

Coupled THINC and level set method: A conservative interface capturing scheme with high-order surface representations

Longgen Qian^{a,b}, Yanhong Wei^{a,*}, Feng Xiao^{b,*}

^a College of Materials Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, 211100, China

^b Department of Mechanical Engineering, Tokyo Institute of Technology, 2-12-1(i6-29) Ookayama, Meguro-ku, Tokyo, 152-8550, Japan

ARTICLE INFO

Article history:

Received 7 March 2018

Received in revised form 15 May 2018

Accepted 29 June 2018

Available online 3 July 2018

Keywords:

Moving interface

Multiphase flow

VOF

THINC

Level set

High-order interface representation

ABSTRACT

In this paper, we propose a simple and accurate numerical method for capturing moving interfaces on fixed Eulerian grids by coupling the Tangent of Hyperbola Interface Capturing (THINC) method and Level Set (LS) method. The innovative and practically-significant aspects of the proposed method, so-called THINC/LS method, lie in (1) representing the interface with polynomial of high-order (arbitrary order in principle) rather than the plane representation commonly used in the Piecewise Linear Interface Calculation (PLIC) Volume of fluid (VOF) methods, (2) conserving rigorously the mass of the transported VOF field, (3) being able to resolving fine interface structures under mesh resolution, and (4) providing a straightforward and easy-to-code algorithm for 3D implementation. We verified the proposed scheme with the widely used benchmark tests. Numerical results show that this new method can obtain high-order accuracy for interface reconstruction and can produce more accurate results in capturing moving interfaces compared to the classical VOF methods.

© 2018 Elsevier Inc. All rights reserved.

1. Introduction

Moving interface phenomena, such as multiphase flows, moving fronts with chemical reactions or phase change, are widely found in nature and engineering fields. Numerical modeling and computation of moving interfaces have been intensively studied in the past decades with large amount of effort devoted, but still remain a challenging and unresolved topic. Numerous methods have been proposed to identify and compute moving interfaces by using either marking points [1–5] or identification functions. The latter approach solves the identification function for moving interface on fixed Eulerian grids, and is thus efficient and robust for interfaces which undergo even topological changes, like merging and breaking. The most popularly used identification functions are the volume fraction or volume of fluid (VOF) function [6–11,13] and the level set (LS) function [14–16].

The idea behind the VOF methods is to use the volume fraction (VOF function) of the target fluid in each grid cell and identify the interface by the cells where the volume fraction values fall between 0 and 1. The VOF methods update the VOF function by solving the advection equation with the finite volume scheme, which ensures the mass conservation of the VOF function. A class of schemes have been devised so far to explicitly reconstruct the interface with straight or curved line segments in 2D (in 3D cases the surface polygons are limited to the planer geometry in nearly all schemes) from the given volume fraction values. For example, in the original VOF scheme [6] and the earlier Simple Line Interface Calculation

* Corresponding authors.

E-mail addresses: nuaadw@126.com (Y. Wei), xiao.faa@m.titech.ac.jp (F. Xiao).

(SLIC) scheme [17], the interface is represented by a straight line parallel to the grid axis, which results in at best the first-order accuracy. As a more accurate algorithm, Piecewise Linear Interface Calculation (PLIC) [7,8,11,13,18–22] algorithm was proposed where the orientation (the normal direction) of the cell-wise interface segment is taken into account. To improve the accuracy of the orientation computation, the Least squares Volume-of-fluid Interface Reconstruction Algorithm (LVIRA) [12] and the Efficient Least squares Volume-of-fluid Interface Reconstruction Algorithm (ELVIRA) [13] were developed. An efficient formulation to determine the interface from volume fraction values was presented in [11] for Cartesian grid. So far, different variants of VOF method with PLIC reconstructions have been explored and successfully implemented in numerical models for practical use, such as the Gerris and Basilisk open source codes [23–26] which facilitate multi-scale simulations with adaptive mesh refinement technique.

Rather than the linear approximation in the SLIC and PLIC schemes, efforts to fit the interface with higher order surface representations have been also made in the past years, for instance, the Least-Square Fit (LSF) [19,20] algorithm, the Spline Interface Reconstruction (SIR) [27] algorithm, piecewise-planar interface reconstruction with Cubic-Bézier fit (CBIR) [28] algorithm, and the Quadratic Spline based Interface (QUASI) [29] algorithm. Although these algorithms show some improvements compared to the PLIC method, to our knowledge, all these algorithms are limited to 2D and none of them can realize interface reconstruction beyond second-order. The above-mentioned VOF methods, known as the geometrical type VOF methods, explicitly use geometrical elements, such as line or surface segments cutting through grid cells, to formulate the interface, thus involve complicated algorithmic procedure which makes the following tasks formidably difficult in 3D case: (1) approximating interface with higher-order surface representation, and (2) implementing for three-dimensional cases.

In contrast to the geometrical VOF schemes mentioned above, the Tangent of Hyperbola Interface Capturing (THINC) [30–35] schemes use an implicit approach to embed the geometrical information in the THINC function which is a regularized hyperbolic tangent function to mimic the VOF field. Given the VOF values in all cells, the THINC function can be cell-wisely constructed as a continuous function in each cell, which is then used to compute the numerical fluxes needed in the finite volume formulation to update the VOF function. A THINC scheme can be simply used as a normal finite volume advection scheme to capture moving interfaces in practice. Some variants of THINC method have been developed, including multi-dimensional THINC reconstructions in Cartesian grid [32] and unstructured grids in both 2 and 3D [33,35] with linear and quadratic surface representations. It is revealed that the schemes with multi-dimensional THINC reconstructions are able to give numerical results competitive to existing PLIC type VOF methods and show great practical significance and potential for further extension.

It should be noted that being a discontinuous (or steep-gradient) field function, the VOF function cannot provide accurate and reliable geometrical information for the interface, while the level set method shows great superiority as discussed next.

Being another methodology for interface capturing, the level set methods have gained increasing popularity in the past decades. Unlike the VOF methods, the level set methods employ the smooth signed distance function to represent the interface. The signed distance function, also known as the LS function, is a set of smooth hypersurfaces or manifolds of higher dimensions, which can be solved by high-order schemes for advection equation. Furthermore, given the signed distance function, the level set methods can provide adequate geometrical information of the interface, such as normal vector and curvature, needed in formulating physical models for interface and reconstructing the interface as explored later in this paper. During the evolution of moving interface the values of LS function in the computational domain may not remain always the distance to the interface, so the reinitialization process [16,36] is needed in practice to modify the values of the level set function on grid cells. In principle, as the numerical conservativeness is not guaranteed in either advection (evolution) or reinitialization computation, mass loss or gain may occur to the fluid of interest, which is the fatal drawback of level set method. Several efforts have been made to remedy the conservation error for level set methods, for example, the coupled Particle/Level Set (Particle LS) [37] and the Coupled Level Set/VOF (CLSVOF) methods [38–41]. Unfortunately, despite algorithmic complexity, none of these methods realizes high-order interface representation.

In this paper, we present a novel interface capturing scheme, so-called THINC/LS (THINC scheme coupled with the LS method), which possesses the following major new properties in comparison with other state-of-the-art methods.

- Instead of the plane and quadratic surface representations used in the existing geometrical VOF methods, higher-order (arbitrary order in principle) surface polynomials can be determined from the level set function, and then used to build the THINC or VOF function.
- The interface in the level set field, i.e. the 0-level surface in the level set function, is synchronized with the interface identified from the THINC reconstruction with the mass conservation constraint. Thus, the numerical conservativeness is ensured.
- The function pair of THINC and level set are updated and coupled simultaneously through simple and straightforward solution procedure without the algorithmic complexity in the geometrical VOF schemes.
- With a surface polynomial higher than 2nd order, the THINC/LS method is able to capture fine interface structures which are under the grid resolution, for example, the thin film with a thickness less than the size of mesh cell as shown later in numerical tests.

We have extensively examined the THINC/LS method regarding interface reconstruction and transport of moving interfaces. The numerical results verify the superiority of the method in comparison with other existing methods. It should be

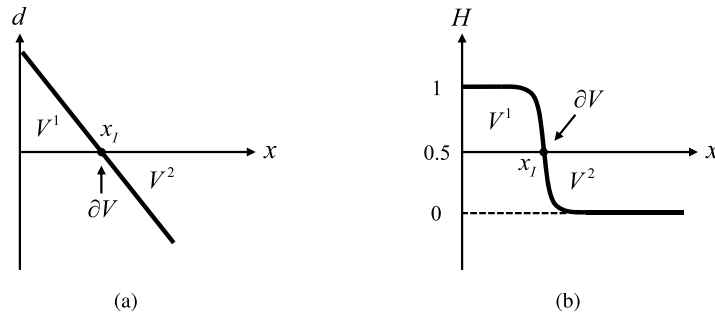


Fig. 1. Schematic diagram of the interface indicator functions in one-dimension: (a) The LS function; (b) The THINC function.

noted that unlike CLSVOF [38,42] in which VOF is used to ameliorate the LS results, THINC/LS method realizes simultaneously rigorous numerical conservativeness for LS and high-order surface representation for VOF solutions.

This paper is organized as follows. The details of the proposed THINC/LS scheme are described in section 2. Interface reconstruction tests are presented in section 3. Extensive numerical verifications with widely used benchmark tests are presented in section 4. We end this paper with some conclusion remarks in section 5.

2. Numerical method

2.1. The interface indication functions

Although the idea presented in this work can be in principle extended to any structured and unstructured mesh, we limit our presentation to Cartesian grid. The computational domain is divided by mesh cell $\Omega_{ijk} = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] \times [y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}] \times [z_{k-\frac{1}{2}}, z_{k+\frac{1}{2}}]$, where i , j and k denote the cell index in x , y and z directions respectively. We also denote the six surface segments of Ω_{ijk} by

$$\begin{aligned} S_{i\pm\frac{1}{2}jk} &: x_{i\pm\frac{1}{2}} \cap [y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}] \times [z_{k-\frac{1}{2}}, z_{k+\frac{1}{2}}], \\ S_{ij\pm\frac{1}{2}k} &: y_{j\pm\frac{1}{2}} \cap [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] \times [z_{k-\frac{1}{2}}, z_{k+\frac{1}{2}}], \\ S_{ijk\pm\frac{1}{2}} &: z_{k\pm\frac{1}{2}} \cap [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] \times [y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}]. \end{aligned} \quad (1)$$

In the present work, we use a mesh of uniform spacing ($\Delta x = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}} = \Delta y = y_{j+\frac{1}{2}} - y_{j-\frac{1}{2}} = \Delta z = z_{k+\frac{1}{2}} - z_{k-\frac{1}{2}}$) for sake of simplicity. The center of each cell locates at (x_i, y_j, z_k) .

Considering an interface ∂V lying between two regions, V^1 and V^2 filled with different fluids in \mathbb{R}^2 or \mathbb{R}^3 , we introduce two interface indicator functions, i.e. the level set (LS) function $\phi(\mathbf{x}, t)$ and the THINC function $H(\mathbf{x}, t)$, as illustrated in Fig. 1 for the one-dimensional case.

- LS function (Fig. 1(a)):

The distance from a point $\mathbf{x} = (x, y, z)$ to the interface ∂V is defined by

$$\text{dist}(\mathbf{x}, \partial V) = \inf_{\mathbf{x}_I \in \partial V} \|\mathbf{x} - \mathbf{x}_I\|, \quad (2)$$

where \mathbf{x}_I represents any point on the interface, and is called the interface point hereafter.

The LS function is a signed distance function to the interface, given by

$$\phi(\mathbf{x}, t) = \begin{cases} \text{dist}(\mathbf{x}, \partial V) & \text{if } \mathbf{x} \in V^1 \\ 0 & \text{if } \mathbf{x} \in \partial V, \\ -\text{dist}(\mathbf{x}, \partial V) & \text{if } \mathbf{x} \in V^2 \end{cases} \quad (3)$$

where ∂V denotes the interface, V^1 the region for fluid 1 and V^2 the region for fluid 2. The LS function is differentiable and facilitates accurate computations for the geometrical properties of the interface.

- THINC function (Fig. 1(b)):

The THINC function is designed to represent a jump-like solution using the hyperbolic tangent function,

$$H(\mathbf{x}, t) = \frac{1}{2} (1 + \tanh(\beta \phi(\mathbf{x}, t))), \quad (4)$$

where β is a parameter to control the thickness of the interface, i.e. the width of the jump in the hyperbolic tangent function. See a more detailed discussion on β in [31]. There is an arbitrariness in choosing β , and we use $\beta = 3.0/\Delta_{\min}$

with $\Delta_{\min} = \min(\Delta x, \Delta y, \Delta z)$ in this work unless otherwise specified. As shown in the numerical tests, such a value of β keeps the thickness of the interface within 2 or 3 mesh cells. It should be also noted that a large β is beneficial to resolve fine interface structures, but might affect in some cases the convergence of the Newton iterative method to solve the nonlinear algebraic equation shown later.

Being essentially a continuous function, the THINC function is able to approximate the Heaviside step function by increasing the value of β . The cell-average of THINC function

$$\bar{H}_{ijk}(t) = \frac{1}{|\Omega_{ijk}|} \int_{\Omega_{ijk}} H(\mathbf{x}, t) dxdydz \quad (5)$$

stands for the volume fraction of a specified fluid (fluid 1 in Fig. 1(b)) in the VOF method. Thus, the rigorous conservation of mass/volume can be automatically guaranteed if the finite volume formulation is applied to solve the THINC function as in the classical VOF schemes.

From the above definitions, the interface can be equivalently identified by either $\phi(\mathbf{x}, t) = 0$ or $H(\mathbf{x}, t) = 0.5$.

The proposed THINC/LS method updates the LS function $\phi(\mathbf{x}, t)$ and the THINC function $H(\mathbf{x}, t)$ simultaneously and synchronizes them to realize both high-order interface representation and numerical conservativeness.

2.2. The evolution equations

In the present work, we assume that the free interface is passively transported by fluid motion. The advection equation of the LS function $\phi(\mathbf{x}, t)$ reads

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = 0, \quad (6)$$

where $\mathbf{u} = (u, v, w)$ is the flow velocity.

The THINC function $H(\mathbf{x}, t)$ follows the same advection equation,

$$\frac{\partial H}{\partial t} + \mathbf{u} \cdot \nabla H = 0. \quad (7)$$

In order to implement the finite volume method, we rewrite (7) into the flux form as

$$\frac{\partial H}{\partial t} + \nabla \cdot (\mathbf{u}H) - H \nabla \cdot \mathbf{u} = 0. \quad (8)$$

It is obvious that (8) reduces to

$$\frac{\partial H}{\partial t} + \nabla \cdot (\mathbf{u}H) = 0 \quad (9)$$

for incompressible flows. Shown latter, the cell-average of THINC function $\bar{H}_{ijk}(t)$, i.e. the volume fraction or VOF function defined in (5), is updated via the finite volume method so as to ensure rigorous numerical conservation.

2.3. The THINC/LS method

As the central part of this paper, we describe the numerical algorithm of the THINC/LS method in this subsection.

We suppose that at time step n ($t = t^n$), we are given with the VOF values \bar{H}_{ijk}^n of the interface cells Ω_{ijk} , where $\epsilon \leq \bar{H}_{ijk}^n \leq 1 - \epsilon$ and ϵ is a small positive and set to 10^{-8} in the present work, as well as the values of LS function, $\phi_{i_l j_l k_l}^n$, at the center of the supporting cells of Ω_{ijk} , $\{\Omega_{i_l j_l k_l} : l = 0, 1, \dots, p\}$. Here we re-index the supporting cells by (i_l, j_l, k_l) with $l = 0, 1, \dots, p$, and p denotes the order of the surface polynomial. Fig. 2 shows the union of the supporting cells for constructing 2nd ($p = 2$) and 4th-order ($p = 4$) surface polynomials.

2.3.1. Retrieve the high-order surface polynomial from LS function

We approximate the cell-wise LS function within cell Ω_{ijk} by

$$\mathcal{P}_{ijk}(x, y, z) = \sum_{s,t,r=0}^p a_{str} X^s Y^t Z^r, \quad (10)$$

where \mathcal{P}_{ijk} is a polynomial of p th order and (X, Y, Z) is the local coordinates for cell Ω_{ijk} , i.e. $(X, Y, Z) = ((x - x_i), (y - y_j), (z - z_k))$. $\mathcal{P}_{ijk}(x, y, z) = 0$, referred to as the surface polynomial in our context, represents the 0-level surface of the level set function cutting through the target cell. The coefficients a_{str} are computed from the supporting cells $\{\Omega_{i_l j_l k_l} : l = 0, 1, \dots, p\}$ using Lagrange interpolation,

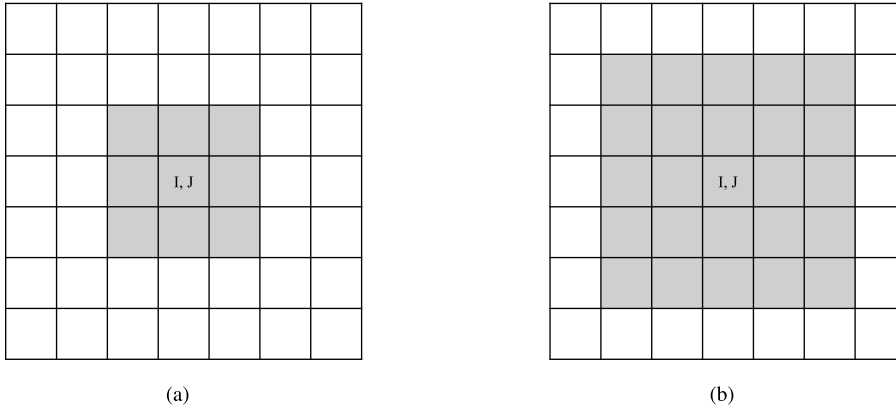


Fig. 2. The 2D supporting cells for constructing surface polynomials of 2nd ($p = 2$) order (a) and 4th ($p = 4$) order (b).

$$\mathcal{P}_{ijk}(x_{i_l}, y_{j_l}, z_{k_l}) = \phi_{i_l j_l k_l}^n, \quad l = 0, 1, \dots, p. \quad (11)$$

From (10) into (11), we get a system of linear equations to find the coefficients a_{str} . The system in matrix–vector form reads

$$\begin{bmatrix} 1 & X_{i_0} & Y_{j_0} & Z_{k_0} & \dots & X_{i_0}^p Y_{j_0}^p Z_{k_0}^p \\ 1 & X_{i_1} & Y_{j_1} & Z_{k_1} & \dots & X_{i_1}^p Y_{j_1}^p Z_{k_1}^p \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ 1 & X_{i_p} & Y_{j_p} & Z_{k_p} & \dots & X_{i_p}^p Y_{j_p}^p Z_{k_p}^p \end{bmatrix} \begin{bmatrix} a_{000} \\ a_{100} \\ \vdots \\ a_{ppp} \end{bmatrix} = \begin{bmatrix} \phi_{i_0 j_0 k_0}^n \\ \phi_{i_1 j_1 k_1}^n \\ \vdots \\ \phi_{i_p j_p k_p}^n \end{bmatrix}. \quad (12)$$

It is sensible to set the stencil of the supporting cells symmetrical with respect to the center of the targeted cell Ω_{ijk} . For convenience (not necessarily), we consider the polynomials of even orders ($p = 2, 4$) in this paper. In order to compare with the classical VOF methods, the linear surface polynomial ($p = 1$) is also evaluated and the coefficients of the first-order terms, i.e. the unit normal of the interface, are determined from the quadratic or second-order surface polynomial ($p = 2$) in our numerical tests.

In our experience, the condition number of the Vandermonde matrix in (12) may be ill-conditioned especially in some 3D tests. In such a case, we extend the stencil of supporting cells in each direction from $p + 1$ to $p + 3$. The resulting over-determined linear system is solved by the least square method.

In practice, (10) can be simplified as

$$\mathcal{P}_{ijk}(x, y, z) = \sum_{\substack{s, t, r=0 \\ s+t+r \leq p}}^p a_{str} X^s Y^t Z^r, \quad (13)$$

which greatly reduces the computational cost.

2.3.2. Enforce mass/volume conservation from the volume fraction constraint

In the solution procedure of the present THINC/LS algorithm shown latter, the LS values ϕ_{ijk}^n are obtained by transporting the LS function with the advection equation, thus the interface defined by $\phi(x, y, z, t^n) = 0$ does not guarantee mass/volume conservation. We use the volume fraction \bar{H}_{ijk}^n as the constraint condition to enforce the conservation.

Given the surface polynomial \mathcal{P}_{ijk} computed by (10), as well as the volume fraction \bar{H}_{ijk}^n , we write the THINC function (4) as,

$$H(x, y, z, t) = \frac{1}{2} \left(1 + \tanh \left(\beta \left(\mathcal{P}_{ijk}(x, y, z) + \phi_{ijk}^\Delta \right) \right) \right), \quad (14)$$

where ϕ_{ijk}^Δ is the correction to the level set function due to the volume fraction constraint, which is determined so that

$$\frac{1}{|\Omega_{ijk}|} \int_{\Omega_{ijk}} \frac{1}{2} \left(1 + \tanh \left(\beta \left(\mathcal{P}_{ijk}(x, y, z) + \phi_{ijk}^\Delta \right) \right) \right) = \bar{H}_{ijk}^n \quad (15)$$

is satisfied.

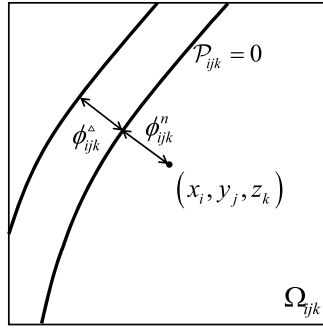


Fig. 3. Schematic diagram of the surface polynomial and the volume fraction constraint.

Following [35], we use Gaussian quadrature to approximate the spatial integration in (15), which yields

$$\sum_{g=1}^G w_g \frac{1}{2} \left(1 + \tanh \left(\beta \left(\mathcal{P}_{ijk}(x, y, z) + \phi_{ijk}^{\Delta} \right) \right) \right) = \bar{H}_{ijk}^n, \quad (16)$$

where w_g is the normalized weights at Gaussian quadrature points satisfying $\sum_{g=1}^G w_g = 1$. The number of quadrature points (G) should be chosen according to the desired numerical accuracy. See [35] for more discussions. In the present work, we use 3 Gaussian points in each dimension. Newton iteration method can be used to solve (16). Using 0 as the first guess, the iteration converges within a few steps. As long as ϕ_{ijk}^{Δ} is obtained, the interface segment in cell Ω_{ijk} can be identified as

$$\mathcal{P}_{ijk}(x, y, z) + \phi_{ijk}^{\Delta} = 0. \quad (17)$$

Remarks. We make use of the LS function to find the cell-wise surface polynomial \mathcal{P}_{ijk} that includes all geometrical information of the interface. The surface polynomial \mathcal{P}_{ijk} can be viewed as a local LS function for the target cell. However, the interface represented by

$$\mathcal{P}_{ijk} = 0 \quad (18)$$

does not necessarily satisfy the mass/volume conservation. As shown in Fig. 3, using the constraint condition (15), we adjust (or synchronize) the LS function by a distance of ϕ_{ijk}^{Δ} in the normal direction of the interface so as to enforce the conservation.

2.3.3. Reinitialization

After the conservative LS function $\phi_{ijk}^{nc} = \phi_{ijk}^n + \phi_{ijk}^{\Delta}$ are obtained for all interface cells, we need to reinitialize ϕ^{nc} . The reinitialization process is equivalent to solving the Eikonal equation,

$$|\nabla \phi| = 1. \quad (19)$$

In the present work, we use the Fast Sweeping Method (FSM) [43] for reinitialization. During the reinitialization, the values ϕ_{ijk}^{nc} are fixed in the interface cells. We identify the interface cell by either $\phi_{ijk}^{nc} \phi_{i'j'k'}^{nc} \leq 0$ with $\phi_{i'j'k'}^{nc}$ standing for the LS value in any neighboring cell of Ω_{ijk} , or $\epsilon_1 \leq \bar{H}_{ijk}^n \leq 1 - \epsilon_2$ where ϵ_1 and ϵ_2 are small positives. Hence, the global LS function after reinitialization (19), ϕ^{nc} , satisfies the mass/volume conservation condition.

2.3.4. Update the level set and THINC functions

Given the mass-conserving LS function ϕ^{nc} at time step n (t^n), we solve the advection evolution (6) to update the LS function to ϕ^{n+1} at step $n+1$ ($t^{n+1} = t^n + \Delta t$). In the present work, we use the fifth-order Hamilton–Jacobi WENO scheme [44] for spatial discretization and the 3rd-order TVD Runge–Kutta scheme [45,46] for time integration. It is noted that ϕ^{n+1} computed from the advection equation does not necessarily satisfy the mass/volume conservation.

The cell-average values of the THINC function, i.e. the volume fractions are updated via the finite volume formulation as follows

$$\frac{d\bar{H}_{ijk}(t)}{dt} = - \left(\frac{1}{\Delta x} \left(\mathcal{F}_{i+\frac{1}{2}jk}^{<x>} - \mathcal{F}_{i-\frac{1}{2}jk}^{<x>} \right) + \frac{1}{\Delta y} \left(\mathcal{F}_{ij+\frac{1}{2}k}^{<y>} - \mathcal{F}_{ij-\frac{1}{2}k}^{<y>} \right) + \frac{1}{\Delta z} \left(\mathcal{F}_{ijk+\frac{1}{2}}^{<z>} - \mathcal{F}_{ijk-\frac{1}{2}}^{<z>} \right) \right), \quad (20)$$

where $\mathcal{F}^{<\alpha>}$, $\alpha = x, y, z$, denote the numerical fluxes in x, y, z directions respectively, which are computed over each cell surface by Gaussian quadrature. We show the numerical formula for $\mathcal{F}_{i+\frac{1}{2}jk}^{<x>}$, the numerical flux in x direction through cell surface segment $S_{i+\frac{1}{2}jk}$, as an example,

$$\mathcal{F}_{i+\frac{1}{2}jk}^{<x>} = \sum_{g=1}^G w_g \left(u(x_{i+\frac{1}{2}}, y_g, z_g) H_{i^{up}jk}(x_{i+\frac{1}{2}}, y_g, z_g) \right), \quad (21)$$

where (y_g, z_g) stands for the Gaussian points on cell surface $S_{i+\frac{1}{2}jk}$. The subscript i^{up} indicates the THINC reconstruction function in the upwinding cell which is determined by

$$i^{up} = \begin{cases} i, & \text{if } u(x_{i+\frac{1}{2}}, y_g, z_g) > 0; \\ i+1, & \text{otherwise.} \end{cases} \quad (22)$$

Given the THINC function (14) and the velocity specified at the Gaussian points, we get the numerical flux (21) immediately. The same applies to numerical fluxes on other surface segments of Ω_{ijk} .

After the numerical fluxes are obtained, the 3rd-order TVD Runge–Kutta scheme is used to update the volume fraction to $n+1$ time step (\bar{H}_{ijk}^{n+1}).

Note that the THINC reconstruction and the computation to update the volume fraction are limited only to the interface cells. For other cells, the solution of the VOF field is trivially either 0 or 1.

2.3.5. Summary of the solution procedure

In order to facilitate the implementation of the present method, we summarize the computational steps to update the function pair from (ϕ^n, \bar{H}^n) to $(\phi^{n+1}, \bar{H}^{n+1})$ as follows.

- Step 1: Calculate the surface polynomial \mathcal{P}_{ijk} in (10) or (13) for interface cells ($\epsilon \leq \bar{H}_{ijk}^n \leq 1 - \epsilon$) with the coefficients computed from (12) or the least square method using the values of LS function ϕ^n in the supporting cells;
- Step 2: Compute the correction to LS function, ϕ_{ijk}^Δ , for interface cells ($\epsilon \leq \bar{H}_{ijk}^n \leq 1 - \epsilon$) to satisfy the mass conservation constraint (15) from the volume fraction \bar{H}_{ijk}^n ;
- Step 3: Find the LS values, $\phi_{ijk}^{nc} = \phi_{ijk}^n + \phi_{ijk}^\Delta$, for interface cells, and reinitialize the LS values by (19) for other cells to synchronize the global LS function from ϕ_{ijk}^n to ϕ_{ijk}^{nc} that fulfill the mass/volume conservation;
- Step 4: Update the LS function from ϕ^{nc} to ϕ^{n+1} by the fifth-order Hamilton–Jacobi WENO scheme [44] and the 3rd-order TVD Runge–Kutta scheme [45,46];
- Step 5: Update the volume fraction by (20) where the numerical fluxes of interface cells ($\epsilon \leq \bar{H}_{ijk}^n \leq 1 - \epsilon$) are computed from the THINC function (14) using Gaussian quadrature formula (21). The 3rd-order Runge–Kutta scheme is used to predict the volume fraction \bar{H}^{n+1} at new time level;
- Step 6: Go back to step 1 to repeat the computation for next time level.

It is noted that the above solution procedure is straightforward and easy to implement in 3D and unstructured grids.

3. Interface reconstruction tests

We now assess the interface reconstruction accuracy and convergence rates of the proposed THINC/LS method for specific geometries. In order to quantify the numerical accuracy and convergence rates, we identify the reconstruction error E_r as the difference between the area (or volume in 3D) encompassed by the numerical reconstructed interface and the exact interface [10]

$$E_r = \sum_{ijk} \iiint_{\Omega_{ijk}} |H(x, y, z) - H^0(x, y, z)| dx dy dz, \quad (23)$$

where $H(x, y, z)$ and $H^0(x, y, z)$ denote the numerical and exact solutions respectively to the VOF function.

Given the interface function (17) as well as the THINC reconstruction function, we have two approaches to numerically evaluate the value of VOF field \bar{H}_{ijk} for each cell. The first approach is using dense sample points, e.g. 100×100 points, within each cell to sample the value of 0 or 1 according to the location of the sample point with respect to the reconstructed interface (17). This method treats the interface as a one-cell sharp interface. The second approach simply calculates Gaussian quadrature of the THINC function for each cell. Our numerical experiments reveal that the computed VOF values of the two approaches are very close. We thus adopt the Gaussian quadrature of the THINC function to evaluate the numerical errors in interface reconstruction tests as well as the advection tests shown later.

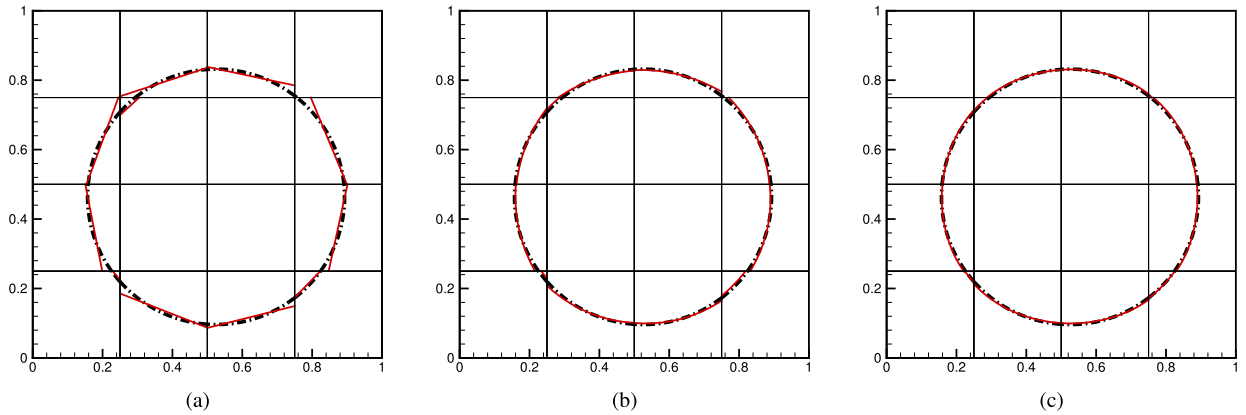


Fig. 4. Reconstructed circle curves using P1 (a), P2 (b) and P4 (c) surface polynomials on a 4×4 mesh. Black dash dot line is the exact solution and red solid line is the numerical solution that indicates the cellwise segment $\mathcal{P}_{ijk} + \phi_{ijk}^{\Delta} = 0$ and is generated from the LS values at the 5×5 uniformly distributed sample points within each mesh cell. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

Table 1

L_1 -norm errors and convergence rates for different methods to reconstruct a circle.

Methods	10^2	Order	20^2	Order	40^2	Order	80^2	Order	160^2	Order	320^2
Youngs [27]	2.09×10^{-3}	1.79	6.06×10^{-4}	1.18	2.67×10^{-4}	1.31	1.08×10^{-4}	1.03	5.29×10^{-5}	–	–
CD [27]	2.34×10^{-3}	2.06	5.61×10^{-4}	1.89	1.51×10^{-4}	1.88	4.10×10^{-5}	1.81	1.17×10^{-5}	–	–
Puckett [27]	3.36×10^{-3}	2.27	6.96×10^{-4}	2.05	1.68×10^{-4}	1.94	4.38×10^{-5}	1.95	1.13×10^{-5}	–	–
ELVIRA [27]	2.94×10^{-3}	2.24	6.22×10^{-4}	1.98	1.58×10^{-4}	1.93	4.15×10^{-5}	2.05	1.00×10^{-5}	–	–
SIR [27]	1.94×10^{-3}	2.03	4.76×10^{-4}	1.88	1.29×10^{-4}	2.15	2.91×10^{-5}	1.89	7.86×10^{-6}	–	–
THINC/LS (P1)	2.25×10^{-3}	2.00	5.64×10^{-4}	2.00	1.41×10^{-4}	2.02	3.48×10^{-5}	1.98	8.81×10^{-6}	2.00	2.21×10^{-6}
THINC/LS (P2)	5.27×10^{-4}	3.01	6.56×10^{-5}	2.99	8.24×10^{-6}	3.01	1.02×10^{-6}	2.99	1.28×10^{-7}	3.00	1.60×10^{-8}
THINC/LS (P4)	6.26×10^{-5}	5.09	1.84×10^{-6}	4.99	5.79×10^{-8}	5.00	1.81×10^{-9}	5.00	5.66×10^{-11}	5.00	1.77×10^{-12}

3.1. Circle reconstruction

As described in [10], a circle with the radius of 0.368 is centred at (0.525, 0.464) in a unit domain. We use the first-order ($p = 1$, denoted as P1 hereafter), second-order ($p = 2$, P2) and fourth-order ($p = 4$, P4) surface polynomials to represent the interface. The reconstructed curves are illustrated in Fig. 4. It is found that the linear surface polynomial (P1) reconstruction looks apparently different from the exact solution. High-order surface polynomials (P2 and P4) give more accurate results than the linear surface polynomial.

For a detailed comparison of different order polynomials, we list the numerical errors and convergence rates in Table 1 and compare them with some classical geometrical VOF methods. As shown in Table 1, the numerical errors and convergence rates of the linear surface polynomial (P1) in THINC/LS method are close to the Piecewise Linear Interface Calculation (PLIC) VOF methods. High-order surface polynomials (P2 and P4) fit the curved surface better than the reconstructions with linear (first-order) polynomials as those used in the PLIC VOF methods. We observe the high-order convergence rates as expected, which implies that the THINC/LS method can achieve high order accuracy for interface reconstruction by using high-order surface polynomials.

Moreover, using higher-order surface polynomials, the presented THINC/LS method is able to maintain high-order convergence rate for curvature. To illustrate this, we computed the curvature test presented in [47]. We use the exact LS function and the corrected LS function from the THINC/LS method to calculate the curvature at cell center with a central differencing. A bilinear interpolation is then used to calculate the curvature on the interface. We compare the error norms of the curvature estimated along the circular interface using P1, P2 and P4 polynomial representations. As shown in Fig. 5, we observe a second-order convergence for the P4 and exact LS function. The convergence for P2 degrades from the second-order to a lower order even for finer grids. The P1, which is equivalent to the PLIC reconstruction, cannot reach converged numerical results for curvature.

3.2. Sphere reconstruction

For the three-dimensional case, we conduct the reconstruction test presented in [20]. A sphere is centred at (0.5, 0.5, 0.5) in a unit cube with radius 0.325. We use P1, P2 and P4 surface polynomials to reconstruct the surface segments for the interface cells as shown in Fig. 6. It is observed that the P1 surface polynomial produces less accurate result than the P2 and P4 surface polynomials.

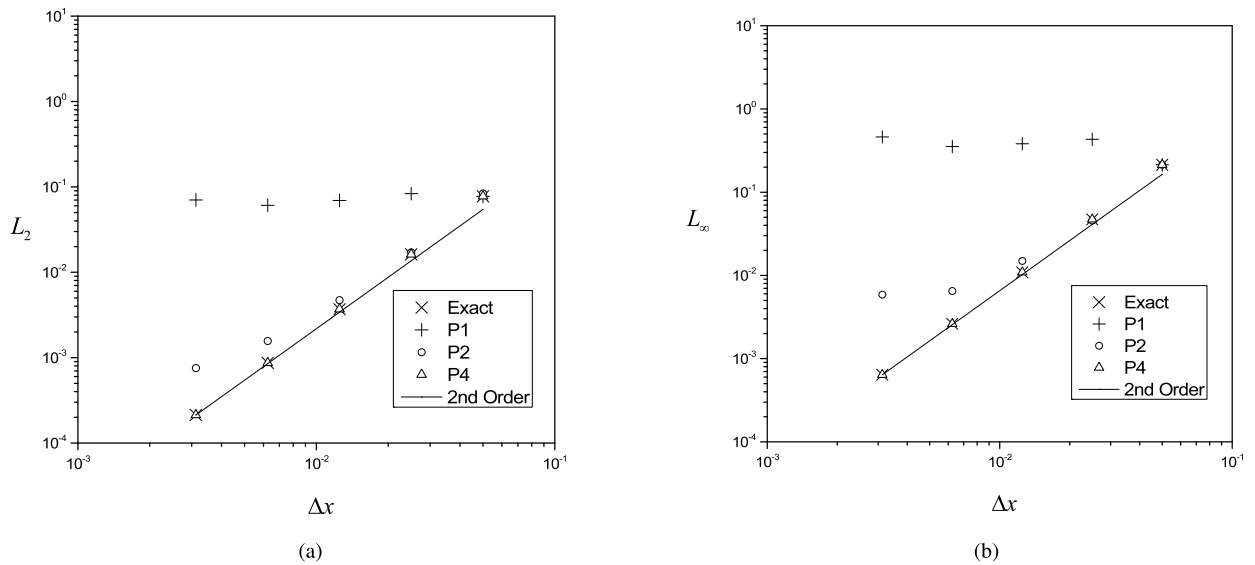


Fig. 5. L_2 (a) and L_∞ (b) error norms for curvature estimated along a circular interface using P1, P2, P4 representations and the exact LS function.

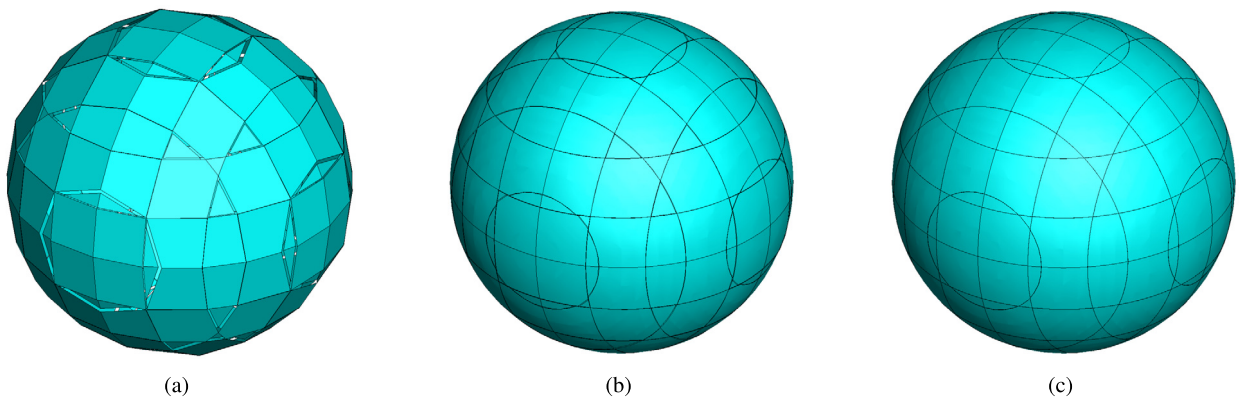


Fig. 6. Reconstructed sphere surfaces using P1 (a), P2 (b) and P4 (c) surface polynomials on a $10 \times 10 \times 10$ mesh. The curve lines indicate the edges of the reconstructed surface segments cut by the cell surfaces. The piecewise surface segment in each cell is defined by (17) which is then used to synchronize the zero-level of the LS function.

Table 2

L_1 -norm errors and convergence rates for different methods to reconstruct a sphere.

Methods	10^2	Order	20^2	Order	40^2	Order	80^2	Order	160^2	Order	320^2
Youngs [28]	1.89×10^{-3}	1.84	5.28×10^{-4}	1.45	1.93×10^{-4}	1.17	8.60×10^{-5}	1.06	4.12×10^{-5}	1.02	2.03×10^{-5}
LSF [20]	1.92×10^{-3}	2.01	4.77×10^{-4}	2.00	1.19×10^{-4}	2.00	2.98×10^{-5}	2.00	7.46×10^{-6}	–	–
LLCIR [28]	2.14×10^{-3}	2.03	5.23×10^{-4}	1.89	1.41×10^{-4}	1.61	4.61×10^{-5}	1.32	1.85×10^{-5}	1.14	8.40×10^{-6}
ELCIR [28]	2.23×10^{-3}	1.99	5.62×10^{-4}	1.98	1.42×10^{-4}	1.88	3.86×10^{-5}	1.69	1.20×10^{-5}	1.42	4.47×10^{-6}
CLCIR [28]	2.38×10^{-3}	2.11	5.50×10^{-4}	2.08	1.30×10^{-4}	2.01	3.23×10^{-5}	2.01	8.00×10^{-6}	2.00	2.00×10^{-6}
CBIR [28]	2.43×10^{-3}	2.11	5.64×10^{-4}	2.12	1.30×10^{-4}	2.03	3.18×10^{-5}	2.02	7.82×10^{-6}	2.00	1.95×10^{-6}
THINC/LS (P1)	2.01×10^{-3}	1.98	5.09×10^{-4}	2.00	1.27×10^{-4}	2.01	3.16×10^{-5}	2.00	7.91×10^{-6}	2.00	1.98×10^{-6}
THINC/LS (P2)	4.77×10^{-4}	3.00	5.96×10^{-5}	2.98	7.53×10^{-6}	3.00	9.44×10^{-7}	3.00	1.18×10^{-7}	3.00	1.47×10^{-8}
THINC/LS (P4)	7.88×10^{-5}	5.02	2.43×10^{-6}	4.98	7.69×10^{-8}	5.00	2.41×10^{-9}	5.00	7.54×10^{-11}	5.00	2.36×10^{-12}

To compare the THINC/LS method quantitatively against other geometrical VOF methods, we list the reconstruction errors and convergence rates on different mesh resolutions in Table 2. It is observed that the P1 surface polynomial is quite similar to the geometrical VOF methods with PLIC reconstructions which employ linear surface representation. High-order surface polynomials (P2 and P4) produce the expected accurate results. It is noted that unlike the geometrical VOF methods which are of significant algorithmic complexity, the interface reconstruction procedure in THINC/LS method is quite simple and straightforwardly extendable to even higher-order in both 2 and 3D.

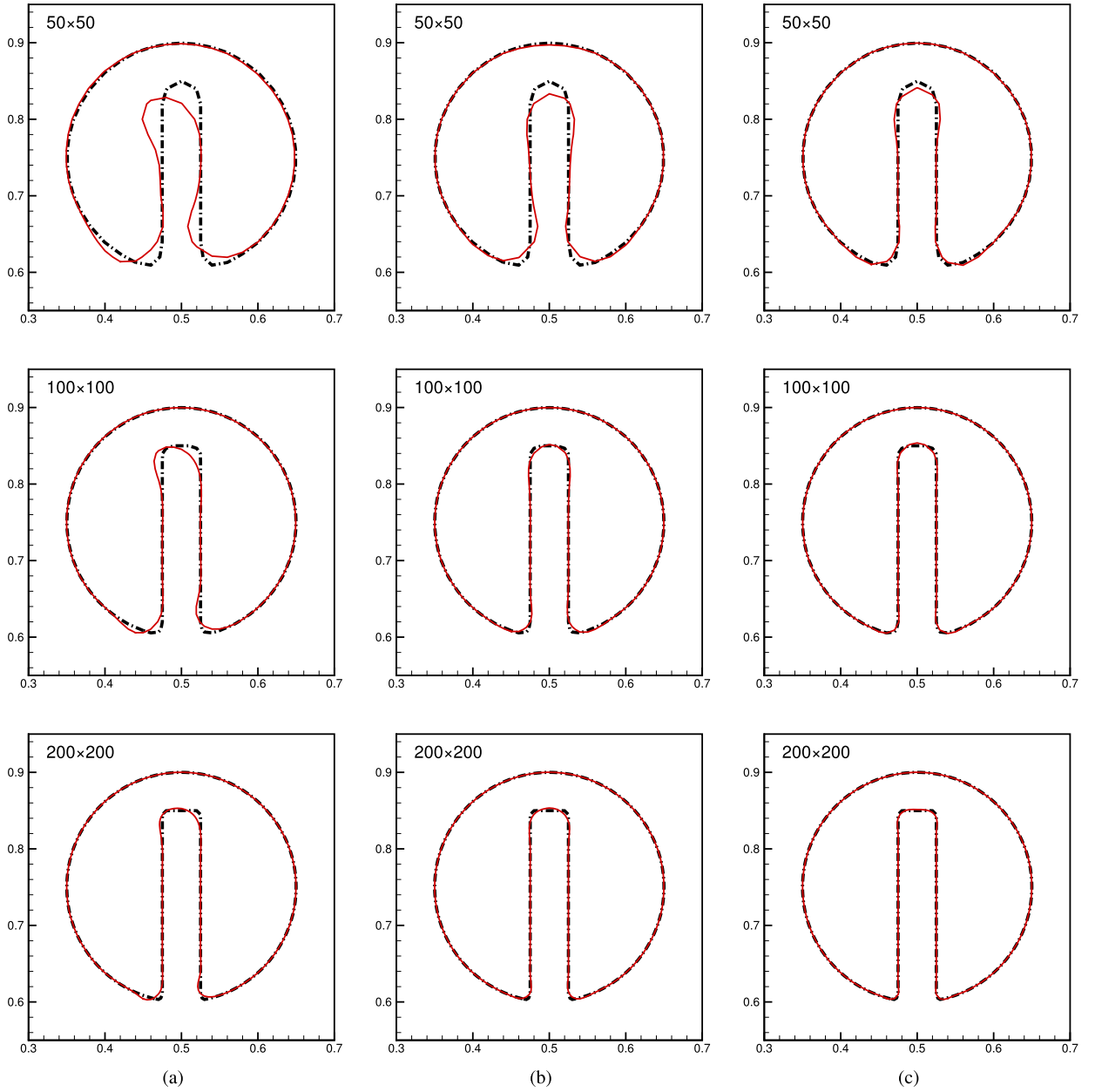


Fig. 7. Numerical results of the Zalesak rotation test after one revolution using P1 (a), P2 (b) and P4 (c) surface polynomials on 50×50 , 100×100 , 200×200 meshes. Black dash dot line is the exact solution and red solid line is the numerical solution. Both lines are the 0.5-contours of the VOF fields.

4. Advection tests

In this section, we evaluated the proposed THINC/LS method as an interface-capturing scheme with some widely used advection benchmark tests. To quantify the accuracy of the numerical results, we define the L_1 error norm for VOF field as

$$E(L_1) = \sum_{ijk} |\bar{H}_{ijk} - \bar{H}_{ijk}^0| |\Omega_{ijk}|, \quad (24)$$

where \bar{H}_{ijk} and \bar{H}_{ijk}^0 stand for the numerical and exact VOF values respectively.

Another widely used error measure for evaluating the VOF methods is the relative error defined by

$$E_r = \frac{\sum_{ijk} |H_{ijk} - H_{ijk}^0|}{\sum_{ijk} H_{ijk}^0}. \quad (25)$$

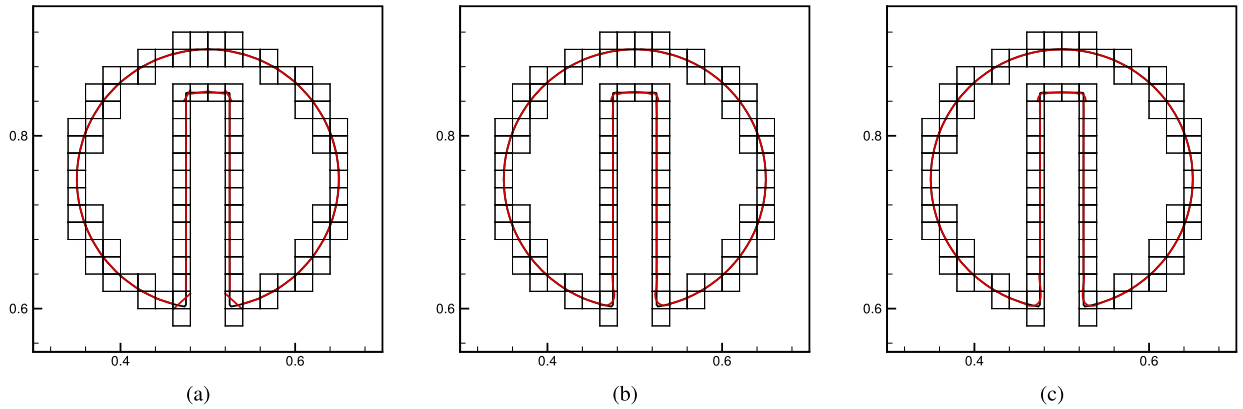


Fig. 8. Reconstructed interfaces in Zalesak solid rotation test using P1 (a), P2 (b) and P4 (c) surface polynomials on a 50×50 mesh at $t = 0$. Black solid lines indicate the exact interfaces and red solid lines indicate the reconstructed interfaces by using polynomials of different orders. The plotted interfaces are generated from the LS values sampled at the 5×5 uniformly distributed sample points within each mesh cell.

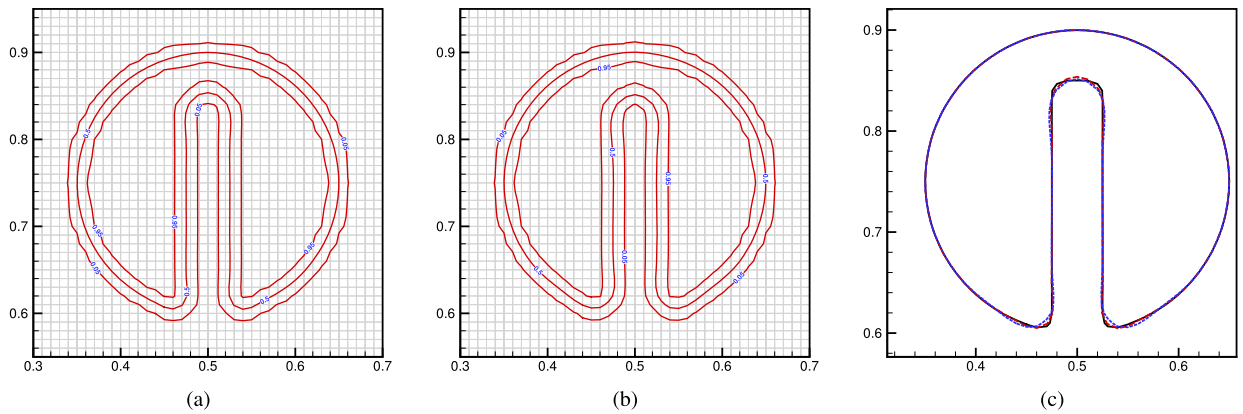


Fig. 9. The numerical results of Zalesak rotation test computed by the THINC/LS scheme using P4 surface polynomial reconstruction on a 100×100 mesh. (a) The enlarged view of 0.05, 0.5, 0.95 contours of the VOF field after one revolution. (b) Same as (a) but after ten revolutions. (c) The 0.5-contours of the solutions after one revolution (red dashed line) and ten revolutions (blue dotted line). The black solid line stands for the exact solution.

4.1. Solid body rotation test

Zalesak's slotted disk test has been widely used to evaluate interface capturing schemes. As described in [48], a slotted circle centred at $(0.5, 0.75)$ with radius 0.15 is rotated in a unit domain. The slot is defined by $(|x - 0.5| \leq 0.025$ and $y \leq 0.85)$ and the velocity field is given by $(0.5 - y, x - 0.5)$. After one revolution period, the slotted circle returns back to its initial state. The CFL number in our computation is about 0.25.

Numerical results of THINC/LS method with different orders of the surface polynomials are illustrated in Fig. 7. As mentioned in [27], the interface reconstruction errors are mainly concentrated at the regions of large interface curvatures, such as the corners of the slot. In most of the VOF methods, as well as the THINC/LS method with linear surface representation, the slot of the circle is found to be distorted after one revolution. However, we do not observe significant distortions of the slot in the results of THINC/LS method with high-order surface polynomials. The same observation is shared by MTHINC scheme [32] which uses a quadratic surface representation. In order to illustrate the difference, we plot the reconstructed interfaces using polynomials of different orders on a 50×50 mesh in Fig. 8, which shows that the deviation of the linear polynomial reconstruction around the sharp corners can be effectively improved by using higher order surface representations. With high-order interface reconstruction, the slotted disk can be accurately resolved even after ten revolutions as shown in Fig. 9 for the case of P4. It reveals that reconstructions with high-order surface polynomials are crucial to get the geometrically faithful solution to the transported interface.

We give the numerical errors after one revolution and compare it with other THINC schemes in Table 3, which shows that the THINC/LS method with high-order surface polynomials produces superior results than other existing schemes.

In order to evaluate the computational cost, we show the elapse time of the THINC/LS method in comparison with the LS method in Table 4. The THINC/LS method takes nearly twice computational time compared to the narrow band LS method, but still within the acceptable range.

Table 3
Numerical errors (E_r) and convergence rates of Zalesak rotation test after one revolution period.

Methods	50^2	Order	100^2	Order	200^2
MTHINC [32]	2.93×10^{-2}	0.86	1.61×10^{-2}	1.03	7.91×10^{-3}
UMTHINC [35]	8.12×10^{-2}	1.63	2.61×10^{-2}	0.97	1.33×10^{-2}
THINC/QQ [35]	8.96×10^{-2}	1.47	3.22×10^{-2}	0.95	1.67×10^{-2}
THINC/LS (P1)	1.23×10^{-1}	1.87	3.37×10^{-2}	1.50	1.19×10^{-2}
THINC/LS (P2)	6.40×10^{-2}	2.12	1.47×10^{-2}	1.06	7.04×10^{-3}
THINC/LS (P4)	3.64×10^{-2}	1.87	9.98×10^{-3}	1.32	3.99×10^{-3}

Table 4
Computation costs of different methods in Zalesak rotation test for one revolution.

Methods	50^2	100^2	200^2
LS (Narrow band)	2.95 s	8.21 s	45.62 s
THINC/LS (P1)	4.51 s	13.90 s	79.74 s
THINC/LS (P2)	4.63 s	15.72 s	84.43 s
THINC/LS (P4)	5.39 s	20.78 s	102.79 s

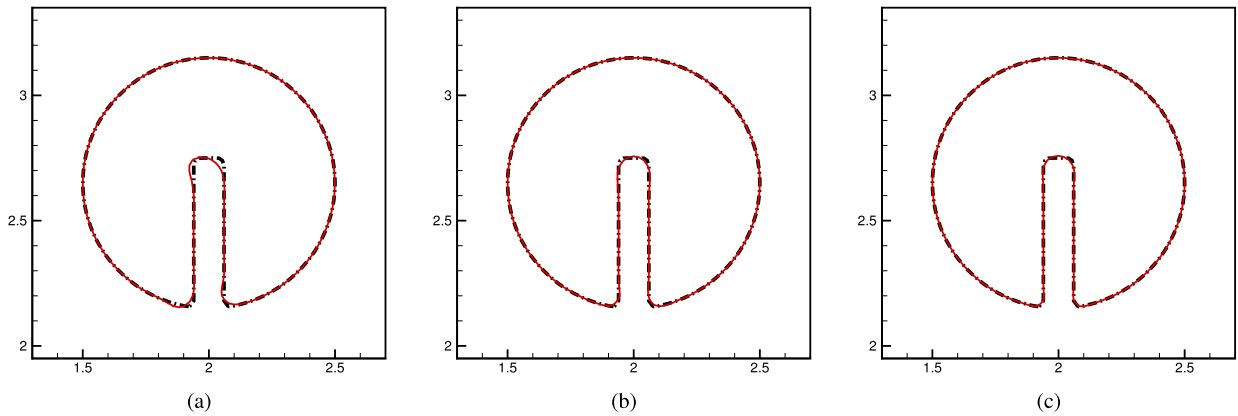


Fig. 10. Numerical results of the Rudman rotation test after one revolution using P1 (a), P2 (b) and P4 (c) surface polynomials on a 200×200 mesh. Black dash dot line is the exact solution and red solid line is the numerical solution. Both lines are the 0.5-contours of the VOF fields.

We also conducted the solid rotation benchmark test proposed by Rudman. As described in [9], a slotted circle with radius 0.5 is centred at (2.0, 2.65) on a $[0, 4] \times [0, 4]$ domain. The slot is defined by $(|x - 2.0| \leq 0.06 \text{ and } y \leq 2.75)$ and the velocity field is given by $(2.0 - y, x - 2.0)$. The computations were conducted on a 200×200 mesh and the maximum CFL number is about 0.25. Fig. 10 shows the numerical results of the THINC/LS method with different order surface polynomials. As discussed before, the linear surface polynomial produces significant distortions near the slot corner. The numerical solutions of high-order surface polynomials in general coincide well with the exact solution. We compute the numerical errors of the THINC/LS method with other VOF methods. Shown in Table 5, the results of THINC/LS method are among the most accurate ones.

4.2. Vortex deformation transport test

We further assessed the THINC/LS method with the moving interface advected by a deformational velocity field. As described in [10], a circle with radius 0.15 is initially centred at (0.5, 0.75) in a unit domain. A time-dependent velocity field is given by the stream function

$$\Psi(x, y, t) = \frac{1}{\pi} \sin^2(\pi x) \sin^2(\pi y) \cos\left(\frac{\pi t}{T}\right). \quad (26)$$

Advected by velocity field (26), the initial circle is firstly distorted and stretched into a spiral shape with a thin tail up to $t = T/2$, and then transported at a reverse velocity field after $T/2$ until restoring its initial shape at T . For a large T , the width of the tail can be thinner than the size of mesh cell, and thus not resolvable by the Eulerian fixed grid at $T/2$. It is well known that the conventional VOF methods tend to generate small droplets or “flotsams” at $t = T/2$ when the tail of the spiral-like VOF field is stretched into a thin film under grid resolution. In this case, the interface losses its geometrical information, and it is hard to restore the initial state at $t = T$.

Table 5
Numerical errors (E_r) of the Rudman rotation test.

Methods	Error
Hirt-Nichols [6]	9.62×10^{-2}
SLIC [17]	8.38×10^{-2}
FCT-VOF [9]	3.29×10^{-2}
Youngs [9]	1.09×10^{-2}
Stream/Youngs [49]	1.07×10^{-2}
Stream/Puckett [49]	1.00×10^{-2}
DDR/Youngs [50]	1.56×10^{-2}
DDR/Puckett [50]	1.50×10^{-2}
EMFPA/Youngs [27]	1.06×10^{-2}
EMFPA/Puckett [27]	9.73×10^{-3}
EMFPA/SIR [27]	8.74×10^{-3}
ELVIRA [21]	1.00×10^{-2}
GPCA [22]	9.79×10^{-3}
CICSAM [51]	2.02×10^{-2}
THINC [30]	3.52×10^{-2}
THINC/WLIC [52]	1.96×10^{-2}
THINC/SW [31]	1.34×10^{-2}
THINC/QQ [35]	1.42×10^{-2}
Linear fit/Lagrangian [19]	9.42×10^{-3}
Quadratic fit/Lagrangian [19]	5.47×10^{-3}
Quadratic fit+continuity/Lagrangian [19]	4.16×10^{-3}
THINC/LS (P1)	9.93×10^{-3}
THINC/LS (P2)	4.93×10^{-3}
THINC/LS (P4)	3.77×10^{-3}

In this test, $\beta = 6.0/\Delta_{\min}$ was used to get a steeper interface jump and resolve the finer interface structures. We tested the case of $T = 8$ on grids with gradually increased resolutions, 32×32 , 64×64 and 128×128 . The maximum CFL number of all computations are set to 0.1. The numerical results on different grid resolutions at $t = T/2$ or $t = T$ are shown in Fig. 11, where the interface is identified by the 0-contour of the surface polynomials in each cell interpolated by 5×5 sample points at $t = T/2$ and 0.5-contour of the VOF fields at $t = T$. It is observed that THINC/LS method with high-order surface polynomials can restore the initial circular shape with adequate accuracy, and the P4 polynomial gives the best solution quality. Without explicit geometrical reconstruction, the THINC/LS method can retrieve the high-order surface polynomial from the smooth LS function even when the interface is under the grid resolution as shown in Fig. 12. It is noted that the THINC/LS method with a surface polynomial higher than 2nd order is able to identify two interface segments in a single cell, which enables to resolve interface structures under mesh resolution. Unlike other more complicated VOF schemes using multiple linear segments [53] or markers [54] to resolve the thin interface in single cell, the THINC/LS method is much easier to implement in 3D and unstructured grids. We further show the L_1 VOF errors and convergence rates of the THINC/LS method in comparison with other VOF methods in Table 6. It is observed that the THINC/LS method can produce competitive results compared to other VOF or hybrid VOF methods.

We also examined the numerical diffusion of the THINC/LS method. Shown in Fig. 13, the interface thickness and sharpness maintain satisfactorily after one period $t = T$.

4.3. 3D rotation of a slotted-sphere test

For the 3D case, we first conduct the numerical test of a rotating slotted-sphere in a unit domain $[0, 1]^3$ as described in [32]. The slotted-sphere is specified as

$$(x - 0.5)^2 + (y - 0.75)^2 + (z - 0.5)^2 \leq 0.15^2 \wedge (|x - 0.5| \geq 0.025 \vee y \geq 0.725), \quad (27)$$

and the velocity field is given by $(0.5 - y, x - 0.5, 0)$.

We computed this test case on $50 \times 50 \times 50$, $100 \times 100 \times 100$, and $200 \times 200 \times 200$ meshes. The CFL number in all computations is about 0.25. The numerical results of the THINC/LS method with different order of the surface polynomials are shown in Fig. 14. It is observed that the shape of the slotted sphere is kept very well after one revolution with high-order surface polynomials, which proves the geometrically faithfulness of the THINC/LS method to preserve the shape of 3D transported interfaces. We further compare the L_1 errors and the corresponding convergence rates of the THINC/LS method with other THINC schemes in Table 7. It is found that the THINC/LS method can produce superior results than other THINC schemes.

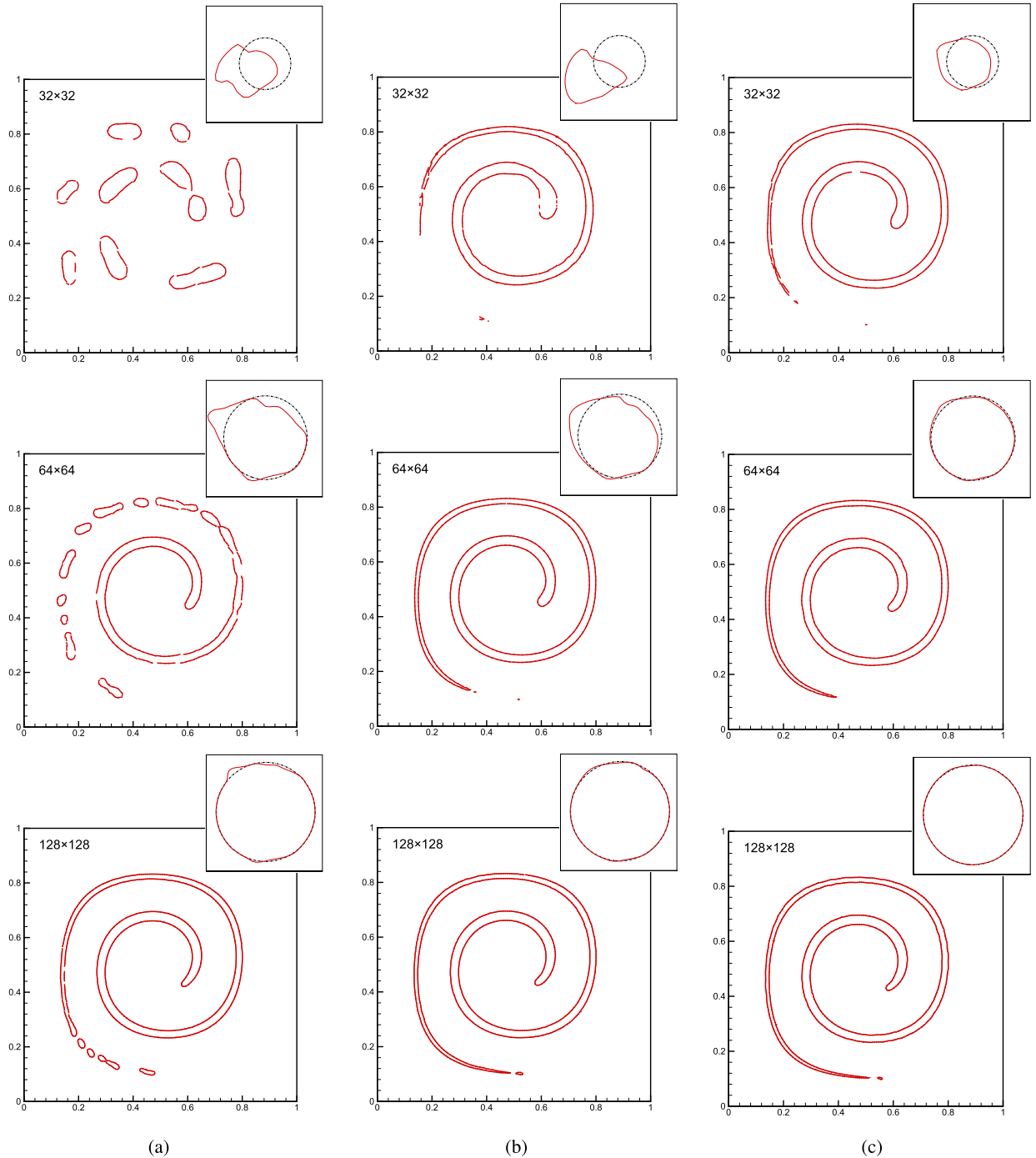


Fig. 11. Numerical results of the Rider-Kothe single vortex test using P1 (a), P2 (b) and P4 (c) surface polynomials on 32×32 , 64×64 , 128×128 meshes. Solid red line at $t = T/2$ is the reconstructed interface and is generated from the LS values sampled at the 5×5 uniformly distributed sample points within each mesh cell. Solid red line and dashed black line at $t = T$ are the numerical results and the exact solution respectively. Both lines at $t = T$ are the 0.5-contours of the VOF fields.

4.4. 3D vortex deformation test

To verify the capability of the THINC/LS scheme to resolve heavily distorted interface in 3D, we study the 3D time-dependent vortex test in a unit domain $[0, 1]^3$ proposed in [56]. A sphere with radius 0.15 is centred at (0.35, 0.35, 0.35) and transported by a time-dependent deformational velocity field:

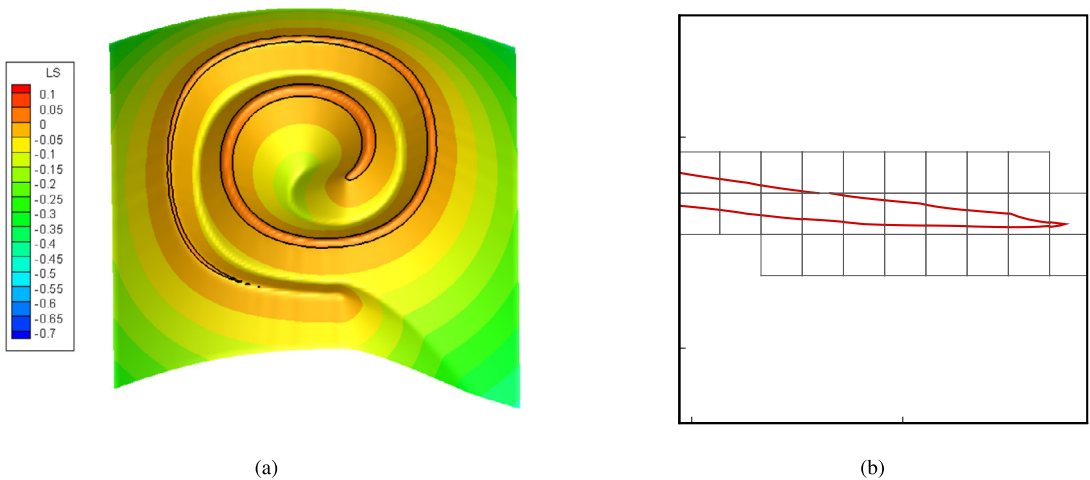


Fig. 12. Numerical results of LS function (a) and reconstructed interface (b) in the Rider-Kothe single vortex test on a 128×128 mesh using P4 surface polynomials at $t = T/2$. It is seen that the spiral tail is stretched into a film thinner than the size of mesh cell that can be still resolved using surface polynomials higher than 2nd order.

Table 6
 L_1 errors and convergence rates of the time-dependent reversing single vortex test.

Methods	32^2	Order	64^2	Order	128^2
Rider-Kothe/Puckett [10]	4.78×10^{-2}	2.78	6.96×10^{-3}	2.27	1.44×10^{-3}
Stream/Puckett [49]	3.72×10^{-2}	2.45	6.79×10^{-3}	2.52	1.18×10^{-3}
Stream/Youngs [49]	3.61×10^{-2}	1.85	1.00×10^{-2}	2.21	2.16×10^{-3}
EMFPA/Puckett [27]	3.77×10^{-2}	2.52	6.58×10^{-3}	2.62	1.07×10^{-3}
DS-MOF [55]	3.45×10^{-2}	1.78	1.00×10^{-2}	3.17	1.11×10^{-3}
THINC/WLIC [52]	4.16×10^{-2}	1.37	1.61×10^{-2}	2.18	3.56×10^{-3}
THINC/SW [31]	3.90×10^{-2}	1.36	1.52×10^{-2}	1.94	3.96×10^{-3}
THINC/QQ [35]	6.70×10^{-2}	2.14	1.52×10^{-2}	2.31	3.06×10^{-3}
Hybrid markers-VOF [54]	2.53×10^{-2}	3.19	2.78×10^{-3}	2.54	4.78×10^{-4}
Markers-VOF [53]	7.41×10^{-3}	1.83	2.12×10^{-3}	2.31	4.27×10^{-4}
DS-CLSVOF [55]	5.45×10^{-2}	2.37	1.05×10^{-2}	2.59	1.74×10^{-3}
DS-CLSMOF [55]	2.92×10^{-2}	2.40	5.51×10^{-3}	2.00	1.37×10^{-3}
THINC/LS (P1)	6.71×10^{-2}	2.13	1.53×10^{-2}	2.75	2.27×10^{-3}
THINC/LS (P2)	1.00×10^{-1}	3.04	1.22×10^{-2}	3.35	1.20×10^{-3}
THINC/LS (P4)	2.85×10^{-2}	3.07	3.39×10^{-3}	2.32	6.79×10^{-4}

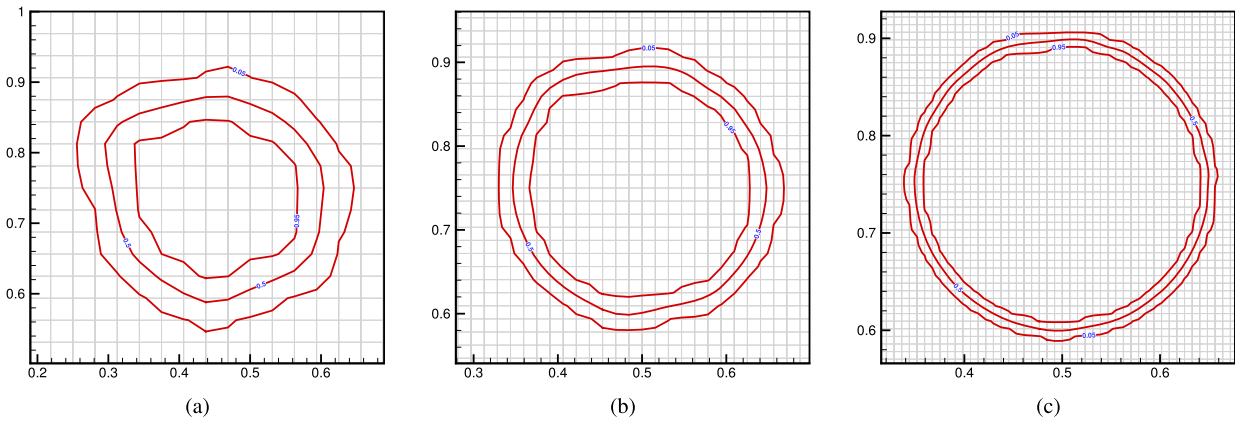


Fig. 13. The 0.05, 0.5 and 0.95 contours of VOF function in the Rider-Kothe single vortex flow test after one period ($t = T$) using the P4 polynomial on 32×32 (a), 128×128 (b) and 128×128 (c) meshes.

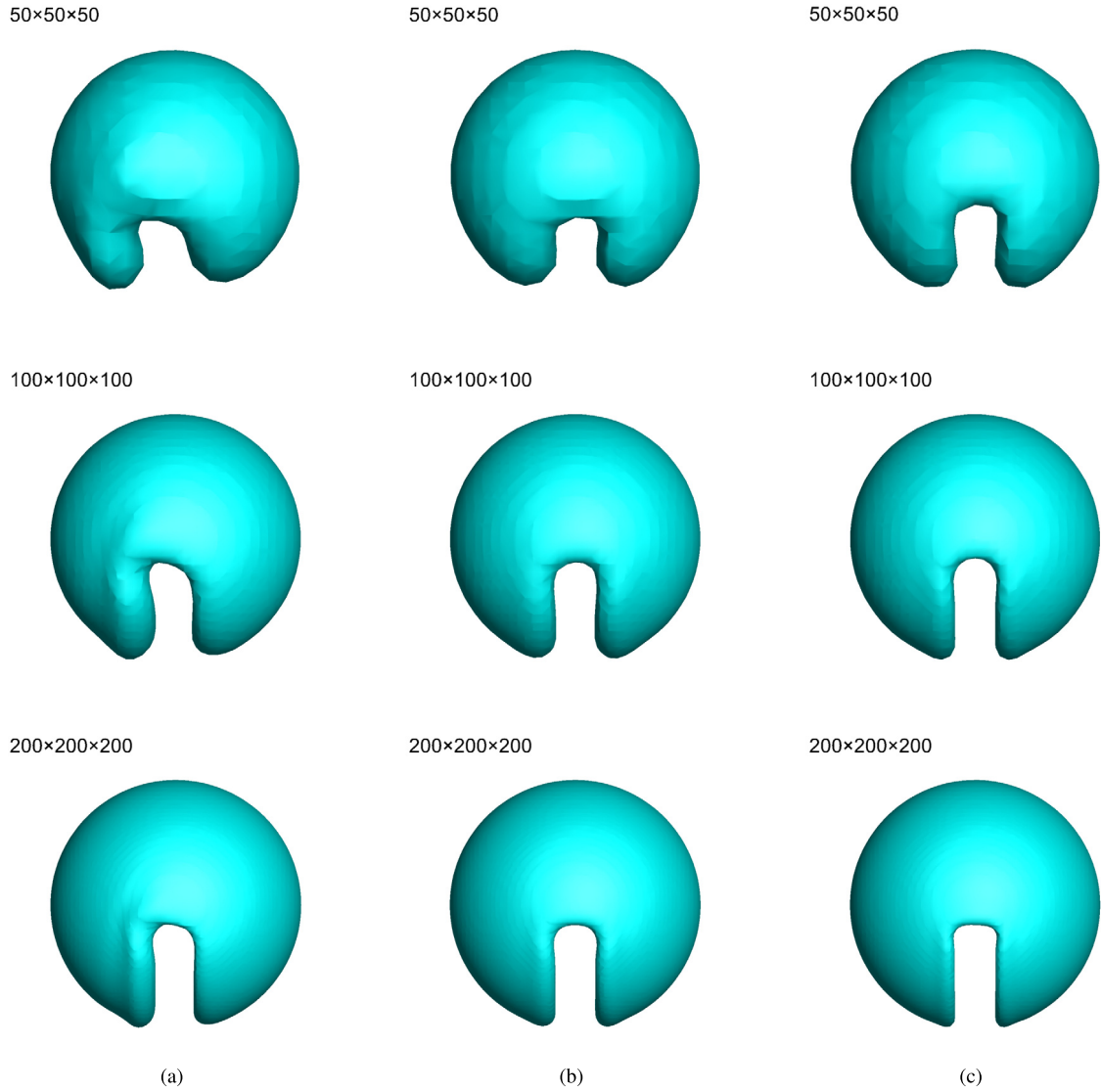


Fig. 14. Numerical results of the 3D Zalesak rotation test after one revolution using P1 (a), P2 (b) and P4 (c) surface polynomials on $50 \times 50 \times 50$, $100 \times 100 \times 100$, $200 \times 200 \times 200$ meshes. Displayed are the 0.5 iso-surface of the VOF fields.

Table 7

Numerical errors (E_r) and convergence rates of 3D Zalesak rotation test after one revolution period.

Methods	50^3	Order	100^3	Order	200^3
MTHINC [32]	1.19×10^{-3}	1.09	5.60×10^{-4}	0.91	2.97×10^{-4}
UMTHINC [35]	1.11×10^{-3}	1.38	4.27×10^{-4}	0.93	2.24×10^{-4}
THINC/QQ [35]	1.19×10^{-3}	1.35	4.67×10^{-4}	0.90	2.51×10^{-4}
THINC/LS (P1)	1.75×10^{-3}	1.68	5.48×10^{-4}	1.36	2.14×10^{-4}
THINC/LS (P2)	6.62×10^{-4}	1.09	3.11×10^{-4}	1.36	1.21×10^{-4}
THINC/LS (P4)	4.38×10^{-4}	1.58	1.46×10^{-4}	1.38	5.61×10^{-5}

$$\begin{cases} u(x, y, z) = 2\sin^2(\pi x) \sin(2\pi y) \sin(2\pi z) \cos(\pi t/T) \\ v(x, y, z) = -\sin(2\pi x) \sin^2(\pi y) \sin(2\pi z) \cos(\pi t/T). \\ w(x, y, z) = -\sin(2\pi x) \sin(2\pi y) \sin^2(\pi z) \cos(\pi t/T) \end{cases} \quad (28)$$

The period T is set to 3 and the maximum CFL number is set to 0.1 in our test. The numerical results of different surface polynomials for $32 \times 32 \times 32$, $64 \times 64 \times 64$, and $128 \times 128 \times 128$ meshes, are shown in Fig. 15. In this test case, we set $\beta = 6.0/\Delta_{\min}$ to get a steeper interface jump and the better resolved fine interface structures. It is observed that

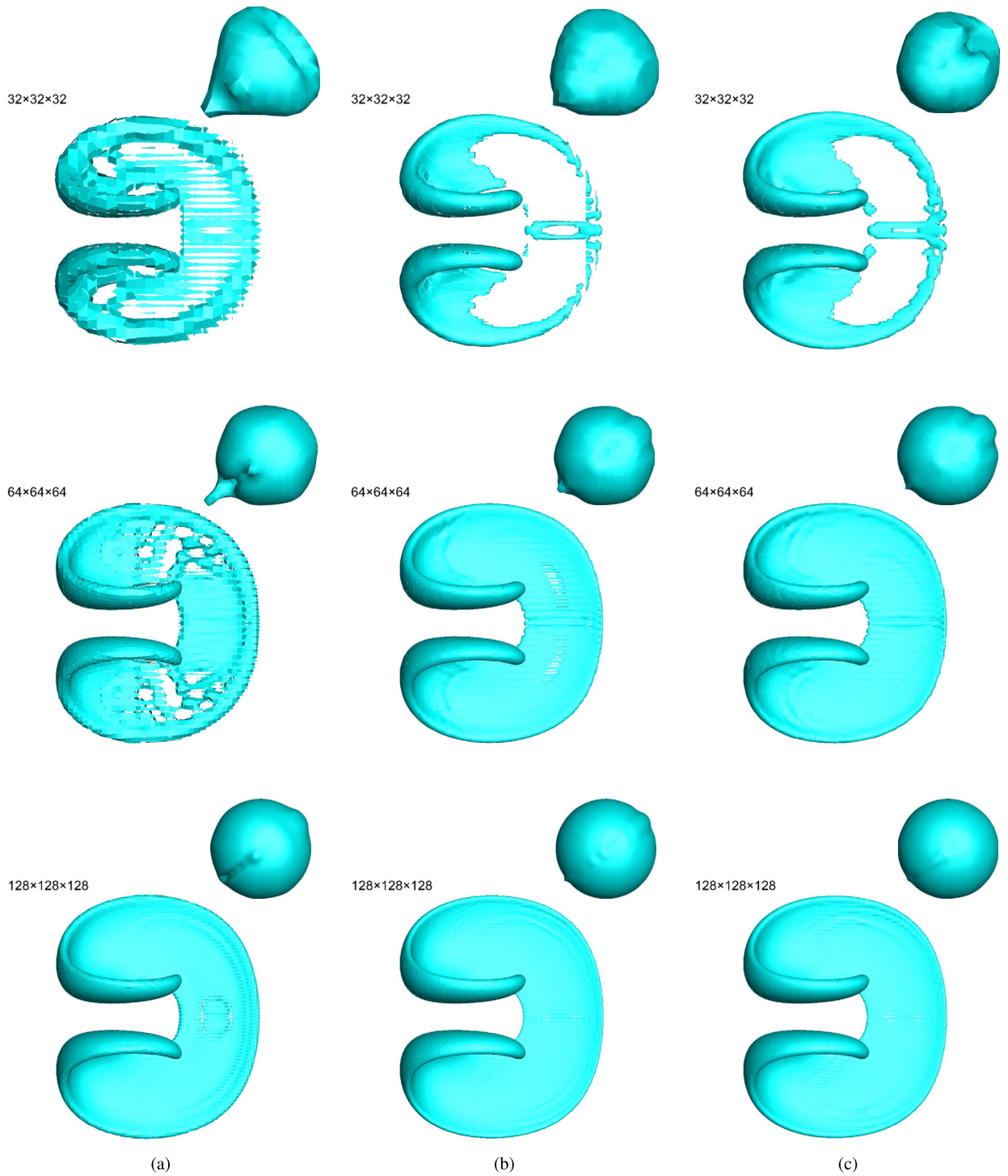


Fig. 15. Numerical results of the 3D vortex deformation test using P1 (a), P2 (b) and P4 (c) surface polynomials on $32 \times 32 \times 32$, $64 \times 64 \times 64$, and $128 \times 128 \times 128$ meshes. Surfaces at $t = T/2$ are the reconstructed interfaces which are generated from the LS values sampled at the $5 \times 5 \times 5$ uniformly distributed sample points within each mesh cell. Numerical surfaces at $t = T$ are displayed by the 0.5 iso-surfaces of the VOF fields.

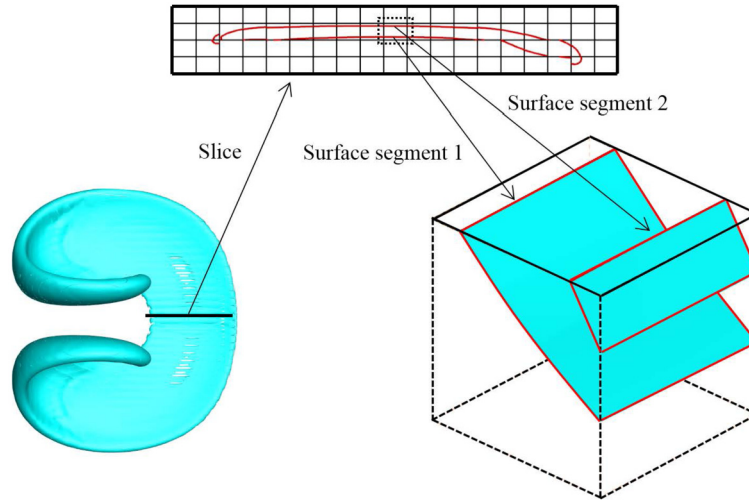


Fig. 16. Reconstructed surfaces of the 3D deformational interface transport test at $t = T/2$ using P2 surface polynomial on a $64 \times 64 \times 64$ mesh. It shows that two surface segments can be retrieved for the thin film under the mesh resolution.

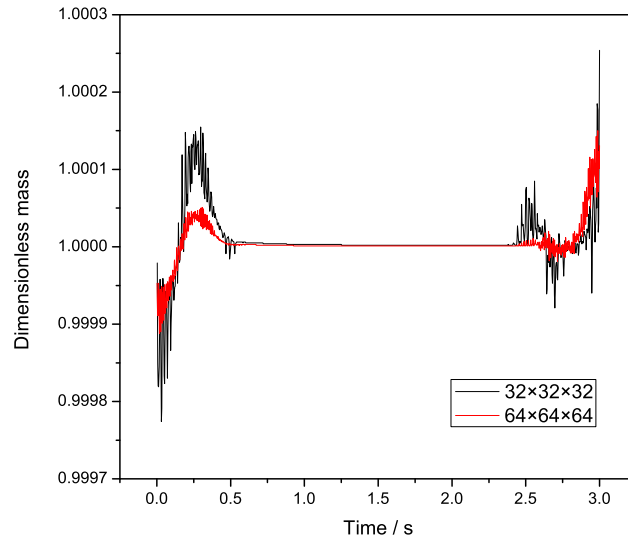


Fig. 17. Variation of the total volume (normalized by the initial volume) calculated from the Heaviside functions generated from the LS using P2 surface polynomial on 32^3 and 64^3 meshes.

high-order surface polynomials can retrieve the thin interfaces and restore the initial sphere shape with a substantially improved solution quality in comparison with the linear polynomial. Using high-order surface polynomials, the THINC/LS method can retrieve the interface under the grid resolutions. As shown in Fig. 16, the THINC/LS method with a surface reconstructions higher than 2nd order is able to identify the two interface segments of a thin film in unresolved regions, as demonstrated before in the 2D case. More relevant to applications, we also examined the volume/mass conservation for the Heaviside function generated from the LS field. We use the Gaussian quadrature to calculate the volume integration of the Heaviside function at different instants. Fig. 17 shows the ratio between the volumes at the initial time and the instants afterward. The volume conserves adequately through the whole process in accordance with the rigorous conservation in the VOF field updated by the THINC scheme.

It is noted that the present THINC/LS method provides simultaneously a VOF function with high-order surface representation and an LS function with an enforced conservativeness, without degrading the accuracy of the original LS method. The resultant LS functions are well suited for high-accurate interface capturing in real case applications, such as the multiphase Navier–Stokes equations including moving interfaces.

We further compare the L_1 errors and the convergence rates of the THINC/LS method against other VOF methods in Table 8. The linear surface polynomial solution of the THINC/LS method is quite close to the VOF methods. The THINC/LS method with high-order surface polynomials can produce more accurate results.

Table 8 L_1 errors and convergence rates of the 3D vortex deformation test ($\beta = 6.0$).

Methods	32^3	Order	64^3	Order	128^3
RK-3D [57]	7.85×10^{-3}	1.51	2.75×10^{-3}	1.89	7.41×10^{-4}
FMFPA-3D [57]	7.44×10^{-3}	1.42	2.79×10^{-3}	1.97	7.14×10^{-4}
Youngs [58]	7.47×10^{-3}	1.43	2.77×10^{-3}	1.77	8.14×10^{-4}
LVIRA [58]	6.92×10^{-3}	1.51	2.43×10^{-3}	1.93	6.37×10^{-4}
Improved ELVIRA-3D [28]	7.35×10^{-3}	1.45	2.69×10^{-3}	2.05	6.51×10^{-4}
DS-MOF [55]	5.72×10^{-3}	1.50	2.02×10^{-3}	–	–
DS-CLSVOF [55]	6.92×10^{-3}	1.70	2.13×10^{-3}	–	–
DS-CLSMOF [55]	4.81×10^{-3}	1.27	1.99×10^{-3}	–	–
UMTHINC [59]	8.06×10^{-3}	1.41	3.04×10^{-3}	1.69	9.40×10^{-4}
THINC/QQ [35]	7.96×10^{-3}	1.46	2.89×10^{-3}	1.68	9.05×10^{-4}
THINC/LS (P1)	7.18×10^{-3}	1.62	2.34×10^{-3}	1.93	6.14×10^{-4}
THINC/LS (P2)	6.81×10^{-3}	1.72	2.07×10^{-3}	2.06	4.51×10^{-4}
THINC/LS (P4)	5.54×10^{-3}	1.82	1.57×10^{-3}	1.88	3.79×10^{-4}

5. Conclusions

We propose a novel interface capturing scheme, so-called THINC/LS, for capturing moving interfaces. The innovative and practically-significant aspects of the proposed THINC/LS schemes are summarized as follows:

- Higher-order (arbitrary order in principle) surface polynomials can be determined from the smooth level set function, and then used to build the THINC or VOF function.
- The interface in the level set field is synchronized with the interface identified from the THINC reconstruction with the mass conservation constraint. Thus, the numerical conservativeness is ensured.
- A THINC/LS scheme with the surface polynomial higher than 2nd order is able to retrieve interfacial structures under grid resolution.
- The numerical scheme maintains well the interface thickness and sharpness, as well as the superior geometrical faithfulness of the moving interface.
- The THINC/LS method provides an easy-to-code algorithm for implementation and can be straightforwardly applied to 3D and unstructured grids.

Numerical verifications have been conducted via interface reconstruction tests and advection benchmark tests for moving interfaces to verify the THINC/LS method. It is revealed that the method can significantly improve the numerical accuracy in comparison with other existing methods despite much simpler numerical algorithm.

Acknowledgements

The authors gratefully acknowledge the helpful discussions with Xi Deng, Bin Xie, Peng Jin and Akio Ikebata. This work was supported in part by the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD) and the financial support of the project from the Fundamental Research Funds for the Central Universities NP2016204, as well as the funds from JSPS (Japan Society for the Promotion of Science) under Grant Nos. 15H03916, 17K18838 and 18H01366.

References

- [1] S.O. Unverdi, G. Tryggvason, A front-tracking method for viscous, incompressible, multi-fluid flows, *J. Comput. Phys.* 100 (1992) 25–37.
- [2] 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.
- [3] W. Rider, D. Kothe, A marker particle method for interface tracking, in: *Sixth International Symposium on Computational Fluid Dynamics (ISCFD)*, 1995.
- [4] W. Rider, D. Kothe, Stretching and tearing interface tracking methods, in: *12th AIAA Computational Fluid Dynamics Conference*, 1995.
- [5] G. Tryggvason, R. Scardovelli, S. Zaleski, *Direct Numerical Simulations of Gas-Liquid Multiphase Flows*, Cambridge University Press, 2011.
- [6] C.W. Hirt, B.D. Nichols, Volume of fluid (VOF) method for the dynamics of free boundaries, *J. Comput. Phys.* 39 (1981) 201–225.
- [7] D.L. Youngs, Time-Dependent Multi-Material Flow with Large Fluid Distortion, Academic Press, 1982, pp. 273–285.
- [8] B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, G. Zanetti, Modelling merging and fragmentation in multiphase flows with SURFER, *J. Comput. Phys.* 113 (1994) 134–147.
- [9] M. Rudman, Volume-tracking methods for interfacial flow calculations, *Int. J. Numer. Methods Fluids* 24 (1997) 671–691.
- [10] W.J. Rider, D.B. Kothe, Reconstructing volume tracking, *J. Comput. Phys.* 141 (1998) 112–152.
- [11] R. Scardovelli, S. Zaleski, Analytical relations connecting linear interfaces and volume fractions in rectangular grids, *J. Comput. Phys.* 164 (2000) 228–237.
- [12] E.G. Puckett, A volume-of-fluid interface tracking algorithm with applications to computing shock wave refraction, in: *Proceedings of the Fourth International Symposium on Computational Fluid Dynamics*, 1991, pp. 933–938.
- [13] J.E. Pilliod Jr, E.G. Puckett, Second-order accurate volume-of-fluid algorithms for tracking material interfaces, *J. Comput. Phys.* 199 (2004) 465–502.
- [14] S. Osher, J.A. Sethian, Fronts propagating with curvature-dependent speed: algorithms based on Hamilton–Jacobi formulations, *J. Comput. Phys.* 79 (1988) 12–49.

- [15] S. Osher, R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*, Springer, 2003.
- [16] 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.
- [17] W.F. Noh, P. Woodward, SLIC (simple line interface calculation), *Lect. Notes Phys.* 59 (1976) 273–285.
- [18] D.L. Youngs, An Interface Tracking Method for a 3D Eulerian Hydrodynamics Code, Tech. Rep., Atomic Weapons Research Establishment (AWRE) Design Mathematics Division, 1984.
- [19] R. Scardovelli, S. Zaleski, Interface reconstruction with least-square fit and split Eulerian–Lagrangian advection, *Int. J. Numer. Methods Fluids* 41 (2003) 251–274.
- [20] E. Aulisa, S. Manservigi, R. Scardovelli, S. Zaleski, Interface reconstruction with least-squares fit and split advection in three-dimensional Cartesian geometry, *J. Comput. Phys.* 225 (2007) 2301–2319.
- [21] E. Aulisa, S. Manservigi, R. Scardovelli, S. Zaleski, A geometrical area-preserving volume-of-fluid advection method, *J. Comput. Phys.* 192 (2003) 355–364.
- [22] A. Cervone, S. Manservigi, R. Scardovelli, S. Zaleski, A geometrical predictor–corrector advection scheme and its application to the volume fraction function, *J. Comput. Phys.* 228 (2009) 406–419.
- [23] S. Popinet, Gerris: a tree-based adaptive solver for the incompressible Euler equations in complex geometries, *J. Comput. Phys.* 190 (2003) 572–600.
- [24] S. Popinet, An accurate adaptive solver for surface-tension-driven interfacial flows, *J. Comput. Phys.* 228 (2009) 5838–5866.
- [25] S. Popinet, *Gerris Flow Solver*, <http://gfs.sourceforge.net>, 2017.
- [26] S. Popinet, *Basilisk*, <http://basilisk.fr>, 2017.
- [27] J. López, J. Hernández, P. Gómez, F. Faura, A volume of fluid method based on multidimensional advection and spline interface reconstruction, *J. Comput. Phys.* 195 (2004) 718–742.
- [28] J. López, C. Zanzi, P. Gómez, F. Faura, J. Hernández, A new volume of fluid method in three dimensions – Part II: piecewise-planar interface reconstruction with cubic-Bézier fit, *Int. J. Numer. Methods Fluids* 58 (2008) 923–944.
- [29] S. Diwakar, S.K. Das, T. Sundararajan, A quadratic spline based interface (QUASI) reconstruction algorithm for accurate tracking of two-phase flows, *J. Comput. Phys.* 228 (2009) 9107–9130.
- [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] F. Xiao, S. Li, C. Chen, Revisit to the THINC scheme: a simple algebraic VOF algorithm, *J. Comput. Phys.* 230 (2011) 7086–7092.
- [32] S. Li, K. Sugiyama, S. Takeuchi, S. Takagi, Y. Matsumoto, F. Xiao, An interface capturing method with a continuous function: the THINC method with multi-dimensional reconstruction, *J. Comput. Phys.* 231 (2012) 2328–2358.
- [33] S. Li, B. Xie, F. Xiao, An interface capturing method with a continuous function: the THINC method on unstructured triangular and tetrahedral meshes, *J. Comput. Phys.* 259 (2014) 260–269.
- [34] B. Xie, S. Li, F. Xiao, An efficient and accurate algebraic interface capturing method for unstructured grids in 2 and 3 dimensions: the THINC method with quadratic surface representation, *Int. J. Numer. Methods Fluids* 76 (2014) 1025–1042.
- [35] B. Xie, F. Xiao, Toward efficient and accurate interface capturing on arbitrary hybrid unstructured grids: the THINC method with quadratic surface representation and Gaussian quadrature, *J. Comput. Phys.* 349 (2017) 415–440.
- [36] J.A. Sethian, *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*, Cambridge University Press, 1999.
- [37] D. Enright, R. Fedkiw, J. Ferziger, I. Mitchell, A hybrid particle level set method for improved interface capturing, *J. Comput. Phys.* 183 (2002) 83–116.
- [38] 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.
- [39] T. Ménard, S. Tanguy, A. Berlemont, Coupling level set/VOF/ghost fluid methods: validation and application to 3D simulation of the primary break-up of a liquid jet, *Int. J. Multiph. Flow* 33 (2007) 510–524.
- [40] X. Yang, A.J. James, J. Lowengrub, X. Zheng, V. Cristini, An adaptive coupled level-set/volume-of-fluid interface capturing method for unstructured triangular grids, *J. Comput. Phys.* 217 (2006) 364–394.
- [41] D. Sun, W. Tao, A coupled volume-of-fluid and level set (VOSET) method for computing incompressible two-phase flows, *Int. J. Heat Mass Transf.* 53 (2010) 645–655.
- [42] W. Aniszewski, T. Ménard, M. Marek, Volume of Fluid (VOF) type advection methods in two-phase flow: a comparative study, *Comput. Fluids* 97 (2014) 52–73.
- [43] H. Zhao, A fast sweeping method for eikonal equations, *Math. Comput.* 74 (2005) 603–627.
- [44] G.-S. Jiang, D. Peng, Weighted ENO schemes for Hamilton–Jacobi equations, *SIAM J. Sci. Comput.* 21 (2000) 2126–2143.
- [45] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, *J. Comput. Phys.* 77 (1988) 439–471.
- [46] S. Gottlieb, On high order strong stability preserving Runge–Kutta and multi step time discretizations, *J. Sci. Comput.* 25 (2005) 105–128.
- [47] S.J. Cummins, M.M. Francois, D.B. Kothe, Estimating curvature from volume fractions, *Comput. Struct.* 83 (2005) 425–434.
- [48] S.T. Zalesak, Fully multidimensional flux-corrected transport algorithms for fluids, *J. Comput. Phys.* 31 (1979) 335–362.
- [49] D.J. Harvie, D.F. Fletcher, A new volume of fluid advection algorithm: the stream scheme, *J. Comput. Phys.* 162 (2000) 1–32.
- [50] D.J. Harvie, D.F. Fletcher, A new volume of fluid advection algorithm: the defined donating region scheme, *Int. J. Numer. Methods Fluids* 35 (2001) 151–172.
- [51] O. Ubbink, R. Issa, A method for capturing sharp fluid interfaces on arbitrary meshes, *J. Comput. Phys.* 153 (1999) 26–50.
- [52] K. Yokoi, Efficient implementation of THINC scheme: a simple and practical smoothed VOF algorithm, *J. Comput. Phys.* 226 (2007) 1985–2002.
- [53] J. Lopez, J. Hernandez, P. Gomez, F. Faura, An improved PLIC-VOF method for tracking thin fluid structures in incompressible two-phase flows, *J. Comput. Phys.* 208 (2005) 51–74.
- [54] E. Aulisa, S. Manservigi, R. Scardovelli, A mixed markers and volume-of-fluid method for the reconstruction and advection of interfaces in two-phase and free-boundary flows, *J. Comput. Phys.* 188 (2003) 611–639.
- [55] M. Jemison, E. Loch, M. Sussman, M. Shashkov, M. Arienti, M. Ohta, Y. Wang, A coupled level set-moment of fluid method for incompressible two-phase flows, *J. Sci. Comput.* 54 (2013) 454–491.
- [56] R.J. LeVeque, High-resolution conservative algorithms for advection in incompressible flow, *SIAM J. Numer. Anal.* 33 (1996) 627–665.
- [57] J. Hernández, J. López, P. Gómez, C. Zanzi, F. Faura, A new volume of fluid method in three dimensions – Part I: multidimensional advection method with face-matched flux polyhedra, *Int. J. Numer. Methods Fluids* 58 (2008) 897–921.
- [58] L. Jofre, O. Lehmkuhl, J. Castro, A. Oliva, A 3-D volume-of-fluid advection method based on cell-vertex velocities for unstructured meshes, *Comput. Fluids* 94 (2014) 14–29.
- [59] B. Xie, P. Jin, F. Xiao, An unstructured-grid numerical model for interfacial multiphase fluids based on multi-moment finite volume formulation and THINC method, *Int. J. Multiph. Flow* 89 (2017) 375–398.