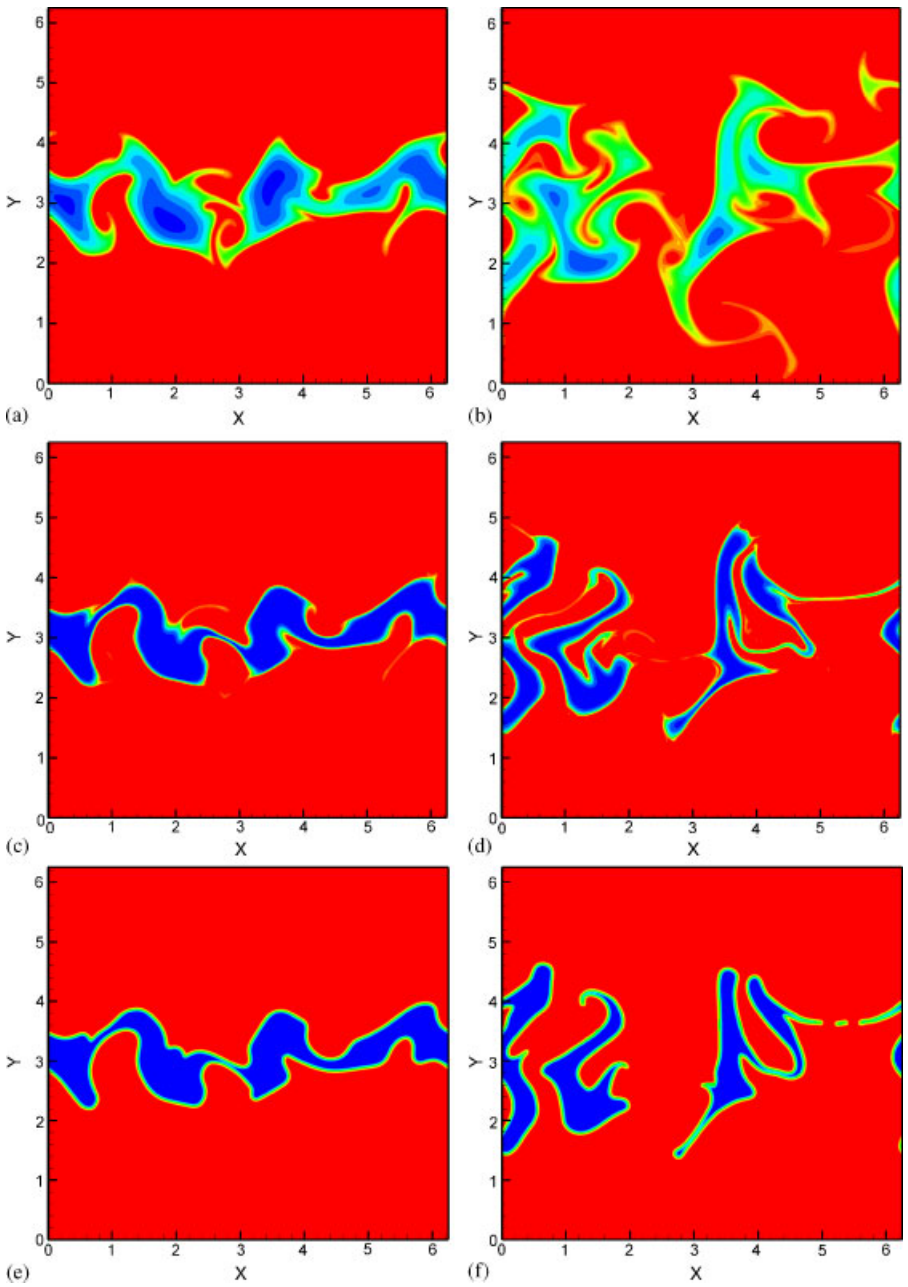

Figure 14. Density contours at different times as obtained by IPLS and ZMA methods in an oil–water two-fluid isotropic turbulent flow. Figures (a)–(d) are obtained by the ZMA method and figures (e) and (f) by the IPLS method. In figures (a) and (b) $Pr=0.7$, and in figures (c) and (d) $Pr=60$. Figures (a), (c) and (e) show the results at $t=1.0$ and figures (b), (d) and (f) show the results at $t=3.0$. In all figures $L_0/L_y \approx 1/10$, $Re=1000$, $\eta=200:1$ and $\lambda=0.8:1$.

DOI: 10.1002/fld

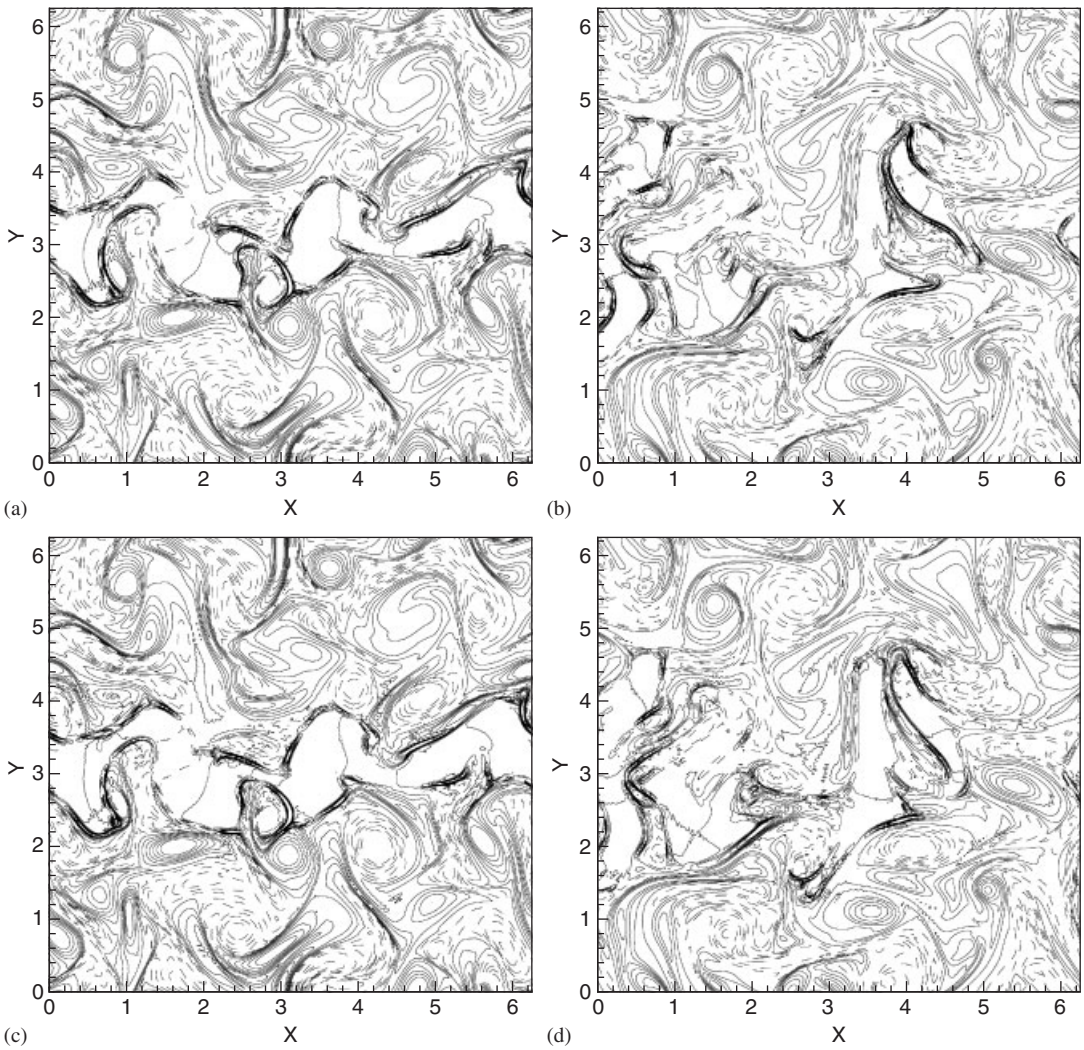

Figure 15. The vorticity contours as obtained by ZMA and IPLS methods in an oil–water two-fluid isotropic turbulent system at different times. Figures (a) and (b) are obtained by the ZMA method and figures (c) and (d) by the IPLS method. Figures (a) and (c) show the results at $t=1.0$ and figures (b) and (d) show the results at $t=3.0$. In all figures $L_0/L_y \approx 1/10$, $Re=1000$, $\eta=200:1$ and $\lambda=0.8:1$.

by the ZMA method for two different Prandtl numbers are shown. Figures 14(a) and (b) show that for $Pr=0.7$ the initial sharp density (or temperature) gradient is smoothed and the minimum density increases from 0.8 to 0.87 due to relatively high values of the thermal diffusivity coefficient in the energy equation. In contrast, the initial density jump across the interface is well maintained when $Pr=60$ and the thermal diffusion is low (Figures 14(c) and (d)). Also for $Pr=60$, the results obtained by the ZMA method are reasonably close to those predicted by the IPLS method when Weber number is large or surface tension is small (Figures 14(e) and (f)). Furthermore, Figure 15

DOI: 10.1002/fld

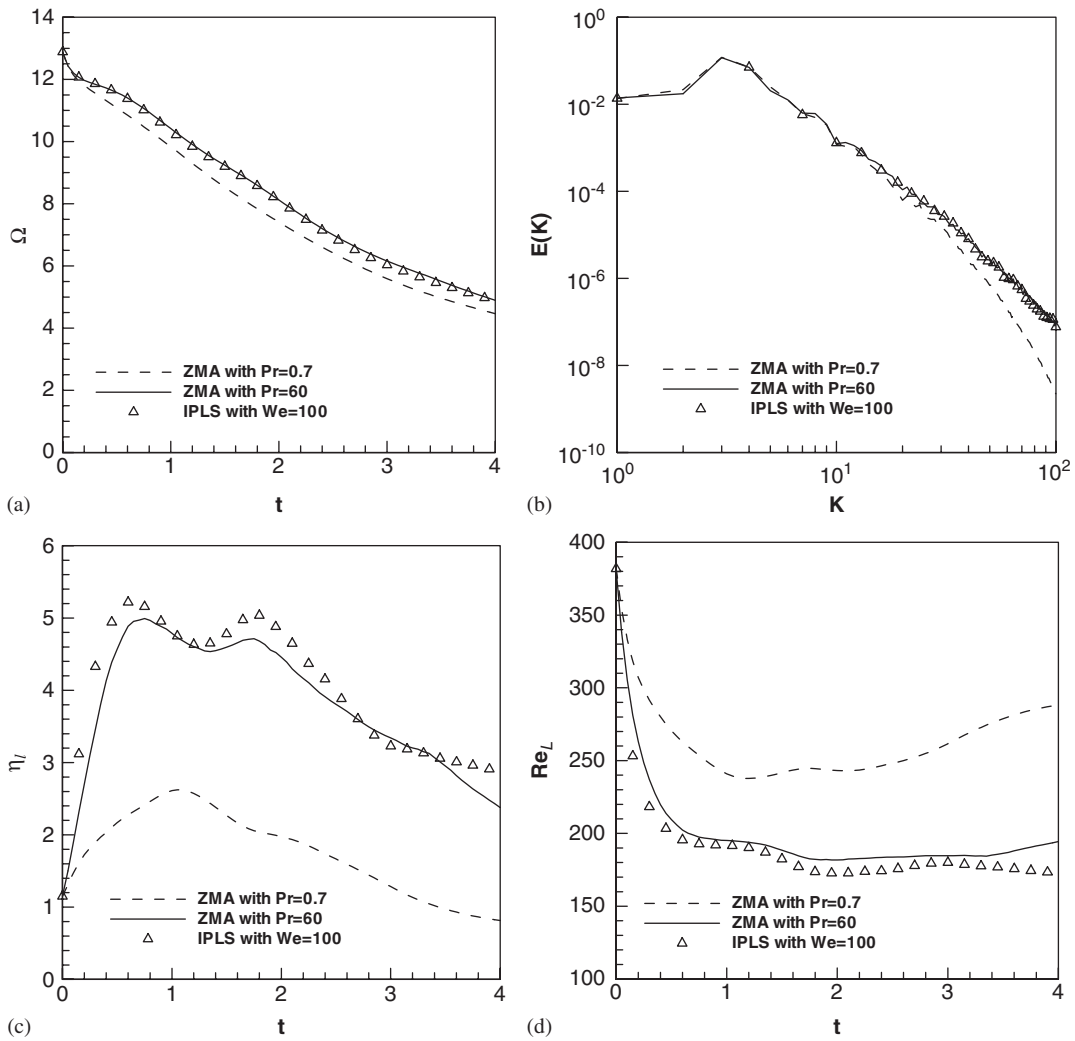

Figure 16. Turbulent statistics calculated by different methods: (a) enstrophy; (b) turbulent energy spectral density at $t = 3.0$; (c) enstrophy dissipation rate; and (d) Reynolds number based on integral scale and turbulent energy.

shows that the computed turbulent vorticity contours *via* IPLS and ZMA methods are very similar; indicating the consistency of these two methods and the accuracy of their numerical solution procedures. Further evidence is provided in Figure 16, where several statistical quantities, obtained from the ZMA and IPLS data are compared.

It is to be emphasized here that the governing equations and the numerical solution procedure in the ZMA and IPLS methods are very different. Nevertheless, the interface evolution, the vorticity field and all the flow/turbulence statistics, predicted by these two methods, are similar when the Prandtl number in the ZMA equations and the Weber number in the IPLS equations are

DOI: 10.1002/fld

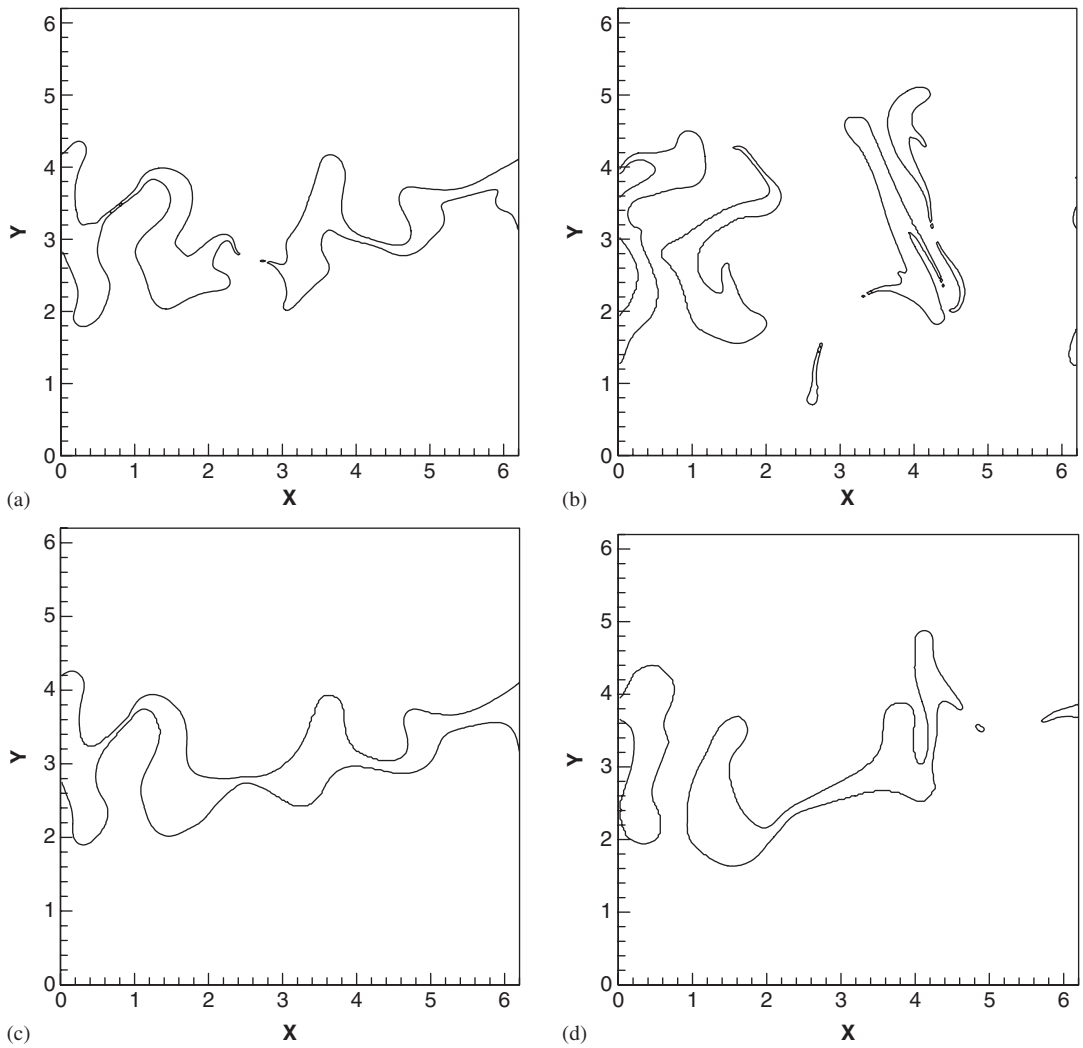

Figure 17. Interface contours in a two-fluid (oil–water) turbulent flow as predicted by the IPLS method. In figures (a) and (b) $We=100$ and in figures (c) and (d) $We=20$. Figures (a) and (c) show the results at $t=2.0$ and figures (b) and (d) show the results at $t=4.0$. In all figures, $\eta=200:1$, $L_0/L_y\approx 1/10$ and $Re=1000$.

sufficiently large. This supports the argument we made in Section 2.2 that when the surface tension and the thermal diffusion are negligible, both IPLS and ZMA methods should yield similar results. Note that, since the numerical approximation in the ZMA method is based on a high-order pseudo-spectral scheme, the similarity of the IPLS and ZMA results indicates that the IPLS method is indeed a high-order numerical method for two-fluid turbulent calculations. The results in Figures 17 and 18 confirm that the IPLS method is able to accurately predict the two-fluid turbulence and interface evolution for a variety of flow parameters. Figure 17 shows the interfaces

DOI: 10.1002/fld

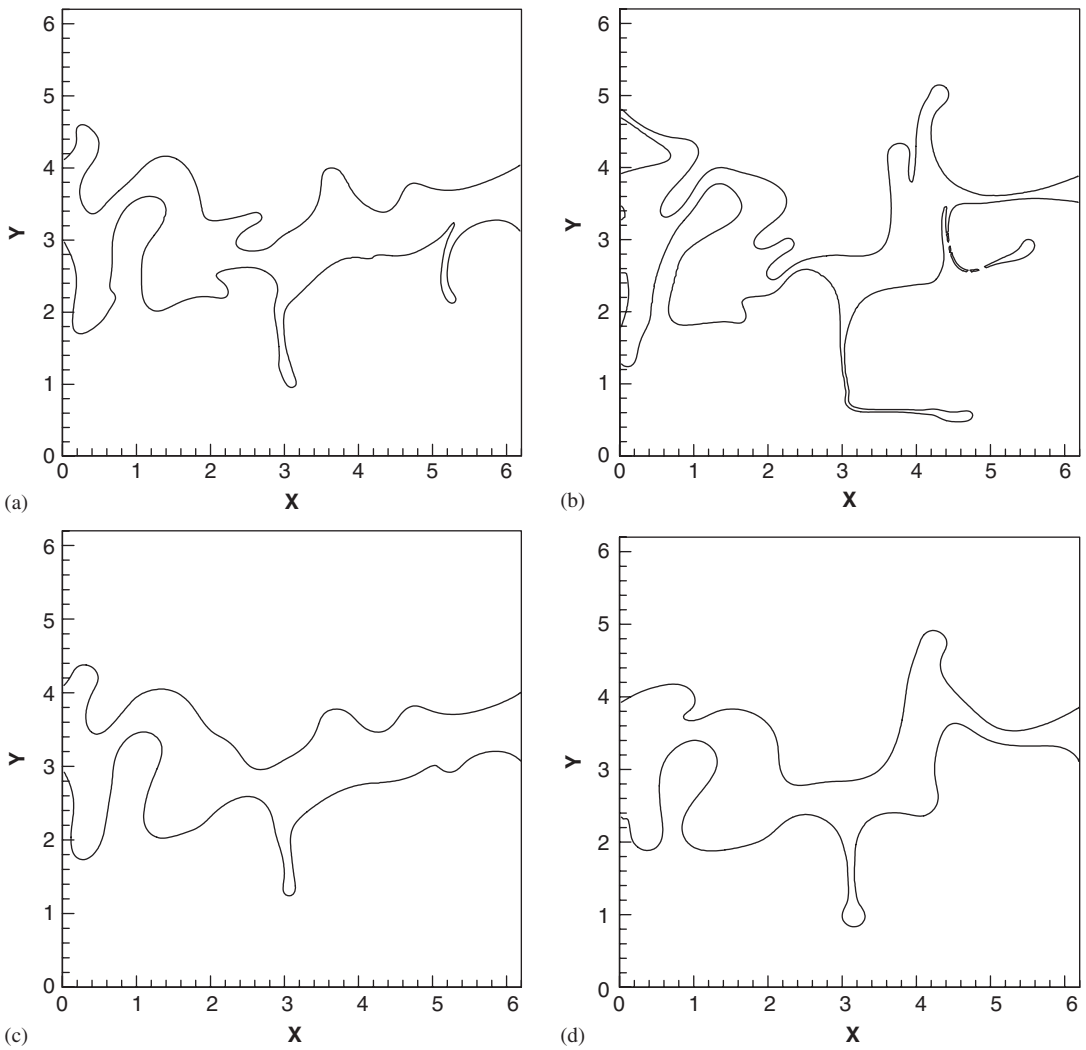

Figure 18. Interface contours in a two-fluid (oil–water) turbulent flow as predicted by the IPLS method. In figures (a) and (b) $We=100$ and in figures (c) and (d) $We=20$. Figures (a) and (c) show the results at $t=2.0$ and figures (b) and (d) show the results at $t=4.0$. In all figures $\eta=200:1$, $L_0/L_y\approx 1/6$ and $Re=1000$.

for various values of the Weber number. It is clear that for a higher Weber numbers, the interface becomes more 'active' and less 'stable' as the surface tension decreases and the interface shape becomes very complex. In fact, in some cases, the interface is broken up into fluid ligaments or droplets. Nevertheless, the IPLS method is able to capture the complicated interactions between the turbulence and the interface. The effects of the oil thickness or L_0/L_y on the interface are shown in Figure 18. A comparison between the results in Figures 17 and 18 indicates that for thinner oil layer ($L_0/L_y=1/10$), the interface is broken up quickly into several portions and the

DOI: 10.1002/fld

oil ligaments entrap in the surrounding water. In this case, the overall effect of the turbulence on the interface is indeed substantial. In contrast, for the thicker oil layer ($L_0/L_y = 1/6$), the large distance between the two neighboring vorticity fields in water reduces the global effect of turbulence and the pinching effect of the turbulence on the oil layer becomes more important. This is shown in Figure 18, where it is observed that some small ligaments or droplets are pinched off from the thick oil layer by small-scale turbulent motions, while the large-scale motions are not able to completely break up the layer. The formation of ligaments and droplets in a liquid–gas system is studied by Boeck *et al*. [52], who discussed the effects of Weber number and Reynolds number in detail. Again the results in Figures 17 and 18 demonstrate the ability of the IPLS method to capture the complicated features of the interface evolution in two-fluid isotropic turbulent flows.

6. CONCLUSIONS

A hybrid Lagrangian–Eulerian interfacial particle-level set (IPLS) method is developed for numerical simulation of two-fluid turbulent flows. In this method, the interface is identified based on the locations of notional (Lagrangian) particles and the geometrical information concerning the interface and the fluid properties are obtained from the (Eulerian) level set function. The Lagrangian particles can accurately represent the interface evolution without any numerical mass loss or gain and the level set function provides smooth geometrical information concerning the interface and its effects on the flow. The new methodology has been assessed by applying it to several standard interface-moving and two-fluid laminar problems. It has also been used for simulation of two-fluid isotropic turbulent flows under various flow/fluid conditions. The numerical results are evaluated by monitoring the mass conservation law, the turbulence energy spectral density function and the consistency between Eulerian and Lagrangian components of the method. It is shown that the IPLS method can handle interfaces with substantial topological complexity, and accurately predict the interface evolution even at the 'sub-grid range' as long as the velocity field is well resolved. The accuracy of the Navier–Stokes flow solver in the IPLS method is established first by comparing the results obtained by the IPLS method for a single-fluid isotropic turbulent flow with those obtained *via* high-order incompressible and compressible pseudo-spectral numerical schemes. The IPLS results for a two-fluid isotropic turbulent flow are also compared with those predicted by a high-order zero Mach number variable density flow solver. Despite the differences in the equations and the numerical schemes, we have found that the IPLS method generates virtually identical results with other methods. The similarity of the results confirm the accuracy of the IPLS flow solver in single-fluid and two-fluid turbulent systems.

Analysis of the velocity statistics and vorticity/interface contours in turbulent flows indicates that the destabilization effect of turbulence and the stability effect of surface tension on the interface motion are strongly dependent on the fluid density and viscosity ratios. As expected, the interface becomes more stable and less 'active' as the surface tension increases or the Weber number decreases. The turbulence can completely break up a thin layer of oil in water. However, for an initially thick layer, turbulence can only pinch off small ligaments or droplets from the layer. Despite the complex and sometimes very significant effects that the turbulence and the interface have on each other, it is shown that the IPLS method can correctly capture the interface evolution and the effect of turbulence on the interface in all the cases considered in this study.

DOI: 10.1002/fld

ACKNOWLEDGEMENTS

This work was sponsored by the National Science Foundation under grant # 0325760. Computational resources were provided by the High Performance Computing Center at Michigan State University.

REFERENCES

1. Li X. Mechanism of atomization of a liquid jet. *Atomization and Sprays* 1995; **5**:89–105.
2. Mamora DD, Sutadiwiria G. Analytical mode of distillation of oil at the steam–oil interface. *The Seventh UNITAR International Conference on Heavy Crude and Tar Sands*, Beijing, China, October 1998; 27–30.
3. Torpey PA. Prevention of air ingestion in a thermal ink-jet device. *Proceedings of the 4th International Congress on Advances in Non-impact Print Technologies*, Springfield, VA, March 1988.
4. Esmaeeli A, Tryggvason G. An inverse energy cascade in two-dimensional low Reynolds number bubble flows. *Journal of Fluid Mechanics* 1996; **314**:315–330.
5. Herrmann M. Modeling primary breakup: a three-dimensional Eulerian level set/vortex sheet method for two-phase interface dynamics. *Annual Research Briefs*. Center for Turbulence Research: Stanford, 2003; 185–195.
6. Reitz RD, Bracco FV. Mechanisms of breakup of round liquid jets. *The Encyclopedia of Fluid Mechanics*, Cheremisnoff N (ed.). Houston: Gulf, 1986; **3**:233–249.
7. Puckett EG, Almgren AS, Bell JB. A high-order projection method for tracking fluid interfaces in variable density incompressible flows. *Journal of Computational Physics* 1997; **130**:269–282.
8. Boulton-Stone JM, Blake JR. Gas bubbles bursting at a free surface. *Journal of Fluid Mechanics* 1993; **254**: 437–466.
9. Strain J. A boundary integral approach to unsteady solidification. *Journal of Computational Physics* 1989; **85**:342–389.
10. Unverdi SO, Tryggvason G. A front-tracking method for viscous, incompressible, multi-fluid flows. *Journal of Computational Physics* 1992; **100**:25–37.
11. Sussman M, Smereka P. Axisymmetric free boundary problems. *Journal of Fluid Mechanics* 1997; **341**:269–294.
12. Helmsen JJ. A comparison of three-dimensional photolithography simulators, *Ph.D. Thesis*, U.C., Berkeley, 1994.
13. Osher S, Sethian JA. Fronts propagating with curvature-dependent speed: algorithms based on Hamilton–Jacobi formulations. *Journal of Computational Physics* 1988; **79**:12–49.
14. Mashayek F, Ashgriz N. Advection of axisymmetric interfaces by volume of fluid method. *International Journal for Numerical Methods in Fluids* 1995; **20**:1337–1361.
15. Mashayek F, Ashgriz N. A hybrid finite element—volume of fluid method for simulating free surface flows and interfaces. *International Journal for Numerical Methods in Fluids* 1995; **20**:1363–1380.
16. Scardovelli R, Zaleski S. Direct numerical simulation of free-surface and interfacial flow. *Annual Review of Fluid Mechanics* 1999; **31**:567–603.
17. Ashgriz N, Poo JY. FLAIR: flux line-segment advection and interface reconstruction. *Journal of Computational Physics* 1991; **93**:449–468.
18. Youngs DL. Time-dependent multi-materials flow with large fluid distortion. In *Numerical Simulation for Fluid Dynamics*, Morton KW, Baines MJ (eds). Academic Press: New York, 1982; 273–285.
19. Pilliod JE, Puckett EG. Second-order accurate volume-of-fluid algorithms for tracking material interfaces. *Technical Report No. LBNL 40744*. Lawrence Berkeley National Laboratory, 1997.
20. Chang YC, Hou TY, Merriman B, Osher S. A level set formulation of Eulerian interface capturing methods for incompressible fluid flows. *Journal of Computational Physics* 1996; **124**:449–464.
21. Chen S, Merriman B, Osher S, Smereka P. A simple level set method for solving Stefan problem. *Journal of Computational Physics* 1997; **135**:8–29.
22. Mulder W, Osher S, Sethian JA. Computing interface motion in compressible gas dynamics. *Journal of Computational Physics* 1992; **100**:209–228.
23. Russo G, Smereka P. A level-set method for the evolution of faceted crystals. *SIAM Journal on Scientific Computing* 2000; **21**:2073–2095.
24. Sussman M, Smereka P, Osher S. A level set approach for computing solution to incompressible two-phase flow. *Journal of Computational Physics* 1994; **114**:146–159.
25. Hou TY, Li Z, Osher S, Zhao H. A hybrid method for moving interface problems with application to the Hele–Shaw flow. *Journal of Computational Physics* 1997; **134**:236–252.

DOI: 10.1002/fld

26. 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.
27. Sussman M. A second order coupled level set and volume of fluid method for computing the growth and collapse of vapor bubbles. *Journal of Computational Physics* 2003; **187**:110–136.
28. Bell JB, Colella P, Glaz HM. A second-order projection method for the incompressible Navier–Stokes equations. *Journal of Computational Physics* 1989; **85**:257–283.
29. Bell JB, Marcus DL. A second-order projection method for variable-density flows. *Journal of Computational Physics* 1992; **101**:334–348.
30. Zhao Y, Tan HH, Zhang B. A high-resolution characteristics-based implicit dual time-stepping VOF method for free surface flow simulation on unstructured grids. *Journal of Computational Physics* 2002; **183**:233–273.
31. Aulisa E, Manservisi S, Scardovelli R. A mixed markers and volume-of-fluid method for reconstruction and advection of interfaces in two-phase and free-boundary flows. *Journal of Computational Physics* 2003; **188**: 611–639.
32. 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.
33. Hieber SE, Koumoutsakou P. A Lagrangian particle level set method. *Journal of Computational Physics* 2005; **210**:342–367.
34. Raad PE, Bidoae R. The three-dimensional Eulerian–Lagrangian marker and micro cell method for the simulation of free surface flows. *Journal of Computational Physics* 2005; **203**:668–699.
35. Sussman M, Almgren AS *et al*. An adaptive level set approach for incompressible two-phase flows. *Journal of Computational Physics* 1999; **148**:81–124.
36. Cook AW, Riley JJ. Direct numerical simulation of a turbulent reactive plume on a parallel computer. *Journal of Computational Physics* 1996; **129**:263–283.
37. Almgren AS, Bell JB, Szymczak WG. A numerical method for the incompressible Navier–Stokes equations based on an approximate projection. *SIAM Journal on Scientific Computing* 1996; **17**:358–369.
38. Kim J, Moin P. Application of a fractional-step method to incompressible Navier–Stokes equations. *Journal of Computational Physics* 1985; **59**:308–323.
39. Yu X. Iterative pressure Poisson equation method for solving the unsteady incompressible N–S equations. *Chinese Journal of Numerical Mathematics and Application* 2001; **23**:447–456.
40. Briggs WL. *A Multigrid Tutorial*. SIAM: Philadelphia, PA, 1987.
41. Tatebe O. The multigrid preconditioned conjugate gradient method. *Sixth Copper Mountain Conference on Multigrid Methods*, Copper Mountain, CO, 4–9 April 1993.
42. Alcouffe RE, Brandt A, Dendy JE, Painter JW. The multigrid method for the diffusion equation with strongly discontinuous coefficient. *SIAM Journal on Scientific Computing* 1981; **2**:430–454.
43. Gamet L, Ducros F, Nicoud F, Poinsot T. Compact finite difference schemes on non-uniform meshes. Application to direct numerical simulations of compressible flows. *International Journal for Numerical Methods in Fluids* 1999; **29**:159–191.
44. Osher S, Fedkiw R. *Level Set Methods and Dynamic Implicit Surfaces*. Applied Mathematical Sciences. Springer: New York, 2002.
45. Lele SK. Compact finite difference schemes with spectral-like resolution. *Journal of Computational Physics* 1992; **103**:16–42.
46. Dupont TF, Liu Y. Back and forth error compensation and correction methods for removing errors induced by uneven gradients of the level set function. *Journal of Computational Physics* 2003; **190**:311–324.
47. Ceniceros HD, Roma AM. A multi-phase flow method with a fast, geometry-based fluid indicator. *Journal of Computational Physics* 2005; **205**:391–400.
48. Najm HN, Wyckoff PS, Knio OM. A semi-implicit numerical scheme for reacting flow. I. Stiff chemistry. *Journal of Computational Physics* 1998; **143**:381–402.
49. Jaberi FA, James S. Effects of chemical reaction on two-dimensional turbulence. *Journal of Scientific Computing* 1999; **14**:31–72.
50. Zalesak ST. Fully multidimensional flux-corrected transport, algorithms for fluid. *Journal of Computational Physics* 1979; **31**:335–362.
51. Ong B. Level set framework for fronts propagating with curvature dependent speeds. *Level Set Research-Curvature Motion*, 2002. http://mathgeek.us/research/papers/curvature/curvature.pdf.
52. Boeck T, Li J, Lopez-Pages E, Yecko P, Zaleski S. Ligament formation in sheared liquid–gas layers. *Theoretical and Computational Fluid Dynamics* 2007; **21**:59–76.

DOI: 10.1002/fld