Short note

An interface capturing method with a continuous function: The THINC method on unstructured triangular and tetrahedral meshes

Satoshi Ii a,*, Bin Xie b, Feng Xiao b

a Graduate School of Engineering Science, Osaka University, 1-3 Machikaneyama, Toyonaka, Osaka, 560-8531, Japan
b Department of Energy Sciences, Tokyo Institute of Technology, 4259 Nagatsuta Midori-ku, Yokohama, 226-8502, Japan

A novel interface-capturing method is proposed to compute moving interfaces on unstructured grids with triangular (2D) and tetrahedral (3D) elements. Different from the conventional VOF (volume of fluid) method which involves geometric reconstructions of the interface, the present method is based on the algebraic reconstruction approach originally developed in the THINC (tangent of hyperbola interface capturing) scheme by Xiao et al. (2005) [17]. A continuous multidimensional hyperbolic tangent function is employed for retrieving the jump-like distribution of the indicator function, which avoids the explicit geometric representation of the interface and thus substantially reduces the algorithmic complexity in unstructured grids. Numerical diffusion and smearing are effectively eliminated, and the compact thickness of the jump transition layer in the volume fraction is retained throughout the computation even for largely deformed interface. The solution quality of the present scheme is comparable to the VOF method with PLIC (piecewise linear interface calculation) algorithm.

© 2013 Elsevier Inc. All rights reserved.

1. Introduction

The one-fluid model [16] has been increasingly used in the simulations of multi-fluid (phase) dynamics due to innovative progress in computer hardwares which makes the interface-resolvable computations affordable for routinely available computers in most laboratories. The interface capturing method plays a key role in a one-fluid model for interfacial multiphase flows. Eulerian type formulation that solves a field function in a fixed grid has been proven powerful and robust for the multiphase flow analyses involving topological changes in the moving interfaces separating different fluids. Among the widely used methods of this kind, the volume of fluid (VOF) method is of particular interest because of its rigorous numerical conservation. Following the underlying concept in [5], more sophisticated algorithms for reconstructing the interface have been proposed in the successive studies [22,23,11,9,10,4,1,13,8].

A VOF method makes use of the volume fraction function to identify the moving interface which is updated through a finite volume formulation. So, the key issue is how to calculate the numerical fluxes. To this end, piecewise reconstructions must be built over the grid cells where the interface falls in. The most popular reconstruction in the VOF method is the piecewise linear interface calculation (PLIC) method found in [22,23] and its advanced variants [11,9,10,4,1,13,8], where a Heaviside function is defined by explicitly creating the interface with a line (2D) or a plane (3D) based on geometric
representations which are usually complicated, even some efforts have been made to simplify the numerical procedure in the PLIC algorithm [12]. A more comprehensive survey of the geometric type VOF method is found in [16].

Above mentioned methods are essentially established on a Cartesian coordinate mesh, and there are a few existing works in the literature to extend the PLIC reconstruction to unstructured grids with triangular element [14,19] and tetrahedral element [19]. However, a geometric reconstruction for an interface in a triangular or tetrahedral grid including the interface advection is not a trivial task, which involves much more complicated numerical procedure compared to the cases in the Cartesian grid. Furthermore, the algorithmic complexity will be significantly increased when higher order approximation, e.g. quadratic surface, is used to represent the interface. To our knowledge, numerical model for interfacial flows with PLIC VOF method in tetrahedral grid has not been well-established. From the point of real-case application, a reliable and efficient interface-capturing scheme for unstructured triangular/tetrahedral grid is highly demanded.

Based on another strategy different from the conventional PLIC VOF schemes, an algebraic type method, so-called tangent of hyperbola interface capturing (THINC) method [17], has been proposed by utilizing a hyperbolic tangent function for the reconstruction. Some variants have been so far developed [20,18,21] to improve the multidimensional solutions when using the 1D solver as the basic building block. These THINC schemes are very simple and can be used as normal advection schemes. In spite of the simplicity, these schemes can give high-quality numerical solutions comparable to the conventional PLIC VOF methods.

Using a continuous analytical function, a THINC scheme uniquely retrieves the interface from given volume fraction values. The geometric manipulations in the PLIC algorithm are not required here, which largely simplifies the numerical procedure and allows the multidimensional extension to be conducted in a straightforward way. A sophisticated scheme, so-called MTHINC (multidimensional THINC) method [6], has been proposed using a multidimensional hyperbolic tangent function for reconstruction. The numerical accuracy is remarkably improved due to the multidimensional reconstruction, and moreover, higher-order interface approximation, like the quadratic interface, can be formulated straightforwardly under the THINC framework.

In this short note, we further extend the numerical procedure presented in [6] to unstructured grids, which leads to an accurate and practical interface-capturing method well-suited for simulating interfacial multi-fluid flows in complex geometry. The new scheme, unstructured MTHINC (UMTHINC) method, is presented on unstructured triangular/tetrahedral meshes in this study. The multidimensional hyperbolic function is constructed piecewisely with the interface represented by a linear approximation. The numerical results of the typical advection benchmark tests show the competitive accuracy of UMTHINC scheme in comparison with other VOF interface-capturing schemes.

2. The numerical algorithm

2.1. Basic definition and equation

The computational domain \( \Omega \) is filled with fluid 1 (\( x \in \Omega^{(1)} \)) and fluid 2 (\( x \in \Omega^{(2)} \)). We assume that both fluids are physically immiscible, but allow the transition layer of the interface (\( x \in \Gamma \)) to have a finite thickness in space. Then, a time-dependent smooth indicator function \( H(x, t) \) of fluid 1 is defined by

\[
H(x, t) = \begin{cases} 
1, & x \in \Omega^{(1)}, \\
0, & x \in \Omega^{(2)}, \\
0 < H(x, t) < 1, & x \in \Gamma,
\end{cases}
\]

on \( \mathbb{R}^3 = \{(x, y, z) \mid x \in \Omega = \Omega^{(1)} \cup \Omega^{(2)} \cup \Gamma\} \), or vice versa, the indicator function of fluid 2 is given by \( 1 - H(x, t) \) on \( \mathbb{R}^3 \).

Consider that the interface between different fluids is moving with the flow velocity, the time evolution of the indicator function is updated by solving the following advection equation,

\[
\frac{\partial H}{\partial t} + \nabla \cdot (\mathbf{v} H) = H \nabla \cdot \mathbf{v},
\]

where \( \mathbf{v} \) is the velocity vector.

The computational domain \( \Omega \) is divided into mesh cells (triangular or tetrahedral element) \( \Omega_i \ (i = 1, 2, \ldots, N) \), where \( N \) is the total number of mesh cells and \( \Omega = \bigcup_{i=1}^{N} \Omega_i \). Then the discrete volume fraction value of \( H(x, t) \) for cell \( \Omega_i \) is defined as

\[
\phi_i(t) \equiv \frac{1}{\Delta V_i} \int_{\Omega_i(x)} H(x, t) \, dx,
\]

where \( \Delta V_i \) is the volume of cell \( \Omega_i \). According to the advection equation (2), the semi-discrete equation to update the volume fraction \( \phi_i(t) \) is cast into a finite volume formulation as,

\[
\frac{\partial \phi_i(t)}{\partial t} + \frac{1}{\Delta V_i} \sum_{j=1}^{J} \left( v_{nj} \int_{S_{ij}} H(x, t) \, dS \right) = \phi_i(t) \frac{1}{\Delta V_i} \sum_{j=1}^{J} (v_{nj} \Delta S_{ij}),
\]
where \( ij \) denotes the index of the \( j \)th surface \( S_{ij} \) of cell \( \Omega_i \). \( f \) the number of surfaces of the cell \((f = 3 \text{ for } 2D \text{ and } f = 4 \text{ for } 3D) \text{ tetrahedron}), \Delta S_{ij} \) the area of surface \( S_{ij} \), \( v_{n_{ij}} = v_{ij} \cdot n_{ij} \) the surface normal velocity on surface \( S_{ij} \) and \( n_{ij} \) the unit outward normal vector of \( S_{ij} \). Here we employ a linear (plane) element and assume that the surface normal velocity \( v_{n_{ij}} \) is a constant on surface \( S_{ij} \).

2.2. Reconstruction of the piecewise indicator function \( H_i(x) \)

The major task in the VOF methods is to retrieve the piecewise indicator function \( H_i(x) \) on each cell \( \Omega_i \) from the discrete volume fraction value \( \phi_i \). Instead of the discontinuous Heaviside function based on the explicit geometric representation of the interface in the conventional VOF methods, we employ a hyperbolic tangent function piecewise defined in each cell,

\[
H_i(x) = \frac{1}{2} \left( 1 + \tanh \left( \beta \left( m_{xi} x + m_{yi} y + m_{zi} z + \hat{d} \right) \right) \right),
\]

where parameter \( \beta \) is used to control the sharpness of the transition jump in function \( H_i(x) \). \( m_i = (m_{xi}, m_{yi}, m_{zi}) \) is the unit normal vector of the interface surface, which is computed from the derivative of the discrete volume fraction value on a union of cells which share surfaces with \( \phi_i \). In this case, \( \phi_i \) is a small value set to 0.01 to avoid the arithmetic failure in computation.

In practice, we only reconstruct the indicator function for the interface cells where the volume fraction satisfies \( \phi_{\text{min}} \leq \phi_i \leq \phi_{\text{max}} \), where \( \phi_{\text{min}} \) is a small value set to 10^{-8} in this study.

Case 1: |\( a_{\xi} \) max(|\( a_{\xi} \), |\( a_{\eta} \), |\( a_{\zeta} \))

An integrated average of \( H_i(x) \) with respect to \( \xi \), namely \( \bar{H}_i^{\xi}(\eta, \zeta) \), is given by an analytical expression as

\[
\bar{H}_i^{\xi}(\eta, \zeta) = \frac{1}{1 - \eta - \zeta} \int_0^{1-\eta-\zeta} H_i(\xi, \eta, \zeta) d\xi
\]

\[
= \frac{1}{2} \left( 1 + \frac{1}{\beta a_\xi (1 - \eta - \zeta)} \ln \left( \frac{\cosh(\beta(a_\xi (1 - \eta - \zeta) + a_\eta \eta + a_\zeta \zeta + d))}{\cosh(\beta(a_\eta \eta + a_\zeta \zeta + d))} \right) \right).
\]
Then \( d \) in (9) is determined by the following constraint:
\[
\vec{R}_{i}^d(1/4, 1/4) = \phi_i,
\]  
which is obtained by employing the mid-point integration for (12). A few algebraic manipulations from (13) with (12) finally give
\[
d = \frac{1}{2\beta'} \ln \left( \frac{1 - AQ}{AB(\phi_1 - A)} \right),
\]  
where
\[
A = \exp \left( \frac{1}{2} \beta' a_{\xi} \right), \quad B = \exp \left( \frac{1}{2} \beta' (a_{\eta} + a_{\zeta}) \right), \quad Q = \exp \left( \frac{1}{2} \beta' a_{\xi} (2\phi_1 - 1) \right).
\]  

**Case 2:** \( |a_{\eta}| = \max(|a_{\xi}|, |a_{\eta}|, |a_{\zeta}|) \)

The unknown \( d \) is determined through the following constraint:
\[
\vec{R}_{i}^d(1/4, 1/4) = \phi_i,
\]  
where \( \vec{R}_{i}^d(\xi, \zeta) = \frac{1}{1-\xi-\zeta} \int_0^{1-\xi-\zeta} H_i(\xi, \eta, \zeta) d\eta \), and finally obtained from (14) but with the coefficients:
\[
A = \exp \left( \frac{1}{2} \beta' a_{\eta} \right), \quad B = \exp \left( \frac{1}{2} \beta' (a_{\xi} + a_{\zeta}) \right), \quad Q = \exp \left( \frac{1}{2} \beta' a_{\eta} (2\phi_1 - 1) \right).
\]  

**Case 3:** \( |a_{\xi}| = \max(|a_{\xi}|, |a_{\eta}|, |a_{\zeta}|) \)

Constraint:
\[
\vec{R}_{i}^d(1/4, 1/4) = \phi_i,
\]  
where \( \vec{R}_{i}^d(\xi, \eta) = \frac{1}{1-\xi-\eta} \int_0^{1-\xi-\eta} H_i(\xi, \eta, \zeta) d\zeta \).

Coefficients in (14):
\[
A = \exp \left( \frac{1}{2} \beta' a_{\xi} \right), \quad B = \exp \left( \frac{1}{2} \beta' (a_{\xi} + a_{\eta}) \right), \quad Q = \exp \left( \frac{1}{2} \beta' a_{\xi} (2\phi_1 - 1) \right).
\]  

See Appendix A in 2D case.

### 2.3. Updating of the volume fraction

The finite volume formulation (4) is used to update the volume fraction,
\[
\frac{d\phi_i(t)}{dt} = -\frac{1}{\Delta V_i} \sum_{j=1}^{f} (v_{ni,j} \Delta S_{ij} \langle H_{i_{up}}(t) \rangle_{S_{ij}}) + \phi_i(t) \frac{1}{\Delta V_i} \sum_{j=1}^{f} (v_{ni,j} \Delta S_{ij}),
\]  
where \( \langle \cdot \cdot \cdot \rangle_{S_{ij}} \) denotes the averaged value over surface \( S_{ij} \), which is numerically obtained by the 2-point Gauss quadrature in 2D and 6-point Gauss quadrature in 3D using the reconstructed indicator function \( H_i(\xi, \eta, \zeta) \) obtained above (see Appendix B for details). \( i_{up} \) is the index of the upwind cell of the surface \( S_{ij} \), that is
\[
i_{up} = \begin{cases} i, & \text{for } v_{ni,j} \geq 0, \\ ij, & \text{otherwise}, \end{cases}
\]  
where \( ij \) (\( j = 1, 2, 3 \) in 2D triangular mesh and \( j = 1, 2, 3, 4 \) in 3D tetrahedral mesh) denotes the index of the neighboring cells of \( \Omega_i \).

With all spatial discretization completed using \( H_i(\xi, \eta, \zeta) \), we obtain a semi-discrete (ordinary differential) equation (20) evolving in time \( t \). We apply the 3-stage TVD Runge–Kutta method [15] to solve (20) and update the volume fraction values.

### 3. Numerical examples

Numerical benchmark tests are presented on unstructured grids with triangular (2D) and tetrahedral (3D) elements generated by the Delaunay algorithm. In the present study, we employ equilateral computational domains, thus the relation between the mesh cell number for each side, \( N_d \), and the total cell number can be approximated by
\[
N_d \approx \begin{cases} (N/2)^{\frac{1}{2}}, & \text{in 2D}, \\ (N/6)^{\frac{1}{3}}, & \text{in 3D}. \end{cases}
\]
The sharpness parameter $\beta'$ in (9) is determined from a normalized sharpness parameter $\beta$ and a representative cell size $h_{\text{ave}} = \frac{1}{N} \sum_{i=1}^{N} h_i$ as

$$\beta' = \frac{\beta}{h_{\text{ave}}},$$

(23)

where

$$h_i = \begin{cases} (2\Delta V_i)^{\frac{1}{3}}, & \text{in 2D}, \\ (6\Delta V_i)^{\frac{1}{3}}, & \text{in 3D}. \end{cases}$$

(24)

We will discuss the influence of $\beta$ on the numerical solution in Section 3.1. A CFL number is defined as

$$\text{CFL} = \max_{i \in [1,N]} \frac{|\mathbf{v}_i| \Delta t}{h_i},$$

(25)

in terms of the velocity vector $\mathbf{v}_i$, the mesh cell size $h_i$ and time increment $\Delta t$.

An initial data of the discrete volume fraction value $\phi_i(t = 0)$ is given by numerically integrating the following smooth indicator function over $\Omega_1$, as

$$H(x, 0) = \frac{1}{2} (1 + \tanh(\beta' \Psi(x))),$$

(26)

where $\Psi(x)$ is a signed distance function of the initial data. The numerical errors in the computed volume fraction function is defined by the $L_1$ norm as

$$\|E_\phi\|_1 = \frac{\sum_{i=1}^{N} \Delta V_i |\phi_i(t^n) - \phi_i(0)|}{\sum_{i=1}^{N} \Delta V_i}. $$

(27)

3.1. Rotation of a slotted-disk on the triangular mesh

Firstly, we perform the Zalesak's solid rotation test [24]. The 2D tests are carried out on a triangular mesh of $N = 10082$ ($N_d = 71$), where we set the time increment $\Delta t$ so as to let the CFL number be approximately 0.17. According to the example in [18], we set $\beta = 2.3$. Fig. 1(a) shows the blow-up views of the transported interface (contour of $\phi = 0.5$) at $t = 2\pi$ (one revolution) and $t = 10\pi$ (five revolutions). Here, we also show the initial solution and computational meshes. The initial profile of the slotted-disk is adequately preserved after one revolution. Although the discrepancy in the shape looks more noticeable around the sharp corners after five revolutions, the numerical solution after long time of calculation is still acceptable. The total volume fraction values over the whole computational domain are exactly conserved. Meanwhile, the jump thickness in volume fraction is kept compact throughout the computation, thus the total volume encompassed by the profile in the transition layer, i.e. $0 < \phi < 1$, is captured in two or three meshes without numerical diffusion. It is also observed that the numerical solution is free of numerical oscillation even the numerical quadrature is employed for reconstructing the indicator function and calculating the numerical flux. Further quantitative evaluation is shown next.

Here, we investigate minimum/maximum values using different sharpness parameters ($\beta = 1.0, 2.3, 3.5, 5.0, 10.0$) in (23) and different CFL numbers (CFL = 0.017, 0.17). The minimum and maximum values of $\phi_i$ for whole computational domain ($i \in [1,N]$) are averaged over time $t \in [0, 2\pi]$ (for one revolution). The results are summarized in Table 1, where we underline the values out the range of [0, 1]. It is found that the increase of $\beta$, which results in a steeper transition layer, is prone to generate numerical oscillations. Meanwhile, a large CFL number also tends to cause numerical oscillations. We may owe these to the approximations involved in the numerical quadrature. In practice, a medium $\beta$ value effectively regulates the oscillations to be negligibly small. It is also noted that the thickness of the transition layer can be controlled by $\beta$ and maintained to a certain level throughout the computation. The accumulation of diffusion errors observed in conventional Eulerian advection schemes, which tends to smear out the sharpness in the volume fraction field, is not seen here.

We further carried out a series of computations with the CFL number gradually increased from 0.05 to 1 on the same mesh ($N = 10082$) to see how CFL number affects the numerical solution and clarify the range of the CFL number which ensures a reasonable numerical accuracy. Here, we set the sharpness parameter $\beta$ to 1.0 and 2.3, respectively. A relationship between the numerical error $\|E_\phi\|_1$ at $t = 2\pi$ (one revolution) and the CFL number is shown in Fig. 2. The numerical error with $\beta$ equal to either 1.0 or 2.3 remains nearly constant until the CFL number approaches 0.15 for $\beta = 1.0$ and 0.25 for $\beta = 2.3$. As the CFL number becomes larger than these values, the numerical error gradually but not significantly increases until the CFL number reaches 0.9 for $\beta = 1.0$ and 0.6 for $\beta = 2.3$. This observation means that a small sharpness
Fig. 1. The transported interface of the Zalesak’s test with an unstructured triangular mesh \((N = 10082)\) at \(t = 2\pi\) and \(10\pi\): (a) contour plot of \(\phi = 0.5\) for the numerical solution (solid line), exact solution (dashed line) and the computational mesh; (b) axial profile of \(\phi\) along cross section \(y = 0.75\).

Table 1
Minimum/maximum values with different \(\beta\) and CFL number for the 2D Zalesak’s rotation test, where the underline in the table indicates undershoot or overshoot of the volume fraction value, i.e. \(\min_{i\in[1,N]}\phi_i < 0\) or \(\max_{i\in[1,N]}\phi_i > 1\). The values are averaged in time: \(t \in [0, 2\pi]\) (during one revolution).

<table>
<thead>
<tr>
<th>(\beta)</th>
<th>(\min_{i\in[1,N]}\phi_i)</th>
<th>(\max_{i\in[1,N]}\phi_i - 1)</th>
<th>(\min_{i\in[1,N]}\phi_i)</th>
<th>(\max_{i\in[1,N]}\phi_i - 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>0</td>
<td>-1.54E-4</td>
<td>0</td>
<td>-1.54E-4</td>
</tr>
<tr>
<td>2.3</td>
<td>0</td>
<td>-7.28E-9</td>
<td>0</td>
<td>-7.11E-9</td>
</tr>
<tr>
<td>3.5</td>
<td>0</td>
<td>-3.91E-9</td>
<td>-1.26E-12</td>
<td>-4.60E-9</td>
</tr>
<tr>
<td>5.0</td>
<td>0</td>
<td>-2.45E-9</td>
<td>-5.29E-4</td>
<td>4.22E-4</td>
</tr>
<tr>
<td>10.0</td>
<td>-2.42E-4</td>
<td>1.91E-4</td>
<td>-1.21E-2</td>
<td>1.05E-2</td>
</tr>
</tbody>
</table>

Fig. 2. A sensitivity test of the numerical error against the CFL number in Zalesak’s rotation test for different sharpness parameters, \(\beta = 1.0\) and 2.3. Parameter \(\beta\) giving a smoother profile of the function \(H(x,t)\), which allows the use of a larger CFL number for numerical accuracy. In both cases, reasonably large CFL number can be used for numerical accuracy.
Table 2
Convergence study in mesh refinement for the 2D Zalesak’s rotation test.

<table>
<thead>
<tr>
<th>Nd</th>
<th>Error</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>64</td>
<td>2.84E−3</td>
<td>–</td>
</tr>
<tr>
<td>128</td>
<td>1.03E−3</td>
<td>1.46</td>
</tr>
<tr>
<td>256</td>
<td>4.88E−4</td>
<td>1.08</td>
</tr>
</tbody>
</table>

Fig. 3. Numerical solutions on the time-dependent single vortex flow at \( t = T_0/2 \) and \( T_0 \) with triangular meshes of \( Nd = 128 \) and 256. The time for one circle calculation is \( T_0 = 8 \). Displayed are the contours of \( \phi = 0.05, 0.5 \) and 0.95.

Table 3
Convergence study in mesh refinement on the 2D time-dependent single vortex flow.

<table>
<thead>
<tr>
<th>Nd</th>
<th>Error</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>64</td>
<td>1.16E−2</td>
<td>–</td>
</tr>
<tr>
<td>128</td>
<td>2.30E−3</td>
<td>2.25</td>
</tr>
<tr>
<td>256</td>
<td>5.72E−4</td>
<td>2.01</td>
</tr>
</tbody>
</table>

We also investigated the convergence rate of the present scheme by solving the Zalesak’s problem on a set of gradually refined meshes. Here we set the maximum CFL number to 0.17 and use meshes of \( Nd = 64, 128 \) and 256. The errors evaluated after one revolution (\( t = 2\pi \)) are summarized in Table 2. It is found that the numerical solution converges at a rate higher than first order in this test, which is comparable to the other existing works formulated in the Cartesian mesh [8,20,6].

From the observations discussed above, we see that the sharpness parameter \( \beta \) and CFL number affect the quality of the numerical solution. A larger \( \beta \), resulting in more compact thickness of the jump transition with less numerical diffusion, is more favorable for capturing the thin structures in interfacial flows. However, care must be taken for the possible numerical oscillation and the restriction on the stable CFL number. In real case applications, it is not difficult for a user to find a sharpness parameter \( \beta \) which gives both adequate accuracy and computational robustness. Hereafter, we employ \( \beta = 2.3 \) in all the following tests and use the CFL number less than 0.2 to ensure the numerical accuracy.

3.2. A time-dependent single vortex flow on the triangular mesh

A time-dependent single vortex flow introduced in [2,10] is carried out on the triangular meshes with \( Nd = 64, 128 \) and 256, where the maximum CFL number ranges from 0.17 to 0.2 (depending on the mesh configuration in different mesh resolution). An initial circular cylinder deforms during \( t \leq T_0/2 \) and then it returns back to the initial shape at \( t = T_0 \) by imposing an inverse velocity field. Here the time for one circle is \( T_0 = 8 \). Fig. 3 shows numerical solutions at \( t = T_0/2 \) and \( T_0 \) on \( Nd = 128 \) and \( Nd = 256 \) meshes. We plotted the contours of \( \phi = 0.05, 0.5 \) and 0.95 and observed that the thickness of transition region of \( \phi \) from 0 to 1 remains compact during the computation even for the heavily distorted interface. We also see a converged solution with a more compact thickness when a mesh of higher spatial resolution (\( Nd = 256 \)) is used, where the numerical solution faithfully duplicates the initial circular shape (exact solution) after one circle calculation. A quantitative numerical errors (\( \|E_\phi\|_1 \)) and the convergence rate at \( t = T_0 \) are shown in Table 3. The error is decreased as the mesh resolution increases at a rate of approximately 2. A direct comparison with [14] reveals that our results are comparable to the PLIC VOF method on the triangular mesh with the same spatial resolution (see Table 8 of [14] for details).

3.3. Rotation of a 3D slotted-sphere on the tetrahedral mesh

Here we extend the Zalesak’s rotation test to 3D. The initial condition and velocity field are detailed in [6]. Computations are run up to \( t = 2\pi \) (one revolution) with the meshes of \( Nd = 20, 40 \) and 80. The CFL number is approximately 0.1. The iso-surface for \( \phi = 0.5 \) at \( t = 0 \) (exact solution) and \( 2\pi \) is shown in Fig. 4. Similar to the 2D results, the initial profile of the slotted-sphere is adequately preserved with a small deformation around the sharp edge of the interface, which is
3.4. Time-dependent 3D vortical deformation flow on the tetrahedral mesh

A sphere is deformed and returned back to the initial position after one period due to a time-dependent vortical deformation flow [7]. We carry out the computations on the tetrahedral meshes of \( N_d = 20, 40 \) and 80, and the maximum CFL number is 0.1. The time for one circle is \( T_0 = 3 \). Instantaneous solutions at \( t = 0, T_0/2 \) and \( T_0 \) are shown in Fig. 5. It is seen that although the thin structure at the most deformed state is not resolvable as other existing VOF schemes on meshes coarser than \( N_d = 80 \), significant improvement in the geometric faithfulness for the solution at \( T_0 \) is obtained if a finer grid is used. Table 5 shows numerical errors and convergence rate. It is again shown that the results nearly retain a second-order accuracy.

4. Concluding remarks

We have proposed a novel interface capturing method, UMTHINC, for both unstructured triangular and tetrahedral meshes. Being the unstructured grid extension of the THINC methods [17,20,18,6] which have been so far limited to the Cartesian grid, UMTHINC method uses the multidimensional hyperbolic tangent function, which gets around the complexity...
of geometric reconstruction as in the conventional VOF schemes and results in an efficient and accurate interface capturing scheme for unstructured grid.

We summarize the major features of the present UMTHINC method as follows.

- The piecewise indicator function can be uniquely constructed from the discrete volume fraction value, without the explicit manipulations of geometric components in the conventional PLIC algorithm where a discontinuous Heaviside function is used.
- Using the hyperbolic tangent function that has a transition jump zone with a controllable thickness, the present scheme essentially removes the numerical diffusion and smearing, and thus keeps the moving interface well regulated with a compact thickness even for the heavily distorted interfaces.
- Given the THINC reconstruction, the numerical procedure involved in the UMTHINC scheme requires only some numerical techniques developed in the standard finite volume method. The UMTHINC can be viewed as a pure advection scheme to transport the volume fraction function, and is very easy to use.
- The present method exhibits a robustness for a wide range of the CFL number, and reasonably large CFL number can be used with acceptable numerical accuracy.

The numerical tests show that the UMTHINC scheme has numerical accuracy comparable to the PLIC-type VOF schemes. For its simplicity, robustness and adequate accuracy for unstructured grid, we believe that the present method will be one of the practical and attractive choices to compute moving interfaces in the simulations of multiphase flows involving complex geometries.

Acknowledgement

FX is supported in part by JSPS KAKENHI (24560187).

Appendix A. Reconstruction of the piecewise indicator function in 2D

A 2D function of $H_i(\xi, \eta)$ is given by

$$H_i(\xi, \eta) = \frac{1}{2} \left( 1 + \tanh \left( \beta' a_\xi \xi + a_\eta \eta + d \right) \right)$$

$$(0 \leq \xi, \eta \leq 1, 0 \leq \xi + \eta \leq 1).\quad (A.1)$$

Similar to 3D case, we consider Case 1 and Case 2 for the magnitude of $|a_\xi|$ and $|a_\eta|$.

Case 1: $|a_\xi| = \max(|a_\xi|, |a_\eta|)$

An analytical expression for integrating $H_i(\xi, \eta)$ with respect to $\xi$ is obtained as

$$\bar{H}_i^\xi(\eta) = \frac{1}{1 - \eta} \int_0^{1-\eta} H_i(\xi, \eta) d\xi$$

$$= \frac{1}{2} \left( 1 + \frac{1}{\beta' a_\xi (1 - \eta)} \ln \left( \frac{\cosh(\beta'(a_\xi (1 - \eta) + a_\eta \eta + d))}{\cosh(\beta'(a_\eta \eta + d))} \right) \right),$$

then $d$ is determined from $\bar{H}_i^\xi(1/3) = \phi_i$ as

$$d = \frac{1}{2\beta'} \ln \left( \frac{1 - A Q}{A B (Q - A)} \right),$$

where,

$$A = \exp\left(\frac{2}{3} \beta' a_\xi\right), \quad B = \exp\left(\frac{2}{3} \beta' a_\eta\right), \quad Q = \exp\left(\frac{2}{3} \beta' (a_\xi (2\phi_i - 1))\right).\quad (A.2)$$

Case 2: $|a_\eta| = \max(|a_\xi|, |a_\eta|)$

The unknown $d$ is similarly determined from (A.3), but with the coefficients:

$$A = \exp\left(\frac{2}{3} \beta' a_\eta\right), \quad B = \exp\left(\frac{2}{3} \beta' a_\xi\right), \quad Q = \exp\left(\frac{2}{3} \beta' (a_\eta (2\phi_i - 1))\right).$$

(A.3)
Appendix B. Numerical integration of $H_i(\xi)$ on a cell surface

We employ the 2-point and 6-point Gauss quadrature [3] for a line edge of a 2D triangular mesh and a triangular surface of a 3D tetrahedral mesh respectively, for $\langle H_{up}/S_{ij} \rangle$ in (20). Here Gauss points $p_i$ in the local coordinate and weights $w_i$ on the cell surface $S_{ij}$ are given by

$$p_1 = \frac{1 - \sqrt{1/3}}{2}, \quad p_2 = \frac{1 + \sqrt{1/3}}{2}, \quad w_1 = w_2 = \frac{1}{2},$$

for the line edge of the triangular element, and

$$
p_1 = (a_1, a_1), \quad p_2 = (1 - 2a_1, a_1), \quad p_3 = (a_1, 1 - 2a_1), \quad p_4 = (a_2, a_2), \quad p_5 = (1 - 2a_2, a_2), \quad p_6 = (a_2, 1 - 2a_2),
$$

$$a_1 = \frac{8 - \sqrt{10 + \sqrt{38 - 44\sqrt{2}/5}}}{18}, \quad a_2 = \frac{8 - \sqrt{10 - \sqrt{38 - 44\sqrt{2}/5}}}{18},$$

$$w_1 = w_2 = w_3 = \frac{620 + \sqrt{213125 - 53320\sqrt{10}}}{3720},$$

$$w_4 = w_5 = w_6 = \frac{620 - \sqrt{213125 - 53320\sqrt{10}}}{3720},$$

for the triangular surface of the tetrahedral element.

References


