Journal of Computational Physics 406 (2020) 109188

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

On the development of LS-assisted VOF method for incompressible interfacial flows

H.L. Wen [a], C.H. Yu [b], Tony W.H. Sheu [a,c,d,*]

[a] Department of Engineering Science and Ocean Engineering, National Taiwan University, Taipei, R.O. China
[b] State key lab of Hydraulics and Mountain River Engineering, Sichuan University, Sichuan 610000, PR China
[c] Institute of Applied Mathematical Sciences, National Taiwan University, Taipei, R.O. China
[d] Center for Advanced Study in Theoretical Sciences, National Taiwan University, Taipei, R.O. China

ARTICLE INFO

Article history:
Received 19 June 2019
Received in revised form 30 October 2019
Accepted 12 December 2019
Available online 31 December 2019

Keywords:
VOF
LS
Moving interface
Bubble bursting
Bubble impacting

ABSTRACT

In this study we are aimed to simulate incompressible fluid flows with moving interface separating liquid and gas phases. To achieve volume/mass conservation and to capture interface, the interface-capturing volume of fluid (VOF) method, which is exercised in compliance with requirement of volume/mass conservation, will be coupled with the other interface-capturing level-set (LS) method, which is suitable to capture interface accurately. In our proposed advection algorithm, VOF is the building block that solves the volume fraction and the level-set function is solely used to assist an accurate calculation of some geometrically relevant quantities at the interface. A high order scheme developed within the optimized compact reconstruction WENO framework has been applied to solve the advection equation. The novelty of this purposed advection algorithm is attributed to its efficient implementation without sacrifice of computational accuracy.

1. Introduction

Modeling two-phase flow motion with sharply evolving interfaces is challenging because of discontinuous physical properties across interfaces and the difficulties in capturing interfaces and maintaining their sharpness [1]. The shape of interface can be predicted using a method developed either in Lagrangian or Eulerian sense. A Lagrangian method adopts moving mesh which is convected with fluid flow to follow the interface, where the interface is treated as a sharp boundary. One major drawback of this class of methods is owing to the difficulty of tracking topological changes, such as bubble breakup and coalescence [2,3]. Besides, this method is computationally expensive. For these reasons, Lagrangian methods have been comparatively less applied.

In contrast to Lagrangian methods implemented in moving mesh, Eulerian methods are performed in fixed grids that remain stationary in space with flow moving through it. A few Eulerian methods have been proposed to predict gas-liquid flow and capture its moving interface, namely, the interface tracking methods and the interface capturing methods. In interface tracking methods, fixed grids are involved to get the velocity field, and the interface is explicitly tracked through the use of Lagrangian-type moving mesh by interface marker particles [4–10]. The need of performing surface remeshing makes this method computationally expensive, in addition to its implementation difficulty [11]. In interface capturing methods, the interface is implicitly predicted using a phase function on fixed grids, and the interface motion is obtained through the

* Corresponding author at: Department of Engineering Science and Ocean Engineering, National Taiwan University, Taipei, R.O. China.
E-mail address: twhsheu@ntu.edu.tw (T.W.H. Sheu).

https://doi.org/10.1016/j.jcp.2019.109188
0021-9991/

advection of phase function. Some well-known interface capturing methods include the Level Set (LS) [12–14], Volume of Fluid (VOF) [15–18,20–22], and Coupled Level Set and Volume of Fluid (CLSVOF) methods [23–27].

LS method was first proposed by Osher and Sethian [12] and it has been further improved by Sussman et al. [13] for achieving a better simulation of two-phase flows. LS method uses a smooth continuous distance function ϕ to describe its moving interface. The function ϕ denotes the shortest signed distance to the interface, and its magnitude is positive in one fluid and negative in the other, implying the presence of zero level set at the interface. LS method can be used to accurately calculate the unit normal vector, mean curvature and, therefore, surface tension force. As a result, it can easily model topological change problems such as wave breakup and bubble merger. Moreover, it can be easily generalized from two dimensions to three dimensions. However, the drawback of LS method is attributed to its poor ability of maintaining volume conservation, which can lead to an erroneously predicted shape of interface.

VOF method, proposed by Hirt and Nichols [15], uses an Eulerian approach to simulate flow motions which are separated by a common interface in fixed grid system. The interface is approximated by either horizontal or vertical lines. VOF method involves using an indicator function F to implicitly capture the moving interface. The value of this function is zero in one fluid and one in the other. There are two major approaches to solve the VOF equation. One involves discretizing the advection equation of indicator function using a high-resolution scheme, which is easier to implement but it would yield a diffusive interface. The other involves reconstructing the geometric interface through Simple Line Interface Calculation (SLIC) [28], Piecewise Linear Interface Calculation (PLIC) [29] and Weighted Line Interface Calculation (WLIC) [31] can be chosen as well. VOF method is characterized by its extremely good volume conservation property, but it has difficulty of accurately computing the geometric quantities such as the unit normal vector, mean curvature and surface tension force, which can altogether render nonphysical flow physics (spurious currents) [16–18].

In light of the above mentioned facts that LS and VOF are complimentary to each other, it seems reasonable to combine them together to construct an attractive refined model. Thus, taking all the good aspects of LS method and VOF method into account, at the same time, makes it possible to establish the proposed LS-assisted VOF advection method. Similar idea was first proposed by Bourlioux [23] and it was further made popular by Sussman and Puckett [24], Son and Hur [25] and Son [26]. This class of methods can be used to calculate the geometric quantities accurately and can retain volume conservation well.

Interface reconstruction is a key issue in improving the prediction accuracy for this class of methods, since it may lead to numerical instability in the simulation. Reconstruction of interface can be done by the modification of level set function from volume fraction. In the previous work [24,33], the level set function is solved iteratively to be the exact signed distance from the location to the reconstructed interface in two-dimensional grids. In the paper presented by Son in [26], the interface is constructed similarly and the geometric relations in Cartesian grids have been generalized. While this type of reconstruction can be implemented accurately with an analytic geometrical relation [34], its geometric relation is not necessarily applicable to all types of mesh under consideration. One can refer to another type of interface reconstruction given in [35], which incorporates Lagrangian transportation of the interface. In the present work, we develop an efficient interface reconstruction algorithm which is applicable to all types of grid system. The proposed interface reconstruction procedure can be easily adopted to the classical level set method, without performing any **if-else** logic during the computer coding. Moreover, reconstruction of an interface with some specific ways helps to gain computational efficiency, without a significant sacrifice of the computational accuracy.

The organization of this paper is as follows. In Section 2, the two-phase incompressible Navier-Stokes equations formulated in primitive-variable form and the proposed LS-assisted VOF coupling algorithm are presented. Section 3 is devoted to the approximation of LS equation using the optimized compact reconstruction weighted essentially non-oscillatory (OCR-WENO) scheme. For the verification and validation sakes, in Section 4, four benchmark problems including two and three dimensional vortex deforming problems and one Rayleigh-Taylor instability problem are investigated using the proposed algorithm. In Section 5, two droplet/bubble hydrodynamic problems are investigated. Concluding remarks are drawn in Section 6.

2. Mathematical model

Our interface capturing model contains two major building blocks, namely, the LS and VOF methods, and they will be described in Section 2.1.1 and 2.1.2, separately. Then, the proposed LS-assisted VOF advection method will be detailed in Section 2.1.3. Besides the description of the interface capturing mathematical model, in Section 2.2 the incompressible Navier-Stokes equations used in this study are detailed as well.

2.1. Interface capturing method

2.1.1. Level set (LS) method

In the LS method, the level-set function ϕ is defined as the signed distance measured from the interface Γ. The value of ϕ is positive in liquid phase Ω_1, negative in gas phase Ω_2, and is defined to be zero on the interface Γ:

$$\phi(\mathbf{x}, t=0) = \begin{cases} \hat{d}; & \text{for } \mathbf{x} \in \Omega_1, \\ 0; & \text{for } \mathbf{x} \in \Gamma, \\ -\hat{d}; & \text{for } \mathbf{x} \in \Omega_2. \end{cases} \tag{1}$$

Note that $\hat{d}$ is the shortest distance from a given point $\mathbf{x}$ to the interface Γ. The following level set equation is solved for ϕ to track the evolution of the interface:

$$\phi_t + \mathbf{u} \cdot \nabla \phi = 0. \tag{2}$$

Take the two-dimensional case as an example, Eq. (2) can be rewritten as follows for incompressible fluid flow

$$\phi_t + \nabla \cdot (\mathbf{u}\phi) = \phi_t + \frac{(u\phi)_{i+1/2,j} - (u\phi)_{i-1/2,j}}{\Delta x} + \frac{(v\phi)_{i,j+1/2} - (v\phi)_{i,j-1/2}}{\Delta y} = 0. \tag{3}$$

After solving the Eq. (3), the level set function ϕ can no longer be kept as a signed distance function because of the introduced dissipation and dispersion discretization errors. To demand ϕ as a signed distance function, a re-initialization step has been devised by calculating the ϕ values at grid points away from the interface [36]:

$$\phi_\tau = \bar{S}(\phi_0)(1 - |\nabla\phi|) + \lambda\delta(\phi)|\nabla\phi|, \tag{4}$$

where τ is the pseudo-time and Δx is the local grid spacing. It is noted here that $\phi_0 = \phi(t, \tau = 0)$. The parameter λ shown in Eq. (4) is as follows [36]

$$\lambda = -\frac{\int_{\Omega_{i,j}} \delta(\phi)(\bar{S}(\phi_0)(1 - |\nabla\phi|))d\Omega}{\int_{\Omega_{i,j}} \delta^2(\phi)|\nabla\phi| d\Omega}. \tag{5}$$

In the above equation, the delta function $\delta(\phi)$ is approximated as:

$$\delta(\phi) = \begin{cases} \frac{1}{2\varepsilon}(1 + cos(\frac{\pi\phi}{\varepsilon})) & ; \text{ if } |\phi| < \varepsilon, \\ 0 & ; \text{ otherwise.} \end{cases} \tag{6}$$

$\bar{S}(\phi_0)$ in Eq. (4) denotes the smoothed sign function

$$\bar{S}(\phi_0) = 2\left(\bar{H}(\phi_0) - 0.5\right), \tag{7}$$

where

$$\bar{H}(\phi) = \begin{cases} 0; & \text{if } \phi < -\varepsilon, \\ \frac{1}{2}[1 + \frac{\phi}{\varepsilon} + \frac{1}{\pi}\sin(\frac{\pi\phi}{\varepsilon})]; & \text{if } |\phi| \le \varepsilon, \\ 1; & \text{if } \phi > \varepsilon, \end{cases} \tag{8}$$

and ε is chosen to be $1.5\Delta x$ in this study.

2.1.2. Volume of fluid (VOF) method

In VOF method, a characteristic function χ is chosen to represent the interface. For the two-dimensional case, as an example, χ is given below:

$$\chi(x, y) = \begin{cases} 1; & \text{if } (x, y) \in \Omega_1, \\ 0; & \text{if } (x, y) \in \Omega_2. \end{cases} \tag{9}$$

Note that $\chi(x, y)$ is the value on (x, y) in the whole computational domain. Therefore, an integral of the function χ is introduced, called as the color function C:

$$C_{i,j} = \frac{1}{\Delta x \Delta y} \int_{x_{i-1/2,j}}^{x_{i+1/2,j}} \int_{y_{i,j-1/2}}^{y_{i,j+1/2}} \chi(x, y) dx dy. \tag{10}$$

The characteristic function χ is advected by virtue of the following equation

$$\frac{\partial \chi}{\partial t} + \nabla \cdot (\mathbf{u}\chi) - \chi \nabla \cdot \mathbf{u} = 0. \tag{11}$$

For Eq. (11), the dimensional splitting algorithm is applied at an interior point (i, j). The flux terms given below are calculated separately along x and y directions:

$$C^*_{i,j} = C^n_{i,j} - \frac{F^n_{x,i+1/2,j} - F^n_{x,i-1/2,j}}{\Delta x} - C^n_{i,j}\frac{u_{i+1/2,j} - u_{i-1/2,j}}{\Delta x}\Delta t, \tag{12}$$

$$C^{n+1}_{i,j} = C^*_{i,j} - \frac{F^*_{y,i,j+1/2} - F^*_{y,i,j-1/2}}{\Delta y} - C^n_{i,j}\frac{v_{i,j+1/2} - v_{i,j-1/2}}{\Delta y}\Delta t, \tag{13}$$

where $F_{x,i+1/2,j}$ and $F_{y,i,j+1/2}$ are

$$F_{x,i+1/2,j} = -\frac{1}{\Delta y}\int_{y_{i,j-1/2}}^{y_{i,j+1/2}}\int_{x_{i+1/2,j}}^{x_{i+1/2,j}-u_{i+1/2,j}\Delta t}\chi_{is,j}(x,y)dxdy, \tag{14}$$

$$F_{y,i,j+1/2} = -\frac{1}{\Delta x}\int_{y_{i,j+1/2}}^{y_{i,j+1/2}-v_{i,j+1/2}\Delta t}\int_{x_{i-1/2,j}}^{x_{i+1/2,j}}\chi_{i,js}(x,y)dxdy. \tag{15}$$

In the above equations, $F_{x,i+1/2,j}$ and $F_{y,i,j+1/2}$ denote the advection fluxes in x and y directions, respectively. The subscripts is and js are defined as

$$is = \begin{cases} i; & \text{if } u_{i+1/2,j} \geq 0, \\ i+1; & \text{if } u_{i+1/2,j} < 0, \end{cases} \tag{16}$$

and

$$js = \begin{cases} j; & \text{if } v_{i,j+1/2} \geq 0, \\ j+1; & \text{if } v_{i,j+1/2} < 0. \end{cases} \tag{17}$$

The tangent of hyperbola for interface capturing (THINC) [30] scheme with the weighed linear interface calculation (WLIC) [31] introduced in Section 3 is used to calculate the VOF flux terms $F_{x,i+1/2,j}$ and $F_{y,i,j+1/2}$.

2.1.3. LS-assisted VOF coupling solution algorithm

While volume conservation can be retained extremely well in VOF method, it has difficulty of getting an accurately calculated geometric quantities on complex interface [48] (see Appendix AA). In LS method, the distance function ϕ is continuous and the geometry of the interface can be easily described by ϕ. However, application of LS method can not guarantee volume conservation. To achieve good volume conservation and to capture interface accurately motivate us to combine the LS and VOF methods. In the proposed method, our coupling strategy takes the advantage of the interface smoothness of the level set method and the excellent volume conservation attribute of the VOF method.

In the proposed method, level set function ϕ and volume fraction C are advected by solving the respective governing equations from t^n to t^{n+1}. To ensure the predicted interfaces from both ϕ and C are sufficiently close to each other, reconstruction of interface is needed after the advections of ϕ and C.

In the reconstruction procedure, an intermediate function $\eta = 2C^{n+1} - 1$ is introduced. The iso-line $\eta = 0$ denotes the position of the interface, which is identical to the interface described by $C^{n+1} = 1/2$ (or $\eta = 0$). Next, to make η to be a distance function at the interface, the following equation is solved iteratively until η reaches its steady-state solution at the interface.

$$\frac{\partial \eta}{\partial \hat{\tau}} + \bar{S}(\eta_0)(|\nabla\eta| - 1) = 0. \tag{18}$$

Another intermediate level set function $\phi^{n+1,*}$, constructed from ϕ^{n+1} and η, is introduced to ensure that $\phi^{n+1,*}$ and C^{n+1} have almost the same position of interface, that is,

$$\phi^{n+1,*} = \begin{cases} \eta, & \text{if } |\phi^{n+1}| \leq \varepsilon, \\ \phi^{n+1}, & \text{if } |\phi^{n+1}| > \varepsilon. \end{cases} \tag{19}$$

To avoid the discontinuity at $|\phi^{n+1,*}| = \varepsilon$, the following equation is solved iteratively to steady state,

$$\frac{\partial \phi^{n+1,*}}{\partial \bar{\tau}} + \bar{S}(\phi^{n+1,*}_0)(|\nabla\phi^{n+1,*}| - 1) = \lambda\delta(\phi^{n+1,*})|\nabla\phi^{n+1,*}|. \tag{20}$$

The parameter λ shown in the above equation is given in Eq. (5). It should be pointed out that although discontinuities occur in the region where $\delta(\phi^{n+1,*}) = 0$, it is still necessary to solve Eq. (20) with a non-zero value of λ to retain good volume conservation ($\frac{\partial \int_\Omega H(\phi^{n+1,*})\, d\Omega}{\partial \bar{\tau}} = 0$) during the iteration [36].

After solving Eq. (20) to steady state, $\phi^{n+1,*}$ is the continuous distance function with the same interface as C^{n+1}. The above procedure describes the way of reconstructing of the level set function from volume fraction, including solving the

initialization, Eq. (18), and the reinitialization, Eq. (20). Note that reconstruction procedure is not necessarily applied only in each step. Rather, it is recommended to apply one reconstruction procedure only after every ten-step calculation. One can refer to Appendix B to see the importance of introduction of η function and to Appendix C for the determination of a proper frequency of performing reconstruction.

In contrast to the previously developed method [23–27], the reconstructed interface in our proposed method has advantages in its implementation, which can be easily adopted to the classical level set solver. Also, three-dimensional solver can be extended directly from the two-dimensional cases for our proposed method. Without performing any **if-else** logic in this solver, the proposed method has also a great potential to be applied in parallel computing for the better computational efficiency.

2.2. Integration of Navier-Stokes equations with level set function and volume fraction

Fluid behaviors on both sides of the interface should be computed when adopting the proposed method to predict the time-evolving of interface. In the simulation of two-phase flow of air and water with the proposed method, the classical three-dimensional Navier-Stokes equation is modified to include variable density, viscosity and surface force terms. The resulting set of Navier-Stokes equation, subject to volume conservation equation (21), is written as follows:

$$\nabla \cdot \mathbf{u} = 0, \tag{21}$$

and

$$\mathbf{u_t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\frac{\nabla p}{\rho(C)} + \frac{\nabla \cdot (2\mu(C)\mathbf{D})}{\rho(C)} - \frac{\sigma\kappa(\phi)\nabla\phi\delta(\phi)}{\rho(C)} + \mathbf{F}, \tag{22}$$

where $\mathbf{u} = (u, v, w)$ is the fluid velocity, p the pressure field, ϕ the level set function, $\mathbf{D} = \frac{1}{2}((\nabla\mathbf{u}) + (\nabla\mathbf{u})^T)$ the strain tensor, and $\mathbf{F} = g\hat{\mathbf{e}}_g$ the gravitational force (in the direction $\hat{\mathbf{e}}_g$). ρ and μ denote the density and viscosity, respectively. Note that $\kappa(\phi) = \nabla \cdot \mathbf{n} = \nabla \cdot (\frac{\nabla\phi}{|\nabla\phi|})$ denotes the curvature of the interface of interest, and σ is the surface tension coefficient.

Non-dimensional analysis is a widely-used technique in fluid mechanics. In this study, the following four characteristic scales are adopted: L_c, U_c, $P_c = \rho_L U_c^2$, $T_c = L_c/U_c$, where L_c is the characteristic length, U_c the characteristic velocity, P_c the characteristic pressure and T_c the characteristic time. By virtue of the above chosen characteristic scales to normalize the variables shown in the volume continuity equation and momentum continuity equation, one can get the following non-dimensionalized variables:

$$\begin{aligned} &x = x^* L_c,\ y = y^* L_c,\ z = z^* L_c,\ t = t^* T_c = t^*(L_c/U_c),\ \rho = \rho^* \rho_L, \\ &u = u^* U_c,\ v = v^* U_c,\ w = w^* U_c,\ p = p^* P_c = p^*(\rho_L U_c^2),\ \mu = \mu^* \rho_L. \end{aligned} \tag{23}$$

Eq. (21) and Eq. (22) can then be rewritten to the following dimensionless equations:

$$\nabla \cdot \mathbf{u} = 0, \tag{24}$$

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\frac{1}{\rho(C)}\nabla p + \frac{1}{Re}\frac{\nabla \cdot (2\mu(C)\mathbf{D})}{\rho(C)} + \frac{1}{Fr^2}\hat{\mathbf{e}}_g - \frac{1}{We}\frac{\kappa(\phi)\delta(\phi)\nabla\phi}{\rho(C)}. \tag{25}$$

Note that all the superscripts * shown above have been omitted for convenience. Dimensionless density ρ^* and viscosity μ^* are given in the following forms, respectively,

$$\begin{aligned} \rho^*(C) &= C + (\frac{\rho_G}{\rho_L})(1 - C), \\ \mu^*(C) &= C + (\frac{\mu_G}{\mu_L})(1 - C), \end{aligned} \tag{26}$$

where C is the volume fraction function, or, the color function. The subscripts G and L in Eq. (26) represent the gas phase and liquid phase, respectively. Three dimensionless parameters are known as the Reynolds number Re for representing the ratio of the magnitudes of the inertial force to viscous force, Weber number We for representing the ratio of the magnitudes of the inertial force to surface tension force, and Froude number Fr for representing the ratio of the magnitudes of the inertial force to gravity. The three flow parameters are expressed as below:

$$Re = \frac{\rho_L L_c U_c}{\mu_L}, \qquad We = \frac{\rho_L L_c U_c^2}{\sigma}, \qquad Fr = \frac{U_c}{\sqrt{g L_c}}. \tag{27}$$

3. Numerical scheme

3.1. Level set solution solver

In the present calculation, the convection term in level set advection and re-initialization equations are discretized using the proposed OCRWENO4 and HJ-WENO [38] schemes, respectively. The third-order total variation diminishing Runge-Kutta (TVD-RK3) scheme is used for time advancement [39].

3.1.1. Approximation of spatial derivatives in level set equation

To get an accurately predicted interface, a numerical scheme of higher accuracy order with less dispersion error in smooth regions shall be chosen. We are also aimed to avoid oscillatory solutions near discontinuities. For achieving the above two goals simultaneously, the optimized compact reconstruction weighted essentially non-oscillatory (OCRWENO4) scheme [41,42] for the convective flux term in the level set equation is developed. In two-dimensional space, the convective flux term in Eq. (3) can be discretized as below

$$\mathbf{U}\cdot\nabla\phi=\nabla\cdot\left(\mathbf{U}\phi\right)=\frac{F_{i+1/2,j}-F_{i-1/2,j}}{\Delta x}+\frac{G_{i+1/2,j}-G_{i-1/2,j}}{\Delta y}. \tag{28}$$

In the above equation, $F_{i+1/2,j}$ and $G_{i,j+1/2}$ are the numerical fluxes reconstructed at the cell face along x, y direction, respectively.

Reconstruction of convective fluxes lies in the use of Lax-Friedrichs splitting method such that the term $F_{i+1/2,j}$ can be written as follows

$$F_{i+1/2,j}=\frac{1}{2}\left(\breve{F}^{L}_{i+1/2,j}+\hat{F}^{R}_{i+1/2,j}\right)=\frac{1}{2}\left((u^{+}\phi)^{L}_{i+1/2,j}+(u^{-}\phi)^{R}_{i+1/2,j}\right). \tag{29}$$

The expression of $G_{i,j+1/2}$ can be derived similarly as well. Note that $u^{+}=u+|u|$ and $u^{-}=u-|u|$, and the superscripts L, R denote the reconstruction of OCRWENO4 scheme from the left- and right-biased interpolations, respectively. The value of $\breve{F}^{L}_{i+1/2,j}=(u^{+}\phi)^{L}_{i+1/2,j}$ can be obtained by solving the following tridiagonal matrix equation [41]

$$\begin{aligned}&\left[\frac{2\omega_1^L+\omega_2^L}{3}\right]\breve{F}^{L}_{i-\frac{1}{2}}+\left[\frac{\omega_1^L+2(\omega_2^L+\omega_3^L)}{3}\right]\breve{F}^{L}_{i+\frac{1}{2}}+\frac{\omega_3^L}{3}\breve{F}^{L}_{i+\frac{3}{2}}\\&=\frac{\omega_1^L}{6}\breve{F}_{i-1}+\left[\frac{5(\omega_1^L+\omega_2^L)+\omega_3^L}{6}\right]\breve{F}_i+\left[\frac{\omega_2^L+5\omega_3^L}{6}\right]\breve{F}_{i+1}.\end{aligned} \tag{30}$$

In the above equation, $\omega_k^L, k=1,2,3$, are the weighting factors associated with the smoothness indicators $\beta_k^L, k=1,2,3$. They are used to detect the degree of discontinuity in grid stencil to properly interpolate the numerical flux at cell face. Expressions of ω_k^L and β_k^L are given as follows

$$\begin{aligned}\omega_k^L&=\frac{\alpha_k^L}{\Sigma_k\,\alpha_k^L},\ \alpha_k^L=c_k\left(1+\frac{|\beta_3^L-\beta_1^L|}{\epsilon+\beta_i^L}\right),\\\beta_1^L&=\frac{13}{12}(\breve{F}_{i-2}-2\breve{F}_{i-1}+\breve{F}_i)^2+\frac{1}{4}(\breve{F}_{i-2}-4\breve{F}_{i-1}+3\breve{F}_i)^2,\\\beta_2^L&=\frac{13}{12}(\breve{F}_{i-1}-2\breve{F}_i+\breve{F}_{i+1})^2+\frac{1}{4}(\breve{F}_{i-1}-\breve{F}_{i+1})^2,\\\beta_3^L&=\frac{13}{12}(\breve{F}_i-2\breve{F}_{i+1}+\breve{F}_{i+2})^2+\frac{1}{4}(3\breve{F}_i-4\breve{F}_{i+1}+\breve{F}_{i+2})^2.\end{aligned} \tag{31}$$

The corresponding tridiagonal matrix equation for $\hat{F}^{R}_{i+1/2,j}=(u^{-}\phi)^{R}_{i+1/2,j}$ is given below

$$\begin{aligned}&\left[\frac{2\omega_1^R+\omega_2^R}{3}\right]\hat{F}^{R}_{i+\frac{3}{2}}+\left[\frac{\omega_1^R+2(\omega_2^R+\omega_3^R)}{3}\right]\hat{F}^{R}_{i+\frac{1}{2}}+\frac{\omega_3^R}{3}\hat{F}^{R}_{i-\frac{1}{2}}\\&=\frac{\omega_1^R}{6}\hat{F}_{i+2}+\left[\frac{5(\omega_1^R+\omega_2^R)+\omega_3^R}{6}\right]\hat{F}_{i+1}+\left[\frac{\omega_2^R+5\omega_3^R}{6}\right]\hat{F}_i.\end{aligned} \tag{32}$$

The weighting factors associated with the smoothness indicators are given as

$$\begin{aligned}
\omega_k^R &= \frac{\alpha_k^R}{\Sigma_k \alpha_k^R},\ \alpha_k^R = c_k\left(1+\frac{|\beta_3^R-\beta_1^R|}{\epsilon+\beta_k^R}\right),\\
\beta_1^R &= \frac{13}{12}(\hat{F}_{i+1}-2\hat{F}_{i+2}+\hat{F}_{i+3})^2+\frac{1}{4}(3\hat{F}_{i+1}-4\hat{F}_{i+2}+\hat{F}_{i+3})^2,\\
\beta_2^R &= \frac{13}{12}(\hat{F}_{i}-2\hat{F}_{i+1}+\hat{F}_{i+2})^2+\frac{1}{4}(\hat{F}_{i}-\hat{F}_{i+2})^2,\\
\beta_3^R &= \frac{13}{12}(\hat{F}_{i-1}-2\hat{F}_{i}+\hat{F}_{i+1})^2+\frac{1}{4}(\hat{F}_{i-1}-4\hat{F}_{i}+3\hat{F}_{i+1})^2.
\end{aligned} \tag{33}$$

The magnitude of the parameter ϵ shown in (31) and (33) is set at 10^{-8} to avoid zero-valued denominator. The optimized coefficients shown in (31) and (33) are given by $c_1 = 0.20891413$, $c_2 = 0.49999999$ and $c_3 = 0.29108586$ which altogether can yield a fourth order accurate approximation with low dispersion error in the approximation of spatial derivatives. One can refer to Appendix D for the detailed derivation of the optimized coefficients.

3.1.2. Approximation of spatial derivatives in initialization equation

The semi-discrete form of the WENO scheme for the initialization equation can be expressed as [38]

$$\frac{d\phi}{dt}+H_G\left(\phi_x^+,\phi_x^-,\phi_y^+,\phi_y^-\right)=0, \tag{34}$$

where $\phi_{x,i}^-$ (left-biased stencil from $i-3$ to $i+2$) and $\phi_{x,i}^+$ (right-biased stencil from $i-2$ to $i+3$) shown in Eq. (34) are known as the WENO approximations of $\frac{\partial\phi}{\partial x}(x_i,y_j)$, and they are given by

$$\begin{aligned}
\phi_{x,i}^- = &\frac{1}{12}\left(-\frac{\Delta^+\phi_{i-2}}{\Delta x}+7\frac{\Delta^+\phi_{i-1}}{\Delta x}+7\frac{\Delta^+\phi_{i}}{\Delta x}-\frac{\Delta^+\phi_{i+1}}{\Delta x}\right)\\
&-\phi^{WENO}\left(\frac{\Delta^-\Delta^+\phi_{i-2}}{\Delta x},\frac{\Delta^-\Delta^+\phi_{i-1}}{\Delta x},\frac{\Delta^-\Delta^+\phi_{i}}{\Delta x},\frac{\Delta^-\Delta^+\phi_{i+1}}{\Delta x}\right),
\end{aligned} \tag{35}$$

and

$$\begin{aligned}
\phi_{x,i}^+ = &\frac{1}{12}\left(-\frac{\Delta^+\phi_{i-2}}{\Delta x}+7\frac{\Delta^+\phi_{i-1}}{\Delta x}+7\frac{\Delta^+\phi_{i}}{\Delta x}-\frac{\Delta^+\phi_{i+1}}{\Delta x}\right)\\
&-\phi^{WENO}\left(\frac{\Delta^-\Delta^+\phi_{i+2}}{\Delta x},\frac{\Delta^-\Delta^+\phi_{i+1}}{\Delta x},\frac{\Delta^-\Delta^+\phi_{i}}{\Delta x},\frac{\Delta^-\Delta^+\phi_{i-1}}{\Delta x}\right).
\end{aligned} \tag{36}$$

In Eqs. (35) and (36), the notations $\Delta^+\phi_k=\phi_{k+1}-\phi_k$, $\Delta^-\phi_k=\phi_k-\phi_{k-1}(k=i-3\sim i+2)$ are introduced. The nonlinear function ϕ^{WENO} can be expressed in terms of $\mathbf{a},\mathbf{b},\mathbf{c},\mathbf{d}$ as

$$\phi^{WENO}(\mathbf{a},\mathbf{b},\mathbf{c},\mathbf{d})=\frac{1}{3}\omega_0(\mathbf{a}-2\mathbf{b}+\mathbf{c})+\frac{1}{6}\left(\omega_2-\frac{1}{2}\right)(\mathbf{b}-2\mathbf{c}+\mathbf{d}). \tag{37}$$

In the above equation, the weighting factors ω_0 and ω_2 are as follows

$$\omega_0=\frac{\alpha_0}{\alpha_0+\alpha_1+\alpha_2},\omega_2=\frac{\alpha_2}{\alpha_0+\alpha_1+\alpha_2} \tag{38}$$

with

$$\alpha_0=\frac{1}{(\epsilon+IS_0)^2},\alpha_1=\frac{6}{(\epsilon+IS_1)^2},\alpha_2=\frac{3}{(\epsilon+IS_2)^2}, \tag{39}$$

and

$$\begin{aligned}
IS_0 &= 13(\mathbf{a}-\mathbf{b})^2+3(\mathbf{a}-3\mathbf{b})^2,\\
IS_1 &= 13(\mathbf{b}-\mathbf{c})^2+3(\mathbf{b}+\mathbf{c})^2,\\
IS_2 &= 13(\mathbf{c}-\mathbf{d})^2+3(3\mathbf{c}-\mathbf{d})^2.
\end{aligned} \tag{40}$$

It is noted that ϵ is chosen as 10^{-8} to avoid division by zero according to the suggestion of Jiang and Peng [38]. We use the following Godunov flux H_G in Eq. (34) [39]

$$H_G(\bar{a},\bar{b},\bar{c},\bar{d}) = \begin{cases} \bar{S}(\phi_0)\left(\sqrt{[\mathbf{max}(\bar{a}^-,\bar{b}^+)]^2+[\mathbf{max}(\bar{c}^-,\bar{d}^+)]^2}-1\right); & \text{if } \phi_0 \geq 0, \\ \bar{S}(\phi_0)\left(\sqrt{[\mathbf{max}(\bar{a}^+,\bar{b}^-)]^2+[\mathbf{max}(\bar{c}^+,\bar{d}^-)]^2}-1\right); & \text{otherwise}, \end{cases} \tag{41}$$

where the notations $\#^+ = \mathbf{max}(\#, 0)$ and $\#^- = \mathbf{min}(\#, 0)$ have been adopted.

3.2. Volume of fluid (VOF) solver

3.2.1. Weighed linear interface calculation (WLIC)

Following the idea of Simple Linear Interface Calculation (SLIC) method given in Ref. [15], Yokoi [31] defined an interface that combines the horizontal interface (interface along x-coordinate) and the vertical interface (interface along y-coordinate) under different weights. In two-dimensional space, the characteristic function $\chi_{i,j}(x, y)$ at each node (i, j) can be expressed as

$$\chi_{i,j}(x,y) = \omega_{x,i,j}(\mathbf{n}_{i,j})\chi_{x,i,j}(x,y) + \omega_{y,i,j}(\mathbf{n}_{i,j})\chi_{y,i,j}(x,y). \tag{42}$$

In the above equation, $\chi_{x,i,j}(x, y)$ and $\chi_{y,i,j}(x, y)$ are the characteristic functions of the vertical interface and horizontal interface. The weighting factors $\omega_{x,i,j}$ and $\omega_{y,i,j}$ can be obtained by calculating $n_{x,i,j}$ and $n_{y,i,j}$, which are the $x-$component and $y-$component of the surface normal $\mathbf{n_{i,j}}$, respectively.

$$\begin{aligned} \omega_{x,i,j} &= \frac{|n_{x,i,j}|}{|n_{x,i,j}|+|n_{y,i,j}|}, \\ \omega_{y,i,j} &= \frac{|n_{y,i,j}|}{|n_{x,i,j}|+|n_{y,i,j}|}. \end{aligned} \tag{43}$$

How to efficiently calculate the values of $n_{x,i,j}$ and $n_{y,i,j}$ is a key issue. In [31], the surface normal near the interface is calculated by using the volume fraction function C:

$$\begin{aligned} n_{x,i,j} &= \frac{1}{4}(n_{x,i+1/2,j+1/2} + n_{x,i-1/2,j+1/2} + n_{x,i+1/2,j-1/2} + n_{x,i-1/2,j-1/2}), \\ n_{y,i,j} &= \frac{1}{4}(n_{y,i+1/2,j+1/2} + n_{y,i-1/2,j+1/2} + n_{y,i+1/2,j-1/2} + n_{y,i-1/2,j-1/2}), \end{aligned} \tag{44}$$

where

$$\begin{aligned} n_{x,i+1/2,j+1/2} &= \frac{1}{2\Delta x}(C_{i+1,j} - C_{i,j} + C_{i+1,j+1} - C_{i,j+1}), \\ n_{y,i+1/2,j+1/2} &= \frac{1}{2\Delta y}(C_{i,j+1} - C_{i,j} + C_{i+1,j+1} - C_{i+1,j}). \end{aligned} \tag{45}$$

It has been pointed out that the above reconstruction of the interface is more accurate than the classical method in [31]. However, the VOF function whose spatial derivatives are not continuous near the interface may lead to an inaccurately predicted geometric quantities. To overcome this drawback in VOF method, the smoothed signed distance function will be employed to calculate surface normal, which can be expressed as follows

$$\mathbf{n}_{i,j} = \frac{(\nabla\phi)_{i,j}}{|\nabla\phi|_{i,j}}. \tag{46}$$

The term $\nabla\phi$ is the gradient of the smoothed distance function, which is approximated by the second-order centered difference scheme

$$(\nabla\phi)_{i,j} = \frac{\phi_{i+1,j} - \phi_{i-1,j}}{2\Delta x}\hat{\mathbf{e}}_x + \frac{\phi_{i,j+1} - \phi_{i,j-1}}{2\Delta y}\hat{\mathbf{e}}_y. \tag{47}$$

3.2.2. Tangent of hyperbola for interface capturing (THINC) scheme

A tangent of hyperbola for interface capturing (THINC) scheme has been proposed in [30]. In the one-dimensional THINC scheme, the piecewise hyperbolic tangent function given below is used to approximate the characteristic function

$$\chi_{x,i} = \frac{1}{2}\left[1 + \alpha\tanh\left(\beta\left(\frac{x - x_{i-1/2}}{\Delta x} - \tilde{x}_i\right)\right)\right], \tag{48}$$

where $\alpha = 1$ for $C_{i-1} < C_{i+1}$ and $\alpha = -1$ for $C_{i-1} > C_{i+1}$. Note that the parameter β $(= 3.5)$ is introduced to control the slope and the span of jump. The center $\tilde{x}_i$ in Eq. (48) is determined from C_i by calculating

$$C_i = \frac{1}{\Delta x_i}\int_{x_{i-1/2}}^{x_{i+1/2}} \chi_{x,i}(x)dx$$
$$= \frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_{i+1/2}} \frac{1}{2}\left[1+\alpha\tanh\left(\beta\left(\frac{x-x_{i-1/2}}{\Delta x}-\tilde{x}_i\right)\right)\right]dx. \tag{49}$$

An improved THINC scheme, called THINC of Slope Weighting (THINC/SW) with an extreme simple expression yet accommodating considerable accuracy, has been further discussed in [32].

3.2.3. THINC/WLIC scheme

The numerical flux $F_{x,i+1/2,j}$ shown in Eq. (12) with an appropriately prescribed characteristic function $\chi_{i,j}(x,y)$ shown in Eq. (42) can be calculated as

$$F_{x,i+1/2,j} = -\frac{1}{\Delta y}\int_{y_{i,j-1/2}}^{y_{i,j+1/2}}\int_{x_{i+1/2,j}}^{x_{i+1/2,j}-u_{i+1/2,j}\Delta t} \chi_{is,j}(x,y)dxdy,$$
$$= -\frac{1}{\Delta y}\int_{y_{i,j-1/2}}^{y_{i,j+1/2}}\int_{x_{i+1/2,j}}^{x_{i+1/2,j}-u_{i+1/2,j}\Delta t} \omega_{x,is,j}\chi_{x,is,j}dxdy$$
$$-\frac{1}{\Delta y}\int_{y_{i,j-1/2}}^{y_{i,j+1/2}}\int_{x_{i+1/2,j}}^{x_{i+1/2,j}-u_{i+1/2,j}\Delta t} \omega_{y,is,j}\chi_{y,is,j}dxdy,$$
$$\equiv F_{x,x,i+1/2,j}(\omega_{x,is,j},\chi_{x,is,j}) + F_{x,y,i+1/2,j}(\omega_{y,is,j},\chi_{y,is,j}). \tag{50}$$

Calculation of the numerical flux $F_{x,i+1/2,j}$ by THINC/WLIC scheme is given below for convenience:

1. Calculate the location of jump in the cell $\tilde{x}_{is}$ by

$$\tilde{x}_{is} = \frac{1}{2\beta}\ln\left(\frac{a_3^2-a_1a_3}{a_1a_3-1}\right), \tag{51}$$

where $a_1 = \mathbf{exp}(\frac{\beta}{\alpha}(2C_{is}-1))$ and $a_3 \equiv \mathbf{exp}(\beta)$.

2. Calculate the numerical flux $F_{x,x,i+1/2,j}$ by using the piecewise modified hyperbolic tangent function in Eq. (48)

$$F_{x,x,i+1/2,j} = -\frac{1}{\Delta y}\int_{y_{i,j-1/2}}^{y_{i,j+1/2}}\int_{x_{i+1/2,j}}^{x_{i+1/2,j}-u_{i+1/2,j}\Delta t} \omega_{x,is,j}\chi_{x,is,j}dxdy,$$
$$= -\frac{1}{\Delta y}\int_{y_{i,j-1/2}}^{y_{i,j+1/2}}dy\int_{x_{i+1/2}}^{x_{i+1/2}-u_{i+1/2}\Delta t} \frac{\omega_{x,is}}{2}\left[1+\alpha\tanh\left(\beta\left(\frac{x-x_{is-1/2}}{\Delta x}-\tilde{x}_i\right)\right)\right]dx,$$
$$= -\frac{\omega_{x,is}}{2}\left[x+\frac{\alpha\Delta x}{\beta}\ln\left(\cosh\left(\beta\left(\frac{x-x_{is-1/2}}{\Delta x}\right)-\tilde{x}_{is}\right)\right)\right]_{x_{i+1/2,j}}^{x_{i+1/2,j}-u_{i+1/2,j}\Delta t},$$
$$= \frac{\omega_{x,is,j}}{2}\left[u_{i+1/2,j}\Delta t - \frac{\alpha\Delta x}{\beta}\ln\left(\frac{a_4}{a_5}\right)\right], \tag{52}$$

where $a_4 = \mathbf{cosh}[\beta(\gamma - u_{i+1/2,j}\Delta t/\Delta x - \tilde{x}_{is})]$, $a_5 = \mathbf{cosh}[\beta(\gamma-\tilde{x}_{is})]$. The value of γ is 1 if $u_{i+1/2,j} > 0$, and $\gamma = 0$ otherwise.

3. Calculate the numerical flux $F_{x,y,i+1/2}$ using the following equation

$$F_{x,y,i+1/2,j} = \omega_{y,is,j}C_{is,j}u_{x,i+1/2,j}\Delta t. \tag{53}$$

4. Generalize the numerical flux $F_{x,i+1/2,j}$ by adding $F_{x,x,i+1/2,j}$ and $F_{x,y,i+1/2,j}$

$$F_{x,i+1/2,j} = F_{x,x,i+1/2,j} + F_{x,y,i+1/2,j}. \tag{54}$$

One can refer to [31] for the numerical flux term $F_{y,i,j+1/2}$ using the THINC/WLIC scheme.

3.3. Navier-Stokes equation solver

3.3.1. Momentum equation solver

Advection terms are discretized using the second-order upwind scheme in this paper. Take the term $u\frac{\partial u}{\partial x}$ in two-dimensional space as an example, we have

$$u\frac{\partial u}{\partial x}=\frac{1}{4\Delta x}\Big(u^{+}(3u_{i,j}-4u_{i-1,j}+u_{i-2,j})+u^{-}(-u_{i+2,j}+4u_{i+1,j}-3u_{i,j})\Big), \tag{55}$$

where $u^{+}=u_{i,j}+|u_{i,j}|$ and $u^{-}=u_{i,j}-|u_{i,j}|$. Other advection terms can be approximated in the same way. The second-order center difference scheme is used for the approximation of the diffusion terms.

3.3.2. Projection method

Projection method can be implemented by splitting Eq. (25) into two steps. In the first step, $\mathbf{A}^n$ given below is defined

$$\mathbf{A}^n\equiv\big(\mathbf{u}^n\cdot\nabla\big)\mathbf{u}^n-\frac{1}{Re}\frac{\nabla\cdot(2\mu^n\mathbf{D}^n)}{\rho^n}-\frac{1}{Fr^2}\hat{\mathbf{e}}_g+\frac{1}{We}\frac{\kappa(\phi^n)\delta(\phi^n)\nabla\phi^n}{\rho^n}. \tag{56}$$

Intermediate velocity $\mathbf{u}^*$ can be solved directly by using the second-order Adams-Bashforth scheme,

$$\frac{\mathbf{u}^*-\mathbf{u}}{\Delta t}+\big(\frac{3}{2}\mathbf{A}^n-\frac{1}{2}\mathbf{A}^{n-1}\big)=0. \tag{57}$$

The second step is for the calculation of $\mathbf{u}^{n+1}$ using the following equation,

$$\frac{\mathbf{u}^{n+1}-\mathbf{u}^*}{\Delta t}=-\frac{1}{\rho^{n+1}}\nabla p^{n+1}. \tag{58}$$

To obtain p^{n+1}, the divergence operator is performed on both sides of Eq. (58) and applying $\nabla\cdot\mathbf{u}^{n+1}=0$, thereby leading to the following Poisson equation:

$$\nabla\cdot\big(\frac{1}{\rho^{n+1}}\nabla p^{n+1}\big)=\frac{\nabla\cdot\mathbf{u}^*}{\Delta t}. \tag{59}$$

If p^{n+1} can be obtained through Eq. (59), the computed velocity field satisfies the continuity equation:

$$\nabla\cdot\mathbf{u}^{n+1}=0. \tag{60}$$

It is worthy to address here that after the calculation of the pressure value, the continuity constraint will be satisfied automatically. Approximation of Eq. (59) leads to the following difference equation:

$$aP_{i-1,j}+bP_{i+1,j}+cP_{i,j}+dP_{i,j-1}+eP_{i,j+1}=(\frac{\nabla\cdot\mathbf{u}^*}{\Delta t})_{i,j}, \tag{61}$$

where $a=\frac{1}{\rho^{n+1}_{i-1/2,j}\Delta x^2}$, $b=\frac{1}{\rho^{n+1}_{i+1/2,j}\Delta x^2}$, $d=\frac{1}{\rho^{n+1}_{i,j-1/2}\Delta y^2}$, $e=\frac{1}{\rho^{n+1}_{i,j+1/2}\Delta y^2}$, and $c=-(a+b+d+e)$. We then apply the point-successive over-relaxation method given below to iterate Eq. (61):

$$P^{m+1}_{i,j}=\omega P^{m+1}_{i,j}+(1-\omega)P^{m}_{i,j}, \tag{62}$$

where the relaxation factor ω is set to 1.5 in this paper. In Eq. (62), m is the number of iterations. Point-wise absolute convergence tolerance is chosen as:

$$|P^{m+1}_{i,j}-P^{m}_{i,j}|\le 1\times10^{-5}. \tag{63}$$

Substitute the computed pressure value into Eq. (58), the velocity solution for $\mathbf{u}^{n+1}$ can then be obtained. The advantage of this method is that the continuity equation is satisfied automatically, and the pressure value can be calculated iteratively. The numerical accuracy order is $O(\Delta t^2,\Delta x^2)$. In summary, a detailed flow chart of the proposed LS-assisted VOF advection method together with the momentum solver is given in Fig. 1.

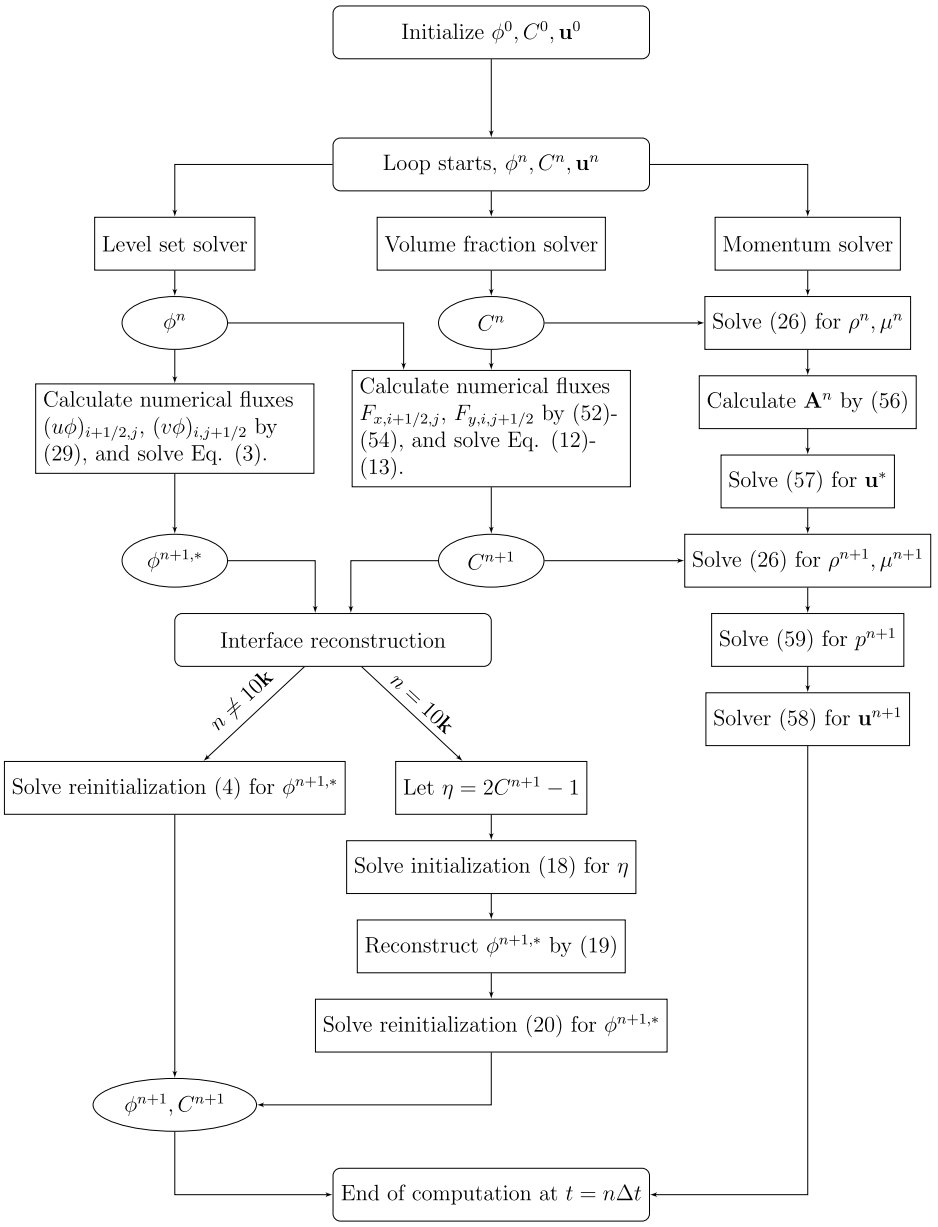


Fig. 1. Flow chart of the proposed solution algorithm, $\mathbf{k} \in \mathbb{N}$.

4. Verification and validation studies

To justify the ability of applying the proposed to capture interface and to conserve volume, three different error norms $\varepsilon_V, \bar{\varepsilon}_V$ and ε_I are introduced as follows:

$$\begin{aligned} \varepsilon_M &= \frac{|V_0 - V_T|}{V_0}, \\ \bar{\varepsilon}_M &= \frac{1}{T}\int_0^T \frac{|V(t) - V_0|}{V_0}\, dt, \\ \varepsilon_I &= \frac{\int_\Omega |H^T_{i,j} - H^T(x_i, y_j)|\, d\Omega}{\int_\Omega d\Omega}. \end{aligned} \tag{64}$$

The computational domain Ω is separated by the interface Γ, with Ω_1 and Ω_2 inside and outside of the interface, respectively. V_0 denotes the total volume of Ω_1 in the beginning and V_T denotes the total volume at the end of the computation, respectively. $V(t)$ is the volume at a time t with $T > t > 0$. $H^T_{i,j}$ is the Heaviside function of the solution at (x_i, y_j) at $t = T$, and $H^T(x_i, y_j)$ is the Heaviside function of the exact solution. It is noted that the total volume obtained by different methods can be expressed below

$$\begin{aligned} &\textbf{total volume by pure level set method} = \int_{\Omega} H(\phi)\, d\Omega, \\ &\textbf{total volume by the proposed method} = \int_{\Omega} C\, d\Omega. \end{aligned} \tag{65}$$

Calculation of the three error norms defined in Eq. (64) for pure level set method and our proposed method is associated with the definition of volume in Eq. (65). ε_M and ε_I can directly reveal the ability of the employed numerical method to capture the interface. ε_M accounts for the loss of volume and ε_I exhibits the accuracy of the predicted interface position. $\bar{\varepsilon}_M$ denotes the averaged loss of volume for the entire computation. For a problem undergoing a sharp topology change, $\bar{\varepsilon}_M$ can provide more information than the information about the amount of decreased volume, like ε_M. These three errors will be calculated and discussed in the four chosen benchmark problems, including two-dimensional and three-dimensional problems.

To make a comparison on the computational efficiency of the two methods, the factor, defined as $\Theta^+ \equiv \frac{\Theta_{\text{present}} - \Theta_{\text{ls}}}{\Theta_{\text{ls}}}$, has been introduced, where Θ_{ls} is the CPU time consumed using the pure level set method and Θ_{present} is the CPU time consumed by the proposed method. The time step Δt and the mesh size h are set as $\Delta t = 0.1h$ in each problem described in Section 4.1.

4.1. Verification studies

4.1.1. Two-dimensional vortex deforming problem

The problem with a circle evolving with a prescribed velocity field can be traced back to the study conducted by Rider et al. [18] and Rudman [19]. In this problem, a circle with the radius $r = 0.15$ is initially located at $(x, y) = (0.5, 0.75)$ in a square box $\Omega : [0, 1] \times [0, 1]$. The prescribed velocity is given below:

$$u(x, y, t) = \sin^2(\pi x) \sin(2\pi y) \cos(\pi t/T), \tag{66}$$

$$v(x, y, t) = -\sin(2\pi x) \sin^2(\pi y) \cos(\pi t/T). \tag{67}$$

The circle starts deforming its shape, and will return back to its initial shape at $t = T$. During the shape deformation, a very thin tail will be formed. The longer the period is, the thinner the filament will be. This problem has been considered as a standard assessment test case for different numerical methods.

This problem has been simulated at three different time periods, $T = 4, 8, 16$, and the corresponding errors $\varepsilon_M, \bar{\varepsilon}_M$ and ε_I obtained at different grid numbers are given in Table 1–3. According to these tables, the proposed method demonstrates a very good ability of conserving the volume. The values of ε_M and $\bar{\varepsilon}_M$ are both smaller than those obtained from the pure level set method. The averaged volume loss using the proposed method at $T = 16$ in the grid number 32^2 is 1.2489×10^{-7} while 6.9462×10^{-1} using the pure level set method. This indicates that volume can be preserved extremely well even in such a coarse mesh using our proposed method. The solution is also compared with the exact solution by calculating ε_I at $t = T$. The solution obtained by the proposed method has a very good agreement with the exact solution. The snapshots of the predicted solutions at different periods of time with two different grid numbers are given in Figs. 2–4. From these figures good agreement with the exact solution has been confirmed for the solution obtained at $t = T$ using the proposed method. It is noticed that the error norms in fine grids seem too small to be properly calculated in double-precision framework. According to the tabulated results in Table 4, it is shown that the proposed method consumed about 35-40 percentages of CPU time more than the pure level set method.

4.1.2. Rotating (Zalesak's) disk problem

The Zalesak's disk problem has been investigated to verify the scheme ability of capturing interface. A slotted circle of radius 0.15 with the slot length of 0.25 and the width of 0.05 is initially located at $(x, y) = (0.5, 0.75)$ in a square $\Omega : [0, 1] \times [0, 1]$. The circle is rotated about the point $(0.5, 0.5)$ at a constant velocity given by:

$$\begin{aligned} u(x, y) &= \frac{2\pi}{T}(0.5 - y), \\ v(x, y) &= \frac{2\pi}{T}(x - 0.5). \end{aligned} \tag{68}$$

Table 1
Comparison of the computed error norms defined in Eq. (64) for the two-dimensional vortex deforming problem at different grid numbers and at period $T = 4$.

Error norms	Grid number				
	32^2	64^2	128^2	256^2	512^2
Proposed method					
ε_M	7.7969×10^{-9}	1.9638×10^{-16}	5.8906×10^{-16}	2.7487×10^{-15}	3.0431×10^{-14}
$\bar{\varepsilon}_M$	2.9735×10^{-9}	2.5409×10^{-16}	8.7345×10^{-16}	4.0627×10^{-15}	1.9727×10^{-14}
ε_I	6.4324×10^{-3}	1.5848×10^{-3}	7.4680×10^{-4}	2.9672×10^{-4}	1.7412×10^{-4}
Pure level set method					
ε_M	2.5157×10^{-1}	6.7310×10^{-2}	1.1492×10^{-2}	2.9732×10^{-3}	8.0280×10^{-4}
$\bar{\varepsilon}_M$	1.6856×10^{-1}	4.2187×10^{-2}	6.7685×10^{-3}	1.6316×10^{-3}	3.8082×10^{-4}
ε_I	2.5201×10^{-2}	6.7419×10^{-3}	1.4351×10^{-3}	4.4390×10^{-4}	2.6008×10^{-4}

Table 2
Comparison of the computed error norms defined in Eq. (64) for the two-dimensional vortex deforming problem at different grid numbers and at period $T = 8$.

Error norms	Grid number				
	32^2	64^2	128^2	256^2	512^2
Proposed method					
ε_M	2.2130×10^{-8}	1.2961×10^{-14}	1.1781×10^{-15}	2.1597×10^{-15}	2.4149×10^{-14}
$\bar{\varepsilon}_M$	1.5484×10^{-8}	2.5541×10^{-15}	5.4199×10^{-16}	3.7515×10^{-15}	3.0528×10^{-14}
ε_I	3.9169×10^{-2}	1.1063×10^{-2}	1.9889×10^{-3}	5.2282×10^{-4}	2.2537×10^{-4}
Pure level set method					
ε_M	6.5916×10^{-1}	7.2638×10^{-1}	1.8404×10^{-1}	3.5970×10^{-2}	8.9098×10^{-3}
$\bar{\varepsilon}_M$	4.5934×10^{-1}	4.7033×10^{-1}	1.1506×10^{-1}	2.1393×10^{-2}	4.7859×10^{-3}
ε_I	9.8008×10^{-2}	5.2966×10^{-2}	1.5121×10^{-2}	3.4015×10^{-3}	1.0145×10^{-3}

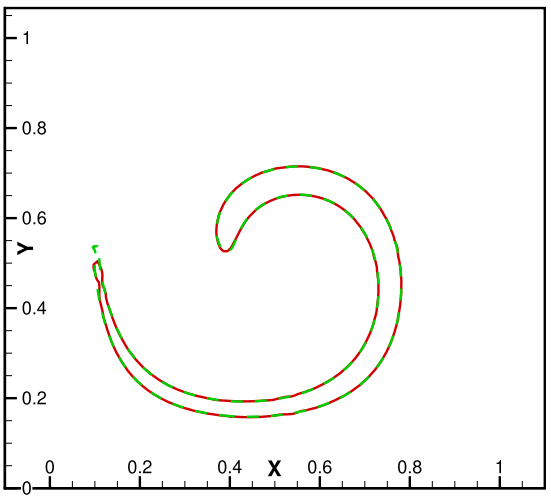
Table 3
Comparison of the computed error norms defined in Eq. (64) for the two-dimensional vortex deforming problem at different grid numbers and at period $T = 16$.

Error norms	Grid number				
	32^2	64^2	128^2	256^2	512^2
Proposed method					
ε_M	3.7134×10^{-7}	1.5318×10^{-13}	1.9635×10^{-16}	2.5524×10^{-15}	2.2578×10^{-14}
$\bar{\varepsilon}_M$	1.2489×10^{-7}	6.1319×10^{-14}	3.8988×10^{-16}	2.0751×10^{-15}	1.6975×10^{-14}
ε_I	3.0783×10^{-2}	4.9155×10^{-2}	2.0170×10^{-2}	3.6789×10^{-3}	7.3369×10^{-4}
Pure level set method					
ε_M	1.1310×10^{0}	2.2455×10^{0}	1.3396×10^{0}	2.6457×10^{-1}	4.1024×10^{-2}
$\bar{\varepsilon}_M$	6.9462×10^{-1}	1.5700×10^{0}	9.3730×10^{-1}	2.0167×10^{-1}	3.5942×10^{-2}
ε_I	1.6421×10^{-1}	1.6741×10^{-1}	9.4898×10^{-2}	2.6491×10^{-2}	6.8054×10^{-3}

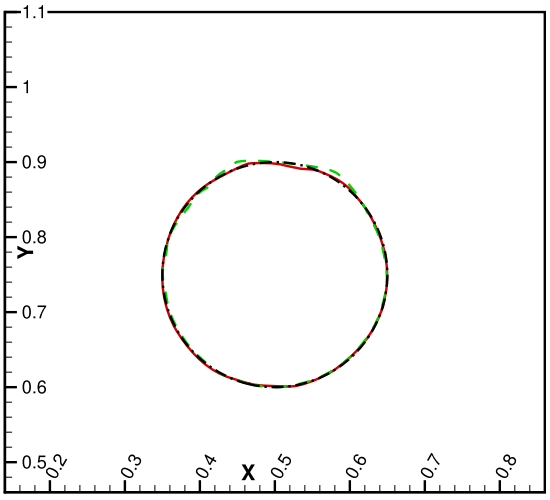
Table 4
Comparison of the CPU time (seconds) for the two-dimensional vortex deforming problem at different grid numbers and at period $T = 16$.

	Grid number				
	32^2	64^2	128^2	256^2	512^2
Proposed method	26.40	57.07	227.11	1061.58	8547.37
Pure level set method	20.00	38.16	163.85	790.82	6225.49
Θ^+	32.00%	49.55%	38.61%	34.23%	37.29%

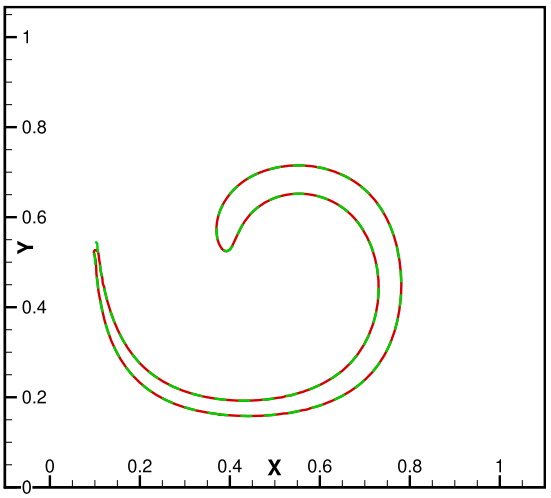
In the above equation, T is the period for a circle to proceed one rotation, which is set to 2π in our study. Error estimations of the proposed method and the pure level set method have been performed as shown in Table 5. Solutions computed under different grid numbers are compared with the exact solution after one and two revolutions, or at $t = T$ and $t = 2T$. As expected, the proposed method conserves the volume extremely well in comparison with the degree of volume conservation using the pure level set method. According to Table 5, interface predicted by the proposed method is better than the pure level set method carried out in coarser grids ($32^2, 64^2$ and 128^2). In finer grids with the resolutions of 256×256 and 512×512, for the predicted values of ε_I we found that the pure level set method performs slightly



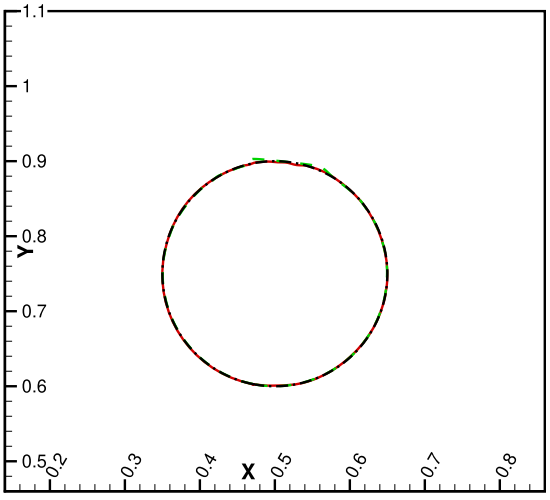
(a) At $t = 2.0$, coarse grid 128^2



(b) At $t = 4.0$, coarse grid 128^2



(c) At $t = 2.0$, fine grid 256^2



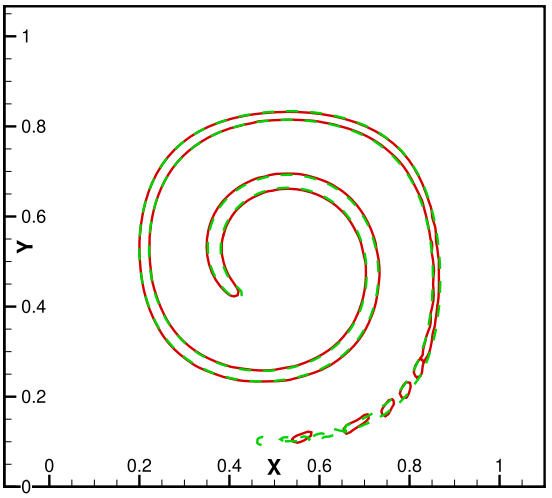
(d) At $t = 4.0$, fine grid 256^2

Fig. 2. Comparison of the predicted interfaces for the two-dimensional vortex deforming problem with the period of $T = 4$ at different grid numbers. (Red solid denotes the solution obtained by the proposed method, green dash denotes the solution obtained by the pure level set method, and black dash-dot denotes the exact solution at $t = T$.) (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

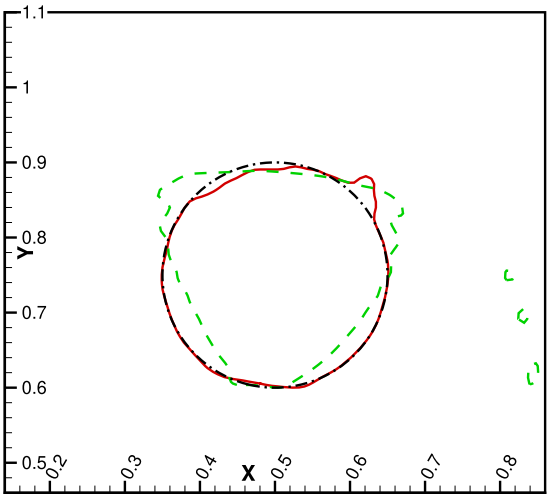
better than the proposed method. In general, we can conclude that the ability of retaining volume conservation using our proposed method is much better than that using the pure level set method. However, an additional CPU time is need in our proposed method as it shown in Table 6. Our proposed method costs about 40 percentages of the additional CPU time than the classical level set method. Solution snapshots are given in Fig. 5.

4.1.3. Three-dimensional vortex deforming problem

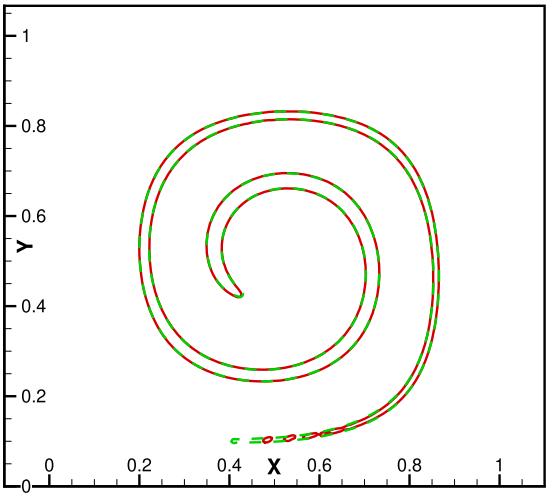
In order to show the scheme ability of retaining volume conservation in three dimensional simulation, the single vortex deforming problem will be simulated using the proposed method and the pure level set method. This problem was first introduced by LeVeque [45]. A sphere of radius $r = 0.15$ is located at $(x, y, z) = (0.35, 0.35, 0.35)$ in a cubic domain Ω : $[0, 1] \times [0, 1] \times [0, 1]$. Velocity components considered in this problem at any time $0 \leq t \leq T$ are given by



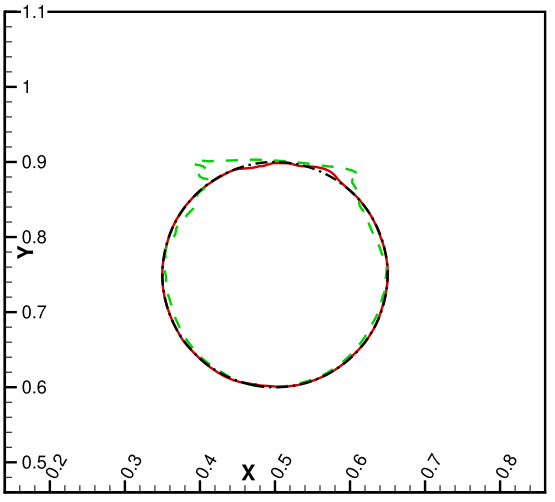
(a) At $t = 4.0$, coarse grid 128^2



(b) At $t = 8.0$, coarse grid 128^2



(c) At $t = 4.0$, fine grid 256^2

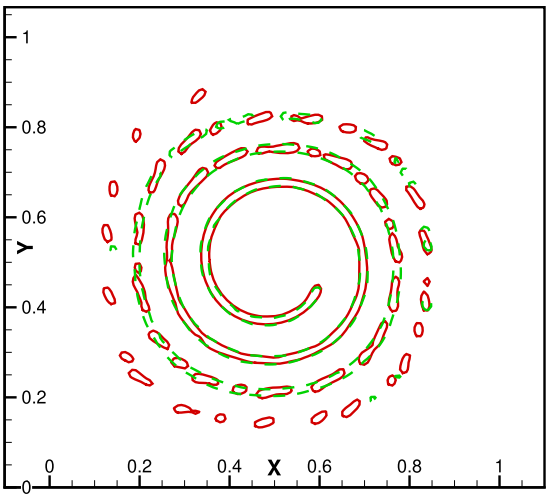


(d) At $t = 8.0$, fine grid 256^2

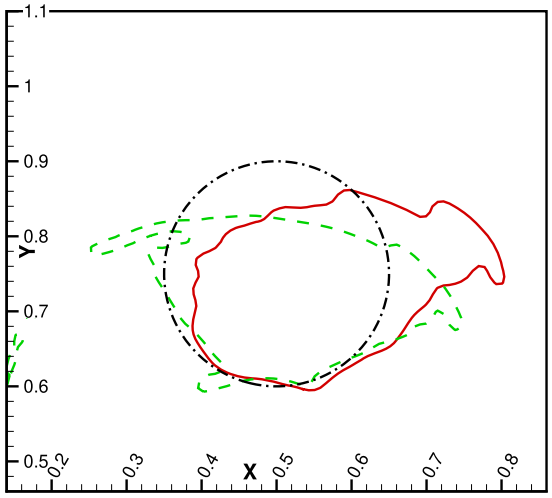
Fig. 3. Comparison of the predicted interfaces for the two-dimensional vortex deforming problem with the period of $T = 8$ at different grid numbers. (Red solid denotes the solution obtained by the proposed method, green dash denotes the solution obtained by the pure level set method, and black dash-dot denotes the exact solution at $t = T$.)

$$
\begin{aligned}
u(x, y, z, t) &= 2\sin^2(\pi x)\sin(2\pi y)\sin(2\pi z)\cos(\pi t/T),\\
v(x, y, z, t) &= -\sin(2\pi x)\sin^2(\pi y)\sin(2\pi z)\cos(\pi t/T),\\
w(x, y, z, t) &= -\sin(2\pi x)\sin(2\pi y)\sin^2(\pi z)\cos(\pi t/T).
\end{aligned}
\tag{69}
$$

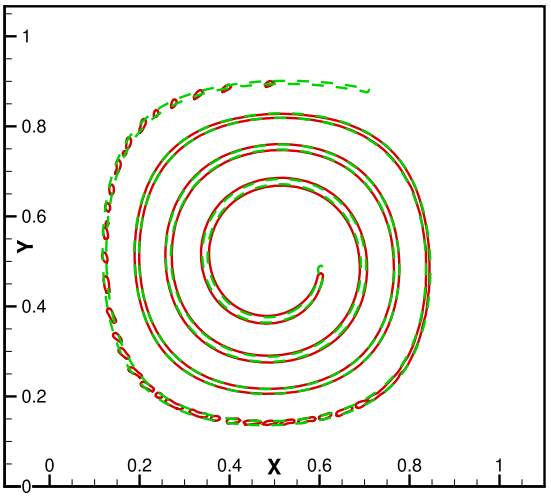
Note that T is the period of the rotating shear vortex, and the sphere should reverse back to its initial shape at $t = T$, which is set at 3.0 in this study. Eq. (69) allows the sphere to start to deform, and, then, to evolve to form two vortices that scoop out the opposite side of the sphere. A thin film can be seen during the evolution as shown in Figs. 6, 7. The error norms are also calculated in various grids as shown in Table 7. According to these tabulated results, the conservation of volume property is well retained using our proposed method in three grid numbers $32^3, 64^3, 128^3$, without having to sacrifice computational accuracy, in the prediction of interface location. Moreover, only 30 percentage of the additional CPU time is needed for our proposed method, as shown in Table 8.



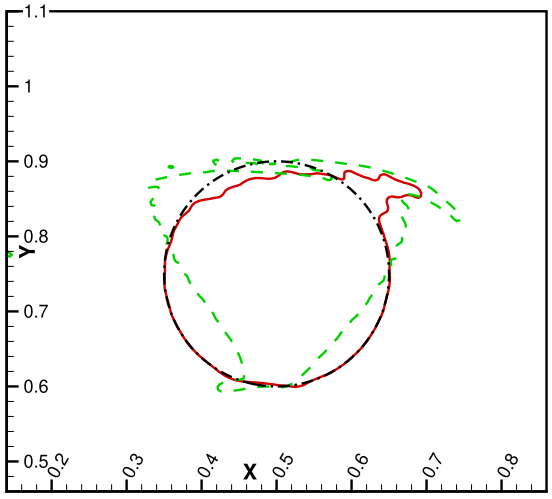
(a) At $t = 8.0$, coarse grid 128^2



(b) At $t = 16.0$, coarse grid 128^2



(c) At $t = 8.0$, fine grid 256^2



(d) At $t = 16.0$, fine grid 256^2

Fig. 4. Comparison of the predicted interfaces for the two-dimensional vortex deforming problem with the period of $T = 16$ at different grid numbers. (Red solid denotes the solution obtained by the proposed method, green dash denotes the solution obtained by the pure level set method, and black dash-dot denotes the exact solution at $t = T$.)

4.1.4. Rotating (Zalesak's) sphere problem

Analogous to the two-dimensional Zalesak's disk problem, a slotted sphere is also chosen in this study to examine the degree of volume conservation and the computational accuracy in capturing the interface using the proposed method in three spatial dimensions. A slotted sphere of radius 0.15 with slot length 0.25 and width 0.05 is initially located at $(x, y, z) = (0.5, 0.75, 0.5)$ in a cubic domain $\Omega : [0, 1] \times [0, 1] \times [0, 1]$. This sphere undergoes a rigid body rotation about the point $(0.5, 0.5, 0.5)$ in the constant velocity field given by:

$$\begin{aligned} u(x, y, z) &= \frac{2\pi}{T}(0.5 - y), \\ v(x, y, z) &= \frac{2\pi}{T}(x - 0.5), \\ w(x, y, z) &= 0. \end{aligned} \tag{70}$$

Table 5
Comparison of the computed error norms for the rotating disk problem after one and two revolutions at different grid numbers.

Error norms	Grid number				
	32^2	64^2	128^2	256^2	512^2
Proposed method					
One revolution					
ε_M	1.9966×10^{-6}	6.9467×10^{-13}	8.0003×10^{-14}	1.5439×10^{-14}	3.1289×10^{-13}
$\bar{\varepsilon}_M$	9.3930×10^{-7}	3.5887×10^{-13}	8.6364×10^{-14}	1.7407×10^{-14}	2.3138×10^{-13}
ε_I	8.5244×10^{-3}	2.3729×10^{-3}	1.0900×10^{-3}	4.7988×10^{-4}	2.6941×10^{-4}
Two revolutions					
ε_M	6.5687×10^{-6}	1.1161×10^{-12}	2.6175×10^{-13}	4.9175×10^{-14}	3.5415×10^{-13}
$\bar{\varepsilon}_M$	3.2921×10^{-6}	8.0847×10^{-13}	2.1386×10^{-13}	6.3525×10^{-14}	3.8095×10^{-13}
ε_I	1.0073×10^{-2}	3.3466×10^{-3}	1.4420×10^{-3}	5.8798×10^{-4}	3.1982×10^{-4}
Pure level set method					
One revolution					
ε_M	2.8083×10^{-2}	1.2157×10^{-2}	1.3399×10^{-2}	1.1621×10^{-3}	6.9726×10^{-5}
$\bar{\varepsilon}_M$	2.5896×10^{-2}	1.1838×10^{-2}	8.4206×10^{-3}	5.9694×10^{-4}	4.3832×10^{-5}
ε_I	5.2193×10^{-3}	6.7121×10^{-3}	1.0917×10^{-3}	1.9377×10^{-4}	5.6305×10^{-5}
Two revolutions					
ε_M	1.0767×10^{-2}	6.6267×10^{-3}	1.4404×10^{-2}	3.0966×10^{-3}	1.1355×10^{-4}
$\bar{\varepsilon}_M$	3.6370×10^{-3}	1.4920×10^{-2}	1.5563×10^{-2}	1.6482×10^{-3}	9.6142×10^{-5}
ε_I	6.2970×10^{-3}	1.2703×10^{-2}	1.8785×10^{-3}	3.1685×10^{-4}	6.2972×10^{-5}

Table 6
Comparison of the CPU time (seconds) for the rotating disk problem at different grid numbers after two revolutions.

	Grid number				
	32^2	64^2	128^2	256^2	512^2
Proposed method	16.00	40.01	107.53	479.04	3255.67
Pure level set method	10.42	24.75	81.58	345.92	2376.40
Θ^+	53.55%	61.65%	31.81%	38.48%	37.29%

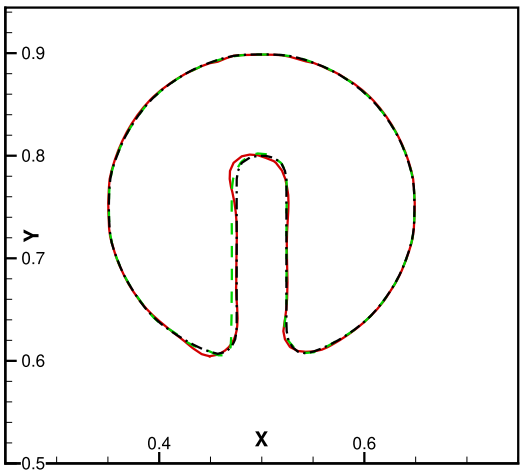
Table 7
Comparison of the computed error norms defined in Eq. (64) for the three-dimensional vortex deforming problem at different grid numbers.

Error norms	Grid number		
	32^3	64^3	128^3
Proposed method			
ε_M	2.9133×10^{-12}	2.3683×10^{-14}	3.7475×10^{-13}
$\bar{\varepsilon}_M$	1.1944×10^{-12}	1.2834×10^{-14}	1.4903×10^{-13}
ε_I	3.8052×10^{-3}	1.7552×10^{-3}	4.1459×10^{-4}
Pure level set method			
ε_M	7.1916×10^{-2}	1.9904×10^{-1}	1.4743×10^{-1}
$\bar{\varepsilon}_M$	1.3558×10^{-1}	1.1656×10^{-1}	5.2291×10^{-2}
ε_I	9.4198×10^{-3}	6.9538×10^{-3}	3.0152×10^{-3}

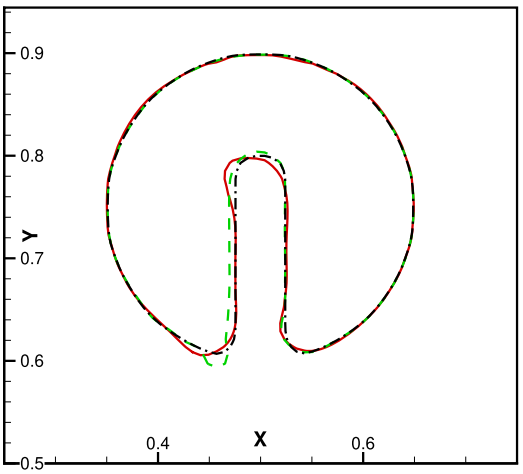
Table 8
Comparison of the CPU time (seconds) for the three-dimensional vortex deforming problem at different grid numbers.

	Grid number		
	32^3	64^3	128^3
Proposed method	19.21	213.69	3495.02
Pure level set method	15.07	169.78	2666.94
Θ^+	27.47%	25.86%	31.05%

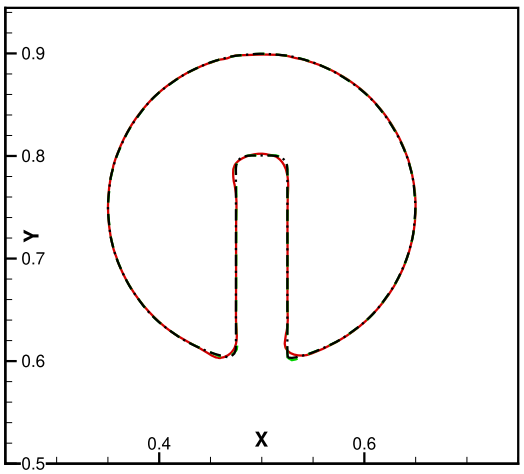
The period T of one revolution is set as 2π in this study. The initial condition and the snapshots of the numerical solutions after one and two revolutions are shown in Figs. 8, 9, and the errors are tabulated in Table 9. According to Figs. 8, 9 and Table 9, the solution obtained by the proposed method has a good match with the exact solution. As expected, the degree of conserving volume is extremely high using our proposed method. Only a 1.1566×10^{-10}% of volume is lost after



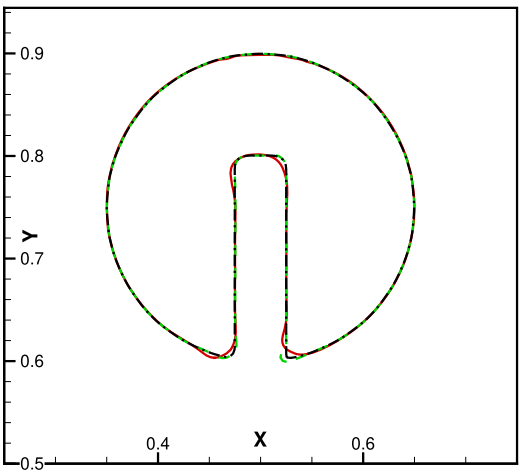
(a) After one revolution, coarse grid 128^2.



(b) After two revolutions, coarse grid 128^2



(c) After one revolution, fine grid 256^2



(d) After two revolutions, fine grid 256^2

Fig. 5. Comparison of the predicted interfaces for the rotating disk problem at different grid numbers. (Red solid denotes the solution obtained by the proposed method, green dash denotes the solution obtained by the pure level set method, and black dash-dot denotes the exact solution.)

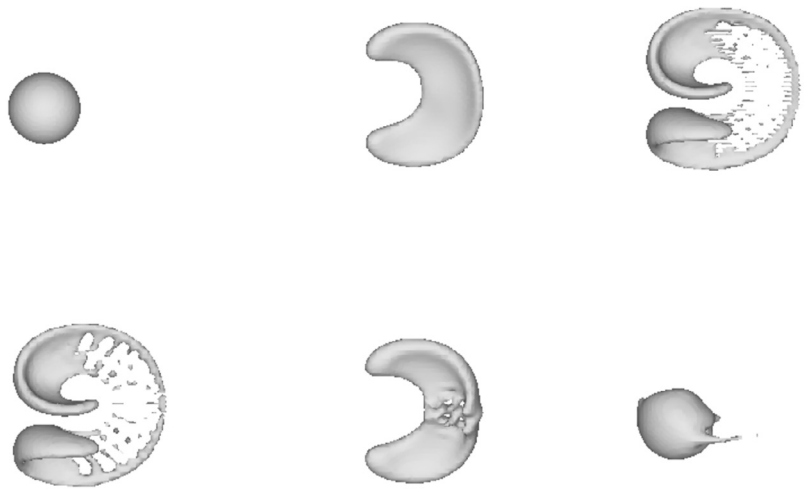
one rotation in the coarse grid 64^3, while a 3.8093% of the volume loss has been seen for the pure level set method. According to Table 10, solutions obtained by using our proposed method consumed about 30 percentages more CPU time than the classical level set method.

4.2. Validation study

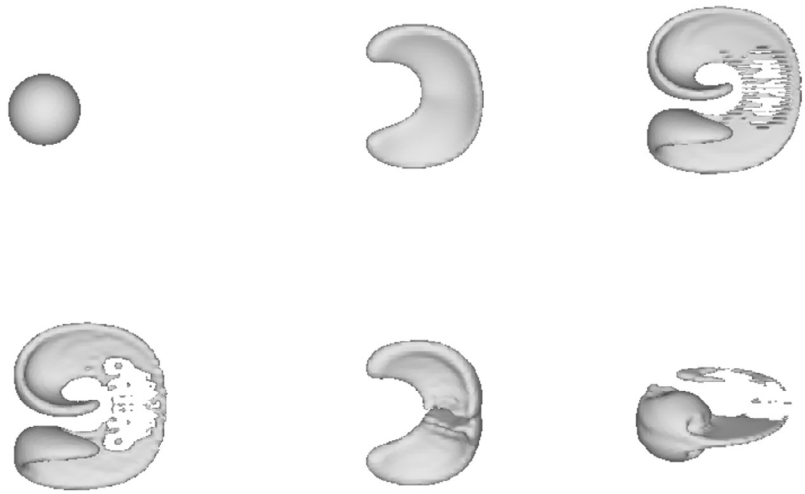
Flow instability of the Rayleigh-Taylor type involves a complicated topology change in the solution. This instability problem is associated with the penetration of a heavy fluid into a light fluid in the direction of gravity, which consists damped oscillatory motions of interface [46]. The Atwood number is used to characterize the density difference of two fluids, which is defined as:

$$At = \frac{\rho_A - \rho_B}{\rho_A + \rho_B} \tag{71}$$

ρ_A and ρ_B are the densities of the heavy and light fluids, respectively. Reynolds number under investigation is 3000 and $At = 0.5$ in this study. Initially, the interface is given by $y(x) = 0.1D\cos(2\pi x/D)$ in a rectangular domain $\Omega : [0, D] \times$



(a) Proposed method

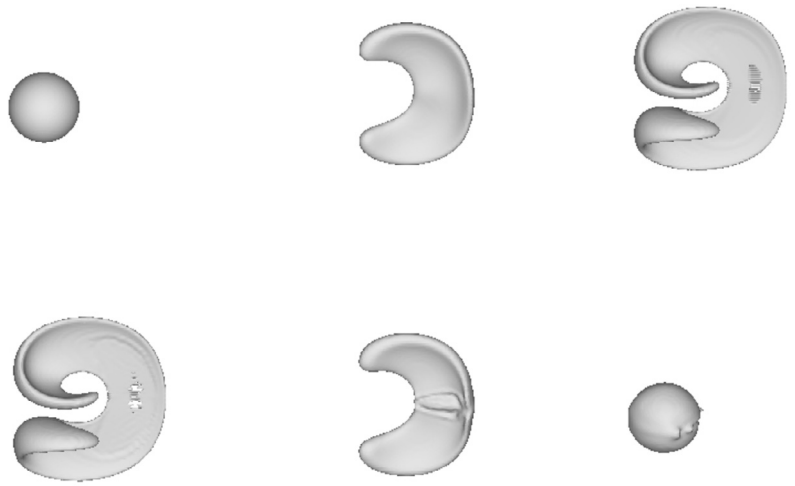


(b) Pure level set method

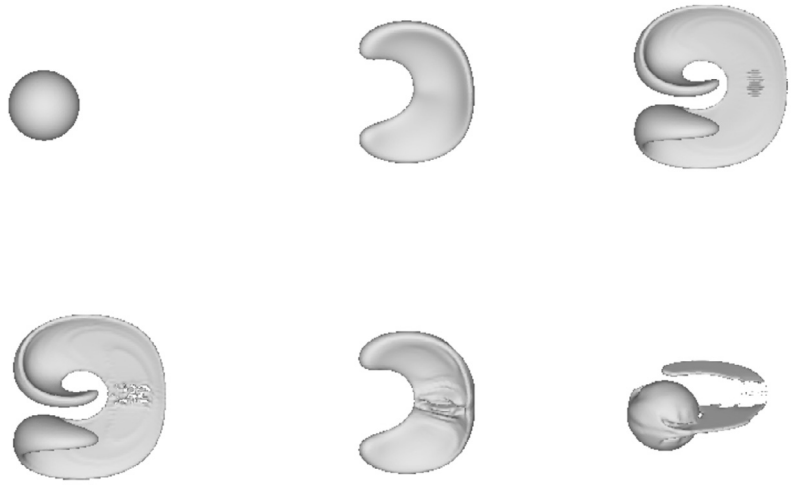
Fig. 6. Snapshot of the predicted interfaces for the three-dimensional vortex deforming problem predicted in fine grid 64^3. From top to down, from left to right, $t = 0, 0.6, 1.2, 1.8, 2.4, 3.0$.

$[-2D, 2D]$, where D is set to 1.0 in this study. Surface tension is ignored in this problem. No-slip condition is applied at the top and the bottom, while slip condition is applied at two sides of the cavity.

The predicted interface profiles in the coarse mesh 50×200 is plotted in Fig. 10 and the fine mesh 200×800 in Fig. 11. The contours of $H(\phi)$ obtained by the pure level set method and C by the proposed method have been shown. The corresponding time step is $\Delta t = 0.01\Delta x$. In meshes with different number of grid points, the percentages of area loss are plotted in Fig. 13. In this figure, the proposed method demonstrates a very good ability to preserve area/volume in comparison with the ability of pure level set method. As a result, the predicted interface profile remains to have a good symmetry during the evolution as shown in Figs. 10 and 11. In Fig. 12, a good match of our results with those of the previous studies [43,44] can be seen at the positions of falling and rising fluid. In Figs. 10-12, our chosen time scale $t_{\text{Tryg}} = t\sqrt{At}$ is related to the time scale chosen in [47].



(a) Proposed method



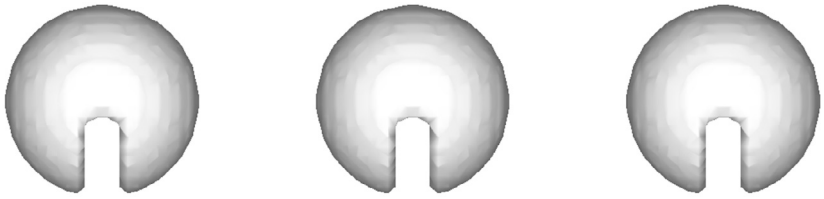
(b) Pure level set method

Fig. 7. Snapshot of the predicted interfaces for the three-dimensional vortex deforming problem predicted in fine grid 128^3. From top to down, from left to right, $t = 0, 0.6, 1.2, 1.8, 2.4, 3.0$.

5. Numerical results

5.1. 3D gas bubble bursting at a free surface

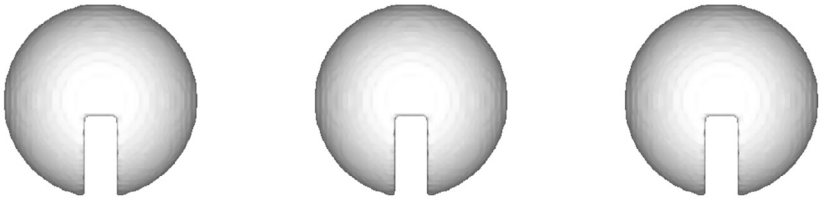
Bubble bursting at a water surface occurs often in our daily life. This phenomenon results in a transport or an exchange of substances across the interface [52]. One of the important applications is the exchange of various substances between the ocean and the atmosphere, including the transports of heat, mass, CO_2 and other contaminants. The phenomena of bubble bursting occur at the moment that the bubble reaches the interface and the film separating the bubble from the atmosphere starts collapsing. The resulting rapture of the film can generate hundreds of tiny droplets, whose diameters can be the scale of micrometer-sized. Such a formation of droplets may cause a large fraction of media transfer to occur [53]. The second major phenomenon subsequent to the rupture of the film results from surface tension and the formation of buoyancy force along the film. By the combined effects of both surface-tension and buoyancy, a narrow vertical jet can be formed and it



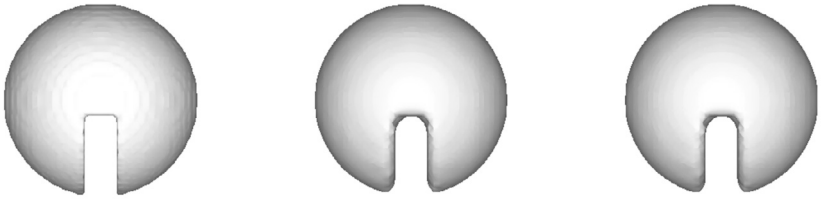
(a) Proposed method

(b) Pure level set method

Fig. 8. Snapshot of the predicted interfaces for the rotating sphere problem predicted in coarse grid 64^3. From left to right, $t = 0, T, 2T$.



(a) Proposed method



(b) Pure level set method

Fig. 9. Snapshot of the predicted interfaces for the rotating sphere problem predicted in fine grid 128^3. From left to right, $t = 0, T, 2T$.

will eventually break into several droplets (see Fig. 14). Droplet ejection owing to bubble impacting to thin film is also attributed to an intricate formation of capillary waves, in addition to the properties of the liquid under consideration in the gravity field [60]. Moreover, flow motion and energy dissipation of high speed jets have been studied intensively since it has been pointed out in many research reports that high stresses originating from bubble bursting could even kill live cells [54,55].

Gas bubble bursting at a free surface has been numerically investigated by Boulton-Stone and Blake [37]. However, their simulation did not take the pinch-off liquid jet into account. In this study, Fig. 14 plots the evolution of a liquid jet resulting from the submerged gas bubble. A spherical bubble of unit radius is initially located at $(x, y, z) = (0, 0, -3.2)$ in a box domain $\Omega : [-3, 3] \times [-3, 3] \times [-6, 10]$. The simulation is conducted in $90 \times 90 \times 240$ grids. No-slip condition has been

Table 9
Comparison of the computed error norms defined in Eq. (64) for the rotating sphere problem after one and two revolutions at different grid numbers.

Error norms	Grid number		
	32^3	64^3	128^3
Proposed method			
One revolution			
ε_M	9.8321×10^{-7}	1.1566×10^{-12}	2.2773×10^{-11}
$\bar{\varepsilon}_M$	4.7692×10^{-7}	5.0613×10^{-13}	8.7285×10^{-13}
ε_I	1.4356×10^{-3}	6.0994×10^{-4}	2.7426×10^{-4}
Two revolutions			
ε_M	1.8873×10^{-6}	1.4834×10^{-12}	1.2092×10^{-7}
$\bar{\varepsilon}_M$	1.2376×10^{-6}	1.1606×10^{-13}	1.5456×10^{-8}
ε_I	1.7437×10^{-3}	8.6733×10^{-4}	3.5663×10^{-4}
Pure level set method			
One revolution			
ε_M	3.5641×10^{-2}	3.8093×10^{-2}	7.6316×10^{-3}
$\bar{\varepsilon}_M$	2.3865×10^{-2}	2.1679×10^{-2}	4.7209×10^{-3}
ε_I	2.1107×10^{-3}	1.2685×10^{-3}	2.0662×10^{-4}
Two revolutions			
ε_M	6.7406×10^{-2}	4.4715×10^{-2}	8.1937×10^{-3}
$\bar{\varepsilon}_M$	4.9694×10^{-2}	4.4634×10^{-2}	8.8627×10^{-3}
ε_I	4.6010×10^{-3}	2.5634×10^{-3}	3.2132×10^{-4}

Table 10
Comparison of the CPU time (seconds) for the rotating sphere problem at different grid numbers after two revolutions.

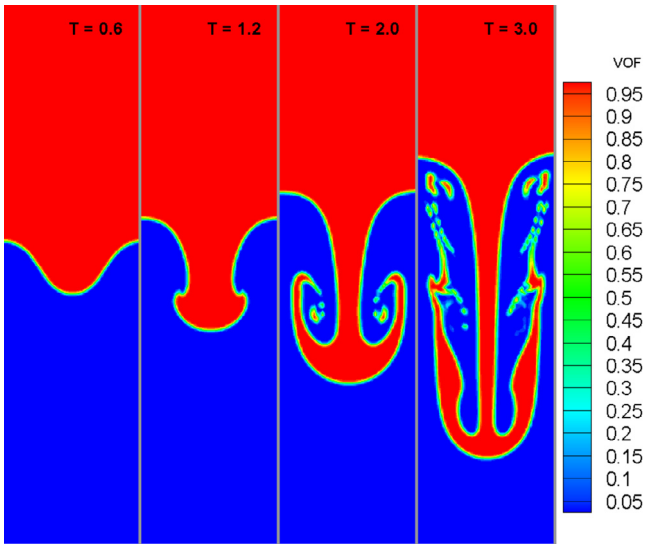
	Grid number		
	32^3	64^3	128^3
Proposed method	71.71	573.06	9564.30
Pure level set method	60.91	489.50	7361.36
Θ^+	17.73%	17.07 %	29.93%

applied in this simulation study. We set the Reynolds number as $Re = 474$, the Weber number as $We = 1$, and the Froude number as $Fr = 0.64$. The air–water density ratio and the viscosity ratio are chosen to be $\rho_g/\rho_l = 0.001$ and $\mu_g/\mu_l = 0.01$. The liquid jet starts breaking up into a droplet at a time around $t = 1.0$ and, then, the second droplet is formed at a time around 1.5. The pinch-off of the liquid jet and the consequent generation of liquid droplets, which are known to be generated by the Rayleigh-Plateau destabilization mechanism [60] and capillary instability [24], have been accurately predicted using the proposed method. According to our simulation results, a very thin jet and tiny droplets can be captured using the proposed method. It is also worthy to point out that the capillary driven focusing wave gives rise to a high-speed upward jet above the free surface [60]. The difference of the two solutions in Fig. 14 is negligibly small. However, the degree of volume conservation using the pure level set method is considerably inferior to that predicted by the proposed method for the simulation of pinch-off liquid jet, as shown in Fig. 15.

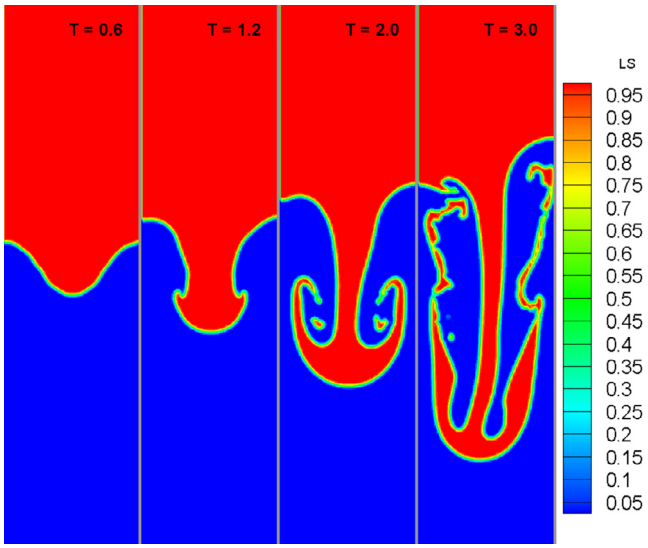
5.2. A single droplet impacting upon thin liquid layer to generate a milkcrown pattern

Droplet collision and impact are omnipresent phenomena occurring in nature and in processing industries. A profound understanding and a better control of the events subsequent to impact are crucial to many applications. For instance, spreading is desirable for coating or ink-jet printing while splashing may improve the efficiency of evaporation and mixing in fuel combustion [57,56]. The outcome of the impact depends on various factors, such as the speed and the type of fluids, and on the surface of substrate. If substrate is dry, results will depend on the wetting ability and the smoothness of the surface. Droplets impacting into a deep pool or a thin layer lead to an extra degree of physical complexity, as evidenced by the presence of a dramatic topology change resulting from the surface tension and capillary instability. A full knowledge of droplet impact into liquid surface is still lacking. The subjects of some unexplored complex dynamics about droplet impact include, for example, the understanding of the formation of singular surface deformation and the accompanying flow instability of different physical origins and the complex transition from splashing to spreading [57]. However, relations among the selected dimensionless parameters and the resulting outcomes have been discussed by many researchers [58,59].

The milkcrown problem has been investigated by many research groups to demonstrate their simulation ability of capturing a complicated topology change, both in two-dimensional [49] and three-dimensional [50,51] simulations. The initial setup in this study is identical to that in [50]. A droplet of diameter 5.33 mm with the initial velocity 2.0 m/s is impacting



(a) Proposed method



(b) Pure level set method

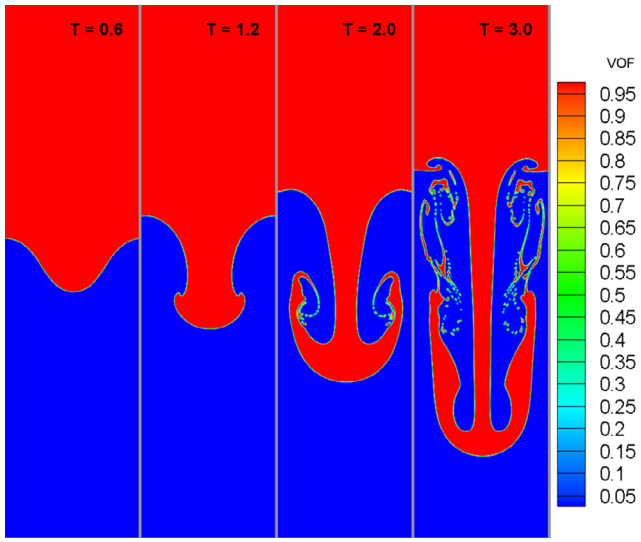
Fig. 10. Comparison of the predicted results for the Rayleigh-Taylor instability problem using different methods in coarse grid (50×200 grid points).

onto a liquid film of 1 mm depth. We set the Reynolds number as $Re = 6270.58$, the Weber number as $We = 426.4$, and the Froude number as $Fr = 8.75$. The air–water density ratio and the viscosity ratio are chosen to be $\rho_g/\rho_l = 0.0013$ and $\mu_g/\mu_l = 0.0006$. The computational domain is set in $\Omega : [-3, 3] \times [-3, 3] \times [0, 3]$.

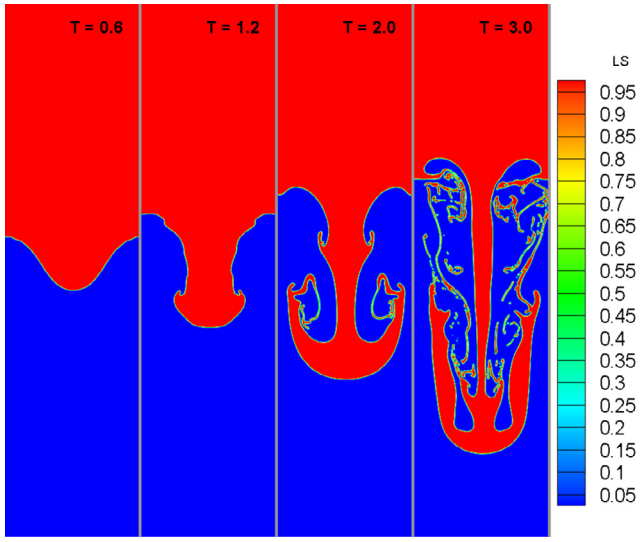
This problem is simulated in three grids with the resolutions of $180 \times 180 \times 90$, $240 \times 240 \times 120$ and $300 \times 300 \times 150$ to get the grid independent solution. The simulation results using the proposed method are shown in Figs. 16–18. In the coarse grid simulation, tiny drops are not observed, possibly due to the discretization error and the underestimation of the surface tension force (mainly by the calculation of curvature), as Kensuke Yokoi pointed out in [50]. We also simulate this problem using the pure level set method, as shown in Fig. 19, in fine grid $300 \times 300 \times 150$. One can see that the solution obtained by the pure level set method has been strongly affected by the poor volume conservation, as shown in Fig. 20. As a result, our proposed method is recommended for application to simulate problems involving complex topology change that results from the consideration of surface tension.

6. Concluding remarks

A LS-assisted VOF advection method is proposed in this study with an aim to capture the interface. In this method, volume fraction function C is used for tracking the time-evolving markers of interest interface, and level set function ϕ is used to assist an accurate calculation of the geometrically relevant quantities at the interface. Advection of C lies in the frame-



(a) Proposed method



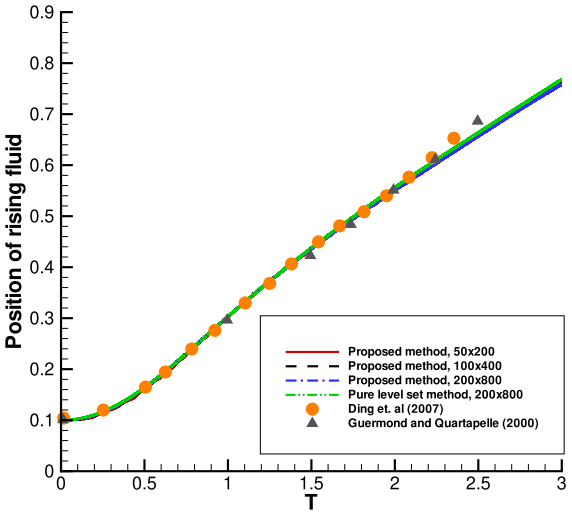
(b) Pure level set method

Fig. 11. Comparison of the predicted results for the Rayleigh-Taylor instability problem using different methods in fine grid. (200×800 grid points).

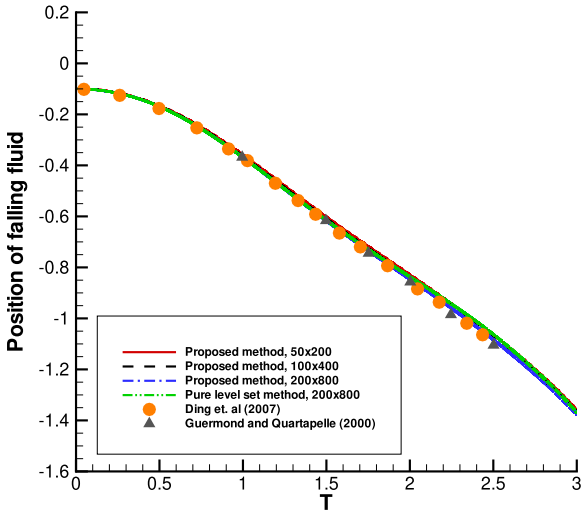
work of THINC (tangent of hyperbola for interface capturing) and WLIC (weighted line interface calculation) schemes, while the advection of ϕ is solved by using OCRWENO (optimized compact reconstruction weighted essentially non-oscillatory) scheme. Reconstruction of ϕ from C has been performed to avoid generating numerical instability in the simulation. For our proposed interface reconstruction procedure, its potential extension to parallel computing and the flexible adoption to the classical level set solver have been also pointed out.

Four verification and one validation studies have been conducted, including two-dimensional and three-dimensional problems. Three different error norms are introduced to justify the accuracy of solution obtained by the proposed scheme. According to the tabulated results, the proposed method can predict interface position accurately and retain the volume extremely well, in comparison with the pure level set method.

The proposed LS-assisted VOF advection scheme has been applied to simulate two-phase flow problems as well for showing its ability in capturing interface and retaining volume conservation. Two different problems were investigated, including the three-dimensional bubble bursting at a free surface and a single droplet impacting upon thin liquid layer. In our simulations, solutions obtained by our proposed method are less deteriorated by the errors leading to volume imbalance, in comparison with the pure level set method. As a result, we recommended application of our proposed method to predict two-phase flow, especially for problems involving a complex topological change.



(a) Rising fluid



(b) Falling fluid

Fig. 12. Comparison of the falling and rising fluid positions for the Rayleigh-Taylor instability problem.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Curvature calculation on a sphere

To show the different ability of applying level set function and volume fraction function to calculate the curvature, we take a sphere of radius 0.5 in a box domain of side length 1.5 as an example. The sphere can be described by the function $f(x, y, z)$ given below

$$f(x, y, z) = -\sqrt{x^2 + y^2 + z^2} + 0.5. \tag{72}$$

The level set function is initialized as $\phi(x_i, y_j, z_k) = f(x_i, y_j, z_k)$, and the volume fraction function $C(x_i, y_j, z_k)$ is initialized as

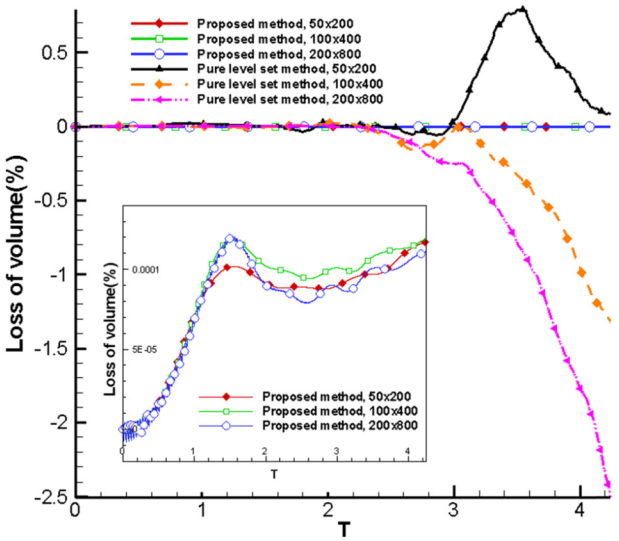


Fig. 13. Comparison of the predicted percentages of the loss of area using different methods under different grid resolutions for the Rayleigh-Taylor instability problem.

$$C(x_i, y_j, z_k) = \frac{1}{\Delta x \Delta y \Delta z} \int_{x_{i-1/2}}^{x_{i+1/2}} \int_{y_{j-1/2}}^{y_{j+1/2}} \int_{z_{k-1/2}}^{z_{k+1/2}} \chi(x, y, z)\, dx dy dz, \tag{73}$$

where the color function is given as

$$\chi(x, y, z) = \begin{cases} 1; \text{if } f(x, y, z) \geq 0, \\ 0; \qquad \text{otherwise.} \end{cases} \tag{74}$$

Firstly, we compute the surface normal vector $\mathbf{n}$ by ϕ and C, and, then, calculate the curvature through $\kappa = -\nabla \cdot \mathbf{n}$ with the second-order central difference scheme. Given the two values of curvature obtained by two different indicator functions, we compute the L_1 error norms at different grid resolutions for each of them, as shown in Table 11.

According to the tabulated results, curvature calculated by using the level set function converges well with the increase of the grid number. However, due to the steep gradient of C across the interface (this issue can not be resolved by increasing the number of grid points), direct calculation of curvature is not possible to get the same degree of accuracy as the level set function.

Appendix B. Error norms calculation and interface reconstruction

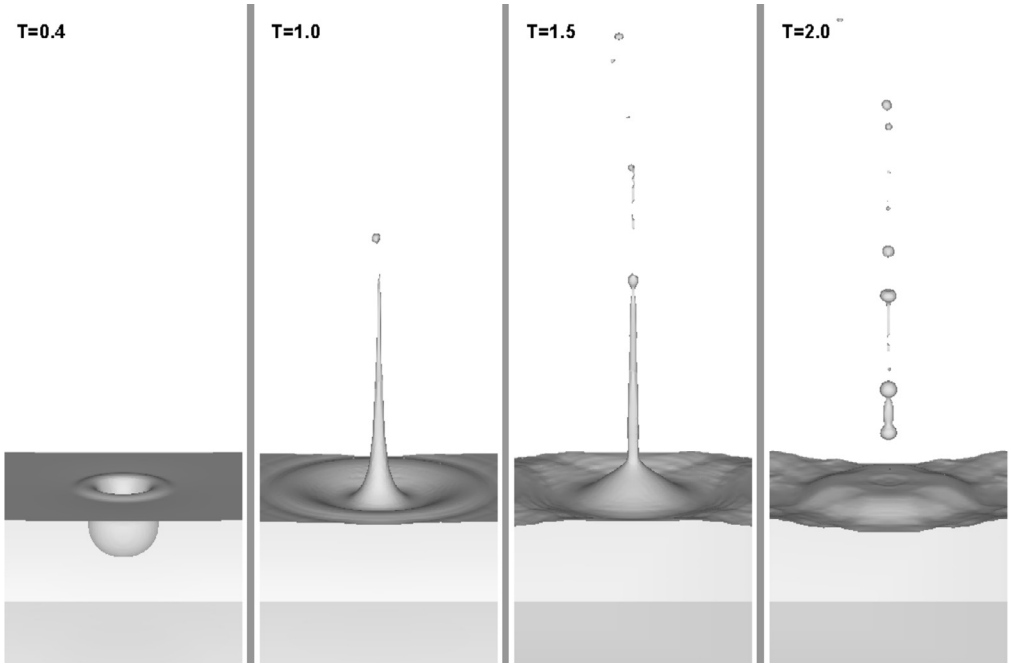
To show the importance of interface reconstruction procedure described in Section 2.1.3, we simulate the problem in Section 5.1 in $60 \times 60 \times 120$ grids for two cases, one is with the consideration of interface reconstruction and the other is not. In order to show the degree of mismatch of the interface position described by level set function and volume fraction function, we define the difference function $\hat{\varepsilon}_{i,j,k}$ at each node

$$\hat{\varepsilon}_{i,j,k} = H(\phi_{i,j,k}) - C_{i,j,k}. \tag{75}$$

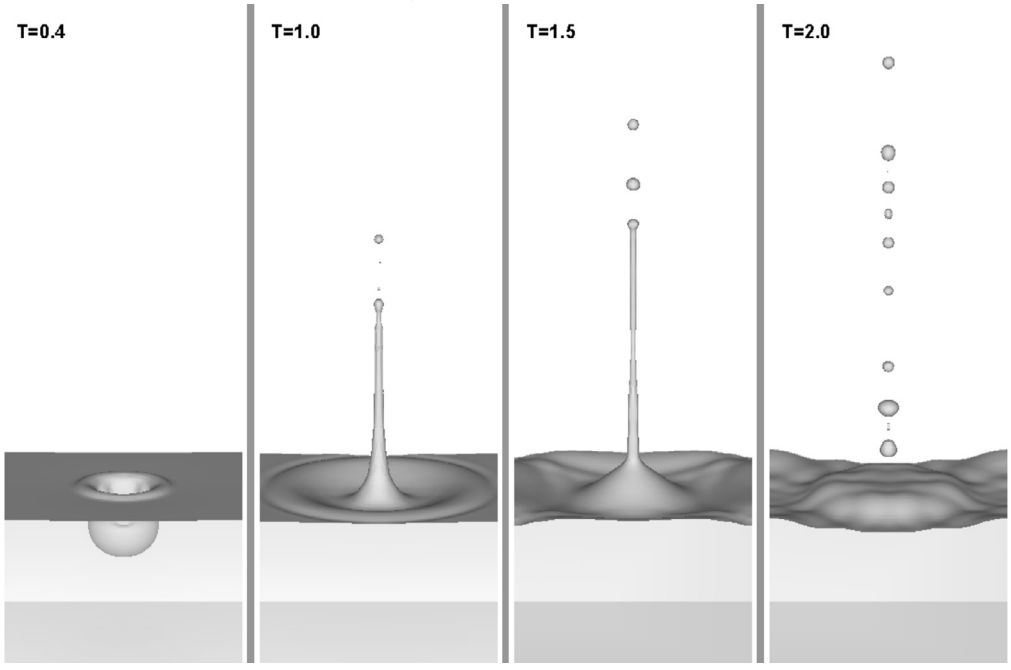
The corresponding L_2 norms of $\hat{\varepsilon}_{i,j,k}$ are plot in Fig. 21 to show its time evolution. According to the figure, solution obtained by considering interface reconstruction has a smaller value of $\hat{\varepsilon}_{i,j,k}$ than that of $\hat{\varepsilon}_{i,j,k}$ without consideration the reconstruction procedure. As a result, interface reconstruction and the consideration of η function in Section 2.1.3 are needed for implementing our proposed method. It is noted that the interface has different degrees of diffusive effect for level set function and volume fraction function. As a result, the norm defined in Eq. (75) is not zero at the beginning of the simulation.

Appendix C. Determination of the frequency of performing interface reconstruction

To determine how frequently the interface reconstruction described in Section 2.1.3 should be performed, we conduct a numerical test by simulating the problem in Section 5.1 under different frequency of applying reconstruction, that is, every 5, 10, 15, 20 steps. The higher the frequency, the higher accuracy of the solution is obtained, but may be at the cost of



(a) Proposed method



(b) Pure level set method

Fig. 14. Comparison of the predicted jetting events following a bubble bursting at a free surface. Simulation has been conducted in $90 \times 90 \times 120$ grid points.

deteriorating the computational efficiency. If the frequency is too low, numerical instabilities will occur and lead to several unpredictable numerical issues.

According to the comparison of CPU times shown in Table 12, the frequency of every 10 steps for the interface reconstruction has a better computational efficiency. Frequencies of every 15 and 20 steps seem to have some numerical instabilities, which seriously slow down the iteration of pressure Poisson equation, please refer to Eq. (59). According to the plot of the predicted percentage loss of volume (see Fig. 22), frequency of every 5 and 10 steps has a similar ability of preserving the volume. As a result, we conduct the interface reconstruction only for every 5 steps.

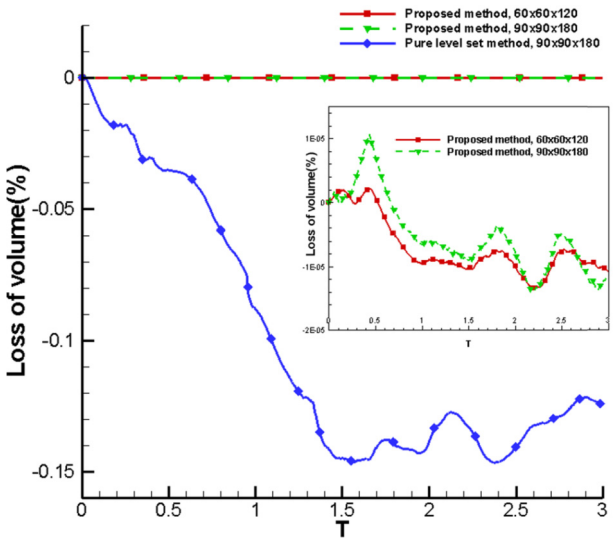


Fig. 15. Comparison of the predicted percentages of the loss of volume using different methods with different grid numbers for the problem of bubble bursting at the free surface.

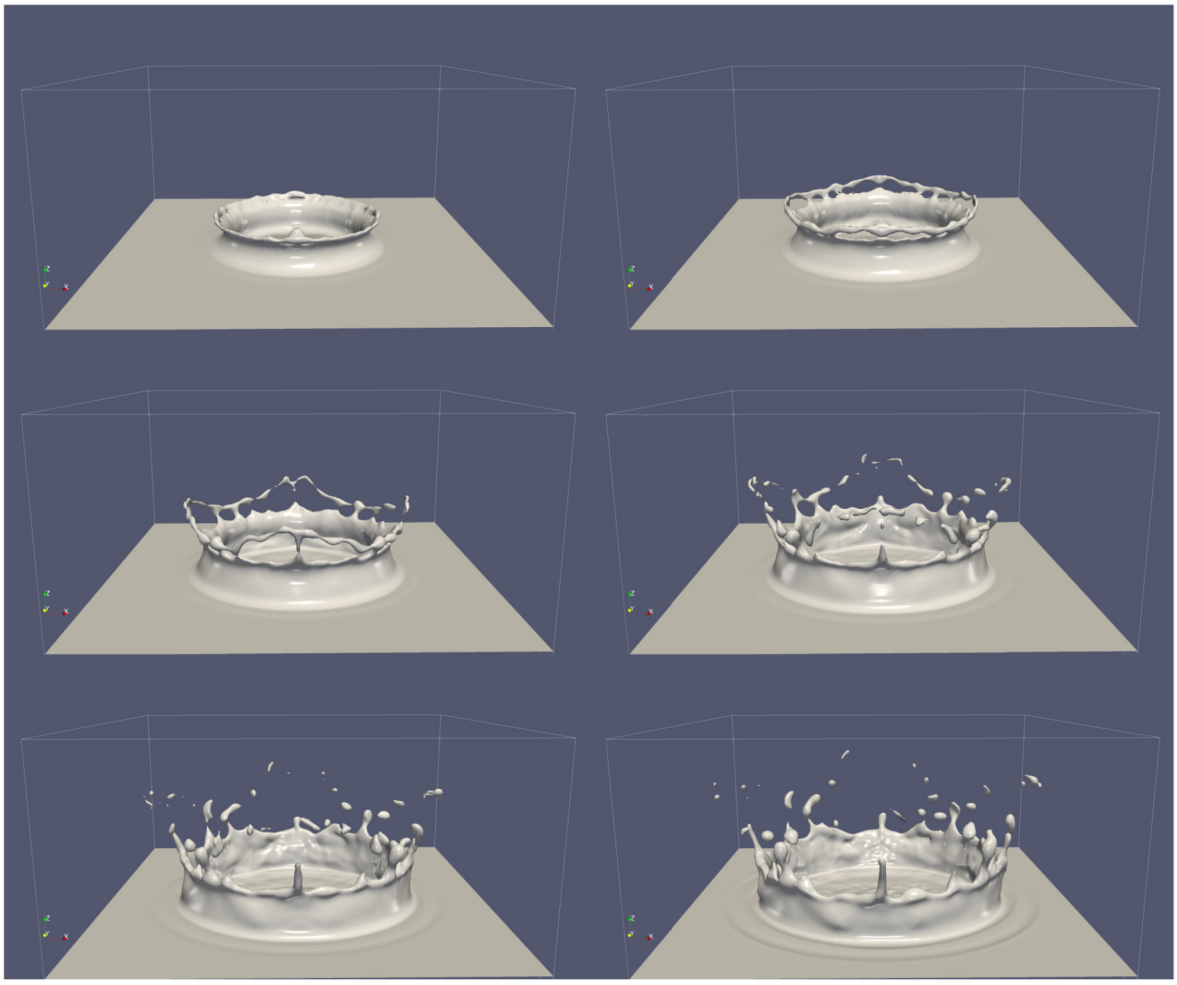


Fig. 16. Plot of the crown forming events (From top to down, left to right, $t = 1.5, 2.0, 2.5, 3.0, 3.5, 4.0$) following a droplet impacting upon a thin liquid layer. The simulation has been conducted in $180 \times 180 \times 90$ grid points using the proposed method.

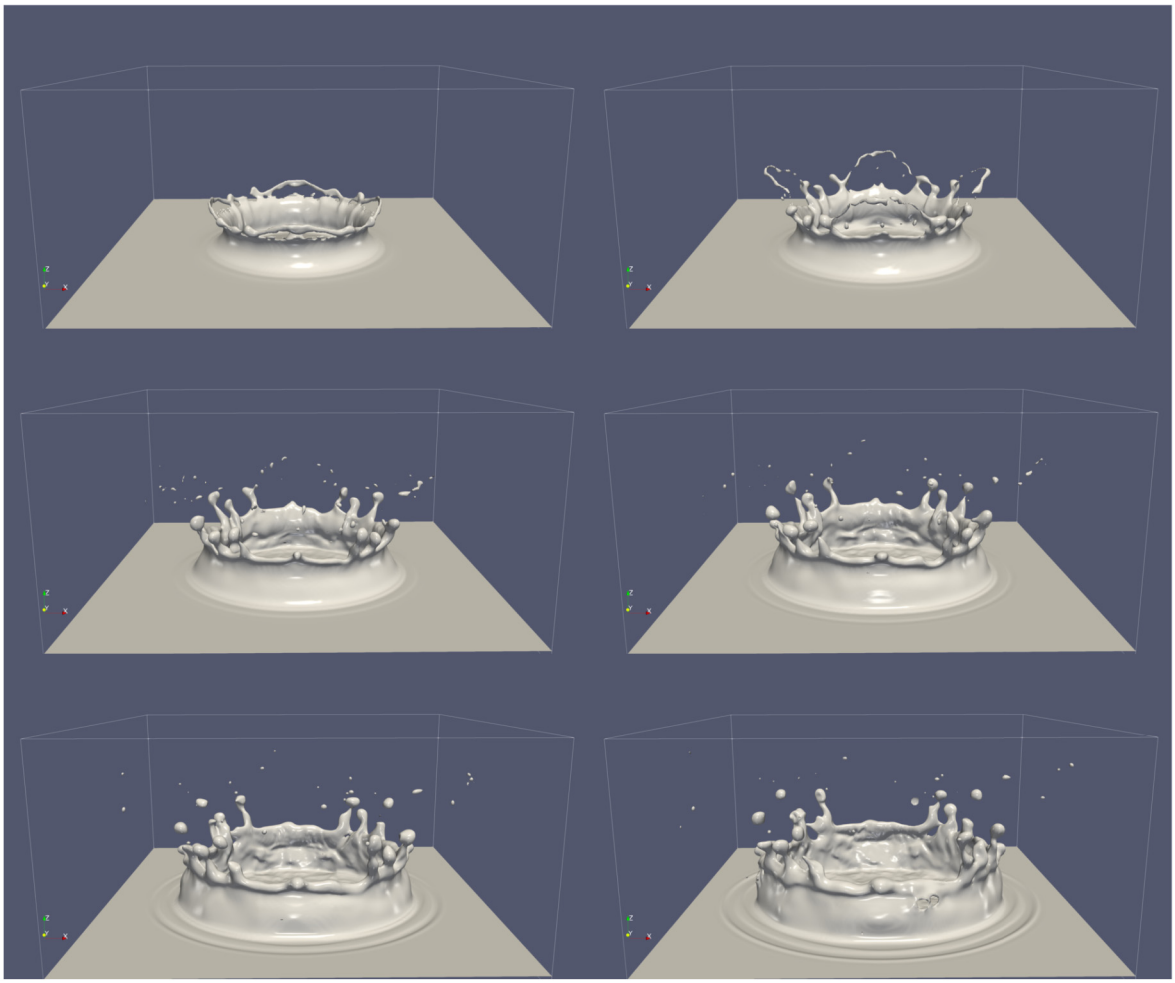


Fig. 17. Plot of the crown forming events (From top to down, left to right, $t = 1.5, 2.0, 2.5, 3.0, 3.5, 4.0$) following a droplet impacting upon a thin liquid layer. The simulation has been conducted in $240 \times 240 \times 120$ grid points using the proposed method.

Table 11
L_1 error norms and the corresponding rates of convergence for curvature calculation on a sphere at different grids using different indicator functions.

Grids	Level set function	Order	Volume fraction function	Order
48^3	1.6397×10^{-2}	–	3.8595×10^{-2}	–
96^3	4.0812×10^{-3}	2.0064	2.9223×10^{-2}	0.4013
192^3	1.0171×10^{-3}	2.0045	2.5831×10^{-2}	0.1780
384^3	2.5616×10^{-4}	1.9893	2.7813×10^{-2}	-0.1067

Table 12
CPU time for different frequency of applying the interface reconstruction procedure for the problem of bubble bursting at the free surface.

Frequency	CPU time (s)
Every 5 steps	29928
Every 10 steps	29688
Every 15 steps	40442
Every 20 steps	38019

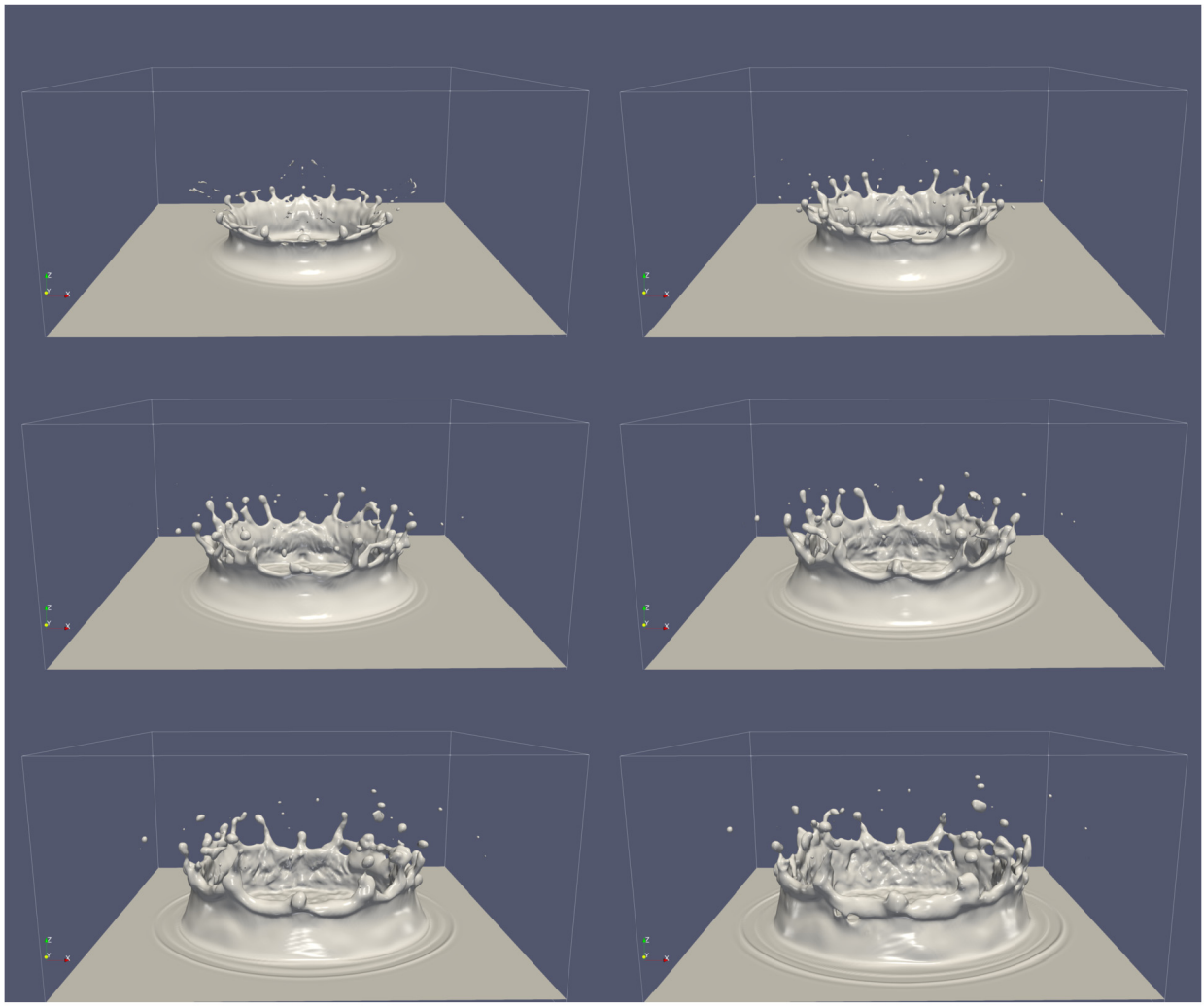


Fig. 18. Plot of the crown forming events (From top to down, left to right, $t = 1.5, 2.0, 2.5, 3.0, 3.5, 4.0$) following a droplet impacting upon a thin liquid layer. The simulation has been conducted in $300 \times 300 \times 150$ grid points using the proposed method.

Appendix D. Derivation of optimized CRWENO scheme

The derivation of this scheme is summarized below in four steps:
(**Step 1**) The upwind compact difference scheme given below is applied

$$\overline{\mathrm{A}}\frac{\partial \phi}{\partial x}|_{i-1} + \frac{\partial \phi}{\partial x}|_i + \overline{\mathrm{C}}\frac{\partial \phi}{\partial x}|_{i+1} = \frac{\overline{\mathrm{a}}\phi_{i-2} + \overline{\mathrm{b}}\phi_{i-1} + \overline{\mathrm{c}}\phi_i + \overline{\mathrm{d}}\phi_{i+1}}{\Delta x}. \tag{76}$$

By performing the Taylor series expansions on $\frac{\partial \phi}{\partial x}|_{i-1}$, $\frac{\partial \phi}{\partial x}|_{i+1}$, ϕ_{i-2}, ϕ_{i-1} and ϕ_{i+1} with respect to ϕ_i, the leading five truncation error terms in the corresponding modified equations are eliminated. The following set of algebraic equations is derived as

$$\begin{aligned}
&\bar{a} + \bar{b} + \bar{c} + \bar{d} = 0,\\
&-2\bar{a} - \bar{b} + \bar{d} - \bar{A} - \bar{C} = 1,\\
&2\bar{a} + \frac{1}{2}\bar{b} + \frac{1}{2}\bar{d} + \bar{A} - \bar{C} = 0,\\
&-\frac{4}{3}\bar{a} - \frac{1}{6}\bar{b} + \frac{1}{6}\bar{d} - \frac{1}{2}\bar{A} - \frac{1}{2}\bar{C} = 0,\\
&\frac{2}{3}\bar{a} + \frac{1}{24}\bar{b} + \frac{1}{24}\bar{d} + \frac{1}{6}\bar{A} - \frac{1}{6}\bar{C} = 0.
\end{aligned} \tag{77}$$

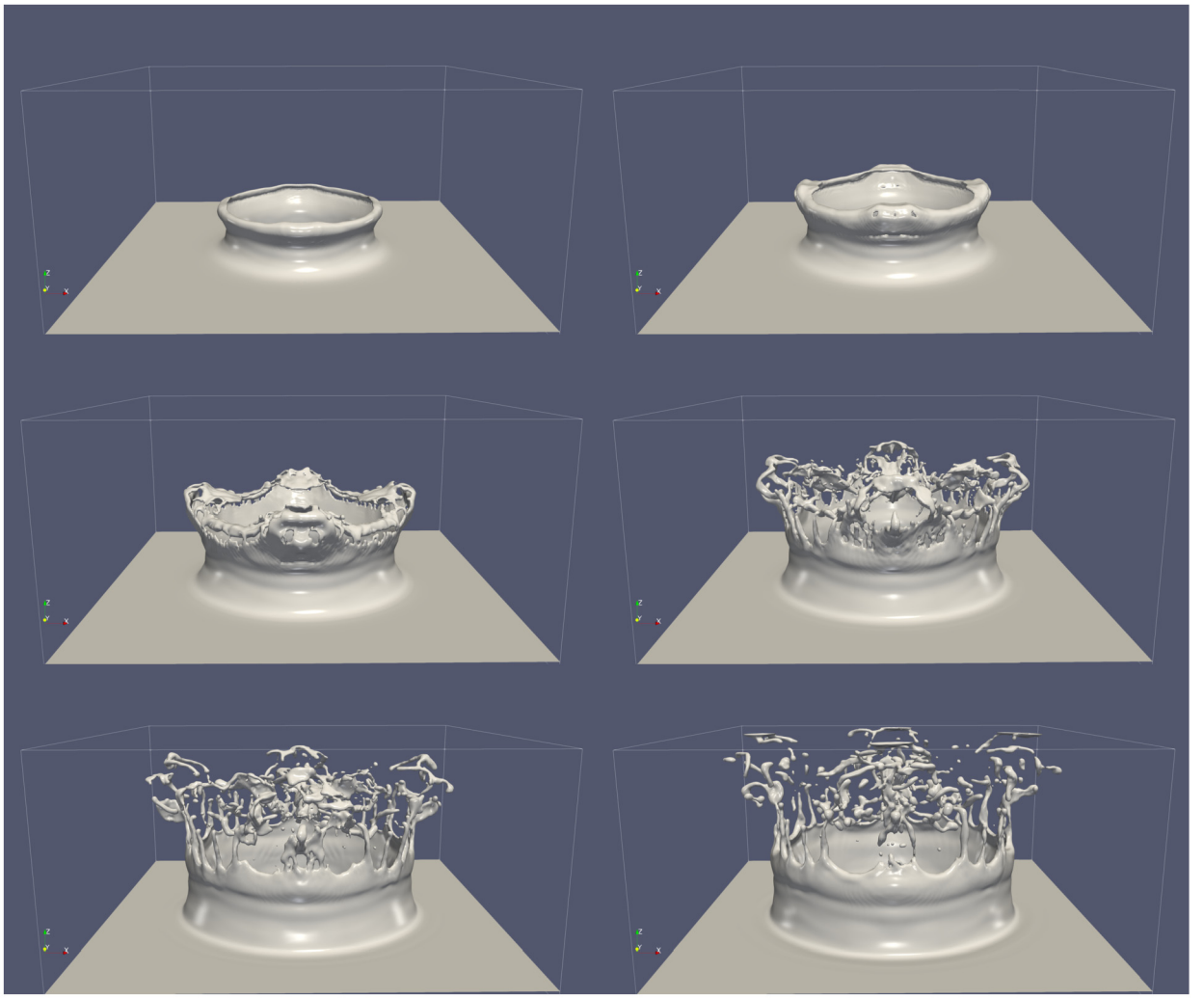


Fig. 19. Plot of the crown forming events (From top to down, left to right, $t = 1.5, 2.0, 2.5, 3.0, 3.5, 4.0$) following a droplet impacting upon a thin liquid layer. The simulation has been conducted in $300 \times 300 \times 150$ grid points using the pure level set method.

To reduce numerical dispersion error generated from this compact scheme for modeling wave propagation over a long distance or a long period of simulation time has elapsed, we need another algebraic equation to uniquely determine all the six introduced coefficients shown in Eq. (76) that can altogether render the smallest dispersion error. Performing the modified wave number analysis [40] on each term shown in Eq. (76), the equation for the corresponding numerical wavenumber $\alpha'\Delta x$ can be derived

$$\mathbf{i}\alpha'\Delta x(\bar{A}e^{-\mathbf{i}\alpha\Delta x} + 1 + \bar{C}e^{\mathbf{i}\alpha\Delta x}) = \bar{a}e^{-2\mathbf{i}\alpha\Delta x} + \bar{b}e^{-\mathbf{i}\alpha\Delta x} + \bar{c} + \bar{d}e^{\mathbf{i}\alpha\Delta x} \tag{78}$$

The expression of $\alpha'\Delta x$ can then be expressed as follows by substituting the coefficients, as shown in Eq. (77), into Eq. (78)

$$\alpha'\Delta x = \frac{\mathbf{i}(-3e^{-2\mathbf{i}\alpha\Delta x} + 4\bar{d}e^{-2\mathbf{i}\alpha\Delta x} - 24e^{-\mathbf{i}\alpha\Delta x} + 18\bar{d}e^{-\mathbf{i}\alpha\Delta x} + 27 - 36\bar{d} + 14\bar{d}e^{\mathbf{i}\alpha\Delta x})}{-14 - 17e^{-\mathbf{i}\alpha\Delta x} + 18\bar{d}e^{-\mathbf{i}\alpha\Delta x} + e^{\mathbf{i}\alpha\Delta x} - 6\bar{d}e^{\mathbf{i}\alpha\Delta x}} \tag{79}$$

To make α' an appropriate representation of α, it is required that the following positive error function $E(\alpha)$ should take the smallest value

$$E(\alpha) = \int_0^{\frac{17\pi}{20}} \left[W_d \cdot (\alpha\,\Delta x - \Re[\alpha'\,\Delta x])\right]^2 d(\alpha\Delta x), \tag{80}$$

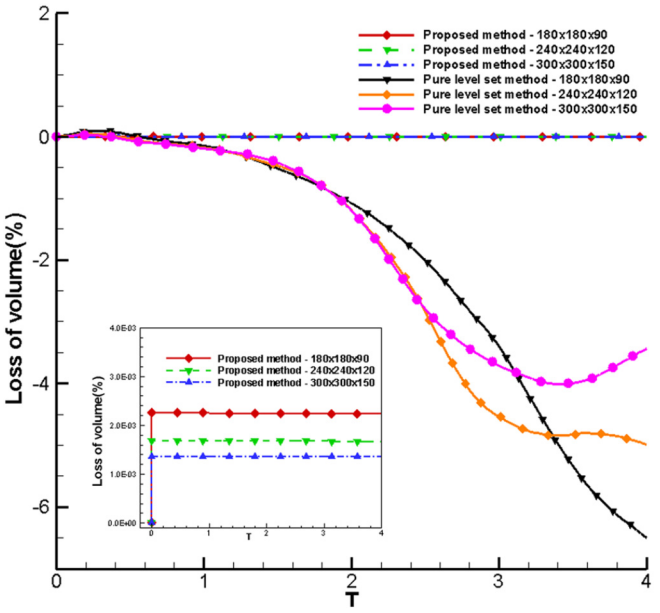


Fig. 20. Comparison of the predicted percentages of the loss of volume using different methods under different grid resolutions for the problem of a single droplet impacting upon a thin liquid layer.

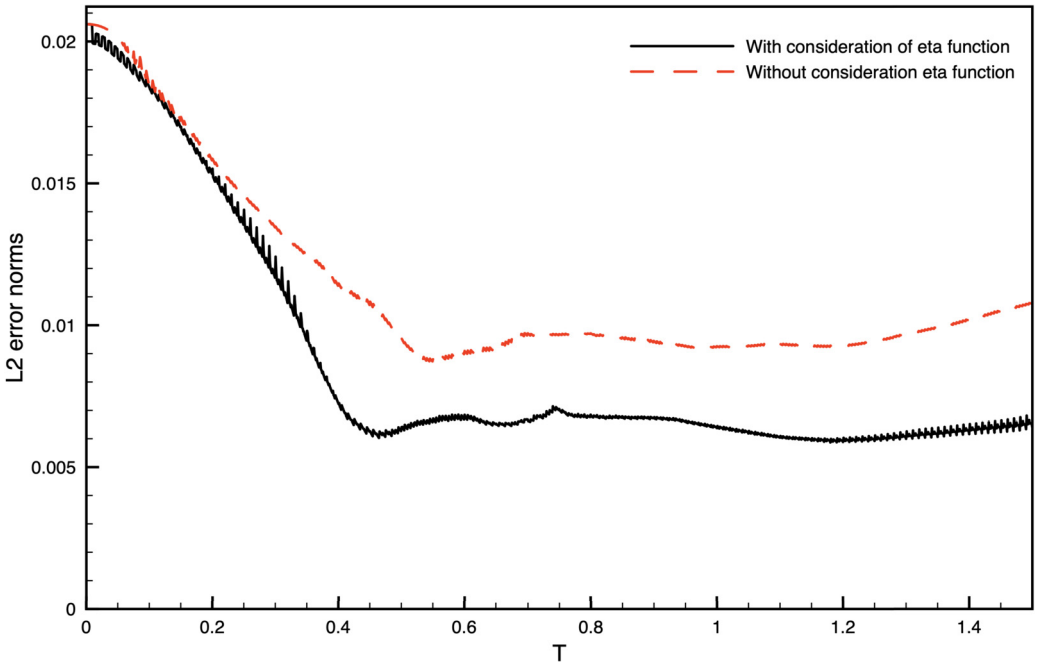


Fig. 21. Comparison of the predicted L_2 difference norms with and without the consideration of η function in the interface reconstruction procedure.

where W_d shown above is chosen to be the denominator of $\left(\alpha\,\Delta x - \Re[\alpha'\,\Delta x]\right)$ so as to make it possible to integrate Eq. (80) analytically. The constraint condition defined below is enforced so that we can minimize the value of $E(\alpha)$

$$\frac{\partial E(\alpha)}{\partial \overline{\mathrm{d}}} = 0. \tag{81}$$

This constraint equation is coupled with the other five algebraic equations shown in Eq. (77) to get the optimized set of coefficients: $\overline{\mathrm{A}} = 0.5418416108$, $\overline{\mathrm{C}} = 0.1527194630$, $\overline{\mathrm{a}} = -0.0648536914$, $\overline{\mathrm{b}} = -1.041841611$, $\overline{\mathrm{c}} = 0.583683223$, and $\overline{\mathrm{d}} = 0.5230120803$.

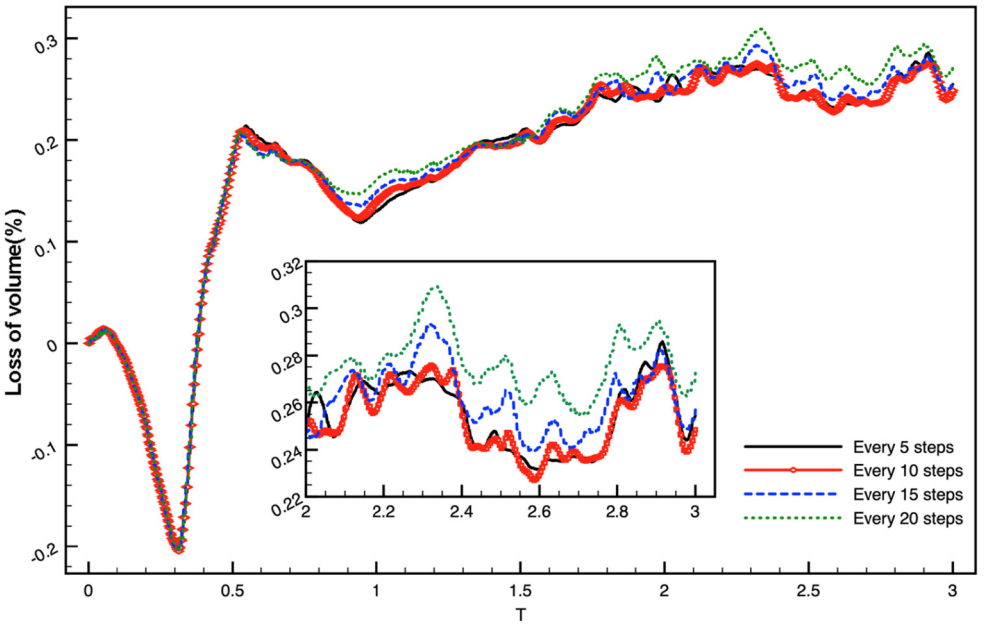


Fig. 22. Comparison of the predicted percentages of the loss of volume using different frequency of applying the interface reconstruction procedure for the problem of bubble bursting at the free surface.

(**Step 2**) Define the values $\hat{\phi}$ at half nodal point $i \pm \frac{1}{2}$ as follows

$$\widetilde{A}\hat{\phi}_{i-\frac{1}{2}} + \hat{\phi}_{i+\frac{1}{2}} + \widetilde{C}\hat{\phi}_{i+\frac{3}{2}} = \widetilde{a}\phi_{i-1} + \widetilde{b}\phi_i + \widetilde{c}\phi_{i+1}, \tag{82}$$

and

$$\widetilde{A}\hat{\phi}_{i-\frac{3}{2}} + \hat{\phi}_{i-\frac{1}{2}} + \widetilde{C}\hat{\phi}_{i+\frac{1}{2}} = \widetilde{a}\phi_{i-2} + \widetilde{b}\phi_{i-1} + \widetilde{c}\phi_i. \tag{83}$$

The coefficients in Eq. (82) and Eq. (83), which are $\widetilde{A} = 0.5418416108$, $\widetilde{C} = 0.1527194630$, $\widetilde{a} = 0.0648536914$, $\widetilde{b} = 1.106695303$ and $\widetilde{c} = 0.5230120803$, are obtained by comparing the optimized coefficients derived in Eq. (76) for $\frac{\partial \phi}{\partial x}|_i = (\hat{\phi}_{i+\frac{1}{2}} - \hat{\phi}_{i-\frac{1}{2}})/\Delta x$.

(**Step 3**) The following three third-order accurate compact interpolations have been used for the left-biased compact reconstruction of $\hat{\phi}_{i+1/2}$ [41]

$$\begin{aligned}
&\frac{2}{3}\hat{\phi}^1_{i-1/2} + \frac{1}{3}\hat{\phi}^1_{i+1/2} = \frac{1}{6}(\phi_{i-1} + 5\phi_i),\\
&\frac{1}{3}\hat{\phi}^2_{i-1/2} + \frac{2}{3}\hat{\phi}^2_{i+1/2} = \frac{1}{6}(5\phi_i + \phi_{i+1}),\\
&\frac{2}{3}\hat{\phi}^3_{i+1/2} + \frac{1}{3}\hat{\phi}^3_{i+3/2} = \frac{1}{6}(\phi_i + 5\phi_{i+1}).
\end{aligned} \tag{84}$$

Applying three different weighting coefficients c_1, c_2, c_3 on Eq. (84), the following tridiagonal matrix equation can be derived

$$\begin{aligned}
&\left[\frac{2c_1 + c_2}{3}\right]\hat{\phi}_{i-\frac{1}{2}} + \left[\frac{c_1 + 2(c_2 + c_3)}{3}\right]\hat{\phi}_{i+\frac{1}{2}} + \frac{c_3}{3}\hat{\phi}_{i+\frac{3}{2}}\\
&= \frac{c_1}{6}\phi_{i-1} + \left[\frac{5(c_1 + c_2) + c_3}{6}\right]\phi_i + \left[\frac{c_2 + 5c_3}{6}\right]\phi_{i+1}.
\end{aligned} \tag{85}$$

After a term-by-term comparison of the coefficients in Eq. (82) and Eq. (85), the coefficients $c_1 = 0.20891413$, $c_2 = 0.49999999$ and $c_3 = 0.29108586$ can then be obtained.

References

[1] I. Chakraborty, G. Biswas, P.S. Ghoshdastidar, A coupled level-set and volume-of-fluid method for the buoyant rise of gas bubbles in liquids, Int. J. Heat Mass Transf. 58 (2013) 240–259.
[2] B. Ramaswamy, Numerical simulation of unsteady viscous free surface flow, J. Comput. Phys. 90 (1990) 396–430.
[3] J.M. Floryan, H. Rasmussen, Numerical methods for viscous flows with moving boundaries, Appl. Mech. Rev. 42 (1989) 323–341.
[4] B.J. Daly, Numerical study of two fluid Rayleigh–Taylor instability, Phys. Fluids 10 (1967) 297–307.
[5] B.J. Daly, A technique for including surface tension effects in hydrodynamics calculations, J. Comput. Phys. 4 (1969) 97–117.
[6] J. Glimm, O. McBryan, R. Menikoff, D.H. Sharp, Front tracking applied to Rayleigh–Taylor instability, SIAM J. Sci. Stat. Comput. 7 (1986) 230–251.
[7] G. Tryggvason, B. Bunner, A. Esmaeeli, D. Juric, N. Al-Rawahi, W. Tauber, J. Han, S. Nas, Y.J. Jan, A front tracking method for the computations of multiphase flow, J. Comput. Phys. 169 (2001) 708–759.
[8] S.O. Univerdi, G. Tryggvason, A front-tracking method for viscous, incompressible multi-fluid flows, J. Comput. Phys. 100 (1992) 25–37.
[9] M. van Sint Annaland, W. Dijkhuizen, N.G. Deen, J.A.M. Kuipers, Numerical simulation of behavior of gas bubbles using 3-D front-tracking method, AIChE J. 52 (2006) 99–110.
[10] S. Popinet, S. Zaleski, A front-tracking algorithm for accurate representation of surface tension, Int. J. Numer. Methods Fluids 30 (1999) 775–793.
[11] B.M. Ningegowda, B. Premachandran, A coupled level set and volume of fluid method with multi-directional advection algorithms for two-phase flows with and without phase change, Int. J. Heat Mass Transf. 79 (2014) 532–550.
[12] S. Osher, J.A. Sethian, Fronts propagating with curvature-dependent speed: algorithm based on Hamilton–Jacobi formulations, J. Comput. Phys. 79 (1988) 12–49.
[13] 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.
[14] J.A. Sethian, Level Set Methods and Fast Marching Methods, Cambridge University Press, Cambridge, UK, 1999.
[15] C.W. Hirt, B.D. Nichols, Volume of fluid (VOF) method for the dynamics of free boundaries, J. Comput. Phys. 39 (1981) 201–225.
[16] S.W.J. Welch, J. Wilson, A volume of fluid based method for fluid flows with phase change, J. Comput. Phys. 160 (2000) 662–682.
[17] D.K. Agarwal, S.W.J. Welch, G. Biswas, F. Durst, Planar simulation of bubble growth in film boiling in near-critical water using a variant of the VOF method, J. Heat Transfer 126 (2004) 329–338.
[18] W.J. Rider, D.B. Kothe, Reconstructing volume tracking methods, J. Comput. Phys. 141 (1998) 112–152.
[19] M. Rudman, A volume-tracking method for incompressible multifluid flows with large density variations, Int. J. Numer. Methods Fluids 28 (1998) 357–378.
[20] M. Rudman, Volume tracking methods for interfacial flow calculations, Int. J. Numer. Methods Fluids 24 (1997) 671–691.
[21] L. Chen, S.V. Garimella, J.A. Reizes, E. Leonardi, The development of a bubble rising in a viscous liquid, J. Fluid Mech. 387 (1999) 61–96.
[22] M. van Sint Annaland, N.G. Deen, J.A.M. Kuipers, Numerical simulation of gas bubbles behavior using a three-dimensional volume of fluid method, Chem. Eng. Sci. 60 (2005) 2999–3011.
[23] A. Bourlioux, Coupled level set volume of fluid algorithm for tracking material interfaces, in: Proceedings of the 6th International Symposium on Computational Fluid Dynamics, Lake Tahoe, vol. 15, 1995.
[24] M. Sussman, E.G. Puckett, A coupled level set and volume of fluid method for computing 3D and axisymmetric incompressible two-phase flows, J. Comput. Phys. 162 (2000) 301–337.
[25] G. Son, N. Hur, A coupled level-set and volume-of-fluid method for the buoyancy-driven motion of fluid particles, Numer. Heat Transf., Part B 42 (2002) 523–542.
[26] G. Son, Efficient implementation of a coupled level-set and volume-of-fluid method for three-dimensional incompressible two-phase flows, Numer. Heat Transf., Part B 43 (2003) 549–565.
[27] D. Gerlach, G. Tomar, G. Biswas, F. Durst, Comparison of volume-of-fluid methods for computing surface tension-dominant two-phase flows, Int. J. Heat Mass Transf. 49 (2006) 740–754.
[28] W.F. Noh, P.R. Woodward, SLIC (simple line interface method), in: A.I. van der Vooren, P.J. Zandbergen (Eds.), Lecture Notes in Physics, vol. 59, Springer-Verlag, Berlin, New York, 1976, pp. 330–340.
[29] D.L. Youngs, Time-dependent multi-material flow with large fluid distortion, in: W. Morton, M.J. Baines (Eds.), Numerical Methods for Fluid Dynamics, Academic Press, New York, 1982, pp. 273–285.
[30] F. Xiao, Y. Honma, T. Kono, A simple algebraic interface capturing scheme using hyperbolic tangent function, Int. J. Numer. Methods Fluids 48 (2005) 1023–1040.
[31] K. Yokoi, Efficient implementation of THINC scheme: a simple and practical smoothed VOF algorithm, J. Comput. Phys. 226 (2007) 1985–2002.
[32] F. Xiao, S. Ii, C. Chen, Revisit to the THINC scheme: a simple algebraic VOF algorithm, J. Comput. Phys. 230 (2011) 7086–7092.
[33] G. Tomar, G. Biswas, A. Sharma, A. Agrawal, Numerical simulation of bubble growth in film boiling using CLSVOF method, Phys. Fluids 17 (2005) 112103.
[34] T. Menard, S. Tanguy, A. Berlemont, Coupling level set/VOF/ghost fluid methods: validation and application to 3D simulation of the primary breakup of a liquid jet, Int. J. Multiph. Flow 33 (2007) 510–524.
[35] 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, Int. J. Multiph. Flow 35 (2009) 227–246.
[36] M. Sussman, E. Fatemi, An efficient, interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow, SIAM J. Sci. Comput. 20 (1999) 1165–1191.
[37] J.M. Boulton-Stone, J.R. Blake, Gas bubbles bursting at a free surface, J. Fluid Mech. 254 (1993) 437–466.
[38] G.S. Jiang, D. Peng, Weighted ENO schemes for Hamilton-Jacobi equations, SIAM J. Sci. Comput. 21 (1997) 2126–2143.
[39] C.W. Shu, Total-variation diminishing time discretizations, SIAM J. Sci. Stat. Comput. 9 (1988) 1073–1984.
[40] C.K.W. Tam, J.C. Webb, Dispersion-relation-preserving finite difference schemes for computational acoustics, J. Comput. Phys. 107 (1993) 262–281.
[41] D. Ghosh, J.D. Baeder, Compact reconstruction schemes with weighted ENO limiting for hyperbolic conservation laws, SIAM J. Sci. Comput. 34 (2012), A1678-A1706.
[42] Z.H. Gu, H.L. Wen, C.H. Yu, Tony W.H. Sheu, Interface-preserving level set method for simulating dam-break flows, J. Comput. Phys. 374 (2018) 249–280.
[43] H. Ding, Peter D.M. Spelt, C. Shu, Diffuse interface model for incompressible two-phase flows with large density ratios, J. Comput. Phys. 226 (2007) 2078–2095.
[44] J.L. Guermond, L. Quartapelle, A projection FEM for variable density incompressible flows, J. Comput. Phys. 165 (2000) 167–188.
[45] R. LeVeque, High-resolution conservative algorithms for advection in incompressible flow, SIAM J. Numer. Anal. 33 (1996) 627–665.
[46] A. Prosperetti, Motion of two superposed viscous fluids, Phys. Fluids 24 (1981) 1217.
[47] G. Tryggvason, Numerical simulations of the Rayleigh-Taylor instability, J. Comput. Phys. 75 (1988) 253–282.
[48] B. Ray, G. Biswas, A. Sharma, S.W.J. Welch, CLSVOF method to study consecutive drop impact on liquid pool, Int. J. Numer. Methods Heat Fluid Flow 23 (2013) 143–158.
[49] F.H. Harlow, J.P. Shannon, The splash of a liquid drop, J. Appl. Phys. 38 (1967) 3855–3866.

[50] Kensuke Yokoi, A numerical method for free-surface flows and its application to droplet impact on a thin liquid layer, J. Sci. Comput. 35 (2008) 372–396.
[51] F. Xiao, A. Ikebata, T. Hasegawa, Numerical simulations of free-interface fluids by a multi integrated moment method, Comput. Struct. 83 (2005) 409–423.
[52] L. Duchemin, S. Popinet, C. Josserand, S. Zaleski, Jet formation in bubbles bursting at a free surface, Phys. Fluids 14 (2002) 3000.
[53] M. Coantic, Mass transfer across the ocean-air interface: small scale hydrodynamic and aerodynamic mechanisms, Physicochem. Hydrodyn. 1 (1980) 249–279.
[54] Peter L.L. Walls, Oliver McRae, Venkatesh Natarajan, Chris Johnson, Chris Antoniou, James C. Bird, Quantifying the potential for bursting bubbles to damage suspended cells, Sci. Rep. 7 (2007) 15102.
[55] D. Blanchard, C. Syzde, Concentration of bacteria in jet drops from bursting bubbles, J. Geophys. Res. 77 (1972) 5087–5099.
[56] G. Agbaglah, M.-J. Thoraval, S.T. Thoroddsen, L.V. Zhang, K. Fezzaa, R.D. Deegan, Drop impact into a deep pool: vortex shedding and jet formation, J. Fluid Mech. 764 (2015), R1.
[57] C. Josserand, P. Ray, S. Zaleski, Droplet impact on a thin liquid film: anatomy of the splash, J. Fluid Mech. 802 (2016) 775–805.
[58] R. Rioboo, C. Bauthier, J. Conti, M. Voue, J. De Coninck, Experimental investigation of splash and crown formation during single drop impact on wetted surfaces, Exp. Fluids 35 (2003) 648–652.
[59] A.-B. Wang, C.-C. Chen, Splashing impact of a single drop onto very thin liquid films, Phys. Fluids 12 (2000) 2155.
[60] E. Ghabache, A. Antkowiak, C. Josserand, T. Séon, On the physics of fizziness: how bubble bursting controls droplets ejection, Phys. Fluids 26 (2014) 121701.