

A high order positivity-preserving conservative WENO remapping method on 2D quadrilateral meshes

Nuo Lei¹, Juan Cheng² and Chi-Wang Shu³

Abstract

In this paper, we present a [high order](#) accurate positivity-preserving conservative remapping algorithm which is based on the multi-resolution weighted essentially non-oscillatory (WENO) reconstruction. [We use a third-order method on 2D quadrilateral meshes as an example to present the algorithm.](#) The method can effectively remap the physical variables after mesh rezoning in the ALE algorithm. By calculating the intersection exactly, this method does not require the same connectivity between the old and new meshes. By reconstructing a quadratic polynomial and a zero-order polynomial for each cell in a two-dimensional domain, this method assigns nonlinear weights for these polynomials accordingly after calculating the smoothness indicators over the integration area, yielding [third order](#) accuracy without numerical oscillations. After calculating the overlaps between the old and new meshes and integrating the polynomials over the intersections, the remapping is completed. Furthermore, to ensure the positivity-preserving property of relevant physical variables in hydrodynamics numerical simulation such as density and internal energy, a simple and efficient positive-preserving limiter is adopted to slightly modify the reconstructed polynomials, which can maintain the original order of accuracy and conservation. [The algorithm can be extended to higher order accuracy using higher order reconstruction and higher order integration formula over the intersection areas.](#) A series of numerical experiments are performed to test the properties of the multi-resolution WENO conservative remapping algorithm. Numerical results show that the algorithm is conservative, positivity-preserving, highly efficient, third-order accurate for smooth problems, and essentially non-oscillatory for discontinuous problems.

Keywords: Remapping; High-order accuracy; Non-oscillatory; Positivity-preserving; Conservative; Multi-resolution WENO

¹Graduate School, China Academy of Engineering Physics, Beijing 100088, China. E-mail: leinuol19@gscaep.ac.cn.

²Corresponding author. Laboratory of Computational Physics, Institute of Applied Physics and Computational Mathematics, Beijing 100088, China and Center for Applied Physics and Technology, Peking University, Beijing 100871, China. E-mail: cheng_juan@iapcm.ac.cn. Research is supported in part by NSFC grants 12031001, 11871111 and U1630247, Science Challenge Project No. TZ2016002, and CAEP Foundation No. CX20200026.

³Division of Applied Mathematics, Brown University, Providence, RI 02912. E-mail: chi-wang_shu@brown.edu. Research is supported in part by NSF grants DMS-1719410 and DMS-2010107, and AFOSR grant FA9550-20-1-0055.

1 Introduction

The Lagrangian framework and Eulerian framework are two strategies in numerical simulations of fluid flow, both of them have advantages and disadvantages. Hirt et al. developed Arbitrary Lagrangian-Eulerian (ALE) framework which combines the best properties from the above two methods [16]. The typical indirect ALE method can be described in the following three steps:

- Lagrangian step: update the solution and move the mesh with the fluid flow;
- Rezoning step: move the nodes of the computational mesh to more optimal positions to ensure the quality of the mesh;
- Remapping step: conservatively transfer variables from the old distorted mesh to a new rezoned mesh or some arbitrarily defined mesh.

In this paper, we focus on the remapping step in the indirect ALE method. In computational fluid dynamics by the ALE method, physical variables are frequently transferred between the old and new meshes. The remapping method is required to keep the ALE method efficient, high order accurate, essentially non-oscillatory (without code crashing), positivity-preserving and other relevant properties. In numerical simulations of fluid flow, fluid variables should obey the physics laws such as the law of conservation of mass, momentum and total energy. Furthermore, some variables such as density and internal energy should be positive. Dukowicz et al. [10, 12] pointed out that the remapping procedure could be regarded as reconstructing physical variables in the old mesh then integrating them in the new mesh. In this context, conservative remapping means transferring the variables at the intersection region from the old mesh to the new mesh such that the total integrals over the old mesh and over the new mesh are equal. Based on this, there are two typical classes of methods, namely the flux-based (or swept-based) and the intersection-based remapping methods, to deal with this problem.

The flux-based remapping method utilizes the divergence theorem to convert the two-dimensional area integral to one-dimensional curve integrals, then estimates fluid variables exchanged with cells at their common interfaces [10, 34, 29, 23, 30, 27, 25]. By using the divergence theorem, this method greatly reduces the complexity of the problem for two-dimensional general quadrilateral meshes. However, it does require the connectivity between the old and the new meshes to be the same, and the movement of each node in the rezoning step should not exceed the size of its neighboring cells [34, 10]. The latter restriction could be relaxed by solving a fictitious advection equation to march from the old mesh to the new mesh via multiple pseudo time steps, while ensuring the restriction being enforced for each time step. Local ALE remapping methods such as [32, 11, 24] are constructed by adopting such advection algorithm which does not need to compute the intersections.

The intersection-based (or overlay-based) method is another typical remapping strategy [14, 12, 13, 31, 28]. It computes the overlaps between the old and new cells exactly, and then integrates the given fluid variable, reconstructed on the old mesh, over the intersection region exactly. For example, by constructing a supermesh, Menon et al. presented a second-order remapping algorithm, which is generally applicable to any polyhedral source or target meshes [13, 31]. [Powell et al. \[33, 36\] proposed a robust remeshing method to intersect two convex polyhedra in 3D.](#)

This type of methods has a wider application which includes structured or unstructured meshes, and it does not require the new mesh to have the same connectivity as the old one, hence the movement of the nodes can be arbitrary. Therefore, under the arbitrary moving mesh, the intersection-based method can easily achieve conservation and high-order accuracy. When a higher order accurate reconstruction is applied, the intersection-based method can get the corresponding higher order accuracy, since the error in the integration step can be ignored compared with the reconstruction error. However, the intersection-based method is relatively more costly, as it needs the explicit determination of the overlapping region between the old and new polygonal meshes in two-dimension or the old and new polyhedral

meshes in three-dimension. It is usually adopted when higher order of accuracy is desired or multi-material problem is involved or there is no connectivity between the old and the new meshes.

In the remapping process for multi-material ALE simulations, the authors from [1, 20, 21, 22, 18, 19] utilized the hybrid conservative remapping method and achieved second-order accuracy. In [18, 19], the authors compared and analyzed the local error and the behaviors between two different types of methods, the resource-intensive methods utilizing intersections and the faster and simpler flux-based methods. In the cells consisting of one material, a computationally inexpensive flux-based method was used, while for the cells consisting of multi-materials the intersection-based method was adopted. In this way, the method combined the advantages of the above two methods. But it would require the connectivity of the new mesh to be the same as that of the old mesh, and the mesh movement should not exceed the size of its neighboring cells, just like the flux-based methods.

In recent years, some higher order remapping algorithms have been developed. For example, Cheng and Shu [6] proposed a framework of conservative remapping algorithm which can be extended to arbitrary order of accuracy based on essentially non-oscillatory (ENO) reconstruction in one and two-dimensional cases. By solving an advection equation, Lipnikov and Morgan created a high-order remapping algorithm on curvilinear polygonal meshes using the discontinuous Galerkin schemes [24].

For the sake of ensuring the monotonicity property of variables, some bound-preserving remapping algorithms have been proposed. Based on the multidimensional positive definite advection transport algorithm (MPDATA), the authors in [29] proposed a second-order, sign-preserving remapping algorithm for scalar functions. By correcting high-order remapping fluxes with low-order fluxes [25] or redistributing the remapping results conservatively with the neighboring cells [23, 27], the remapping methods can achieve bound-preserving but may lose accuracy near smooth extrema. For high-order remapping algorithms, Burton et al. contributed a method for hydrodynamic fields associated with energy conservation,

entropy production, and bounds preservation [4]. By adding a posteriori multi-dimensional optimal order detection (MOOD) limiting for the high order accurate remapping method in [2], the remapping algorithm retains the physical properties and leads to robustness. The MOOD limiting was proposed in the limitation process of high-order finite volume methods which consists of detecting problematic cells and decrementing the degree of polynomials, in which way the accuracy may be reduced near the problematic cells [9].

In this paper, we will try to design a high order remapping algorithm which can maintain physical properties such as conservative and positivity-preserving without losing accuracy. In general, positivity-preserving is not easy to achieve without losing accuracy near smooth extrema or in problematic cells by using a limiter. Recently, Zhang and Shu proposed a high-order positivity-preserving limiter for finite volume and discontinuous Galerkin schemes [39, 40] which can preserve the positivity property of density, internal energy or pressure for compressible Euler equations. Meanwhile this limiter can keep the conservation and the original high order accuracy.

The performance of the reconstructed polynomial affects the final remapping results in the intersection-based method. For the sake of avoiding numerical oscillations and keeping high order accuracy, Cheng and Shu [6] adopted the ENO reconstruction proposed in [15]. In the ENO reconstruction, by comparing the smoothness of the candidate stencils, one can construct polynomials by the smoothest stencils, selectively avoiding stencils from containing large gradients and discontinuous points. After that, extensive progresses have been made for high order non-oscillatory reconstructions. Liu et al. [26] improved the process of selecting stencils by the ENO scheme, forming the weighted essentially non-oscillatory (WENO) scheme, by utilizing nonlinear weights and using the convex combination of all the candidate stencils. After that, Jiang and Shu [17] proposed a more effective and general WENO scheme by optimizing the calculation of the smoothness indicators, leading to better robustness and higher order accuracy. Recently, Zhu and Shu [41] designed a new class of WENO schemes on the quadrilateral meshes based on multi-resolution ideas, in which the

linear weights can be arbitrary and the stencil combination method is more efficient and simpler. They also extended this kind of WENO schemes to triangular meshes [42].

In this paper, following the intersection-based remapping method, we propose a third order accurate and positivity-preserving remapping algorithm based on the multi-resolution WENO reconstruction in two-dimension. First, we reconstruct polynomials on the old mesh by the multi-resolution WENO reconstruction. Then, the exact polygon clipping algorithm [35] is used to calculate the intersections. After that, a positivity-preserving limiter is added to modify the problematic polynomials. Finally, we integrate the reconstruction polynomials over the intersections in the new mesh. This remapping algorithm can satisfy the requirements of third order accuracy, non-oscillation, conservation, positivity-preserving, and high efficiency. Although we give a third order remapping algorithm on structured quadrilateral meshes as an example, our remapping algorithm is not limited to third order accuracy and structured meshes. It can achieve higher order accuracy with higher order reconstruction and integration formula, and can be extended to unstructured meshes by following the same reconstruction idea in [42]. Since our algorithm is intersection-based, it can ignore the connectivity between the old and new meshes, making it suitable for wider applications. Compared with other remapping algorithms, there are two highlights of our new algorithm. First, the new multi-resolution WENO procedure, which produces a WENO polynomial over each cell rather than just an approximation at quadrature points for classical WENO procedures, and which could use arbitrary positive linear weights provided that they sum to one, is particularly suitable for this remapping task which would need integration over partial cells and both the new and old meshes before and after remapping are subject to change from time step to time step in an ALE algorithm. Second, by adding a positivity-preserving limiter, intrinsic physical properties can be retained without losing accuracy and conservation.

The outline of this paper is as follows. Section 2 will describe the main procedures of the algorithm in details, by dividing it into four parts: reconstructing polynomials on the old mesh by the multi-resolution WENO reconstruction, calculating overlaps between the new

and old meshes, applying a positivity-preserving limiter, and integrating polynomials on the new mesh in the last step. Section 3 will give a series of numerical experiments, to verify the performance of the algorithm on several types of moving meshes and to compare the results with those obtained by the unlimited quadratic reconstruction polynomials and the ENO reconstruction algorithm. Furthermore, we apply our remapping algorithm in an indirect ALE method and show its performance on certain classical fluid flow benchmarks such as the Sedov, Saltzman and Noh problems. The final section contains further comments and conclusions.

2 A high-order accurate positivity-preserving conservative remapping algorithm based on multi-resolution WENO reconstruction

2.1 Basic concepts

The ALE method needs to transfer variables from the old distorted mesh $\mathcal{M}$ to a new rezoned mesh $\tilde{\mathcal{M}}$, which is called the remapping step. The variables for the fluid flow are usually from conservation laws, such as density, momentum, and total energy which should obey the physics laws. In the remapping step, we suppose that we have the information about the mesh $\mathcal{M}$, $\tilde{\mathcal{M}}$ and the cell-averages of the conserved variables on $\mathcal{M}$. Based on them, we will reconstruct polynomials on the old mesh and calculate the intersections between the old and new meshes. After that, a high-order integration formula will be adopted to generate the cell-averages of the corresponding variables on the new mesh.

This paper will focus on the high order accurate, conservative and positivity-preserving remapping method for the physical variables in two-dimension. For the one-dimensional or the three-dimensional cases, the strategy given in this paper is also applicable. Now, suppose Ω is a connected domain, for the sake of narration, we set it as $[0, 1] \times [0, 1]$. $\mathcal{M}$ is a structured mesh on Ω , consisting of quadrilateral cells $\{I_{i+1/2,j+1/2}\}$, $i = 1, 2, \dots, N_x, j = 1, 2, \dots, N_y$, where N_x and N_y are the number of cells in the x and y directions respectively. Each of the

quadrilateral cells $I_{i+1/2,j+1/2}$ has four nodes $\{P_{i,j}, P_{i+1,j}, P_{i+1,j+1}, P_{i,j+1}\}$ and the coordinate of the node $P_{i,j}$ is $(x_{i,j}, y_{i,j})$. Note that we use quadrilateral meshes as they are the typical choices in Lagrangian and ALE methods. Other types of structured or unstructured meshes can be used with the same recipe.

In the typical indirect ALE method, after the Lagrangian step and the rezoning step, the physical variables from the old mesh should be conservatively remapped to the new rezoned mesh $\tilde{\mathcal{M}}$. Here we do not require the meshes $\mathcal{M}$ and $\tilde{\mathcal{M}}$ to have the same connectivity and the same number of nodes and cells. The new rezoned mesh is noted as $\tilde{\mathcal{M}}$, consisting of quadrilateral cells $\{\tilde{I}_{k+1/2,l+1/2}\}, k = 1, 2, \dots, \tilde{N}_x, l = 1, 2, \dots, \tilde{N}_y$. Each cell has four nodes $\{\tilde{P}_{k,l}, \tilde{P}_{k+1,l}, \tilde{P}_{k+1,l+1}, \tilde{P}_{k,l+1}\}$. For both the old and the new meshes, there are neither overlaps nor gaps between any two neighboring cells, and we have

$$\bigcup_{k=1, l=1}^{\tilde{N}_x, \tilde{N}_y} \tilde{I}_{k+1/2, l+1/2} = \Omega = \bigcup_{i=1, j=1}^{N_x, N_y} I_{i+1/2, j+1/2}. \quad (2.1)$$

Figures 3.1 and 3.2 illustrate examples for the uniform mesh, smoothly moving meshes and randomly moving meshes in $[0, 1] \times [0, 1]$ respectively. Since both $\{I_{i+1/2, j+1/2}\}_{i=1, j=1}^{N_x, N_y}$ and $\{\tilde{I}_{k+1/2, l+1/2}\}_{k=1, l=1}^{\tilde{N}_x, \tilde{N}_y}$ can cover Ω without gaps or overlaps, each cell of $\tilde{\mathcal{M}}$ can be divided into the union of its intersections with cells in $\mathcal{M}$:

$$\tilde{I}_{k+1/2, l+1/2} = \bigcup_{i=1, j=1}^{N_x, N_y} (\tilde{I}_{k+1/2, l+1/2} \cap I_{i+1/2, j+1/2}). \quad (2.2)$$

2.2 A brief description of the positivity-preserving conservative WENO remapping algorithm

First, we take ‘density’ as an example to illustrate the framework of the conservative remapping algorithm. Other conservative variables such as ‘momentum’ and ‘total energy’ can follow the same recipe. Based on the notations above, the conservative remapping algorithm can be restated as “Given the old mesh $\mathcal{M}$, and the cell-average of density $\bar{\rho}_{i+1/2, j+1/2}$ in each cell, find the cell-average of density in each cell of the new mesh $\tilde{\mathcal{M}}$ ”. By reconstructing piecewise polynomials $\rho(x, y)$ on $\mathcal{M}$, and the intersections between meshes, a high order

integration formula can be applied to the intersection regions, yielding new cell-averages of density in $\tilde{\mathcal{M}}$. In order to ensure conservation, $\rho(x, y)$ should satisfy

$$\bar{\rho}_{i+1/2,j+1/2} = \iint_{I_{i+1/2,j+1/2}} \rho_{i+1/2,j+1/2}(x, y) dx dy / S_{i+1/2,j+1/2}, \quad (2.3)$$

where $\rho_{i+1/2,j+1/2}(x, y) = \rho(x, y)|_{I_{i+1/2,j+1/2}}$ and $S_{i+1/2,j+1/2}$ is the area of the cell $I_{i+1/2,j+1/2}$. The remapping algorithm is then automatically conservative. In fact, if we denote

$$\tilde{M}_{k+1/2,l+1/2} = \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \iint_{I_{k+1/2,l+1/2} \cap I_{i+1/2,j+1/2}} \rho_{i+1/2,j+1/2}(x, y) dx dy,$$

where $M_{i+1/2,j+1/2} = \bar{\rho}_{i+1/2,j+1/2} S_{i+1/2,j+1/2}$, $\tilde{M}_{k+1/2,l+1/2}$ are the mass of the cells $I_{i+1/2,j+1/2}$ and $\tilde{I}_{k+1/2,l+1/2}$, respectively, and if we ignore the machine error of the numerical integration and intersections which will be mentioned in Sections 2.4 and 2.5, we can prove that the total mass of $\tilde{\mathcal{M}}$ equals to the total mass of $\mathcal{M}$:

$$\begin{aligned} \sum_{k=1}^{\tilde{N}_x} \sum_{l=1}^{\tilde{N}_y} \tilde{M}_{k+1/2,l+1/2} &= \sum_{k=1}^{\tilde{N}_x} \sum_{l=1}^{\tilde{N}_y} \left(\sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \iint_{I_{k+1/2,l+1/2} \cap I_{i+1/2,j+1/2}} \rho_{i+1/2,j+1/2}(x, y) dx dy \right) \\ &= \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \left(\sum_{k=1}^{\tilde{N}_x} \sum_{l=1}^{\tilde{N}_y} \iint_{I_{k+1/2,l+1/2} \cap I_{i+1/2,j+1/2}} \rho_{i+1/2,j+1/2}(x, y) dx dy \right) \\ &= \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \iint_{I_{i+1/2,j+1/2}} \rho_{i+1/2,j+1/2}(x, y) dx dy \\ &= \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} M_{i+1/2,j+1/2}. \end{aligned}$$

If the intersections are exactly obtained and the numerical integration is also exact for the piecewise polynomials, the remapping error comes only from the reconstruction step.

Besides conservation, a remapping algorithm should also fulfill a few other properties such as ‘high-order accuracy’, ‘non-oscillatory’, ‘high efficiency’ and ‘positivity-preserving’. In particular, the polynomial reconstruction over a discontinuous function may generate numerical oscillations and ‘non-oscillatory’ means **no spurious numerical oscillations** appear

near discontinuities. ‘Positivity-preserving’ means the method can keep the positivity of the relevant physically positive variables such as density and internal energy in the fluid flow. To satisfy these desired properties, our scheme follows a four-step procedure:

Reconstruction: Reconstruct polynomials $\rho_{i+1/2,j+1/2}(x, y)$ in each cell of $\mathcal{M}$.

Intersection: By the polygon clipping algorithm [6, 35], compute the intersections

$$\tilde{I}_{k+1/2,l+1/2} \cap I_{i+1/2,j+1/2}$$

for each cell in $\tilde{\mathcal{M}}$.

Positivity-preserving limiter: Modify the reconstructed polynomial $\rho_{i+1/2,j+1/2}(x, y)$ by the designed positivity-preserving limiter, so that the modified polynomial $\rho_{i+1/2,j+1/2}^{(m)}(x, y)$ could preserve positivity while maintaining the original order of accuracy.

Integration: For each cell of $\tilde{\mathcal{M}}$, calculate the numerical integration over the intersections exactly by a numerical quadrature. Sum the integration in each intersection region of the cells in $\tilde{\mathcal{M}}$ and obtain the cell-averages $\tilde{\bar{\rho}}_{k+1/2,l+1/2}$ on $\tilde{\mathcal{M}}$.

2.3 Multi-resolution WENO reconstruction

For the first step of our remapping algorithm, we use the multi-resolution WENO reconstruction proposed by Zhu and Shu [41]. Based on the multi-resolution idea, they constructed a series of unequal-sized hierarchical central spatial stencils that can use arbitrary positive linear weights. Compared with the original WENO reconstruction, the multi-resolution WENO uses fewer stencils and arbitrary linear weights, but still achieves the desired high-order accuracy and non-oscillatory performance efficiently. Since we are aiming for third order accuracy in this paper as an example, according to [41], we choose a large and a small stencil to construct a quadratic polynomial and a zeroth degree polynomial respectively, and perform a convex combination of them.

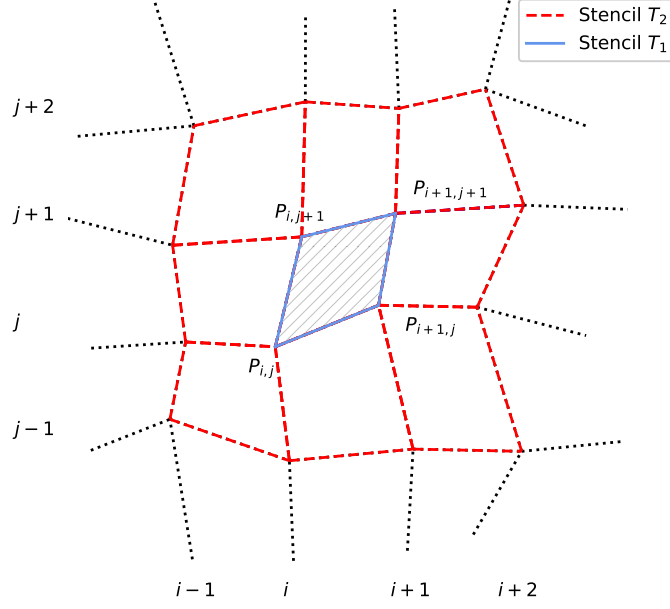


Figure 2.1: Two types of central stencils for $I_{i+1/2,j+1/2}$. The black dotted lines indicate the old mesh $\mathcal{M}$, the blue solid lines indicate the small stencil T_1 and the red dashed lines indicate the big stencil T_2 .

To be more specific, we select $T_1 = \{I_{i+1/2,j+1/2}\}$ as the small stencil and T_2 as the large stencil, where

$$T_2 = \{I_{i-1/2,j-1/2}, I_{i-1/2,j+1/2}, I_{i-1/2,j+3/2}, I_{i+1/2,j-1/2}, I_{i+1/2,j+1/2}, \\ I_{i+1/2,j+3/2}, I_{i+3/2,j-1/2}, I_{i+3/2,j+1/2}, I_{i+3/2,j+3/2}\}.$$

Figure 2.1 shows these two stencils for the cell $I_{i+1/2,j+1/2}$. The reconstruction procedure below is the third-order multi-resolution WENO reconstruction in two-dimension. Following a similar procedure, a higher order reconstruction scheme can be extended [41, 42]. Take density in the fluid flows as an example. Suppose we are given the cell-averages of density in (2.3) for all the cells in $\mathcal{M}$, then the procedure is summarized as follows. We will ignore boundary effects and assume periodic boundary conditions in the reconstruction procedure for a simple description. In fact, for the cells near the boundary, they can be treated in the same way as the inner cells when the corresponding variables in the ghost cells outside the computational domain are given by the boundary conditions such as periodic, reflective, piston and so on.

1. Polynomial reconstruction

For the sake of clarity, we omit the superscripts i, j in $q_1^{i,j}(x, y)$, $q_2^{i,j}(x, y)$ which indicate the reconstruction polynomials in the cell $I_{i+1/2,j+1/2}$. Without loss of generality, all the symbols shown below refer to the cell $I_{i+1/2,j+1/2}$.

Reconstruct the zero degree polynomial $q_1(x, y) = c$ and the second-degree polynomial $q_2(x, y) = a_0(x - x_{i,j}^c)^2 + a_1(x - x_{i,j}^c)(y - y_{i,j}^c) + a_2(y - y_{i,j}^c)^2 + a_3(x - x_{i,j}^c) + a_4(y - y_{i,j}^c) + a_5$, where $(x_{i,j}^c, y_{i,j}^c)$ is the center of $I_{i+1/2,j+1/2}$, which satisfy:

$$\iint_{I_{i+1/2,j+1/2}} q_s(x, y) dx dy = \bar{\rho}_{i+1/2,j+1/2} S_{i+1/2,j+1/2}, \quad s = 1, 2. \quad (2.4)$$

It is easy to reconstruct $q_1(x, y) = \bar{\rho}_{i+1/2,j+1/2}$. $q_2(x, y)$ has six degrees of freedom, but the large stencil has nine cells, so we adopt a constraint least-square procedure to determine $\{a_i, i = 0, 1, \dots, 5\}$:

$$\begin{aligned} q_2(x, y) = \arg \min_{\tilde{q}_2 \in Q} \sum_{m,n} & \left| \iint_{I_{m+1/2,n+1/2}} \tilde{q}_2(x, y) dx dy - \bar{\rho}_{m+1/2,n+1/2} S_{m+1/2,n+1/2} \right|^2, \\ \text{s.t.} \quad & \iint_{I_{i+1/2,j+1/2}} \tilde{q}_2(x, y) dx dy = \bar{\rho}_{i+1/2,j+1/2} S_{i+1/2,j+1/2}, \end{aligned} \quad (2.5)$$

where the sum over (m, n) denotes the sum over the nine cells in the stencil T_2 and the set $Q = \{\tilde{q}_2(x, y) \in P^2\}$ (polynomials of degree at most 2).

2. Linear weights for the reconstruction polynomials

We take $p_1(x, y) = q_1(x, y)$, and

$$p_2(x, y) = \frac{1}{\gamma_2} q_2(x, y) - \frac{\gamma_1}{\gamma_2} p_1(x, y),$$

where $\gamma_1 + \gamma_2 = 1$, $\gamma_2 \neq 0$ are the linear weights. Following [41], we assume $\gamma_l = \frac{\bar{\gamma}_l}{\bar{\gamma}_1 + \bar{\gamma}_2}$, where $\bar{\gamma}_l = 10^{l-1}$, $l = 1, 2$, so we have $\gamma_1 = \frac{1}{11}, \gamma_2 = \frac{10}{11}$. By combining $p_2(x, y)$ and $p_1(x, y)$ with the linear weights, optimal accuracy can be achieved

$$\gamma_1 p_1(x, y) + \gamma_2 p_2(x, y) = q_2(x, y).$$

These linear weights can maintain good accuracy in the smooth region. But near the discontinuity, the quadratic polynomial $q_2(x, y)$ may produce oscillations. Therefore, the WENO idea should be adopted, which can not only maintain the third-order accuracy in the smooth region but also avoid oscillations near the discontinuities.

3. Smoothness indicator

The designed smoothness indicators β_1, β_2 can avoid the numerical oscillation of the polynomial near the discontinuity.

$$\beta_2 = \sum_{l_1+l_2=1, 2} \iint_{I_{i+1/2, j+1/2}} S_{i+1/2, j+1/2}^{|l_1+l_2|-1} \left(\frac{\partial^{l_1+l_2}}{\partial x^{l_1} \partial y^{l_2}} p_2(x, y) \right)^2 dx dy, \quad l_1, l_2 = 0, 1, 2. \quad (2.6)$$

As the smoothness indicator of the zeroth degree polynomial $p_1(x, y)$ obtained by (2.6) would be zero, an alternative definition of β_1 is proposed in [41]:

$$\begin{aligned} \zeta_1 &= (\bar{\rho}_{i+1/2, j-1/2} - \bar{\rho}_{i+1/2, j+1/2})^2, \quad \zeta_2 = (\bar{\rho}_{i-1/2, j+1/2} - \bar{\rho}_{i+1/2, j+1/2})^2, \\ \zeta_3 &= (\bar{\rho}_{i+1/2, j+3/2} - \bar{\rho}_{i+1/2, j+1/2})^2, \quad \zeta_4 = (\bar{\rho}_{i+3/2, j+1/2} - \bar{\rho}_{i+1/2, j+1/2})^2. \end{aligned}$$

β_1 is calculated as

$$\beta_1 = \min\{\zeta_1 + \zeta_2, \zeta_2 + \zeta_3, \zeta_3 + \zeta_4, \zeta_4 + \zeta_1\}.$$

If a shock does not pass the central target cell $I_{i+1/2, j+1/2}$, then at least one of the above 4 quadrants should be smooth, hence the minimum of these 4 smoothness indicators should be small, indicating that β_1 is small.

4. Nonlinear weights

According to the linear weights and smoothness indicators, we obtain the nonlinear weights by [5]

$$\omega_l = \frac{\bar{\omega}_l}{\bar{\omega}_1 + \bar{\omega}_2}, \quad l = 1, 2, \quad (2.7)$$

where

$$\bar{\omega}_l = \gamma_l \left(1 + \frac{\tau}{\beta_l + \varepsilon} \right), \quad \tau = \left(\frac{|\beta_2 - \beta_1|}{2} \right)^2, \quad l = 1, 2, \quad (2.8)$$

and we take $\varepsilon = 10^{-4}$ in our computation.

5. Convex combination

The final reconstruction polynomial for the old cell $I_{i+1/2,j+1/2}$ is given as follows,

$$\rho_{i+1/2,j+1/2}(x, y) = \omega_1 p_1(x, y) + \omega_2 p_2(x, y). \quad (2.9)$$

Generally speaking, the size of the smooth indicators β_1, β_2 , have the same magnitude in smooth regions, yielding ω_l close to the optimal linear weights γ_l for $l = 1, 2$. In the meantime, if the large stencil T_2 contains a discontinuity and the small stencil T_1 does not, we will have $\beta_2 \gg \beta_1$ and the nonlinear weights $\omega_2 \ll \omega_1$, making $p_2(x, y)$ contributing little to the final polynomial and hence achieving the effect of essential non-oscillation. For illustration, we will show $\beta_1, \beta_2, \omega_1, \omega_2$ for the smooth test function in Figure 3.3 and for the discontinuous test function in Figure 3.5.

2.4 Intersections between the old and new meshes

After reconstructing the polynomials on the old mesh, the process of remapping the conserved variables of the old mesh to the new mesh requires the calculation of the intersections. Although compared with the flux-based method, its costs are relatively high, the error of an exact polygon clipping algorithm or the construction of a super-mesh [13, 31] is close to machine zero and the intersection-based method can be applied for meshes with different connectivity or different number of cells, which makes the intersection-based method more widely applicable.

In this paper we will use the Sutherland-Hodgman polygon clipping algorithm [35] to determine the intersections between the old and new meshes. By setting visible and invisible sides, the algorithm divides the polygon into several parts and only polygons in the visible side are what we need. Utilizing this exact clipping algorithm, the error from the intersection step is close to machine zero.

The main component for the Sutherland-Hodgman polygon clipping algorithm is to find the intersection points between two polygons. Figure 2.2 gives an example to show how this

clipping algorithm works. In the figure, we use 9 red quads clip against the blue quad and all of the intersections have been labeled by black solid lines. The Sutherland-Hodgman polygon clipping algorithm is exact theoretically. Although there are still some clipping errors from the calculation of intersection points, the clipping error associates with the machine precision which could be neglected especially compared with the errors from the WENO reconstruction scheme.

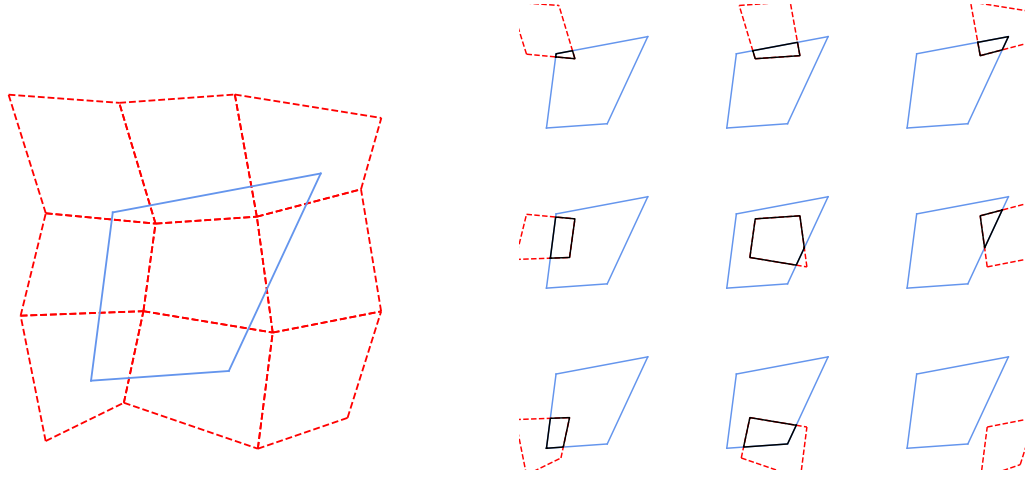


Figure 2.2: Example for the Sutherland-Hodgman polygon clipping algorithm. The cells on the old mesh are labeled in red, the cell on the new mesh is labeled in blue and the intersections are labeled in black.

2.5 Conservative remapping

After obtaining the reconstructed polynomial by the above steps and the intersections between the old and new meshes, the integration of the polynomial on the new mesh could be calculated. Since we provide a second-order polynomial in two-dimension as an example, no matter what the overlapping is, it can be divided into several triangles and the following integration formula (2.10) can be used,

$$\iint_{\Delta} \rho_{i+1/2,j+1/2}(x,y) dx dy = \frac{S_{\Delta}}{3} \sum_{k=1}^3 \rho_{i+1/2,j+1/2}(P_k), \quad (2.10)$$

where P_k are the middle points of the three edges of the triangle. The numerical integration formula (2.10) is exact for second-order polynomials in two-dimension and then the cell-

average of density in the new mesh can be given as:

$$\tilde{\rho}_{k+1/2,l+1/2} = \frac{1}{\tilde{S}_{k+1/2,l+1/2}} \sum_{i=1,j=1}^{N_x,N_y} \iint_{I_{k+1/2,l+1/2} \cap I_{i+1/2,j+1/2}} \rho_{i+1/2,j+1/2}(x,y) dx dy. \quad (2.11)$$

In this way, the total mass of the domain Ω is [conserved](#).

2.6 Positivity-preserving limiter

Some physical variables such as density and internal energy should always be positive for a fluid flow problem, thus the results of the remapping step should keep this property. In the simulation of fluid flow, all the conservative variables such as momentum (m, n) and total energy (E) besides density should also be remapped. During the remapping cycle, density and internal energy $e = E - \frac{m^2+n^2}{2\rho}$ may become negative, which is not physical. Define $\mathbf{w} = (\rho, m, n, E)^T$ and $G = \{\mathbf{w} | \rho > 0, e \geq 0\}$ which can be proven to be a convex set. Following the previous notations, $\bar{\mathbf{w}} = (\bar{\rho}, \bar{m}, \bar{n}, \bar{E})^T$ is the cell-average of $\mathbf{w}$ on the old mesh. The positivity-preserving of density and internal energy means when we input $\bar{\mathbf{w}} \in G$ into the remapping algorithm, then we get $\tilde{\mathbf{w}} \in G$, where $\tilde{\mathbf{w}}$ is the cell-averages of $\check{\mathbf{w}}$ on the new mesh.

As we further want to design our remapping algorithm to have positivity-preserving property, before the procedure of conservative remapping introduced in the above subsection, we will try to modulate the reconstruction polynomials in each cell $I_{i+1/2,j+1/2}$

$$\mathbf{w}_{i+1/2,j+1/2}(x,y) = (\rho_{i+1/2,j+1/2}(x,y), m_{i+1/2,j+1/2}(x,y), n_{i+1/2,j+1/2}(x,y), E_{i+1/2,j+1/2}(x,y))^T,$$

to obtain the new polynomials $\check{\mathbf{w}}_{i+1/2,j+1/2}(x,y) \in G$ at specific quadrature points to be described below, such that $\tilde{\mathbf{w}} \in G$ will hold. In the meantime, we require that

$$\iint_{I_{i+1/2,j+1/2}} \mathbf{w}_{i+1/2,j+1/2}(x,y) dx dy = \iint_{I_{i+1/2,j+1/2}} \check{\mathbf{w}}_{i+1/2,j+1/2}(x,y) dx dy,$$

so this process can preserve conservation. As we have mentioned above, after the reconstructing and clipping steps, we need to calculate the integration over the intersections.

Since we only need to ensure positivity of density and internal energy for the integrals of the conservative variables over the intersections, if we guarantee the values of density and internal energy at all the quadrature points used in the numerical integration formula (2.10) to be positive, then we can achieve our goals, due to the convexity of the set G . We note that the quadrature points in our remapping algorithm are the middle points of the triangle edges in the intersections, according to (2.10).

Following [37, 39], the positivity-preserving limiter for density and internal energy is split into two steps. For simplicity, we ignore the subscript i, j and we denote the set of quadrature points in the cell $I_{i+1/2, j+1/2}$ by $Ga^{(I)}$, which consists of all the middle points of the triangle edges for all the intersection triangles in $I_{i+1/2, j+1/2}$.

1. *Positivity-preserving of density $\rho(x, y)$.*

After the reconstruction step, we obtain the WENO polynomial vector $\mathbf{w}(x, y) = (\rho(x, y), m(x, y), n(x, y), E(x, y))^T$, and its cell-average vector $\bar{\mathbf{w}} = (\bar{\rho}, \bar{m}, \bar{n}, \bar{E})^T \in G$. We modify the reconstructed quadratic polynomial $\rho(x, y)$ by the following formula,

$$\begin{aligned} \hat{\rho}(x, y) &= \theta_1(\rho(x, y) - \bar{\rho}) + \bar{\rho}, \\ \theta_1 &= \min\{1, |\frac{\bar{\rho} - \varepsilon}{\bar{\rho} - b}|\}, \quad b = \min_{(x, y) \in Ga^{(I)}} \rho(x, y), \end{aligned} \quad (2.12)$$

where ε is a very small positive constant which satisfies $\bar{\rho} \geq \varepsilon$ (if $\bar{\rho} < \varepsilon$ in a certain cell, then the limiter returns the cell average, namely $\theta_1 = 0$). In our code, we take $\varepsilon = 10^{-13}$.

Remark 2.1. *By modifying the polynomial $\rho(x, y)$ through (2.12), the values of $\hat{\rho}(x, y)$ at the integration points $(x, y) \in Ga^{(I)}$ are positive. Therefore, the cell average of density obtained by the values of modified polynomials at these integration points through the formula (2.11) are positive.*

Proof. Since $b = \min_{(x, y) \in Ga^{(I)}} \rho(x, y)$, there will be two cases.

(a) If $b < \varepsilon = 10^{-13}$, then $\theta_1 = \min\{1, |\frac{\bar{\rho}-\varepsilon}{\bar{\rho}-b}|\} = \frac{\bar{\rho}-\varepsilon}{\bar{\rho}-b}$. We have

$$\begin{aligned}\hat{\rho}(x, y) &= \theta_1 \rho(x, y) + (1 - \theta_1) \bar{\rho} \\ &= \frac{\bar{\rho} - \varepsilon}{\bar{\rho} - b} \rho(x, y) + \frac{\varepsilon - b}{\bar{\rho} - b} \bar{\rho} \\ &\geq \frac{\bar{\rho} - \varepsilon}{\bar{\rho} - b} b + \frac{\varepsilon - b}{\bar{\rho} - b} \bar{\rho} \\ &= \varepsilon = 10^{-13}, \quad (x, y) \in Ga^{(I)}.\end{aligned}$$

(b) If $b \geq \varepsilon = 10^{-13}$, then we have $\theta_1 = 1$ and $\hat{\rho}(x, y) = \rho(x, y) \geq 10^{-13}$ for all $(x, y) \in Ga^{(I)}$.

□

2. Positivity-preserving of internal energy.

After the first step, we define $\hat{\mathbf{w}}(x, y) = (\hat{\rho}(x, y), m(x, y), n(x, y), E(x, y))^T$, and the internal energy is calculated by $e(\hat{\mathbf{w}}(x, y)) = E(x, y) - \frac{m^2(x, y) + n^2(x, y)}{2\hat{\rho}(x, y)}$, so the limited polynomials are given by:

$$\check{\mathbf{w}}(x, y) = \theta_2(\hat{\mathbf{w}}(x, y) - \bar{\mathbf{w}}) + \bar{\mathbf{w}}, \quad \theta_2 = \min_{(x, y) \in Ga^{(I)}} \frac{e(\bar{\mathbf{w}})}{e(\bar{\mathbf{w}}) - e(\hat{\mathbf{w}}(x, y))}. \quad (2.13)$$

After being modified by θ_2 , $\check{\mathbf{w}}(x, y) \in G$, for all $(x, y) \in Ga^{(I)}$ which can be proven in the similar way as above. Thus we know the cell average of density and internal energy on the new mesh obtained by $\check{\mathbf{w}}(x, y)$ are positive.

Notice that the cost of this positivity-preserving limiter is minimal, since we only need to compute values of the reconstructed polynomials at a few numerical quadrature points in $Ga^{(I)}$. This limiter can keep conservation and accuracy (for a proof of accuracy maintenance, see [38]), which will be verified in the numerical experiments in the next section.

So far, the main procedure of the multi-resolution WENO positivity-preserving high-order conservative remapping algorithm is accomplished step by step, including polynomial reconstruction, intersection, positivity-preserving limiter, and the conservative remapping. In the next section, the accuracy of the algorithm will be verified through numerical experiments,

and its non-oscillatory property will be verified for discontinuous profiles. At the same time, the positivity-preserving property will be tested on several examples. Furthermore, we apply our remapping algorithm in an indirect ALE method and show its performance on certain classical fluid flow benchmarks such as the Sedov, Saltzman and Noh problems. Finally, the runtime efficiency will be compared with the ENO remapping algorithm with the same accuracy.

3 Numerical tests

In this section, a series of numerical examples will be performed to verify the conservative, high order accurate, positivity-preserving, and non-oscillatory natures of the algorithm. At the same time, the numerical results of different conservative remapping algorithms will be given for comparison. Suppose $\mathcal{M}^0$ is a uniform mesh with edges parallel to the axes. Divide by $\Delta x, \Delta y$ equidistantly along the x, y axes, and $0 = x_1 < x_2 < \cdots < x_{N_x+1} = 1$, $0 = y_1 < y_2 < \cdots < y_{N_y+1} = 1$, so the vertices of the mesh are denoted by $(x_i, y_j) = ((i-1)\Delta x, (j-1)\Delta y)$. To mimic the distorted mesh yielding from the ALE method, we generate the smoothly moving meshes and randomly moving meshes from the uniform mesh $\mathcal{M}^0$. Here just for simplicity, we suppose that the old and new meshes have the same number of nodes and cells.

- The smoothly moving mesh $\mathcal{M}_S^n$

$$\begin{aligned} x_{i,j}^n &= x_i + c_M \frac{n}{N} \sin(2\pi x_i) \sin(2\pi y_j), \\ y_{i,j}^n &= y_j + c_M \frac{n}{N} \sin(2\pi x_i) \sin(2\pi y_j), \end{aligned} \quad (3.1)$$

where $c_M = 0.1$, the superscript n refers to the times of remapping and N is the total number of remapping steps. After remapping $n = N$ times, the smoothly moving mesh $\mathcal{M}_S^n$ will be back to the original one $\mathcal{M}_S^N = \mathcal{M}^0$, and the schematic diagrams are given by Figure 3.1.

- The randomly moving mesh $\mathcal{M}_R^n$

$$\begin{aligned} x_{i,j}^n &= x_i + c_R r_{i,j}^n \Delta x, \\ y_{i,j}^n &= y_j + c_R s_{i,j}^n \Delta y, \end{aligned} \quad (3.2)$$

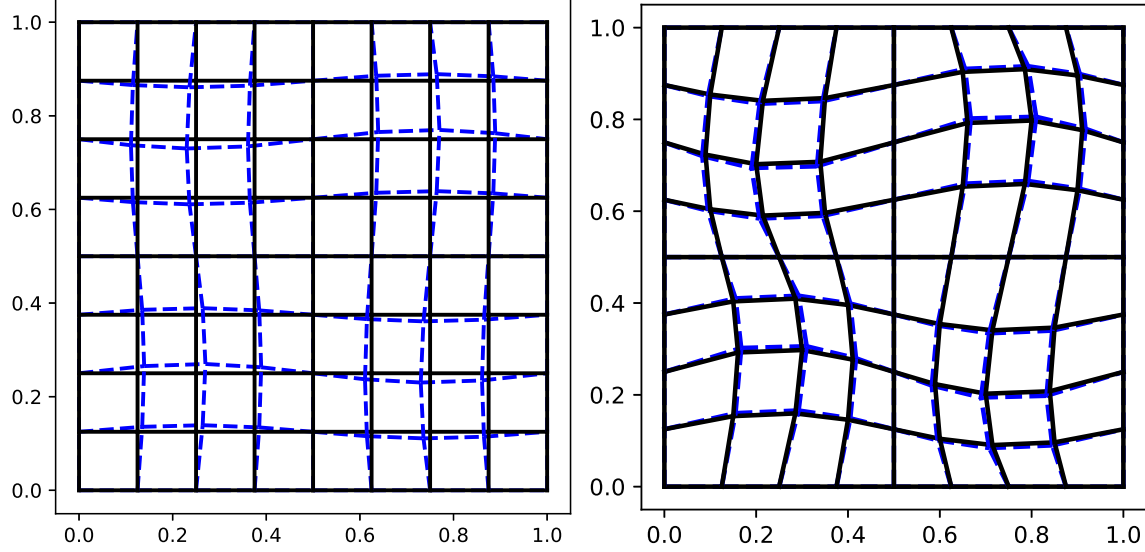


Figure 3.1: Schematic diagrams of the uniform mesh and the smoothly moving mesh, $N_x = N_y = 8$, $N = 10$. Left: the black solid mesh is the uniform mesh $\mathcal{M}^0$ and the blue dashed mesh is the smoothly moving mesh $\mathcal{M}_S^1$; Right: the black solid mesh is the smoothly moving mesh $\mathcal{M}_S^5$ and the blue dashed mesh is the smoothly moving mesh $\mathcal{M}_S^6$.

where $r_{i,j}^n, s_{i,j}^n$ are the random numbers between -0.5 and 0.5 . The randomly moving meshes are obtained by adding random perturbations to inner vertices of the uniform mesh. In the later experiments, we take $c_R = 0.5$ and Figure 3.2 shows the schematic diagrams of “uniform mesh $\mathcal{M}^0$ to randomly moving mesh $\mathcal{M}_R^1$ ” and “randomly moving mesh $\mathcal{M}_R^5$ to randomly moving mesh $\mathcal{M}_R^6$ ”.

In the following numerical experiments, the remapping step starts from a uniform mesh and uses conservative remapping algorithms to remap physical variables from the uniform mesh $\mathcal{M}^0$ to the smoothly moving mesh $\mathcal{M}_S^1$ or the randomly moving mesh $\mathcal{M}_R^1$, and then to the next mesh, until the remapping is performed N times. For the sake of verifying the accuracy, after the final remapping step (Step N), the mesh will be back to $\mathcal{M}^0$, hence the accuracy of the algorithm can be easily verified. The errors from the clipping step [can be ignored compared with the errors from the reconstruction step](#), and the integration formula (2.10) is exact for the third-order algorithm, so the error of this remapping algorithm is mainly determined by the reconstruction step and the positivity-preserving step.

Suppose the initial cell-averages of density in the cell $I_{i+1/2,j+1/2}$ is $\bar{\rho}_{i+1/2,j+1/2}^0$ and

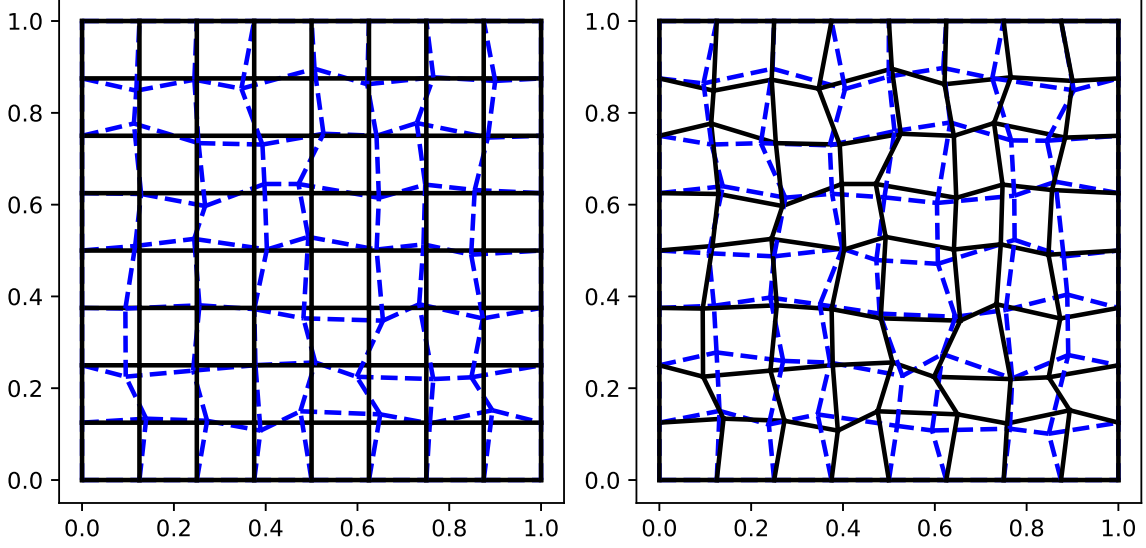


Figure 3.2: Schematic diagrams of the uniform mesh and the randomly moving mesh, $N_x = N_y = 8$, $N = 10$. Left: the black solid mesh is the uniform mesh $\mathcal{M}^0$ and the blue dashed mesh is the randomly moving mesh $\mathcal{M}_R^1$; Right: the black solid mesh is the randomly moving mesh $\mathcal{M}_R^5$ and the blue dashed mesh is the randomly moving mesh $\mathcal{M}_R^6$.

$\tilde{\rho}_{i+1/2,j+1/2}^N$ is the cell-average of density after remapping for N times in the same cell, the norms of the error ϵ are given by

$$\begin{aligned} \|\epsilon\|_1 &= \frac{\sum_{i=1,j=1}^{N_x,N_y} |(\tilde{\rho}_{i+1/2,j+1/2}^N - \bar{\rho}_{i+1/2,j+1/2}^0) S_{i+1/2,j+1/2}|}{\sum_{i=1,j=1}^{N_x,N_y} S_{i+1/2,j+1/2}}, \\ \|\epsilon\|_\infty &= \max_{1 \leq i \leq N_x, 1 \leq j \leq N_y} |\tilde{\rho}_{i+1/2,j+1/2}^N - \bar{\rho}_{i+1/2,j+1/2}^0|. \end{aligned} \quad (3.3)$$

3.1 Accuracy tests

To verify the accuracy of the algorithm, we compare the performance of the following four third order remapping algorithms: the unlimited polynomial reconstruction $q_2(x, y)$ given by the formula (2.5) denoted as P2, the ENO remapping algorithm [6], the multi-resolution WENO remapping algorithm without the positivity-preserving limiter (WENO) and the multi-resolution WENO remapping algorithm with the positivity-preserving limiter (P-WENO). Under different sizes of the meshes, we remap the physical variables on the smoothly moving meshes and the randomly moving ones, then calculate the obtained errors in the L^1 and L^∞ norms respectively. First we consider a test on the following smooth

function with periodic boundary condition

$$\rho(x, y) = \sin^2(2\pi x) \sin^2(2\pi y). \quad (3.4)$$

Here we give two versions of positivity-preserving limiter noted as ‘P-WENO-A’ and ‘P-WENO-B’ respectively. The first version (version A) is applying the limiter as mentioned above, namely the limiter is applied whenever some of the values at the integration quadrature points inside an old cell are negative. The second version (version B) is to apply the limiter in an old cell only when at least one of the cell averages of a new cell which intersects with this old cell is negative after reconstruction and integration. Clearly, the second version is less restrictive in applying the limiter, hence can be expected to be more efficient. We will compare the cost of these two versions in Section 3.5.

Table 3.1: Error and order of the P2, ENO, WENO and P-WENO remapping algorithms on the 2D smoothly moving meshes, $N = 10$.

	P2		ENO		WENO		P-WENO-A		P-WENO-B	
Mesh	L^1		L^1		L^1		L^1		L^1	
8×8	1.69E-2		1.73E-2		3.98E-2		3.98E-2		3.98E-2	
16×16	2.77E-3	2.61	2.83E-3	2.61	4.16E-3	3.26	4.16E-3	3.26	4.16E-3	3.26
32×32	2.84E-4	3.29	2.86E-4	3.31	2.84E-4	3.87	2.84E-4	3.87	2.84E-4	3.87
64×64	2.11E-5	3.75	2.14E-5	3.74	2.11E-5	3.75	2.11E-5	3.75	2.11E-5	3.75
Mesh	L^∞		L^∞		L^∞		L^∞		L^∞	
8×8	3.54E-2		3.59E-2		9.04E-2		9.04E-2		9.04E-2	
16×16	1.25E-2	1.50	1.31E-2	1.45	3.76E-2	1.26	3.76E-2	1.26	3.76E-2	1.26
32×32	1.10E-3	3.51	1.25E-3	3.40	1.28E-3	4.88	1.28E-3	4.88	1.28E-3	4.88
64×64	6.53E-5	4.07	6.89E-5	4.18	6.53E-5	4.29	6.53E-5	4.29	6.53E-5	4.29

Table 3.1 and Table 3.2 show the remapping results. Table 3.3 shows the percentage of the cells with negative integration points noted as N_C ,

$$N_C = \frac{N_{PP}}{N \times N_x \times N_y},$$

where N_{PP} is the total number of the cells with negative integration points during the remapping. From Table 3.1 and Table 3.2, we can see all the four remapping algorithms reach third-order accuracy on both types of meshes. As expected, the P2 remapping algorithm

Table 3.2: Error and order of the P2, ENO, WENO and P-WENO remapping algorithms on the 2D randomly moving mesh, $N = 10$.

	P2		ENO		WENO		P-WENO-A		P-WENO-B	
Mesh	L^1		L^1		L^1		L^1		L^1	
8×8	1.59E-2		1.60E-2		3.61E-2		3.61E-2		3.61E-2	
16×16	1.86E-3	3.09	1.97E-3	3.01	3.48E-3	3.37	3.49E-3	3.37	3.48E-3	3.37
32×32	1.63E-4	3.51	1.78E-4	3.47	1.68E-4	4.38	1.69E-4	4.37	1.68E-4	4.38
64×64	1.87E-5	3.12	2.07E-5	3.11	1.88E-5	3.16	1.90E-5	3.15	1.88E-5	3.16
Mesh	L^∞		L^∞		L^∞		L^∞		L^∞	
8×8	5.14E-2		5.15E-2		1.14E-1		1.14E-1		1.14E-1	
16×16	7.25E-3	2.83	7.99E-3	2.69	2.25E-2	2.34	2.25E-2	2.34	2.25E-2	2.34
32×32	1.09E-3	2.73	1.32E-3	2.60	1.10E-3	4.36	1.10E-3	4.36	1.10E-3	4.36
64×64	1.31E-4	3.06	1.42E-4	3.21	1.31E-4	3.07	1.31E-4	3.07	1.31E-4	3.07

Table 3.3: Percentage of cells with negative integration points (noted as Nc) obtained by the P2, ENO, WENO and P-WENO remapping algorithms on the 2D smoothly moving mesh and randomly moving meshes, $N = 10$.

	$Nc(\%)$ smoothly moving meshes				$Nc(\%)$ randomly moving meshes			
	8×8	16×16	32×32	64×64	8×8	16×16	32×32	64×64
P2	0	7.23	5.47	3.81	0	9.38	8.45	5.53
ENO	0	7.03	5.47	3.76	0	16.21	11.58	6.42
WENO	0	6.84	5.27	3.81	0	8.59	8.15	5.53
P-WENO-A	0	0	0	0	0	0	0	0
P-WENO-B	0	6.84	5.27	3.81	0	8.59	8.15	5.53

has the smallest error for this smooth problem. There is only a little difference of the L^1 , L^∞ norms of errors between the WENO and P-WENO remapping algorithms. After adding the positivity-preserving (noted as *PP* below) limiter, the P-WENO remapping algorithm can still preserve third-order accuracy. The error of the WENO and P-WENO remapping algorithms becomes closer to the results of the P2 remapping algorithm with the increase of the mesh size. From the above two tables, we observe that the accuracy of these two P-WENO versions are nearly the same.

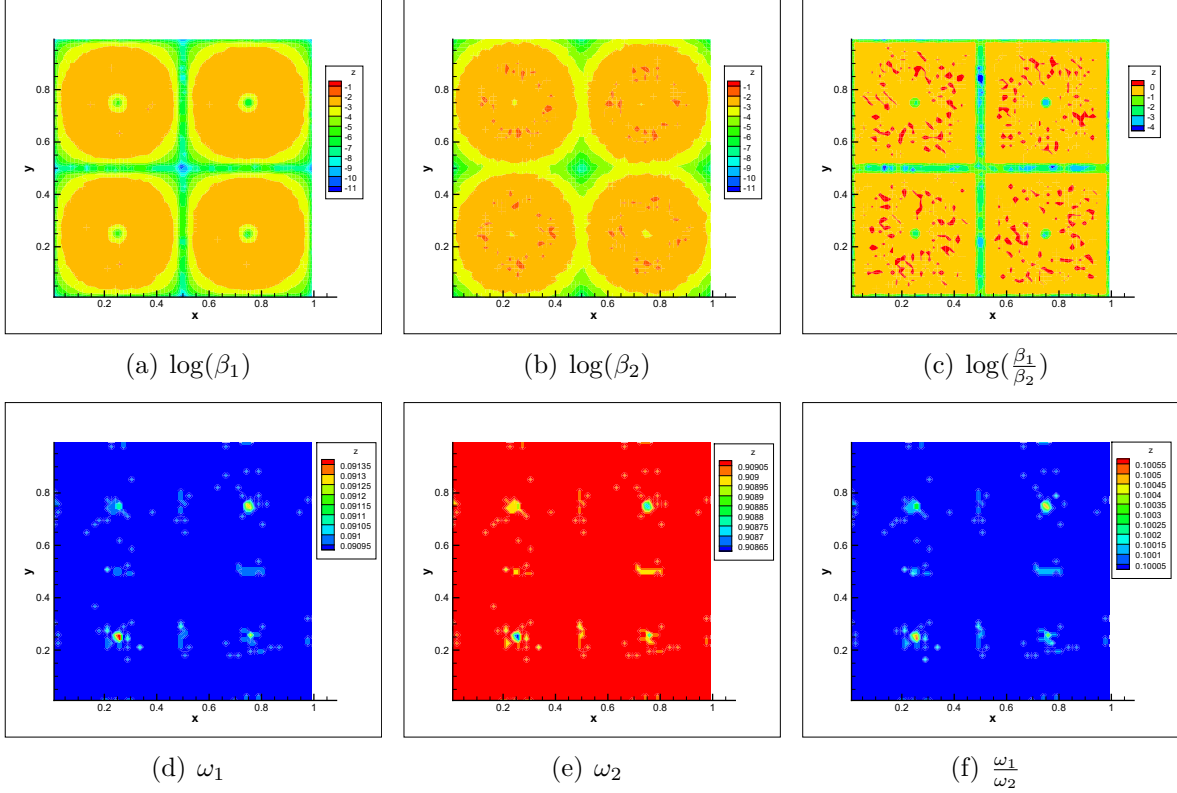


Figure 3.3: The parameters $\beta_1, \beta_2, \omega_1, \omega_2$ in the reconstruction step of the third-order WENO remapping algorithm in the accuracy test, $N_x = N_y = 64$.

In Figure 3.3, we present the smoothness indicators β_1, β_2 and the nonlinear weights ω_1, ω_2 in the WENO reconstruction step for this smooth test function. From the top figures of Figure 3.3, we can see β_1 and β_2 are of the same magnitude of sizes. From the bottom three figures of Figure 3.3, we can observe $\omega_1 \approx \gamma_1, \omega_2 \approx \gamma_2$, therefore, the final reconstruction polynomial $\rho_{i+1/2, j+1/2}(x, y) = \omega_1 p_1(x, y) + \omega_2 p_2(x, y) \approx \gamma_1 p_1(x, y) + \gamma_2 p_2(x, y) = q_2(x, y)$,

which makes it approximate the optimal third order accuracy.

Next we consider the following system (3.5) in $[0, 1] \times [0, 1]$, and compare the performance of the WENO and P-WENO remapping algorithms under the smoothly moving mesh

$$\begin{cases} \rho(x, y) &= e^{-25[(x-0.5)^2(y-0.5)^2]}, \\ u(x, y) &= \sin(\pi x), \\ v(x, y) &= \sin(\pi y), \\ E(x, y) &= \sin^2(2\pi x) \sin^2(2\pi y), \end{cases} \quad (3.5)$$

where (u, v) is the velocity in the x, y directions respectively. Before the remapping procedure, we utilize a high order integral formula to calculate the cell-averages of density $\bar{\rho}$, momentum $\bar{\rho}u, \bar{\rho}v$ and total energy $\bar{E}$ on the initial mesh.

Table 3.4 shows the errors of the four conservative variables in L^1 and L^∞ norms obtained by the P2, WENO, P-WENO remapping algorithms on the smoothly moving meshes with $N = 10$. All of the three remapping algorithms are third-order accurate, and the positivity-preserving limiter does not reduce the order of accuracy which is consistent with the theoretical result. Again, we observe that the accuracy of the two versions of the P-WENO algorithm are nearly the same. To save space, we will adopt the more efficient version P-WENO-B in the following numerical experiments for the discontinuous problems and will simply denote it as P-WENO.

3.2 Non-oscillatory test

In this section, we use the step function (3.6) as the test function to verify the essentially non-oscillatory property of the multi-resolution WENO remapping algorithm with or without the positivity-preserving limiter.

$$\begin{cases} \rho(x, y) &= 100 & y > \frac{10}{3}(x - 0.4), \\ \rho(x, y) &= 1 & y \leq \frac{10}{3}(x - 0.4). \end{cases} \quad (3.6)$$

The P2, WENO and P-WENO remapping algorithms have been used in this experiment, under the randomly moving mesh set with $N_x = N_y = 32$ and remapping for $N = 10$ times.

Table 3.4: The P2, WENO and P-WENO conservative remapping algorithms on the smoothly moving meshes, $N = 10$. Nc is the percentage of cells which conclude negative integration points.

	Mesh	ρ		ρu		ρv		E		$Nc(\%)$
L^1 P2	8×8	1.72E-3		1.74E-3		1.74E-3		1.58E-2		0
	16×16	3.17E-4	2.44	3.43E-4	2.34	3.44E-4	2.34	2.62E-3	2.59	10.74
	32×32	3.69E-5	3.10	4.03E-5	3.09	4.03E-5	3.09	2.71E-4	3.27	8.55
	64×64	3.46E-6	3.42	3.56E-6	3.50	3.56E-6	3.50	1.99E-5	3.77	5.16
L^1 WENO	8×8	2.34E-3		2.43E-3		2.52E-3		2.55E-2		0
	16×16	3.17E-4	2.88	3.43E-4	2.83	3.44E-4	2.87	2.86E-3	3.16	5.47
	32×32	3.69E-5	3.10	4.03E-5	3.09	4.03E-5	3.09	2.71E-4	3.40	6.35
	64×64	3.46E-6	3.42	3.56E-6	3.50	3.56E-6	3.50	1.99E-5	3.77	4.00
L^1 P-WENO-A	8×8	2.02E-3		2.26E-3		2.31E-3		2.71E-2		0
	16×16	3.17E-4	2.70	3.43E-4	2.72	3.44E-4	2.75	2.87E-3	3.24	0
	32×32	3.70E-5	3.10	4.05E-5	3.08	4.03E-5	3.09	2.71E-4	3.40	0
	64×64	3.47E-6	3.41	3.58E-6	3.50	3.56E-6	3.50	2.00E-5	3.76	0
L^1 P-WENO-B	8×8	2.34E-3		2.43E-3		2.52E-3		2.55E-2		0
	16×16	3.17E-4	2.88	3.43E-4	2.83	3.44E-4	2.87	2.86E-3	3.16	5.47
	32×32	3.69E-5	3.10	4.03E-5	3.09	4.03E-5	3.09	2.71E-4	3.40	6.35
	64×64	3.46E-6	3.42	3.56E-6	3.50	3.56E-6	3.50	1.99E-5	3.77	4.00
L^∞ P2	8×8	7.19E-3		7.72E-3		7.82E-3		3.62E-2		0
	16×16	1.88E-3	1.93	2.03E-3	1.93	1.97E-3	1.99	1.25E-2	1.54	10.74
	32×32	2.65E-4	2.83	2.99E-4	2.76	2.93E-4	2.75	1.07E-3	3.55	8.55
	64×64	2.81E-5	3.24	3.28E-5	3.19	3.30E-5	3.15	6.45E-5	4.05	5.16
L^∞ WENO	8×8	1.62E-2		1.67E-2		1.76E-2		7.13E-2		0
	16×16	1.84E-3	3.14	2.03E-3	3.04	2.18E-3	3.02	2.38E-2	1.58	5.47
	32×32	2.65E-4	2.79	2.98E-4	2.77	2.93E-4	2.90	1.28E-3	4.22	6.35
	64×64	2.81E-5	3.24	3.28E-5	3.18	3.30E-5	3.15	6.45E-5	4.31	4.00
L^∞ P-WENO-A	8×8	1.62E-2		1.67E-2		1.76E-2		7.98E-2		0
	16×16	1.84E-3	3.14	2.03E-3	3.04	2.18E-3	3.02	2.38E-2	1.75	0
	32×32	2.65E-4	2.79	2.99E-4	2.77	2.93E-4	2.90	1.28E-3	4.22	0
	64×64	2.81E-5	3.24	3.28E-5	3.19	3.30E-5	3.15	6.45E-5	4.31	0
L^∞ P-WENO-B	8×8	1.62E-2		1.67E-2		1.76E-2		7.13E-2		0
	16×16	1.84E-3	3.14	2.03E-3	3.04	2.18E-3	3.02	2.38E-2	1.58	5.47
	32×32	2.65E-4	2.79	2.98E-4	2.77	2.93E-4	2.90	1.28E-3	4.22	6.35
	64×64	2.81E-5	3.24	3.28E-5	3.18	3.30E-5	3.15	6.45E-5	4.31	4.00

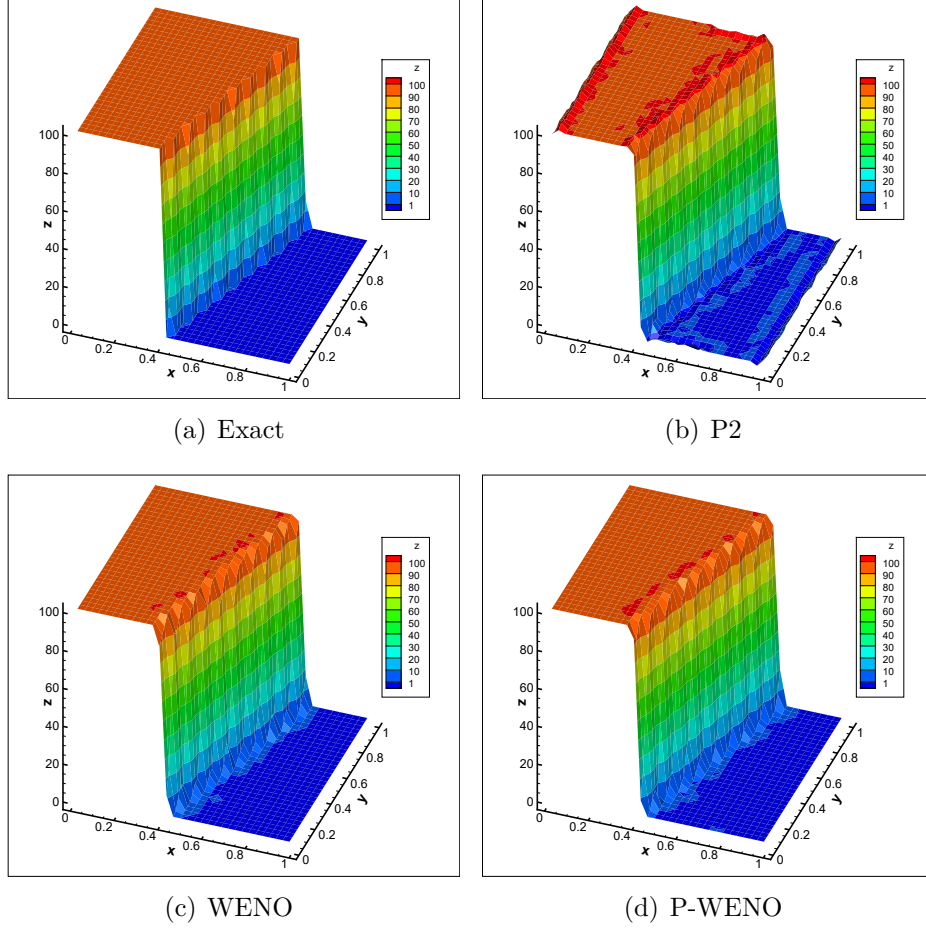


Figure 3.4: The non-oscillatory test: remapping the step function by the third-order P2, WENO and P-WENO remapping algorithms, cell-averages of density, $N = 10$, $N_x = N_y = 32$. Top left: exact; Top right: P2; Bottom left: WENO; Bottom right: P-WENO.

Figure 3.4 shows the results of the P2, WENO and P-WENO remapping algorithms compared with the exact results. We can see that there are some oscillations near the discontinuity in the top right figure of Figure 3.4, but there is no oscillation in the bottom two figures, which demonstrates that the WENO and P-WENO remapping algorithms are robust and essentially non-oscillatory when dealing with shock or contact discontinuities in the ALE method.

Figure 3.5 shows the parameters $\beta_1, \beta_2, \omega_1, \omega_2$ in the reconstruction step of the third-order WENO remapping algorithms in this step function test. Near the discontinuity, we can see $\beta_1 \ll \beta_2$ which demonstrates the polynomial $q_2(x, y)$ is less smooth. $\omega_2 \ll \omega_1$ leads

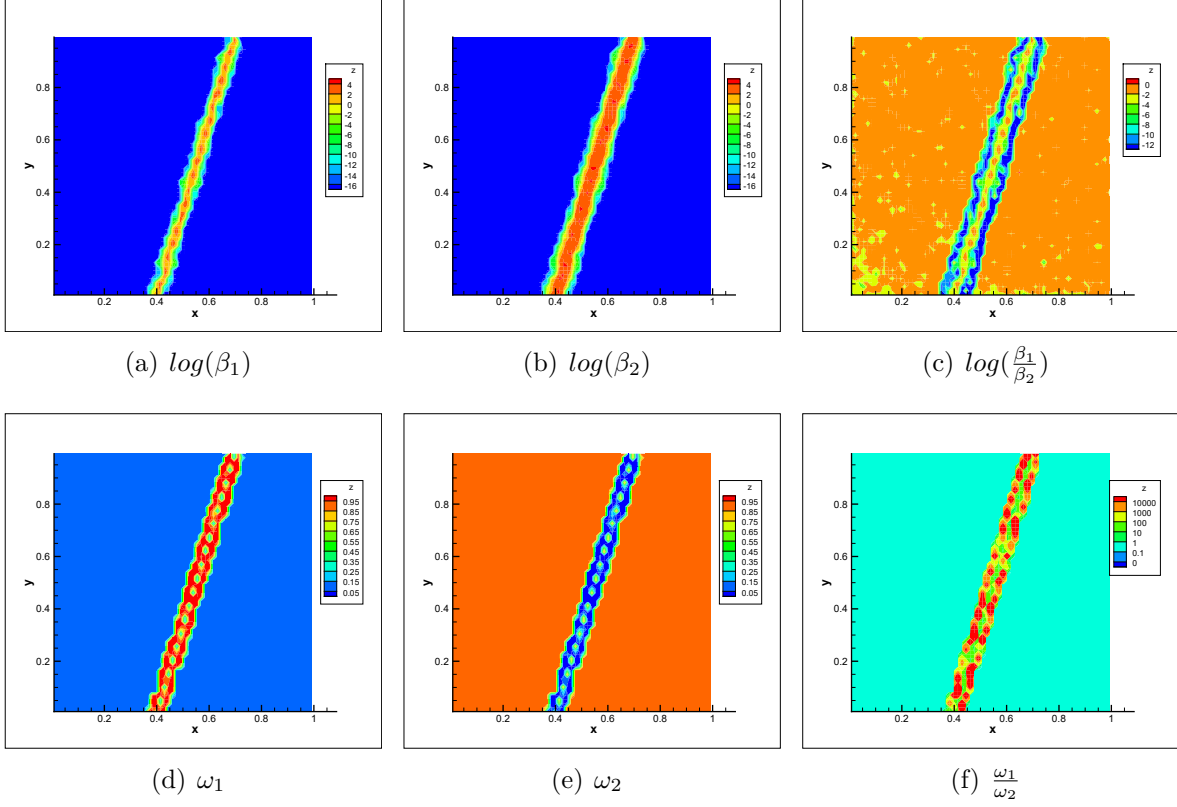


Figure 3.5: The parameters $\beta_1, \beta_2, \omega_1, \omega_2$ in the reconstruction step of the third-order WENO remapping algorithms in the step function test, $N_x = N_y = 64$.

to $p_1(x, y)$ having a greater influence on the final polynomial $p(x, y)$ which can avoid the numerical oscillation. In the smooth region, the nonlinear weights ω_1 and ω_2 are close to the linear weights, thus yielding to highly accurate results.

3.3 Positivity-preserving tests

The positivity-preserving limiter is added after the reconstruction step to maintain the positivity of density and internal energy at all stages of the remapping. We design several numerical experiments to verify that the P-WENO remapping algorithm can preserve positivity. The process is still remapping from a uniform mesh to a randomly moving mesh, after several remapping times between randomly moving meshes, the final remapping brings the mesh back to the initial uniform mesh $\mathcal{M}^0$.

3.3.1 The step function for the positivity-preserving test

We make a small modification on the step function in Section 3.2 to test the positivity-preserving property of our remapping algorithms.

$$\begin{cases} \rho(x, y) = 100 & y > \frac{10}{3}(x - 0.4), \\ \rho(x, y) = 0 & y \leq \frac{10}{3}(x - 0.4). \end{cases} \quad (3.7)$$

In this test, the values of density is zero when $y \leq \frac{10}{3}(x - 0.4)$, which brings difficulties for the remapping algorithm since negative cell-averages may be generated.

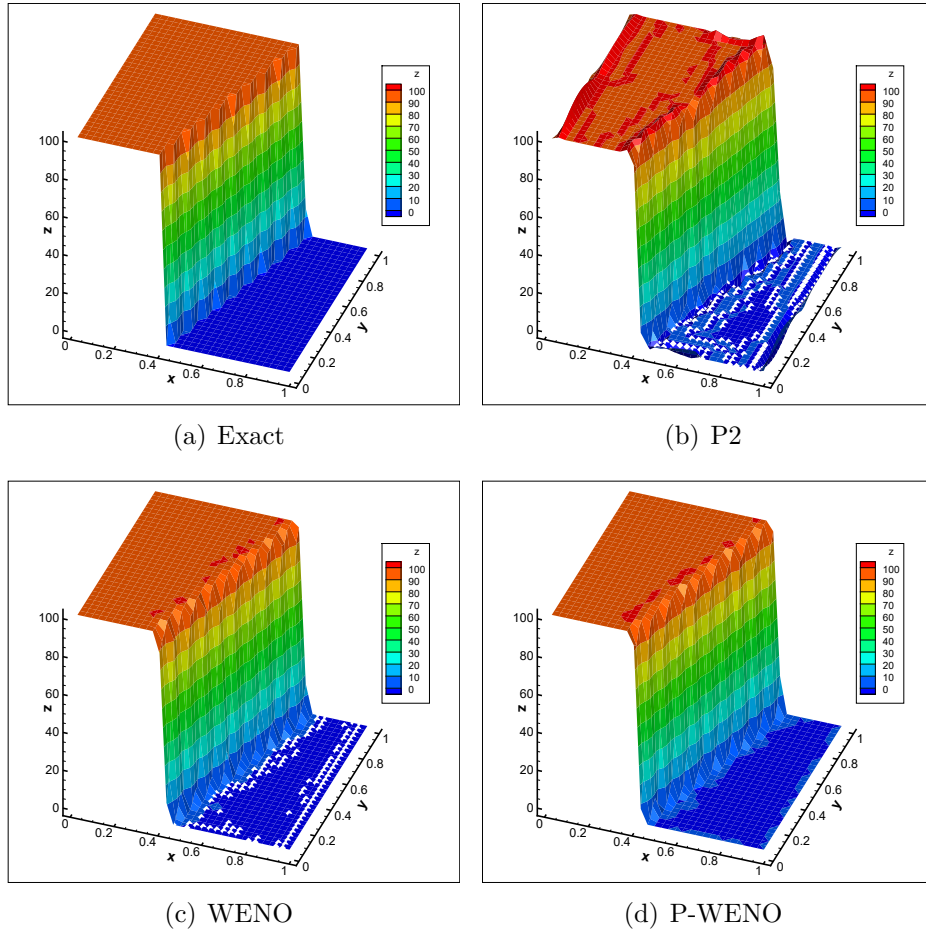


Figure 3.6: The positivity-preserving test on the step function by the P2, WENO and the P-WENO remapping algorithms, cell-averages of density, $N = 10$, $N_x = N_y = 32$. Top left: exact; Top right: P2; Bottom left: WENO; Bottom right: P-WENO. The white symbols in the top right and bottom left profiles represent the cells where the cell-averages are negative.

Figure 3.6 shows the results of the P2, WENO and P-WENO remapping algorithms. To see more clearly, we cut Figure 3.6 along $j = 16$ and show the results in Figure 3.7, where

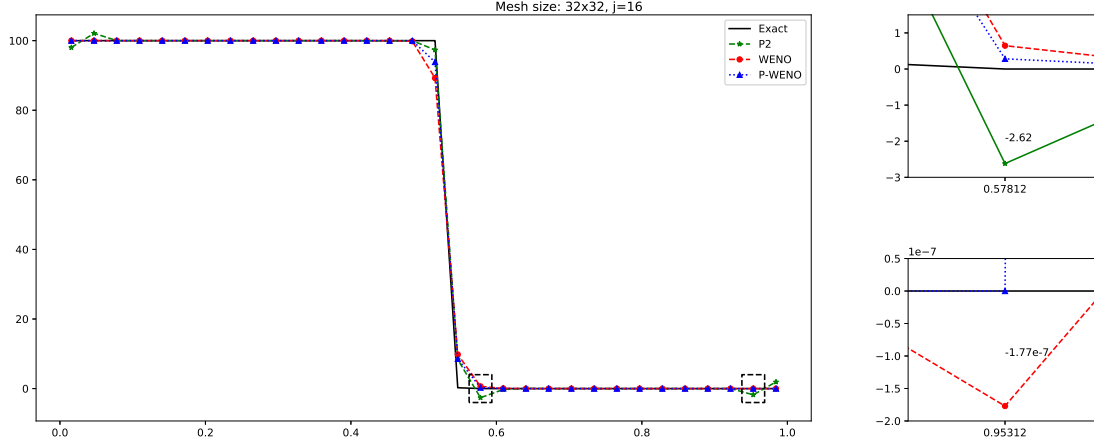


Figure 3.7: The positivity-preserving test on the step function by the P2, WENO and the P-WENO remapping algorithms, cell-averages of density at $j = 16$, zoomed-in figures at $i = 19$ and $i = 31$ on the right, $N = 10$, $N_x = N_y = 32$. Top right: $i = 19$; Bottom right: $i = 31$.

we observe that the WENO remapping algorithm produces negative cell-average value of -1.77×10^{-7} at $i = 31$ and the P2 remapping algorithm produces negative cell-average value of -2.62 at $i = 19$ near the discontinuity but the result from the P-WENO remapping algorithm can always preserve positivity.

3.3.2 The cylinder function

We construct a circular discontinuous function noted as ‘the cylinder function’ in $[0, 1] \times [0, 1]$, and take $C(0.5, 0.5)$ as the center of the circle

$$\begin{cases} \rho(x, y) = 10 & d(x, y) < r \\ \rho(x, y) = 0 & d(x, y) \geq r \end{cases},$$

where

$$d(x, y) = \sqrt{(x - 0.5)^2 + (y - 0.5)^2} \quad (3.8)$$

is the Euclidean distance between the center $C(0.5, 0.5)$ and the point (x, y) , the radius r is given as 0.25 in the numerical experiments. Obviously, the cylinder function is not continuous, and the cell-averages far away from the central circle are zero which may cause negative results.

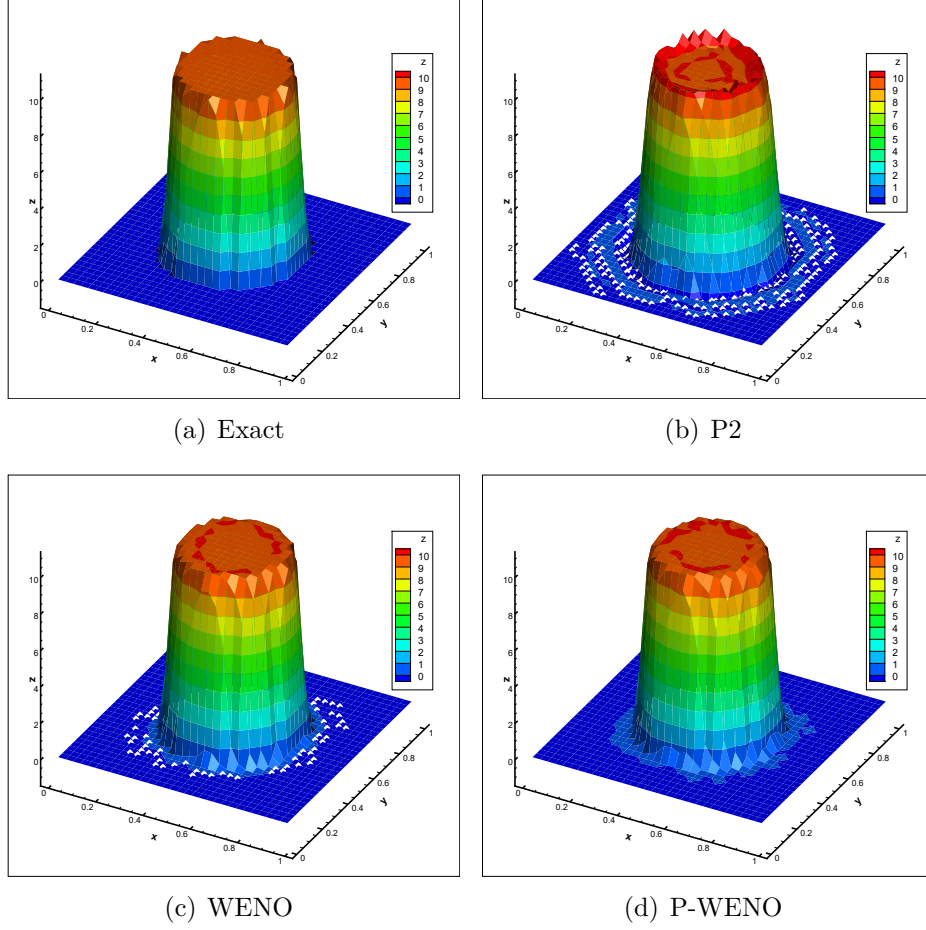


Figure 3.8: The positivity-preserving test on the cylinder function by the P2, WENO and the P-WENO remapping algorithms, cell-averages of density, $N = 10$, $N_x = N_y = 32$. Top left: exact; Top right: P2; Bottom left: WENO; Bottom right: P-WENO. The white symbols in the top right and bottom left profiles represent the cells where the cell-averages are negative.

Figure 3.8 shows the remapping results of the P2, WENO and P-WENO remapping algorithms against the exact solution with the mesh size 32×32 . For the sake of seeing the discrepancy between the three algorithms for positivity-preserving more clearly, we cut Figure 3.8 along $i = 16$ and show the results in Figure 3.9. Differently from the results of P-WENO, negative cell-averages from the P2 remapping algorithm at $j = 7$ and from the WENO remapping algorithm at $j = 27$ can be observed near the discontinuity just as before.

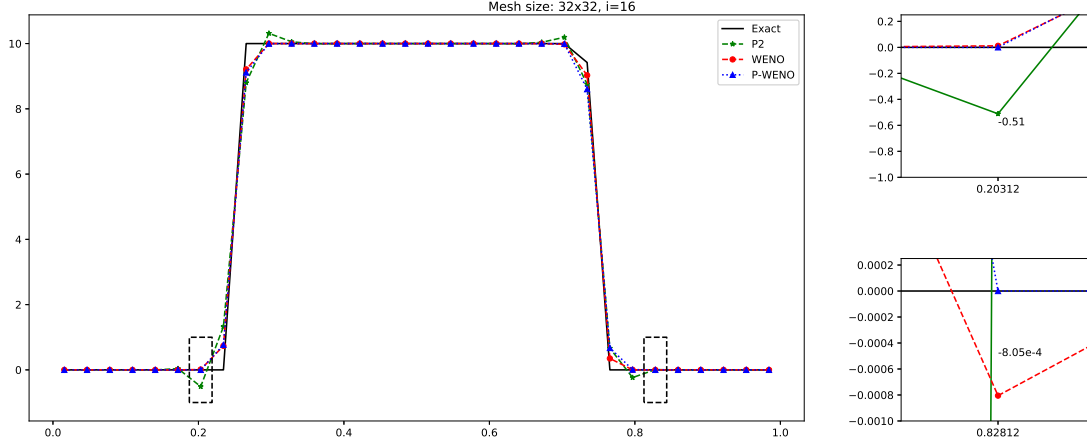


Figure 3.9: The positivity-preserving test on the cylinder function by the P2, WENO and the P-WENO remapping algorithms, cell-averages of density at $i = 16$, zoomed-in figures at $j = 7$ and $j = 27$ on the right, $N = 10$, $N_x = N_y = 32$. Top right: $j = 7$; Bottom right: $j = 27$.

3.3.3 The cone function

Next, we use the cone function to test the positivity-preserving property,

$$\rho(x, y) = 10 \max(1 - 4d(x, y), 0).$$

Figure 3.10 shows the remapping results with or without the *PP* limiter compared with the exact function and Figure 3.11 shows the cuts along $i = 16$. It is obvious that the P2, WENO remapping algorithms generate negative cell-averages, however the P-WENO does not.

From the above figures, we can see that the P2 remapping algorithm and the WENO remapping algorithm may generate negative density in the area where the cell-averages are close to 0 and change drastically. But with the *PP* limiter, P-WENO can handle this problem, no negative cell-averages appear.

3.3.4 The mimetic Sedov problem

Last, we mimic the Sedov problem to test the performance of our remapping algorithms on the fluid flow problem. u and v refer to the velocity in the x and y directions, $\delta = 10^{-6}$ is a small constant number and we set $r_0 = 0.75$ in the definition below.

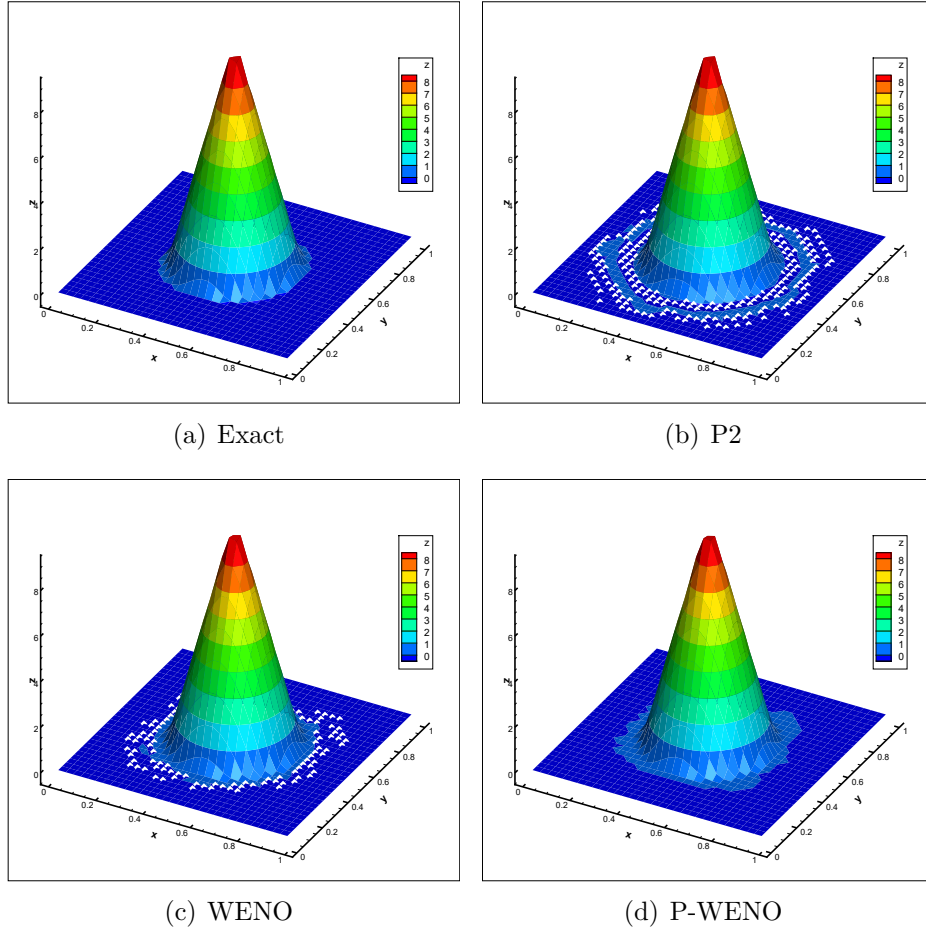


Figure 3.10: The positivity-preserving test on the cone function by the P2, WENO and P-WENO remapping algorithms, cell-averages of density, $N = 10$, $N_x = N_y = 32$. Top left: exact; Top right: P2; Bottom left: WENO; Bottom right: P-WENO. The white symbols in the top right and bottom left profiles represent the cells where the cell-averages are negative.

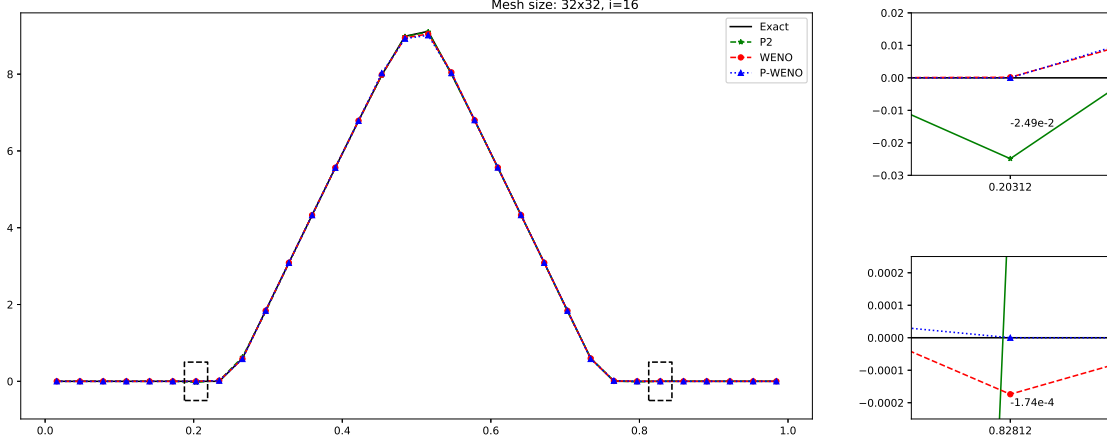


Figure 3.11: The positivity-preserving test on the cone function by the P2, WENO and P-WENO remapping algorithms, cell-averages of density at $i = 16$, zoomed-in figures at $j = 7$ and $j = 27$ on the right, $N = 10$, $N_x = N_y = 32$. Top right: $j = 7$; Bottom right: $j = 27$.

$$\begin{cases} \rho(x, y) &= \begin{cases} \delta + 6\left(\frac{r}{r_0}\right)^8 & \text{if } r = \sqrt{x^2 + y^2} \leq r_0, \\ \delta & \text{else.} \end{cases} \\ \begin{pmatrix} u(x, y) \\ v(x, y) \end{pmatrix} &= \begin{cases} \begin{pmatrix} 0.83x \\ 0.83y \end{pmatrix} & \text{if } r = \sqrt{x^2 + y^2} \leq r_0, \\ \begin{pmatrix} 0 \\ 0 \end{pmatrix} & \text{else.} \end{cases} \\ e(x, y) &= \begin{cases} 0.25 + 3\left(1 - \frac{r}{r_0}\right) & \text{if } r = \sqrt{x^2 + y^2} \leq r_0, \\ \delta & \text{else.} \end{cases} \end{cases} \quad (3.9)$$

Figures 3.12-3.13 show the exact profile and the remapping results by the P2, WENO, P-WENO remapping algorithms of density and internal energy corresponding to the cell-averages of the conserved variables for the mimetic Sedov problem respectively. From these figures, we can see that without the positivity-preserving limiter, some negative values of density and/or internal energy corresponding to the cell-averages of the conserved variables emerge in the figures, which have been highlighted by the white dots. In Figure 3.12, we can see some negative cell-averages of density by the WENO remapping algorithm at the region where the exact solution $\rho \approx \delta = 10^{-6}$ is close to zero. Similarly, in Figure 3.13, we observe some negative internal energy corresponding to the cell-averages of the conserved variables by the WENO remapping algorithm at $r > r_0$, where the exact solution $e \approx \delta = 10^{-6}$ is close to zero. Since $\rho(x, y)$, $e(x, y)$ in the mimetic Sedov problem are radial functions, we take the cuts at $i = 1$ shown as Figures 3.14-3.15. From the above two figures we can see

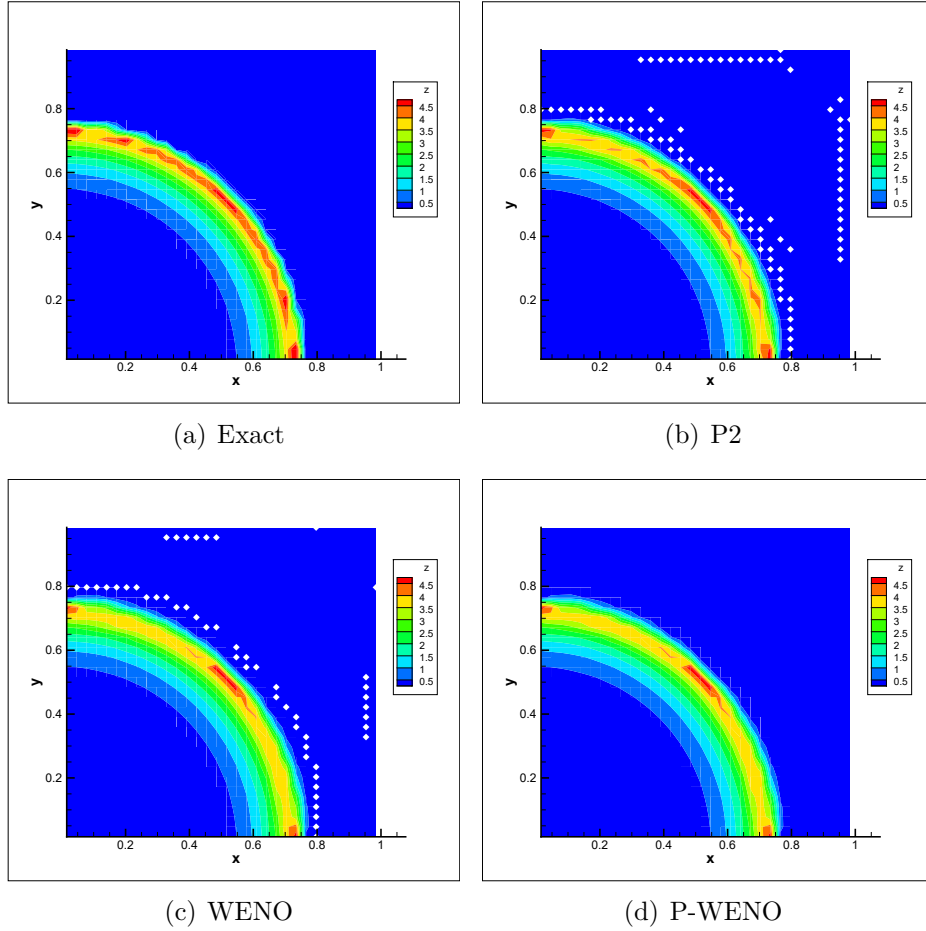


Figure 3.12: The remapping results for the mimetic Sedov problem: contour of density, $N = 10$, $N_x = N_y = 32$. Top left: exact; Top right: P2; Bottom left: WENO; Bottom right: P-WENO. The white symbols in the top right and bottom left profiles represent the cells where the cell-averages are negative.

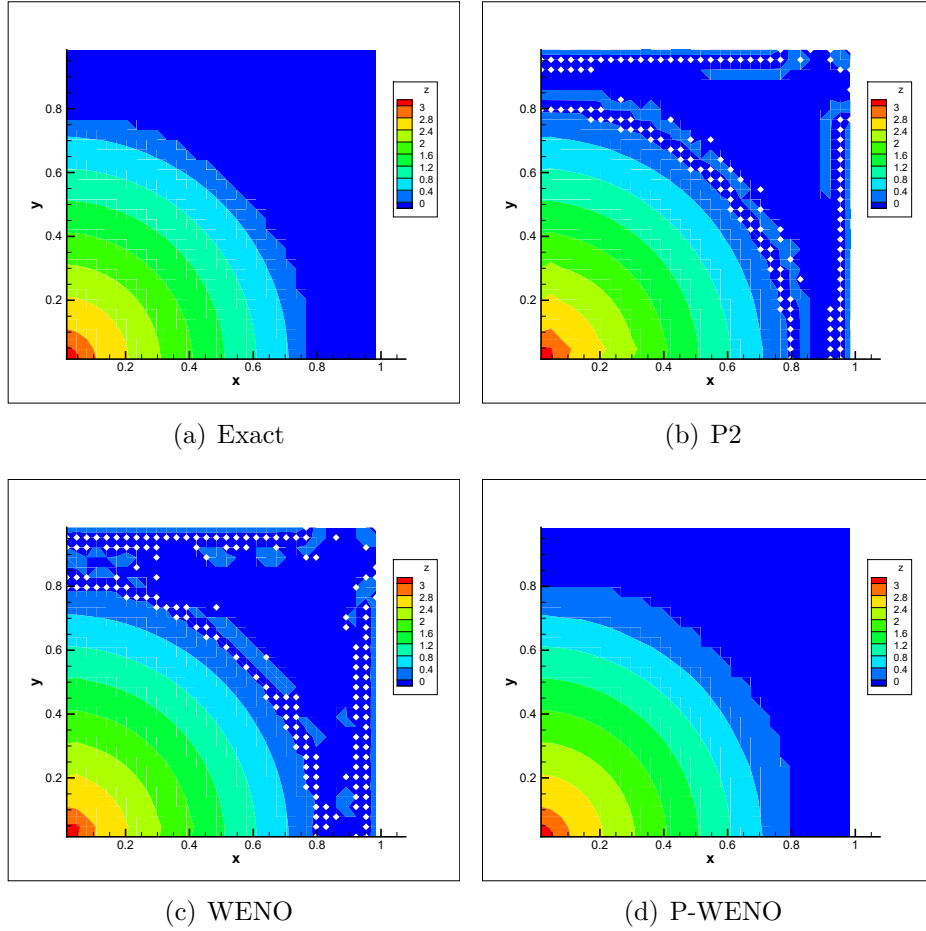


Figure 3.13: The remapping results for the mimetic Sedov problem: contour of internal energy, $N = 10$, $N_x = N_y = 32$. Top left: exact; Top right: P2; Bottom left: WENO; Bottom right: P-WENO. The white symbols in the top right and bottom left profiles represent the cells where the internal energy from the cell-averages are negative.

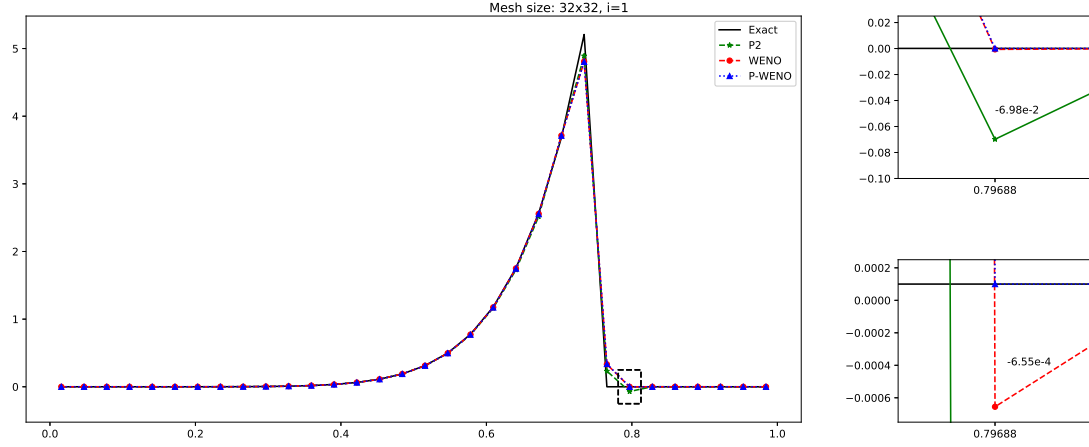


Figure 3.14: The positivity-preserving test on the mimetic Sedov problem by the P2, WENO and P-WENO remapping algorithms, cell-averages of density at $i = 1$, zoomed-in figures at $j = 27$ on the right, $N = 10$, $N_x = N_y = 32$.

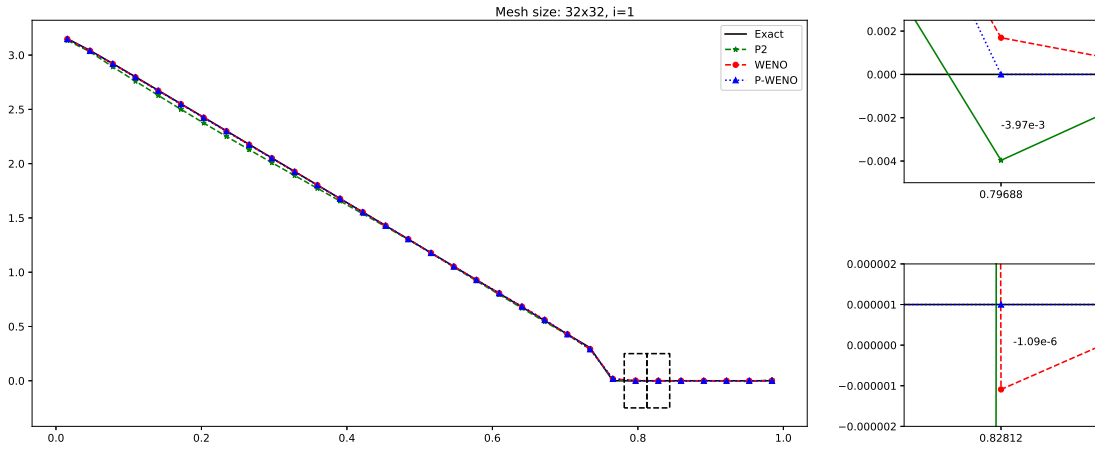


Figure 3.15: The positivity-preserving test on the mimetic Sedov problem by the P2, WENO and P-WENO remapping algorithms, cell-averages of internal energy at $i = 1$, zoomed-in figures at $j = 26$ and $j = 27$ on the right, $N = 10$, $N_x = N_y = 32$. Top right: $j = 26$; Bottom right: $j = 27$.

that without the *PP* limiter some negative values of density and internal energy emerge at $j = 26$ and $j = 27$ when the exact values are close to zero.

3.4 The tests in the ALE simulation

We further apply our P-WENO conservative remapping algorithm in an indirect ALE method after the Lagrangian step and the rezoning step. The classical WENO reconstruction has already been used to design the Lagrangian and ALE methods in the literatures, for example [7, 3]. Here, we modify the 2D second order ENO positivity-preserving Lagrangian scheme in [8] by using the multi-resolution WENO reconstruction introduced in this paper. Then this modified Lagrangian scheme is adopted in the Lagrangian step of the ALE method. We take three classical fluid flow examples, i.e., the Sedov problem, the Saltzman problem and the Noh problem for numerical tests. We will compare the results of the above mentioned purely positivity-preserving Lagrangian scheme and the positivity-preserving ALE method, which will demonstrate the good performance of the ALE method with our P-WENO remapping algorithm.

3.4.1 The Sedov problem

First we test the Sedov problem by the purely positivity-preserving Lagrangian scheme and the ALE method with the P-WENO conservative remapping algorithm. The initial condition is given as:

$$\begin{cases} \rho &= 1, \\ u &= 0, \\ v &= 0, \end{cases} \quad (3.10)$$

where the internal energy e of the system is 10^{-14} almost everywhere except the cell near the origin where we set $e = 182.09$. $p = (\gamma - 1)\rho e$. $\gamma = 1.4$. The simulation is performed on the initially uniform grid consisting of 30×30 rectangular cells in $[0, 1.1] \times [0, 1.1]$. Reflective boundary conditions are given for the four boundaries and the numerical results are shown in Figure 3.16 at time $t = 1$.

In this case, we change the purely positivity-preserving Lagrangian scheme to the positivity-

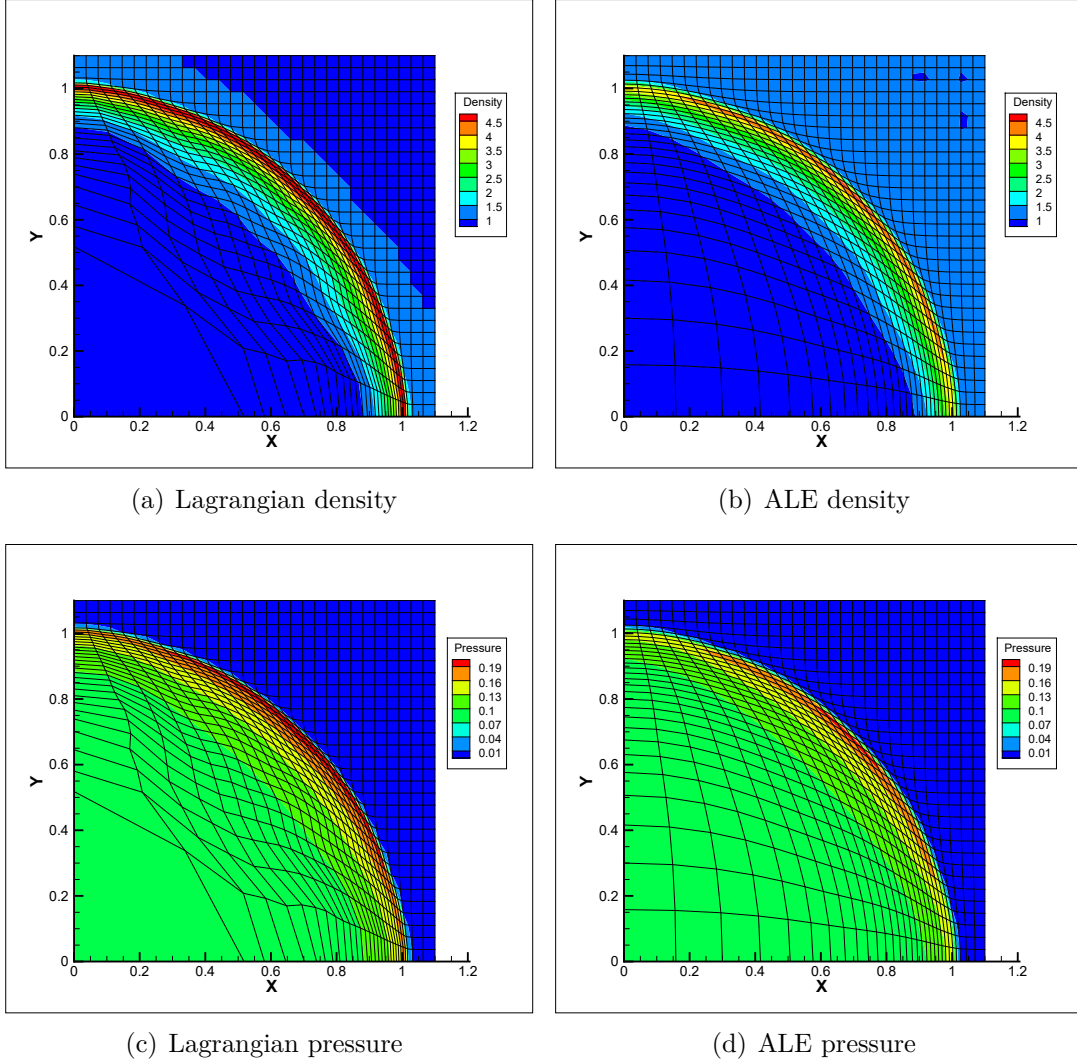


Figure 3.16: The results for the Sedov problem with $N_x = N_y = 30$, $t = 1$. Top left: grid and density by the Lagrangian scheme; Top right: grid and density by the ALE method. Bottom left: grid and pressure by the Lagrangian scheme; Bottom right: grid and pressure by the ALE method.

preserving ALE method when $t = 0.5$ and perform the rezoning and remapping algorithm every 20-time steps to make the grid deformation not too serious. Both of the two methods can handle this case well, while the grid in the ALE method is better than that of the Lagrangian scheme.

3.4.2 The Saltzman problem

Next, we test the Saltzman problem with $N_x = 100, N_y = 10$ in $[0, 1] \times [0, 0.1]$ by the purely positivity-preserving Lagrangian scheme and the ALE method with the P-WENO remapping algorithm. The left boundary of the computation domain is a piston with a constant velocity of 1, and reflective boundary conditions are taken for the other three boundaries. In this test, the initial condition is:

$$\begin{cases} \rho &= 1, \\ u &= 0, \\ v &= 0, \\ p &= 10^{-10}. \end{cases} \quad (3.11)$$

$\gamma = \frac{5}{3}$. The initial grid is shown in Figure 3.17. In this case, a purely positivity-preserving Lagrangian scheme will blow up at the later time due to the distorted grid. So we only compare the results at time $t = 0.6$ in Figure 3.18. The ALE method with our P-WENO remapping algorithm can produce the results at any time. Figure 3.18 shows its results at $t = 0.6, 0.8$ respectively. In the figures, we can see the piston have arrived at the right boundary and returned when $t = 0.8$. By comparing with the purely Lagrangian scheme, we can see the more robustness of the ALE method with our remapping algorithm.

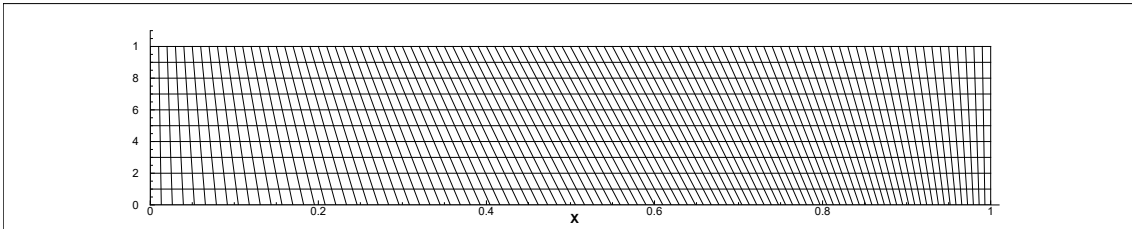


Figure 3.17: Initial grid for the Saltzman problem, $N_x = 100, N_y = 10$.

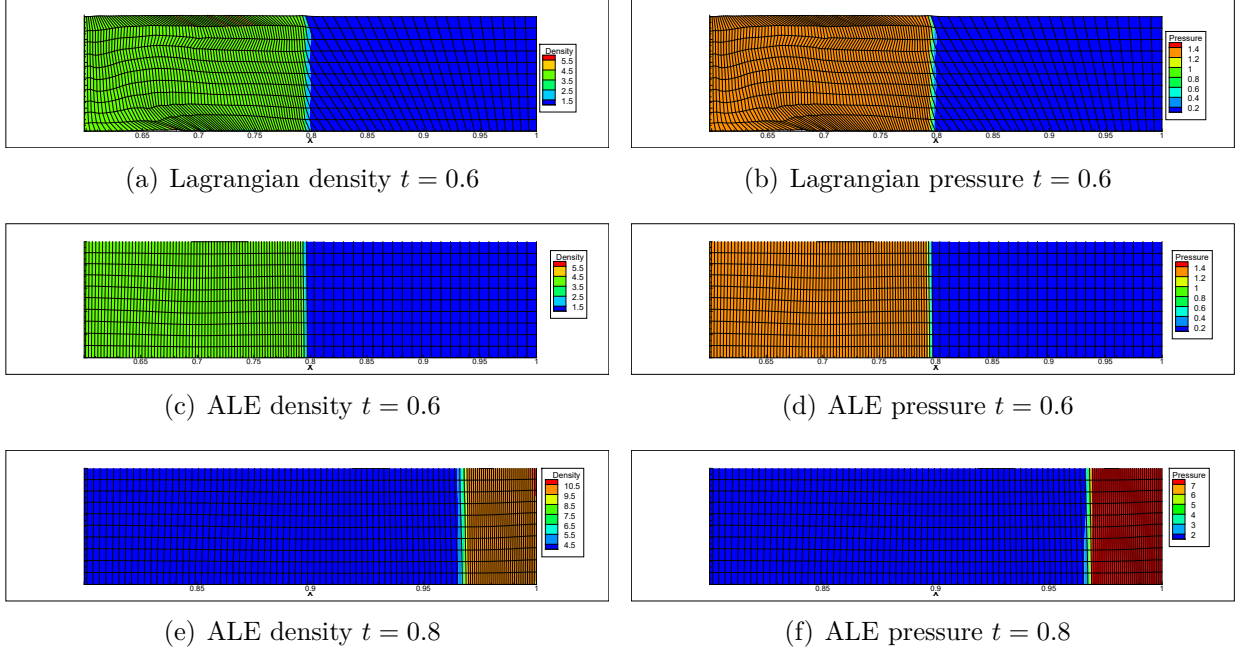


Figure 3.18: The results for the Saltzman problem with $N_x = 100, N_y = 10$. Top left: grid and density by the Lagrangian scheme at time $t = 0.6$; Top right: grid and pressure by the Lagrangian scheme at time $t = 0.6$. Middle: grid, density and pressure by the ALE method at time $t = 0.6$; Bottom: grid, density and pressure by the ALE method at time $t = 0.8$;

3.4.3 The Noh problem

Finally, we test the Noh problem with the initially uniform Cartesian grid in $[0, 1] \times [0, 1]$, and it is difficult for the Lagrangian scheme to simulate because the grid is prone to be distorted near the origin with the time marching. In our test, we take a 35×35 uniform grid as our initial grid and calculate up to time $t = 0.6$. The initial condition is:

$$\begin{cases} \rho &= 1, \\ u_r &= -1, \\ e &= 10^{-14}, \end{cases} \quad (3.12)$$

where u_r is the radial velocity at the cell center. $\gamma = 5/3$. We take the free boundary condition for the right and upper boundaries and take the reflective boundary condition for the lower and left boundaries. As an example, we show the results of the Noh problem under the purely positivity-preserving Lagrangian scheme at $t = 0.54$. After that time, the time step tends to zero causing the failure of the simulation. This problem can be solved by the ALE method. By adding the rezoning step and the P-WENO conservative remapping step

after $t = 0.5$, the ALE method can produce the results at the later time. We show its results at time $t = 0.6$ in Figures 3.19. In the figure, we can see the grid with the better quality.

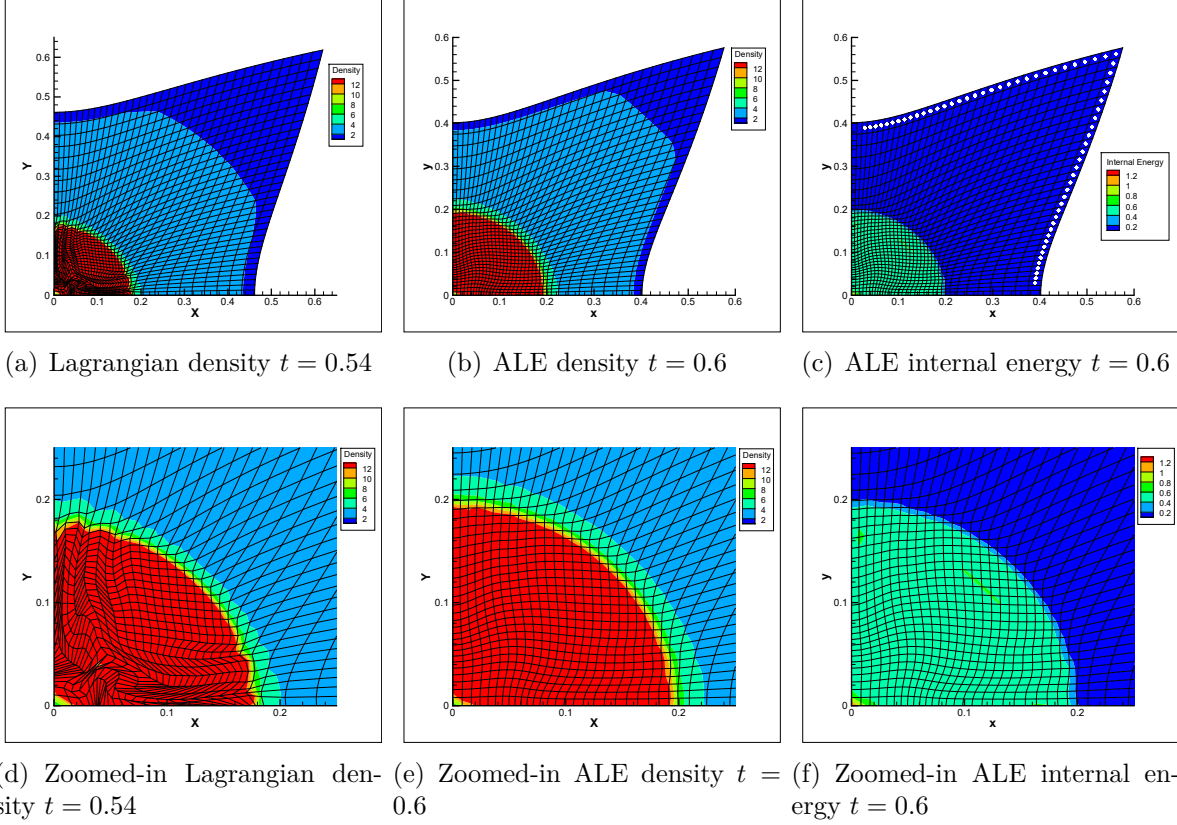


Figure 3.19: The results for the Noh problem with $N_x = N_y = 35$. Top left: grid and density by the Lagrangian scheme at $t = 0.54$; Top middle: grid and density by the ALE method at $t = 0.6$; Top right: grid and internal energy by the ALE method at $t = 0.6$. Bottom: the zoomed grid, density and internal energy near the origin. The white symbols in the top right figure represent the cells where the cell-averages of internal energy are negative.

Without the *PP* limiter in the WENO remapping algorithm used in the ALE method, the code can not continue since negative internal energy and pressure emerge. According to statistics, there are about 3.55% cells which have been modified by the *PP* limiter, these cells have been labeled in white color in the top right figure of Figure 3.19.

In the above three examples, we can observe that with the help of our P-WENO remapping algorithm, the ALE method is positivity-preserving and more robust than the purely Lagrangian scheme. Especially in the Noh problem, there are some cells modified by our positivity-preserving limiter, without which the ALE method will fail due to the negative

internal energy.

3.5 The cost comparison of three types of high order remapping algorithms

In this subsection, the computational cost of the P2, ENO, WENO, P-WENO-A and P-WENO-B remapping algorithms with the same accuracy are compared. An efficient high-accuracy remapping algorithm should ensure that it does not cost too much when increasing the accuracy, and the added positivity-preserving limiter should not bring too much extra cost. We test five remapping algorithms on the function (3.4), described in the accuracy test. As shown in Table 3.5, we record the cost of polynomial reconstruction (Reconst.) step and the positivity-preserving limiter (Limiter.) step in the remapping procedure, and then give the corresponding total computational cost on three different mesh sizes.

Table 3.5: Cost of five third order remapping algorithms in different sizes of the smoothly moving meshes, $N = 10$. The last column T_M indicates the total cost of time versus the unlimited P2 remapping algorithm.

		Reconst.	Limiter.	Total	T_M
Mesh size 16×16	P2	2.60E+2	0	3.43E+2	-
	WENO	3.40E+2	0	4.24E+2	1.24
	P-WENO-A	3.41E+2	8.12E+1	5.06E+2	1.48
	P-WENO-B	3.36E+2	7.00E+0	4.32E+2	1.26
	ENO	5.88E+3	0	5.96E+3	17.38
Mesh size 32×32	P2	1.13E+3	0	1.51E+3	-
	WENO	1.44E+3	0	1.82E+3	1.21
	P-WENO-A	1.50E+3	3.69E+2	2.26E+3	1.50
	P-WENO-B	1.49E+3	3.17E+1	1.93E+3	1.28
	ENO	2.13E+4	0	2.17E+4	14.37
Mesh size 64×64	P2	5.51E+3	0	7.14E+3	-
	WENO	6.75E+3	0	8.34E+3	1.17
	P-WENO-A	7.14E+3	1.59E+3	1.04E+4	1.46
	P-WENO-B	7.31E+3	1.39E+2	9.37E+3	1.31
	ENO	7.79E+4	0	7.96E+4	11.15

It can be seen from Table 3.5 that adding the *PP* limiter does not bring too much extra cost to the program. Moreover, compared with the ENO remapping algorithm with the same accuracy, the P-WENO and the WENO remapping algorithms have significantly

higher efficiency. While the two parts of the numerical integration and the calculation of the intersection region are nearly the same, the ENO remapping algorithm needs to construct several polynomials on different stencils and compare the coefficients of these polynomials to select the optimal stencil during the reconstruction process, which brings significant extra cost to the reconstruction step, resulting in a total cost several times higher than that of the P-WENO and WENO remapping algorithms. We also compare the cost of the two P-WENO versions. The P-WENO-B remapping algorithm, which applies the limiter only when the reconstructed cell averages on the new mesh become negative, is more efficient than the P-WENO-A remapping algorithm.

4 Conclusion

Aiming at the remapping step in the ALE method simulating the fluid flows, we develop a non-oscillatory, positivity-preserving, conservative, third order accurate remapping algorithm based on the multi-resolution WENO reconstruction. For the remapping step in the ALE framework, conservative physical variables such as mass, momentum, and total energy need to be transferred between old and new meshes. Take mass as an example, the detailed analysis is performed based on the cell-average of density. In this paper, the multi-resolution WENO reconstruction is used to reconstruct the polynomials on the old mesh, the overlapping area between old and new meshes is calculated by the Sutherland-Hodgman clipping algorithm. After that, a positivity-preserving limiter is added, ensuring that the remapping results guarantee the positivity of density and internal energy. Finally, using numerical quadrature with sufficient accuracy, the reconstruction polynomial is integrated over the intersections exactly to achieve conservation. A series of accuracy tests, non-oscillatory tests, and positivity-preserving tests verify that the algorithm is conservative, positivity-preserving, essentially non-oscillatory and high order accurate.

Specifically, in this paper, the multi-resolution WENO reconstruction is applied to reconstruct the third-order accurate polynomials. Higher order reconstruction polynomials can

also be considered by following the same recipe, as detailed in [41, 42]. Furthermore, if we combine the higher order WENO reconstruction and higher order integration formula, then the higher order P-WENO remapping algorithm can be designed as well. The cost of the simple multi-resolution WENO reconstruction method is much less than the ENO reconstruction of the same accuracy. After using the Sutherland-Hodgman clipping algorithm to calculate the overlapping area, we utilize the exact numerical integration formula to calculate the integration of reconstruction polynomial over the overlapping area. For higher order reconstruction polynomials, the corresponding higher order accurate quadrature formula is necessary at this step. In the process of numerical integration, a positivity-preserving limiter is added. If the reconstructed polynomials do not meet the requirements of the positivity-preserving property, we modify the polynomial by the *PP* limiter while maintaining the high order accuracy and conservation.

In the numerical experiments, we verify the properties of the algorithm on different moving meshes. The results show that the algorithm can keep the designed third-order accuracy for smooth problems and is essentially non-oscillatory for discontinuous problems, both on smoothly moving meshes and on randomly moving meshes. We also test the performance of our remapping algorithm by the true ALE simulation. By comparing with the purely Lagrangian scheme, the ALE method with our remapping algorithm shows its positivity-preserving property and more robustness to handle the problem where the Lagrangian scheme fails. Based on the advantages of the intersection-based method, our remapping algorithm can be applied to unstructured meshes without the same connectivity.

The design of high order conservative positivity-preserving remapping algorithm in three dimensions and high order conservative positivity-preserving ALE method constitutes our future work.

References

- [1] M. Berndt, J. Breil, S. Galera, M. Kucharik, P.-H. Maire, M. Shashkov, *Two-step hybrid conservative remapping for multimaterial arbitrary Lagrangian-Eulerian methods*, Journal of Computational Physics, 230 (2011), 6664-6687.
- [2] G. Blanchard, R. Loubere, *High order accurate conservative remapping scheme on polygonal meshes using a posteriori MOOD limiting*, Computers and Fluids, 136 (2016), 83-103.
- [3] W. Boscheri, M. Dumbser, *A direct Arbitrary-Lagrangian Eulerian ADER-WENO finite volume scheme on unstructured tetrahedral meshes for conservative and non-conservative hyperbolic systems in 3D*, Journal of Computational Physics, 275 (2014), 484-523.
- [4] D. E. Burton, N. R. Morgan, M. R. J. Charest, M. A. Kenamond, J. Fung, *Compatible, energy conserving, bounds preserving remap of hydrodynamic fields for an extended ALE scheme*, Journal of Computational Physics, 355 (2018), 492-533.
- [5] M. Castro, B. Costa, W. S. Don, *High order weighted essentially non-oscillatory WENO-Z schemes for hyperbolic conservation laws*, Journal of Computational Physics, 230 (2011), 1766-1792.
- [6] J. Cheng, C.-W. Shu, *A high order accurate conservative remapping method on staggered meshes*, Applied Numerical Mathematics, 58 (2008), 1042-1060.
- [7] J. Cheng, C.-W. Shu, *A third order conservative Lagrangian type scheme on curvilinear meshes for the compressible Euler equation*, Communications in Computational Physics, 4 (2008), 1008-1024.
- [8] J. Cheng, C.-W. Shu, *Positivity-preserving Lagrangian scheme for multi-material compressible flow*, Journal of Computational Physics, 257 (2014), 143-168.

- [9] S. Clain, S. Diot, R. Loubere, *A high-order finite volume method for hyperbolic systems: Multi-dimensional Optimal Order Detection (MOOD)*, Journal of Computational Physics, 230 (2011), 4028-4050.
- [10] J. K. Dukowicz, *Conservative rezoning (remapping) for general quadrilateral meshes*, Journal of Computational Physics, 54 (1984), 411-424.
- [11] J. K. Dukowicz, J. R. Baumgardner, *Incremental remapping as a transport/advection algorithm*, Journal of Computational Physics, 160 (2000), 318-335.
- [12] J. K. Dukowicz, J. W. Kodis, *Accurate conservative remapping (rezoning) for arbitrary Lagrangian-Eulerian computations*, SIAM Journal on Scientific and Statistical Computing, 8 (1987), 305-321.
- [13] P. E. Farrell, M. D. Piggott, C. C. Pain, G. J. Gorman, C.R. Wilson, *Conservative interpolation between unstructured meshes via supermesh construction*, Computer Methods in Applied Mechanics and Engineering, 198 (2009), 2632-2642.
- [14] J. Grandy, *Conservative remapping and region overlays by intersecting arbitrary polyhedra*, Journal of Computational Physics, 148 (1999), 433-466.
- [15] A. Harten, B. Engquist, S. Osher, S. Chakravarthy, *Uniformly high order accurate essentially non-oscillatory schemes, III*, Journal of Computational Physics, 71 (1986), 231-303.
- [16] C. W. Hirt, A. Amsden, J. L. Cook, *An arbitrary Lagrangian-Eulerian computing method for all flow speeds*, Journal of Computational Physics, 14 (1974), 227-253.
- [17] G.-S. Jiang, C.-W. Shu, *Efficient implementation of weighted ENO schemes*, Journal of Computational Physics, 126 (1996), 202-228.

- [18] M. Klima, M. Kucharik, M. Shashkov, *Local error analysis and comparison of the swept- and intersection- based remapping methods*, Communications in Computational Physics, 21 (2017), 526-558.
- [19] M. Klima, M. Kucharik, M. Shashkov, *Combined swept region and intersection-based single-material remapping method*, International Journal for Numerical Methods in Fluids, 85 (2017), 363-382.
- [20] M. Kucharik, J. Breil, S. Galera, P.-H. Maire, M. Berndt, M. Shashkov, *Hybrid remap for multi-material ALE*, Computers and Fluids, 46 (2011), 293-297.
- [21] M. Kucharik, M. Shashkov, *One-step hybrid remapping algorithm for multi-material arbitrary Lagrangian-Eulerian methods*, Journal of Computational Physics, 231 (2012), 2851-2864.
- [22] M. Kucharik, M. Shashkov, *Conservative multi-material remap for staggered multi-material arbitrary Lagrangian-Eulerian methods*, Journal of Computational Physics, 258 (2014), 268-304.
- [23] M. Kucharik, M. Shashkov, B. Wendroff, *An efficient linearity-and-bound-preserving remapping method*, Journal of Computational Physics, 188 (2003), 462-471.
- [24] K. Lipnikov, N. Morgan, *A high-order conservative remap for discontinuous Galerkin schemes on curvilinear polygonal meshes*, Journal of Computational Physics, 399 (2019), 108931.
- [25] R. Liska, M. Shashkov, P. Vachal, B. Wendroff, *Optimization-based synchronized flux-corrected conservative interpolation (remapping) of mass and momentum for arbitrary Lagrangian-Eulerian methods*, Journal of Computational Physics, 229 (2010), 1467-1497.
- [26] X.-D. Liu, S. Osher, T. Chan, *Weighted essentially non-oscillatory schemes*, Journal of Computational Physics, 115 (1994), 200-212.

- [27] R. Loubere, M. Shashkov, *A subcell remapping method on staggered polygonal grids for Arbitrary-Lagrangian-Eulerian methods*, Journal of Computational Physics, 209 (2005), 105-138.
- [28] A. Marboeuf, [A. Claisse](#), P. L. Tallec, *Conservative and entropy controlled remap for multi-material ALE simulations with space-staggered schemes*, Journal of Computational Physics, 390 (2019), 66-92.
- [29] L. G. Margolin, M. Shashkov, *Second-order sign-preserving conservative interpolation (remapping) on general grids*, Journal of Computational Physics, 184 (2003), 266-298.
- [30] L. G. Margolin, M. Shashkov, *Remapping, recovery and repair on a staggered grid*, Computer Methods in Applied Mechanics and Engineering, 193 (2004), 4139-4155.
- [31] S. Menon, D. P. Schmidt, *Conservative interpolation on unstructured polyhedral meshes: An extension of the supermesh approach to cell-centered finite-volume variables*, Computer Methods in Applied Mechanics and Engineering, 200 (2011), 2797-2804.
- [32] P. J. O'Rourke, M. S. Sahota, *A variable explicit/implicit numerical method for calculating advection on unstructured meshes*, Journal of Computational physics, 143 (1998), 312-345.
- [33] [D. Powell](#), [T. Abel](#), *An exact general remeshing scheme applied to physically conservative voxelization*, Journal of Computational Physics, 297 (2015), 340-356.
- [34] J. D. Ramshaw, *Conservative rezoning algorithm for generalized two-dimensional meshes*, Journal of Computational Physics, 59 (1985), 193-199.
- [35] I. Sutherland, G. W. Hodgman, *Reentrant Polygon Clipping*, Communications of the ACM, 17 (1974), 32-42.
- [36] [K. Sugihara](#), *A robust and consistent algorithm for intersecting convex polyhedra*, Comput. Graph. Forum, 13 (3) (1994), 45-54.

- [37] C. Wang, X. Zhang, C.-W. Shu, J. Ning, *Robust high order discontinuous Galerkin schemes for two-dimensional gaseous detonations*, Journal of Computational Physics, 231 (2012), 653-665.
- [38] X. Zhang, C.-W. Shu, *On maximum-principle-satisfying high order schemes for scalar conservation laws*, Journal of Computational Physics, 229 (2010), 3091-3120.
- [39] X. Zhang, C.-W. Shu, *On positivity preserving high order discontinuous Galerkin schemes for compressible Euler equations on rectangular meshes*, Journal of Computational Physics, 229 (2010), 8918-8934.
- [40] X. Zhang, C.-W. Shu, *Maximum-principle-satisfying and positivity-preserving high-order schemes for conservation laws: survey and new developments*, Proceedings of the Royal Society A, 467 (2011), 2752-2776.
- [41] J. Zhu, C.-W. Shu, *A new type of multi-resolution WENO schemes with increasingly higher order of accuracy*, Journal of Computational Physics, 375 (2018), 659-683.
- [42] J. Zhu, C.-W. Shu, *A new type of multi-resolution WENO schemes with increasingly higher order of accuracy on triangular meshes*, Journal of Computational Physics, 392 (2019), 19-33.