mathematics

MDPI

Article

New Third-Order Finite Volume Unequal-Sized WENO Lagrangian Schemes for Solving Euler Equations

Yan Tan [1], Hui Lv [1,2] and Jun Zhu [1,*]

[1] State Key Laboratory of Mechanics and Control for Aerospace Structures, Key Laboratory of Mathematical Modelling and High Performance Computing of Air Vehicles (NUAA), Ministry of Industry and Information Technology (MIIT), Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China; tanyan@nuaa.edu.cn (Y.T.); lvhui207@gmail.com (H.L.)
[2] School of Finance and Mathematics, Huainan Normal University, Huainan 232038, China
* Correspondence: zhujun@nuaa.edu.cn

Abstract: In this paper, new third-order finite volume unequal-sized weighted essentially non-oscillatory (US-WENO) Lagrangian schemes are designed to solve Euler equations in two and three dimensions. The spatial reconstruction procedures are implemented by using a convex combination of a quadratic polynomial with several linear polynomials specified on unequal-sized stencils, so the new US-WENO Lagrangian schemes can achieve the designed third-order accuracy and maintain an essentially non-oscillatory property near strong discontinuities in multi-dimensions. Unlike the traditional WENO reconstruction procedures specified on unstructured meshes, the linear weights of these new two-dimensional and three-dimensional US-WENO spatial reconstructions can be selected as any positive numbers as long as their summation equals one and they are not related to the local mesh topology or the location of quadrature points. Moreover, the linear weights do not have to be recalculated even if the grid moves with the fluid, avoiding the appearance of negative linear weights, thus improving computation efficiency and robustness in multi-dimensional Lagrangian numerical simulations. Finally, extensive benchmark numerical cases are employed to display the excellent capability of the presented US-WENO Lagrangian schemes.

Keywords: Lagrangian scheme; finite volume; US-WENO reconstruction; unequal-sized stencil; high-order accuracy

MSC: 65M60; 35L65

Citation: Tan, Y.; Lv, H.; Zhu, J. New Third-Order Finite Volume Unequal-Sized WENO Lagrangian Schemes for Solving Euler Equations. *Mathematics* **2023**, *11*, 4842. https://doi.org/10.3390/math11234842

Academic Editor: Manuel Gadella

Received: 30 October 2023
Revised: 25 November 2023
Accepted: 29 November 2023
Published: 1 December 2023

1. Introduction

In this paper, new third-order finite volume unequal-sized weighted essentially non-oscillatory (US-WENO) Lagrangian schemes are developed to solve Euler equations on quadrilateral and hexahedral meshes. The new finite volume US-WENO Lagrangian schemes have some features and advantages. First, the spatial reconstruction procedures are implemented using a convex combination of different degrees of polynomials specified on associated unequal-sized spatial stencils. Compared with the high-order essentially non-oscillatory (ENO) Lagrangian scheme in [1,2], smaller spatial stencils are utilized in this new US-WENO Lagrangian scheme. Moreover, the proposed scheme can reach third-order accuracy and simulate strong discontinuity problems without introducing any additional procedures. Compared with the traditional weighted ENO (WENO) spatial reconstruction procedures specified on triangular and tetrahedral meshes [3,4], the linear weights can be set as any positive numbers as long as their sums are one. Furthermore, associated linear weights do not rely on the mesh topology or the location of quadrature points, so the designed US-WENO schemes have better computational efficiency and robustness in the multi-dimensional Lagrangian framework. Lastly, such new US-WENO spatial reconstruction procedures are simple to implement in high dimensions with high-order accuracy.

In computational fluid dynamics (CFD), there are two main perspectives: the Eulerian framework and the Lagrangian framework. Each framework has its own strengths and limitations. The Eulerian method describes the fluid motion with time-independent meshes and has been widely used in various CFD simulations. The Lagrangian method can obtain sharper contact discontinuity solutions [1,5], as the grid moves with the fluid. A more general framework is arbitrary Lagrangian–Eulerian (ALE), which combines the best properties of the above two methods. Generally, there are two types of Lagrangian methods: the staggered method and the cell-centered method. The staggered method [6–11] specifies all of the fluid's thermodynamic variables in the center of cells while defining velocity at the vertex. The cell-centered method [1,2,6,12–19] specifies physical variables and velocity in the center of cells, making it simple to achieve high-order accuracy. Therefore, we will construct the scheme based on the cell-centered Lagrangian method for its simplicity in achieving high-order accuracy. The first finite volume Lagrangian scheme was developed by Munz [20] in 1994. Later, Després et al. [21,22] developed the finite volume cell-centered Lagrangian scheme in multi-dimensions. In 2007, Maire et al. [16] constructed a cell-centered Lagrangian scheme in two dimensions by introducing additional freedoms on each edge and extending it to second-order accuracy [17]. Furthermore, the high-order scheme has many superior properties, such as smaller numerical errors and sharper shock transitions, so Lagrangian schemes with high-order accuracy need to be developed. ENO schemes [23,24] and WENO schemes [3,4,25–27], as classical high-order schemes, have achieved successful applications in many fields due to their superior properties. Therefore, depending on high-order ENO reconstruction, Cheng and Shu [1] constructed a class of Lagrangian schemes for Euler equations in Cartesian and cylindrical coordinates. In addition, they obtained a scheme up to second-order accuracy in two dimensions owing to geometric errors in the straight-line edge. Later, Cheng and Shu [2] achieved third-order accuracy by using curved quadrilateral meshes with high-order ENO and crude WENO reconstructions. Dumbser et al. [28–32] developed a class of high-order Lagrangian one-step ADER-WENO finite volume schemes for hyperbolic systems on triangular and tetrahedral meshes, and achieved spatial accuracy mainly by using associated robust WENO reconstructions [33,34]. Meanwhile, the ALE method has also been greatly developed, such as is shown in [35–43].

However, the traditional WENO schemes [3,4,44] attain high-order accuracy by combining lower-order polynomials in a nonlinear fashion. The linear weights are decided by a linear system related to the selection of stencils, the topological mesh relation, and the position of quadrature points. The high-order accuracy relying on linear weights leads to a number of problems. One is that the negative weights will appear if the quadrature points are not appropriately selected [44,45], so some additional procedures are needed to deal with such drawbacks [44]. Another is that the linear weights do not even exist when the computational meshes [4] are stiff enough. The robust WENO scheme [33,34] is a new way to deal with this, as its linear weights can be artificially set when the optimal order is not achieved and too many stencils are introduced. The above problems are more serious when the traditional WENO reconstruction [3,4,44] is applied to the Lagrangian method, in which the computation mesh moves with the fluid, inevitably worsening the mesh quality. In addition, the linear weights must be updated whenever the mesh moves since the linear weights rely on the mesh topological relation. So, the efficiency of multi-dimensional Lagrangian simulations will significantly decrease. To solve the drawbacks mentioned above and improve the computational efficiency and robustness without destroying the designed high-order accuracy, we proposed high-order US-WENO Lagrangian schemes by adopting the ideas of unequal-sized spatial reconstructions [46–48]. By constructing the polynomials of different degrees specified on unequal-sized stencils, the linear weights can be chosen artificially, and they do not depend on local mesh topology or the position of quadrature points. Meanwhile, the linear weights do not need to be updated even if the grid moves with the fluid, avoiding the appearance of negative linear weights and thus improving the computation efficiency and robustness of multi-dimensional Lagrangian simulations. This paper constructs a new third-order finite volume US-WENO Lagrangian scheme for Euler

equations on two-dimensional quadrilateral and three-dimensional hexahedral meshes. The proposed US-WENO Lagrangian scheme has high-order accuracy, computational efficiency, and robustness. Extensive numerical examples are given to evaluate the superior capabilities of the designed US-WENO Lagrangian schemes.

The remainder of this paper is organized as follows. The algorithm of the US-WENO Lagrangian schemes for solving multi-dimensional Euler equations is introduced in Section 2. In Section 3, numerical examples are presented to illustrate the capability of the scheme developed in this paper. Finally, Section 4 gives some concluding remarks.

2. New US-WENO Lagrangian Scheme

This section proposes a new finite volume US-WENO Lagrangian scheme for solving Euler equations in two and three dimensions. Like in [1,5], the Euler equations can be expressed as follows:

$$\frac{d}{dt}\int_{\Omega(t)} \mathbf{U} d\Omega + \int_{\Gamma(t)} \mathbf{F} d\Gamma = 0, \tag{1}$$

where $\Omega(t)$ denotes the moving control volume enclosed by its boundary $\Gamma(t)$. The conserved variables vector $\mathbf{U}$ and the flux vector $\mathbf{F}$ are defined as follows:

$$\mathbf{U} = \begin{pmatrix} \rho \\ \mathbf{M} \\ E \end{pmatrix}, \qquad \mathbf{F} = \begin{pmatrix} (\mathbf{u} - \dot{\mathbf{x}}) \cdot \mathbf{n}\rho \\ (\mathbf{u} - \dot{\mathbf{x}}) \cdot \mathbf{n}\mathbf{M} + p \cdot \mathbf{n} \\ (\mathbf{u} - \dot{\mathbf{x}}) \cdot \mathbf{n}E + p\mathbf{u} \cdot \mathbf{n} \end{pmatrix}, \tag{2}$$

where ρ denotes the density, $\mathbf{u}$ denotes the velocity, $\mathbf{M} = \rho\mathbf{u}$ denotes the momentum, E denotes the total energy, and p denotes the pressure. $\dot{\mathbf{x}}$ is the velocity of the control volume boundary $\Gamma(t)$ and $\mathbf{n}$ denotes the unit outward normal vector of $\Gamma(t)$.

The Euler equation, Equation (1), is closed by adding an equation of state (EOS), with the general form shown below

$$p = p(\rho, e), \tag{3}$$

where $e = \frac{E}{\rho} - \frac{1}{2}|\mathbf{u}|^2$ is the specific internal energy. When considering the ideal gas, the equation of state has a simple form:

$$p = (\gamma - 1)\rho e, \tag{4}$$

where γ is the ratio of the specific heats. In this paper, the governing equation, Equation (1), is considered under the Lagrangian framework, and it can be assumed that $\dot{\mathbf{x}} = \mathbf{u}$, so the vectors (2) have the following simple form:

$$\mathbf{U} = \begin{pmatrix} \rho \\ \mathbf{M} \\ E \end{pmatrix}, \qquad \mathbf{F} = \begin{pmatrix} 0 \\ p \cdot \mathbf{n} \\ p\mathbf{u} \cdot \mathbf{n} \end{pmatrix}. \tag{5}$$

Next, the two-dimensional and three-dimensional algorithms for solving Euler equations will be described separately in the following.

2.1. Two-Dimensional Case

Two-dimensional spatial domain Ω can be dividied into $N_x \times N_y$ computational cells. A quadrilateral cell $\Omega_{i,j}$ consists of four vertices: $(x_{i,j}, y_{i,j})$, $(x_{i+1,j}, y_{i+1,j})$, $(x_{i+1,j+1}, y_{i+1,j+1})$, and $(x_{i,j+1}, y_{i,j+1})$, and $S_{i,j}$ denotes its area. The velocity of vertex $(x_{i,j}, y_{i,j})$ is denoted as $(u_{i,j}, v_{i,j})$, and other variables are saved in cell averages, which form in the center of cells. For instance, $\bar{\rho}_{i,j}$, $\bar{M}^x_{i,j}$, $\bar{M}^y_{i,j}$, and $\bar{E}_{i,j}$ are defined below:

$$\begin{aligned}
\bar{\rho}_{i,j} &= \frac{1}{S_{i,j}} \iint_{\Omega_{i,j}} \rho(x,y)dxdy, \\
\bar{M}^x_{i,j} &= \frac{1}{S_{i,j}} \iint_{\Omega_{i,j}} M_x(x,y)dxdy, \\
\bar{M}^y_{i,j} &= \frac{1}{S_{i,j}} \iint_{\Omega_{i,j}} M_y(x,y)dxdy, \\
\bar{E}_{i,j} &= \frac{1}{S_{i,j}} \iint_{\Omega_{i,j}} E(x,y)dxdy.
\end{aligned} \tag{6}$$

2.1.1. Finite Volume Discretization

The conservative semi-discrete formula for two-dimensional Euler equations is given below [1,2]:

$$\frac{d}{dt}\begin{pmatrix} \bar{\rho}_{i,j}S_{i,j} \\ \bar{M}^x_{i,j}S_{i,j} \\ \bar{M}^y_{i,j}S_{i,j} \\ \bar{E}_{i,j}S_{i,j} \end{pmatrix} = L(\mathbf{U}_{i,j}) = -\int_{\partial\Omega_{i,j}} \hat{\mathbf{F}} \cdot \mathbf{n} dl. \tag{7}$$

We use the Gaussian numerical quadrature rules to calculate the integration, then obtain

$$\int_{\partial\Omega_{i,j}} \hat{\mathbf{F}} \cdot \mathbf{n} dl \approx \sum_{\ell=1}^{4} \Delta l^{\ell} \sum_{m=1}^{2} \omega_m \hat{\mathbf{F}}(\mathbf{U}^-(G^{\ell}_m), \mathbf{U}^+(G^{\ell}_m)) \cdot \mathbf{n}_{\ell}, \tag{8}$$

where Δl^{ℓ} represents the length of the boundary edge ℓ and $G^{\ell}_m = (x_{G^{\ell}_m}, y_{G^{\ell}_m})$ is the m-th Gaussian quadrature point on this edge, ω_m are the two-point Gaussian quadrature weights, and $\mathbf{n}_{\ell}$ represents outward an unit normal of the ℓ-th edge. Here $\hat{\mathbf{F}}(\mathbf{U}^-(G^{\ell}_m), \mathbf{U}^+(G^{\ell}_m))$ is numerical flux, and we apply an HLLC flux [5,49] to guarantee without mass across the interface. $\mathbf{U}^-(G^{\ell}_m)$ and $\mathbf{U}^+(G^{\ell}_m)$ are reconstructed values of $\mathbf{U}$ on different quadrature points. The following subsection will describe the procedures of spatial reconstruction.

2.1.2. Unequal-Sized WENO Reconstruction

The procedures for obtaining the high-order approximation values of $\mathbf{U}^-(G^{\ell}_m)$ by the unequal-sized WENO spatial reconstruction are given in this subsection. To facilitate presentation, as shown in Figure 1, we rename cell $\Omega_{i,j} = I_5$ and its eight adjacent cells as $I_1, \ldots, I_9$.

I_1	I_2	I_3
I_4	I_5	I_6
I_7	I_8	I_9

Figure 1. The target cell $\Omega_{i,j} = I_5$ and its eight adjacent cells.

Step 1. We first select a big stencil as $T_0 = \{I_1, \ldots, I_9\}$ and construct a quadratic polynomial $q_0(x,y) \in span\{1, \frac{(x-x_0)}{|I_5|^{\frac{1}{2}}}, \frac{(y-y_0)}{|I_5|^{\frac{1}{2}}}, \frac{(x-x_0)^2}{|I_5|}, \frac{(x-x_0)(y-y_0)}{|I_5|}, \frac{(y-y_0)^2}{|I_5|}\}$ on T_0, in which (x_0, y_0) represents the barycenter of target cell I_5, and $|I_5|$ denotes the cell's area. The quadratic polynomial $q_0(x,y)$ is obtained by the least squares method [3]:

$$\frac{1}{|I_5|} \int_{I_5} q_0(x,y)dxdy = \bar{U}_5, \tag{9}$$

and

$$q_0(x,y) = argmin \sum_{I_l \in T_0 \setminus I_5} (\frac{1}{|I_l|} \int_{I_l} q_0(x,y)dxdy - \bar{U}_l)^2. \tag{10}$$

Then, four linear polynomials $q_\ell(x,y) \in span\{1, \frac{x-x_0}{|I_5|^{\frac{1}{2}}}, \frac{y-y_0}{|I_5|^{\frac{1}{2}}}\}$ are constructed, which are defined on sub-stencils $T_\ell, \ell = 1,2,3,4$. They satisfy

$$\begin{aligned} &\frac{1}{|I_l|}\int_{I_l} q_\ell(x,y)dxdy = \bar{U}_l, \quad I_l \in T_\ell, \quad \ell = 1,2,3,4, \\ &T_1 = \{I_2, I_4, I_5\}, T_2 = \{I_2, I_5, I_6\}, T_3 = \{I_4, I_5, I_8\}, T_4 = \{I_5, I_6, I_8\}. \end{aligned} \tag{11}$$

Step 2. We take $p_\ell(x,y) = q_\ell(x,y), \ell = 1,2,3,4$, and

$$p_0(x,y) = \frac{1}{\gamma_0}q_0(x,y) - \sum_{\ell=1}^{4}\frac{\gamma_\ell}{\gamma_0}p_\ell(x,y), \tag{12}$$

to obtain the equal expressions for different degree polynomials, where γ_ℓ are the linear weights and they satisfy $\sum_{\ell=0}^{4}\gamma_\ell = 1, \gamma_0 \neq 0$. And quadratic polynomial $q_0(x,y)$ can be rewritten as

$$q_0(x,y) = \sum_{\ell=0}^{4}\gamma_\ell p_\ell(x,y), \tag{13}$$

so the designed third-order accuracy can be recovered. The linear weights need to be replaced by associated nonlinear weights to repress the spurious oscillations near strong discontinuities.

Step 3. We compute the smoothness indicators $\beta_\ell(\ell = 0,\ldots,4)$, representing the smooth measurement of polynomial $p_\ell(x,y), \ell = 0,\ldots,4$, and are calculated following the same methodology in [3,25]:

$$\beta_\ell = \sum_{|\alpha|=1}^{r}\int_{I_5}|I_5|^{|\alpha|-1}\left(\frac{\partial^{|\alpha|}}{\partial x^{\alpha_1}\partial y^{\alpha_2}}p_\ell(x,y)\right)^2 dxdy, \ell = 0,\ldots,4, \tag{14}$$

where $\alpha = (\alpha_1, \alpha_2)$, $|\alpha| = \alpha_1 + \alpha_2$. We set $r = 2$ when $\ell = 0$, and $r = 1$ when $\ell = 1,\ldots,4$.

Step 4. We calculate the nonlinear weights depending on the linear weights and smoothness indicators. Following [46–48], they are computed by

$$\tau = \left(\frac{\sum_{\ell=1}^{4}|\beta_0 - \beta_\ell|}{4}\right)^2, \ \bar{\omega}_\ell = \gamma_\ell\left(1 + \frac{\tau}{\beta_\ell + \varepsilon}\right), \ \omega_\ell = \frac{\bar{\omega}_\ell}{\sum_{ll=0}^{4}\bar{\omega}_{ll}}, \ \ell = 0,\ldots,4. \tag{15}$$

Here, $\varepsilon = 10^{-6}$ is a small positive number to avoid a zero denominator.

Step 5. The final reconstruction polynomial is

$$\mathcal{Q}(x,y) = \sum_{\ell=0}^{4}\omega_\ell p_\ell(x,y), \tag{16}$$

and the reconstructed values on different quadrature points can be obtained:

$$\begin{aligned} &\mathbf{U}^-(G_m^\ell) = \mathbf{U}^-(x_{G_m^\ell}, y_{G_m^\ell}) = \mathcal{Q}(x_{G_m^\ell}, y_{G_m^\ell}), \\ &m = 1,2, \quad \ell = 1,\ldots,4. \end{aligned} \tag{17}$$

The reconstruction of polynomials on the neighboring cells can be carried out similarly to obtain the reconstructed values of $\mathbf{U}^+(G_m^\ell)$.

2.1.3. The Velocity of Vertex

In the Lagrangian method, the computational mesh needs to change with fluids, so the velocity of the vortex must be considered. We calculate the vertex velocity by the simple Roe-average [50] method. It shows good performance in numerical experiments and has a simple form, as follows:

$$u_{i,j} = \frac{\sum\limits_{\ell \in \mathcal{A}} \sqrt{\rho_\ell} u_\ell}{\sum\limits_{\ell \in \mathcal{A}} \sqrt{\rho_\ell}}, \quad v_{i,j} = \frac{\sum\limits_{\ell \in \mathcal{A}} \sqrt{\rho_\ell} v_\ell}{\sum\limits_{\ell \in \mathcal{A}} \sqrt{\rho_\ell}}, \tag{18}$$

where $\mathcal{A}$ is the cell set that contains the vortex (i, j).

2.1.4. Time Discretization

The SSP Runge–Kutta method [51] is used for time marching of semi-discrete schemes (7). The computation mesh changes as time advances, so we need to consider the mesh changes at each stage. Once the mesh moves at each stage, the location of vertexes and each cell's area must be updated accordingly. Hence, the third-order Runge–Kutta time discretization method in the Lagrangian framework is given as:

$$\begin{cases} x^{(1)} = x^n + u^n \Delta t^n, y^{(1)} = y^n + v^n \Delta t^n, \\ S^{(1)} = S(x^{(1)}, y^{(1)}), \\ \mathbf{U}^{(1)} S^{(1)} = \mathbf{U}^n S^n + \Delta t^n L(\mathbf{U}^n), \\ \\ x^{(2)} = \frac{3}{4} x^n + \frac{1}{4}[x^{(1)} + u^{(1)} \Delta t^n], y^{(2)} = \frac{3}{4} y^n + \frac{1}{4}[y^{(1)} + v^{(1)} \Delta t^n], \\ S^{(2)} = S(x^{(2)}, y^{(2)}), \\ \mathbf{U}^{(2)} S^{(2)} = \frac{3}{4} \mathbf{U}^n S^n + \frac{1}{4}[\mathbf{U}^{(1)} S^{(1)} + \Delta t^n L(\mathbf{U}^{(1)})], \\ \\ x^{n+1} = \frac{1}{3} x^n + \frac{2}{3}[x^{(2)} + u^{(2)} \Delta t^n], y^{n+1} = \frac{1}{3} y^n + \frac{2}{3}[y^{(2)} + v^{(2)} \Delta t^n], \\ S^{n+1} = S(x^{n+1}, y^{n+1}) \\ \mathbf{U}^{n+1} S^{n+1} = \frac{1}{3} \mathbf{U}^n S^n + \frac{2}{3}[\mathbf{U}^{(2)} S^{(2)} + \Delta t^n L(\mathbf{U}^{(2)})]. \end{cases} \tag{19}$$

As used in [1], the computation time step Δt^n is decided by

$$\Delta t^n = CFL \cdot min \frac{\Delta l_{i,j}^n}{c_{i,j}^n}, \quad i = 1, \ldots, N_x, j = 1, \ldots, N_y, \tag{20}$$

where $\Delta l_{i,j}^n$ is the shortest edge of cell $\Omega_{i,j}$ and $c_{i,j}^n$ denotes the sound speed.

2.2. Three-Dimensional Case

Three-dimensional spatial domain Ω can be divided into $N_x \times N_y \times N_z$ computational cells. $\Omega_{i,j,k}$ is a hexahedral cell, which consists of eight vertices: $(x_{i,j,k}, y_{i,j,k}, z_{i,j,k})$, $(x_{i+1,j,k}, y_{i+1,j,k}, z_{i+1,j,k})$, $(x_{i+1,j+1,k}, y_{i+1,j+1,k}, z_{i+1,j+1,k})$, $(x_{i,j+1,k}, y_{i,j+1,k}, z_{i,j+1,k})$, $(x_{i,j,k+1}, y_{i,j,k+1}, z_{i,j,k+1})$, $(x_{i+1,j,k+1}, y_{i+1,j,k+1}, z_{i+1,j,k+1})$, $(x_{i+1,j+1,k+1}, y_{i+1,j+1,k+1}, z_{i+1,j+1,k+1})$, and $(x_{i,j+1,k+1}, y_{i,j+1,k+1}, z_{i,j+1,k+1})$, and $V_{i,j,k}$ denotes its volume. The fluid velocity $(u_{i,j,k}, v_{i,j,k}, w_{i,j,k})$ is defined at the vertex $(x_{i,j,k}, y_{i,j,k}, z_{i,j,k})$, and other variables are saved at the cell center in cell averages form. For instance, $\bar{\rho}_{i,j,k}$, $\bar{M}_{i,j,k}^x$, $\bar{M}_{i,j,k}^y$, $\bar{M}_{i,j,k}^z$, and $\bar{E}_{i,j,k}$ are given as follows:

$$\begin{aligned} \bar{\rho}_{i,j,k} &= \frac{1}{V_{i,j,k}} \iiint_{\Omega_{i,j,k}} \rho(x,y,z) dx dy dz, \\ \bar{M}_{i,j,k}^x &= \frac{1}{V_{i,j,k}} \iiint_{\Omega_{i,j,k}} M_x(x,y,z) dx dy dz, \\ \bar{M}_{i,j,k}^y &= \frac{1}{V_{i,j,k}} \iiint_{\Omega_{i,j,k}} M_y(x,y,z) dx dy dz, \\ \bar{M}_{i,j,k}^z &= \frac{1}{V_{i,j,k}} \iiint_{\Omega_{i,j,k}} M_z(x,y,z) dx dy dz, \\ \bar{E}_{i,j,k} &= \frac{1}{V_{i,j,k}} \iiint_{\Omega_{i,j,k}} E(x,y,z) dx dy dz. \end{aligned} \tag{21}$$

2.2.1. Finite Volume Discretization

The conservative semi-discrete formula for three-dimensional Euler equations can be given as

$$\frac{d}{dt}\begin{pmatrix} \bar{\rho}_{i,j,k}V_{i,j,k} \\ \bar{M}^x_{i,j,k}V_{i,j,k} \\ \bar{M}^y_{i,j,k}V_{i,j,k} \\ \bar{M}^z_{i,j,k}V_{i,j,k} \\ \bar{E}_{i,j,k}V_{i,j,k} \end{pmatrix} = L(\mathbf{U}_{i,j,k}) = -\int_{\partial\Omega_{i,j,k}} \hat{\mathbf{F}}\cdot\mathbf{n}ds. \tag{22}$$

We apply the Gaussian numerical quadrature rules to calculate the integration, and obtain

$$\int_{\partial\Omega_{i,j,k}} \hat{\mathbf{F}}\cdot\mathbf{n}ds \approx \sum_{\ell=1}^{6} |\partial\Omega^{\ell}_{i,j,k}| \sum_{m=1}^{4} \omega_m \hat{\mathbf{F}}(\mathbf{U}^-(G^{\ell}_m), \mathbf{U}^+(G^{\ell}_m))\cdot\mathbf{n}_{\ell}, \tag{23}$$

where $|\partial\Omega^{\ell}_{i,j,k}|$ is the area of the face ℓ and $G^{\ell}_m = (x_{G^{\ell}_m}, y_{G^{\ell}_m}, z_{G^{\ell}_m})$ is the m-th quadrature point on this face, ω_m are the four-point Gaussian quadrature weights, and $\mathbf{n}_{\ell}$ represents outward unit normal of face $\partial\Omega^{\ell}_{i,j,k}$. Here $\hat{\mathbf{F}}(\mathbf{U}^-(G^{\ell}_m), \mathbf{U}^+(G^{\ell}_m))$ is numerical flux, and we also apply an HLLC numerical flux [5,49] to guarantee without mass across the interface. $\mathbf{U}^-(G^{\ell}_m)$ and $\mathbf{U}^+(G^{\ell}_m)$ are reconstructed values of $\mathbf{U}$ on different quadrature points. The following subsection will give the procedures of spatial reconstruction.

2.2.2. Unequal-Sized WENO Reconstruction

The three-dimensional unequal-sized WENO reconstruction procedures are given in this subsection to obtain the high-order approximation values $\mathbf{U}^-(G^{\ell}_m)$. For the sake of understanding, let us make a notation here: $\Omega_{i+i_0,j+j_0,k+k_0}$ represents the cell that moves the target cell $\Omega_{i,j,k}$ by i_0 cells in x positive direction, j_0 cells in y positive direction, and k_0 cells in z positive direction. And the target cell $\Omega_{i,j,k}$ is denoted as Ω_0.

Step 1. We first select a big stencil,

$$T_0 = \{\Omega_{i+i_0,j+j_0,k+k_0, i_0,j_0,k_0=-1,0,1, i_0\cdot j_0\cdot k_0\neq\pm1}\} = \{\Omega_l, l = 0,\ldots,18\}, \tag{24}$$

which is shown in Figure 2, and construct a quadratic polynomial $q_0(x,y,z) \in span\{1, \frac{(x-x_0)}{|\Omega_0|^{\frac{1}{3}}}, \frac{(y-y_0)}{|\Omega_0|^{\frac{1}{3}}}, \frac{(z-z_0)}{|\Omega_0|^{\frac{1}{3}}}, \frac{(x-x_0)^2}{|\Omega_0|^{\frac{2}{3}}}, \frac{(x-x_0)(y-y_0)}{|\Omega_0|^{\frac{2}{3}}}, \frac{(x-x_0)(z-z_0)}{|\Omega_0|^{\frac{2}{3}}}, \frac{(y-y_0)^2}{|\Omega_0|^{\frac{2}{3}}}, \frac{(y-y_0)(z-z_0)}{|\Omega_0|^{\frac{2}{3}}}, \frac{(z-z_0)^2}{|\Omega_0|^{\frac{2}{3}}}\}$ based on T_0, in which (x_0, y_0, z_0) is the barycenter of target cell Ω_0, and $|\Omega_0|$ is the cell's volume. The quadratic polynomial $q_0(x,y,z)$ is also obtained by the least squares method [3]:

$$\frac{1}{|\Omega_0|}\int_{\Omega_0} q_0(x,y,z)dxdydz = \bar{U}_0, \tag{25}$$

and

$$q_0(x,y,z) = argmin \sum_{\Omega_{\ell}\in T_0\setminus\Omega_0} (\frac{1}{|\Omega_{\ell}|}\int_{\Omega_{\ell}} q_0(x,y,z)dxdydz - \bar{U}_{\ell})^2. \tag{26}$$

In this paper, the quadratic polynomial $q_0(x,y,z)$ is constructed by using the associated nineteen hexahedral cells to maintain a balance performance between numerical stability and computational cost. Then, we construct eight linear polynomials $q_{\ell}(x,y,z) \in span\{1, \frac{(x-x_0)}{|\Omega_0|^{\frac{1}{3}}}, \frac{(y-y_0)}{|\Omega_0|^{\frac{1}{3}}}, \frac{(z-z_0)}{|\Omega_0|^{\frac{1}{3}}}\}$ defined on smaller stencils $T_{\ell}, \ell = 1,\ldots,8$, respectively. And the eight smaller stencils are

$$\begin{aligned} T_1 &= \{\Omega_{i,j,k}, \Omega_{i-1,j,k}, \Omega_{i,j-1,k}, \Omega_{i,j,k+1}\}, \quad T_2 = \{\Omega_{i,j,k}, \Omega_{i,j-1,k}, \Omega_{i+1,j,k}, \Omega_{i,j,k+1}\}, \\ T_3 &= \{\Omega_{i,j,k}, \Omega_{i+1,j,k}, \Omega_{i,j+1,k}, \Omega_{i,j,k+1}\}, \quad T_4 = \{\Omega_{i,j,k}, \Omega_{i,j+1,k}, \Omega_{i-1,j,k}, \Omega_{i,j,k+1}\}, \\ T_5 &= \{\Omega_{i,j,k}, \Omega_{i-1,j,k}, \Omega_{i,j-1,k}, \Omega_{i,j,k-1}\}, \quad T_6 = \{\Omega_{i,j,k}, \Omega_{i,j-1,k}, \Omega_{i+1,j,k}, \Omega_{i,j,k-1}\}, \\ T_7 &= \{\Omega_{i,j,k}, \Omega_{i+1,j,k}, \Omega_{i,j+1,k}, \Omega_{i,j,k-1}\}, \quad T_8 = \{\Omega_{i,j,k}, \Omega_{i,j+1,k}, \Omega_{i-1,j,k}, \Omega_{i,j,k-1}\}. \end{aligned} \tag{27}$$

These linear polynomials satisfy

$$\frac{1}{|\Omega_l|}\int_{\Omega_l} q_\ell(x,y,z)dxdydz = \bar{U}_l, \quad \Omega_l \in T_\ell, \quad \ell = 1,\ldots,8. \tag{28}$$

Step 2. We take $p_\ell(x,y,z) = q_\ell(x,y,z), \ell = 1,\ldots,8$, and

$$p_0(x,y,z) = \frac{1}{\gamma_0}q_0(x,y,z) - \sum_{\ell=1}^{8}\frac{\gamma_\ell}{\gamma_0}p_\ell(x,y,z), \tag{29}$$

to obtain the equal expressions for different degree polynomials, where γ_ℓ are the linear weights, and they satisfy $\sum_{\ell=0}^{8}\gamma_\ell = 1, \gamma_0 \neq 0$. Moreover, we can rewrite $q_0(x,y,z)$ as

$$q_0(x,y,z) = \sum_{\ell=0}^{8}\gamma_\ell p_\ell(x,y,z), \tag{30}$$

So, the designed third-order accuracy can be recovered. The linear weights need to be replaced by associated nonlinear weights to repress the oscillations near strong discontinuities.

Step 3. We compute the smoothness indicators $\beta_\ell(\ell = 0,\ldots,8)$, representing the smooth measurement of polynomial $p_\ell(x,y,z), \ell = 0,\ldots,8$, which are calculated following the same methodology in [3,25]:

$$\beta_\ell = \sum_{|\alpha|=1}^{r}|\Omega_0|^{\frac{2|\alpha|}{3}-1}\int_{\Omega_0}\left(\frac{\partial^{|\alpha|}}{\partial x^{\alpha_1}\partial y^{\alpha_2}\partial z^{\alpha_3}}p_\ell(x,y,z)\right)^2 dxdydz, \ell = 0,\ldots,8. \tag{31}$$

where $\alpha = (\alpha_1,\alpha_2,\alpha_3)$ and $|\alpha| = \alpha_1+\alpha_2+\alpha_3$. We define r equals 2 when $\ell = 0$, and r equals 1 when $\ell = 1,\ldots,8$.

Step 4. We calculate the nonlinear weights based on the linear weights and associated smoothness indicators. Following [46–48], they are defined as follows:

$$\tau = \left(\frac{\sum_{\ell=1}^{8}|\beta_0-\beta_\ell|}{8}\right)^2, \ \bar{\omega}_\ell = \gamma_\ell\left(1+\frac{\tau}{\beta_\ell+\varepsilon}\right), \ \omega_\ell = \frac{\bar{\omega}_\ell}{\sum_{ll=0}^{8}\bar{\omega}_{ll}}, \ \ell = 0,\ldots,8. \tag{32}$$

Here, $\varepsilon = 10^{-6}$ is a small positive number to avoid a zero denominator.

Step 5. The final reconstruction polynomial is

$$\mathcal{Q}(x,y,z) = \sum_{\ell=0}^{8}\omega_\ell p_\ell(x,y,z), \tag{33}$$

and the reconstructed values on different quadrature points can be obtained:

$$\begin{aligned} &\mathbf{U}^-(G_m^\ell) = \mathbf{U}^-(x_{G_m^\ell}, y_{G_m^\ell}, z_{G_m^\ell}) = \mathcal{Q}(x_{G_m^\ell}, y_{G_m^\ell}, z_{G_m^\ell}), \\ &m = 1,\ldots,4, \quad \ell = 1,\ldots,6. \end{aligned} \tag{34}$$

The reconstruction of polynomials on the neighboring cells can be carried out similarly to obtain the reconstructed values of $\mathbf{U}^+(G_m^\ell)$.

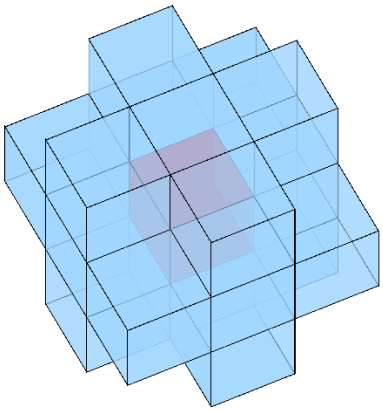

Figure 2. Stencils of target cell Ω_0 (red cubic) for 3D spatial reconstruction.

2.2.3. The Velocity of Vertex

This subsection gives the calculation method of vertex velocity in three-dimensional space. A hexahedral cell has eight vertices, which means that a vertex is shared by eight cells, so the velocity of this vertex is related to the cells that are sharing with it. We also use the simple Roe-average [50] method to calculate the velocity of the vertex, and it takes a simple form, as follows:

$$u_{i,j,k} = \frac{\sum\limits_{\ell \in \mathcal{A}} \sqrt{\rho_\ell} u_\ell}{\sum\limits_{\ell \in \mathcal{A}} \sqrt{\rho_\ell}}, v_{i,j,k} = \frac{\sum\limits_{\ell \in \mathcal{A}} \sqrt{\rho_\ell} v_\ell}{\sum\limits_{\ell \in \mathcal{A}} \sqrt{\rho_\ell}}, w_{i,j,k} = \frac{\sum\limits_{\ell \in \mathcal{A}} \sqrt{\rho_\ell} w_\ell}{\sum\limits_{\ell \in \mathcal{A}} \sqrt{\rho_\ell}}, \tag{35}$$

where $\mathcal{A}$ is the cell set that contains this vertex (i, j, k).

2.2.4. Time Discretization

In three dimensions, the SSP Runge–Kutta method is still used to advance the semi-discrete scheme (22). The variables that rely on the mesh need to be updated accordingly at each stage. So, the location of the vertices and each cell's volume need to be updated once the computational mesh moves. Because it is similar to the two-dimensional case (19), it will not be repeated here for simplicity.

The computation time step Δt^n is decided by

$$\begin{aligned} &\Delta t^n = CFL \cdot min \frac{|\Omega_{i,j,k}|}{c^n_{i,j,k}}, \\ &i = 1, \ldots, N_x, j = 1, \ldots, N_y, k = 1, \ldots, N_z, \end{aligned} \tag{36}$$

where $|\Omega_{i,j,k}|$ is the minimum area among all faces of cell $\Omega_{i,j,k}$, and $c^n_{i,j,k}$ denotes the sound speed.

3. Numerical Results

In this section, numerical examples in two and three dimensions are presented to display the superior capability of the US-WENO Lagrangian schemes. The CFL number is 0.6 for all examples. To reduce spurious oscillations, the reconstructions are carried out in the local characteristic directions [3,52]. We select four types of linear weights to verify the effect of their selection on the accuracy. The two-dimensional accuracy examples are the following: (1) $\gamma_0 = 0.96, \gamma_\ell = 0.01, \ell = 1 \ldots, 4$; (2) $\gamma_0 = 0.6, \gamma_\ell = 0.1, \ell = 1 \ldots, 4$; (3) $\gamma_0 = 0.2, \gamma_\ell = 0.2, \ell = 1 \ldots, 4$; (4) $\gamma_0 = 0.04, \gamma_\ell = 0.24, \ell = 1 \ldots, 4$. The three-dimensional accuracy examples are the following: (1) $\gamma_0 = 0.92, \gamma_\ell = 0.01, \ell = 1 \ldots, 8$; (2) $\gamma_0 = 0.6, \gamma_\ell = 0.05, \ell = 1 \ldots, 8$; (3) $\gamma_0 = 0.2, \gamma_\ell = 0.1, \ell = 1 \ldots, 8$; (4) $\gamma_0 = 0.04$, $\gamma_\ell = 0.12, \ell = 1 \ldots, 8$. Following [46–48], we set the linear weights $\gamma_0 = 0.96$, $\gamma_\ell = 0.01$,

$\ell = 1 \ldots, 4$ and $\gamma_0 = 0.92$, $\gamma_\ell = 0.01$, $\ell = 1 \ldots, 8$ in the latter two and three dimensions examples, respectively.

3.1. Accuracy Test

Example 1. *Two-dimensional Euler equations are first considered to verify the numerical accuracy. The computational domain is* $[0,2] \times [0,2]$ *and periodic boundary conditions are applied in two directions. The initial conditions are* $\rho(x,y,0) = 1 + 0.2\sin(\pi(x+y))$, $u(x,y,0) = 1$, $v(x,y,0) = 1$, *and* $p(x,y,0) = 1$. *All boundaries are defined as periodic boundary conditions. The exact density solution is* $\rho(x,y,t) = 1 + 0.2\sin(\pi(x+y-(u+v)t))$. *The final computing time is* $t = 2.0$. *The errors and numerical orders with different types of linear weights are shown in Table 1. It can be seen that the proposed scheme with different linear weights can obtain the expected third-order accuracy.*

Table 1. Accuracy test for 2D Euler equations. T = 2. L^1 and L^∞ errors.

Mesh	L^1 Error	Order	L^∞ Error	Order	L^1 Error	Order	L^∞ Error	Order
	Linear weights (1)				Linear weights (2)			
20×20	4.03×10^{-4}		1.07×10^{-3}		4.04×10^{-4}		1.07×10^{-3}	
40×40	6.44×10^{-5}	2.65	1.89×10^{-4}	2.50	6.44×10^{-5}	2.65	1.89×10^{-4}	2.50
60×60	1.98×10^{-5}	2.91	5.93×10^{-5}	2.86	1.98×10^{-5}	2.91	5.93×10^{-5}	2.86
80×80	8.42×10^{-6}	2.97	2.54×10^{-5}	2.95	8.42×10^{-6}	2.97	2.54×10^{-5}	2.95
100×100	4.33×10^{-6}	2.98	1.31×10^{-5}	2.97	4.33×10^{-6}	2.98	1.31×10^{-5}	2.97
	Linear weights (3)				Linear weights (4)			
20×20	4.04×10^{-4}		1.07×10^{-3}		4.04×10^{-4}		1.07×10^{-3}	
40×40	6.44×10^{-5}	2.65	1.89×10^{-4}	2.50	6.44×10^{-5}	2.65	1.89×10^{-4}	2.50
60×60	1.98×10^{-5}	2.91	5.93×10^{-5}	2.86	1.98×10^{-5}	2.91	5.93×10^{-5}	2.86
80×80	8.42×10^{-6}	2.97	2.54×10^{-5}	2.95	8.42×10^{-6}	2.97	2.54×10^{-5}	2.95
100×100	4.33×10^{-6}	2.98	1.31×10^{-5}	2.97	4.33×10^{-6}	2.98	1.31×10^{-5}	2.97

Example 2. *Here, we continue to test the numerical accuracy of three-dimensional Euler equations. The computational domain is* $[0,2] \times [0,2] \times [0,2]$. *The initial conditions are* $\rho(x,y,z,0) = 1 + 0.2\sin(\pi(x+y+z))$, $u(x,y,z,0) = 1$, $v(x,y,z,0) = 1$, $w(x,y,z,0) = 1$, *and* $p(x,y,z,0) = 1$ *with periodic boundary conditions. The exact density solution is* $\rho(x,y,z,t) = 1 + 0.2\sin(\pi(x+y+z-(u+v+w)t))$. *The final computing time is* $t = 1.0$. *The errors and numerical orders of accuracy are provided in Table 2, and the theoretical order is reached with four types of linear weights.*

Table 2. Accuracy test for 3D Euler equations. T = 1. L^1 and L^∞ errors.

Mesh	L^1 Error	Order	L^∞ Error	Order	L^1 Error	Order	L^∞ Error	Order
	Linear weights (1)				Linear weights (2)			
$10 \times 10 \times 10$	8.25×10^{-4}		1.62×10^{-3}		8.25×10^{-4}		1.62×10^{-3}	
$20 \times 20 \times 20$	8.52×10^{-5}	3.28	1.99×10^{-4}	3.03	8.53×10^{-5}	3.27	2.00×10^{-4}	3.02
$30 \times 30 \times 30$	2.38×10^{-5}	3.15	6.03×10^{-5}	2.94	2.38×10^{-5}	3.15	6.03×10^{-5}	2.96
$40 \times 40 \times 40$	1.05×10^{-6}	2.84	2.44×10^{-5}	3.14	1.05×10^{-6}	2.84	2.44×10^{-5}	3.14
$50 \times 50 \times 50$	5.23×10^{-6}	3.12	1.26×10^{-5}	2.96	5.23×10^{-6}	3.12	1.26×10^{-5}	2.96
	Linear weights (3)				Linear weights (4)			
$10 \times 10 \times 10$	8.26×10^{-4}		1.62×10^{-3}		8.26×10^{-4}		1.62×10^{-3}	
$20 \times 20 \times 20$	8.53×10^{-5}	3.28	2.00×10^{-4}	3.02	8.53×10^{-5}	3.28	2.00×10^{-4}	3.02
$30 \times 30 \times 30$	2.38×10^{-5}	3.15	6.03×10^{-5}	2.96	2.38×10^{-5}	3.15	6.03×10^{-5}	2.96
$40 \times 40 \times 40$	1.05×10^{-6}	2.84	2.44×10^{-5}	3.14	1.05×10^{-6}	2.84	2.44×10^{-5}	3.14
$50 \times 50 \times 50$	5.23×10^{-6}	3.12	1.26×10^{-5}	2.96	5.23×10^{-6}	3.12	1.26×10^{-5}	2.96

3.2. Two-Dimensional Lagrangian Tests

This subsection displays a series of two-dimensional Lagrangian test problems. The Sod problem [53] and Lax problem [54], Shu–Osher problem [55], Blast wave problem [56],

two-dimensional Sedov problem [1], two-dimensional Sod problem [57], and two-dimensional Saltzman problem [58] are presented to display the capability of the designed scheme.

Example 3. *We first consider two classical shock tube problems. The first is the Sod problem [53], with the initial conditions*

$$(\rho, u, v, p)^T = \begin{cases} (1, 0, 0, 2.5)^T, & x \in [0, 0.5), \\ (0.125, 0, 0, 0.25)^T, & x \in [0.5, 1]. \end{cases} \tag{37}$$

The second is the Lax problem [54], with the initial conditions

$$(\rho, u, v, p)^T = \begin{cases} (0.445, 0.698, 0, 3.528)^T, & x \in [0, 0.5), \\ (0.5, 0, 0, 0.571)^T, & x \in [0.5, 1]. \end{cases} \tag{38}$$

These two problems are calculated in the domain of $[0, 1] \times [-0.05, 0.05]$ *with* 100×10 *uniform initial cells. The density profiles of the Sod problem at* $t = 0.2$ *and the Lax problem at* $t = 0.12$ *are displayed in Figure 3. It can be seen that the designed scheme can obtain a sharp capture of discontinuities.*

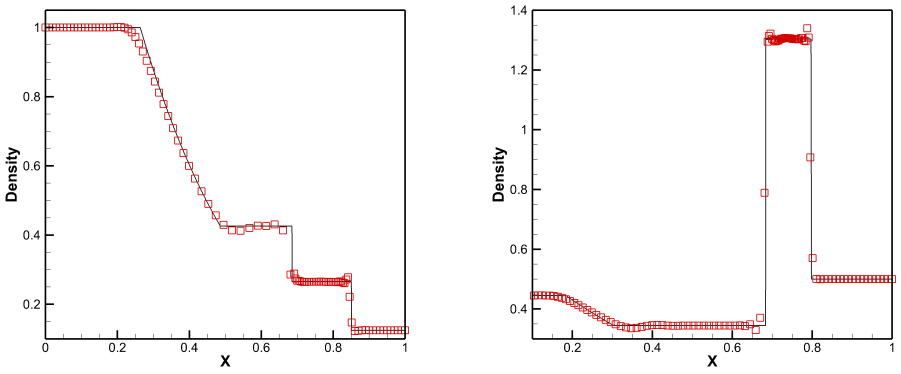

Figure 3. Sod and Lax problem. Density cut at $y = 0$. Solid line: the exact solution; squares: the results of US-WENO Lagrangian scheme.

Example 4. *We now consider the Shu–Osher problem [55], with the following initial conditions*

$$(\rho, u, v, p)^T = \begin{cases} (3.857143, 2.629369, 0, 10.333333)^T, & x \in [-10, -4), \\ (1 + 0.2\sin(5x), 0, 0, 1)^T, & x \in [-4, 5]. \end{cases} \tag{39}$$

The computation domain is $[-10, 5] \times [-0.05, 0.05]$ *with a uniform* 200×10 *cells initially. The final computing time is* $t = 1.8$. *The density* ρ *calculated by the designed scheme is presented in Figure 4, where the reference solution is the results of the fifth-order Eulerian WENO scheme [25] with 2000 cells. It can be seen that the developed scheme obtains the desirable results in a fine structure of density profile.*

Example 5. *We consider the two blast waves problem [56]. The initial conditions are*

$$(\rho, u, v, p)^T = \begin{cases} (1, 0, 0, 1000)^T, & x \in [0, 0.1), \\ (1, 0, 0, 0.01)^T, & x \in [0.1, 0.9), \\ (1, 0, 0, 100)^T, & x \in [0.9, 1.0]. \end{cases} \tag{40}$$

The computation domain is $[0, 1] \times [-0.05, 0.05]$ *with* 200×10 *cells. The computed density* ρ *at* $t = 0.038$ *is displayed in Figure 5, and the results of the fifth-order Eulerian WENO scheme [25] with 2000 cells are also plotted for comparison. The designed US-WENO Lagrangian scheme could obtain comparable results as before. Moreover, some overshoots occurred due to the Lagrangian framework rather than the high-order spatial reconstructions, as mentioned in [1].*

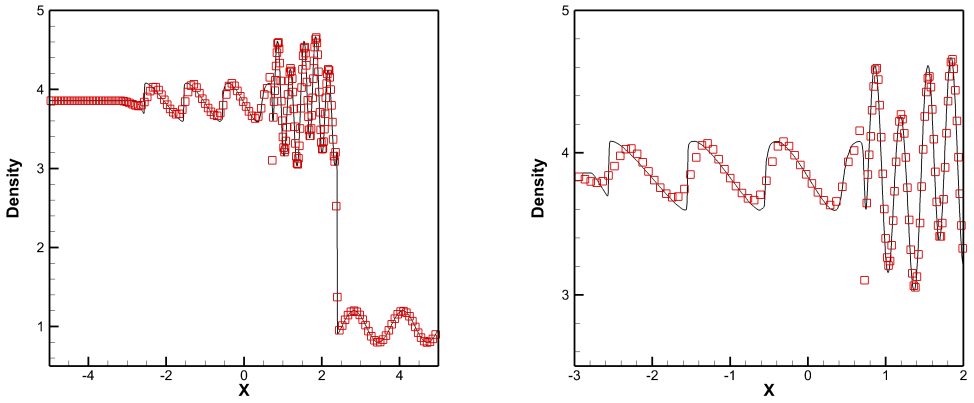

Figure 4. Shu–Osher problem. T = 1.8. Density cut at $y = 0$. Solid line: the reference solution; squares: the results of US-WENO Lagrangian scheme. (**Left**): overall view. (**Right**): enlarged view.

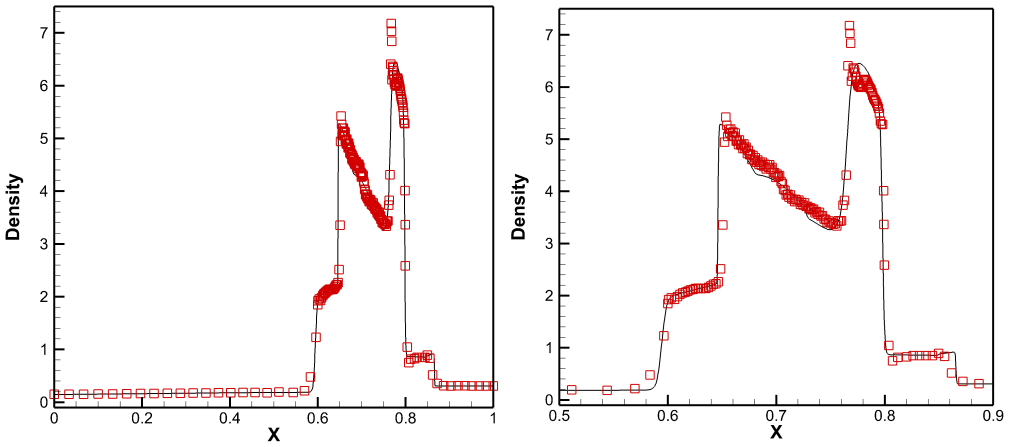

Figure 5. Blast wave problem. T = 0.038. Density cut at $y = 0$. Solid line: the reference solution; squares: the results of US-WENO Lagrangian scheme. (**Left**): overall view. (**Right**): enlarged view.

Example 6. *We test the two-dimensional Sedov problem [1], a typical test example used to verify the capability of the Lagrangian method. The computation is performed on* $[0, 1.1] \times [0, 1.1]$ *with* 30×30 *cells initially. The initial conditions are* $\rho = 1, u = 0, v = 0, \gamma = 1.4$, *and the internal energy is* 10^{-14} *except for the cell containing the origin, where we set* $e = 182.09$. *All boundaries employ the reflection conditions. The analytical solution [59] is that the shock arrives at* $r = 1$ *at time* $t = 1$ *with a peak density of 6. The computation results are presented in Figure 6, and the figure shows that numerical solutions with good resolution are obtained for this benchmark example.*

Example 7. *We consider two-dimensional Sod problem [57]. The initial conditions are:*

$$(\rho, u, v, p)^T = \begin{cases} (1, 0, 0, 1)^T, & 0 \le \sqrt{x^2 + y^2} \le 0.5, \\ (0.125, 0, 0, 0.1)^T, & 0.5 < \sqrt{x^2 + y^2} \le 1. \end{cases} \tag{41}$$

The computation domain is $[-1, 1] \times [-1, 1]$ *and initially divided into* 200×200 *uniform cells. All of the boundaries are employed the non-reflection condition. It is a significant test case since the wave propagation direction does not correspond with the initial grid. The computational results including mesh, density distributions, pressure distributions, and density profiles along* $y = 0$ *are displayed in Figure 7. The calculation results show that the numerical solution of density and pressure agree well with the corresponding reference solution.*

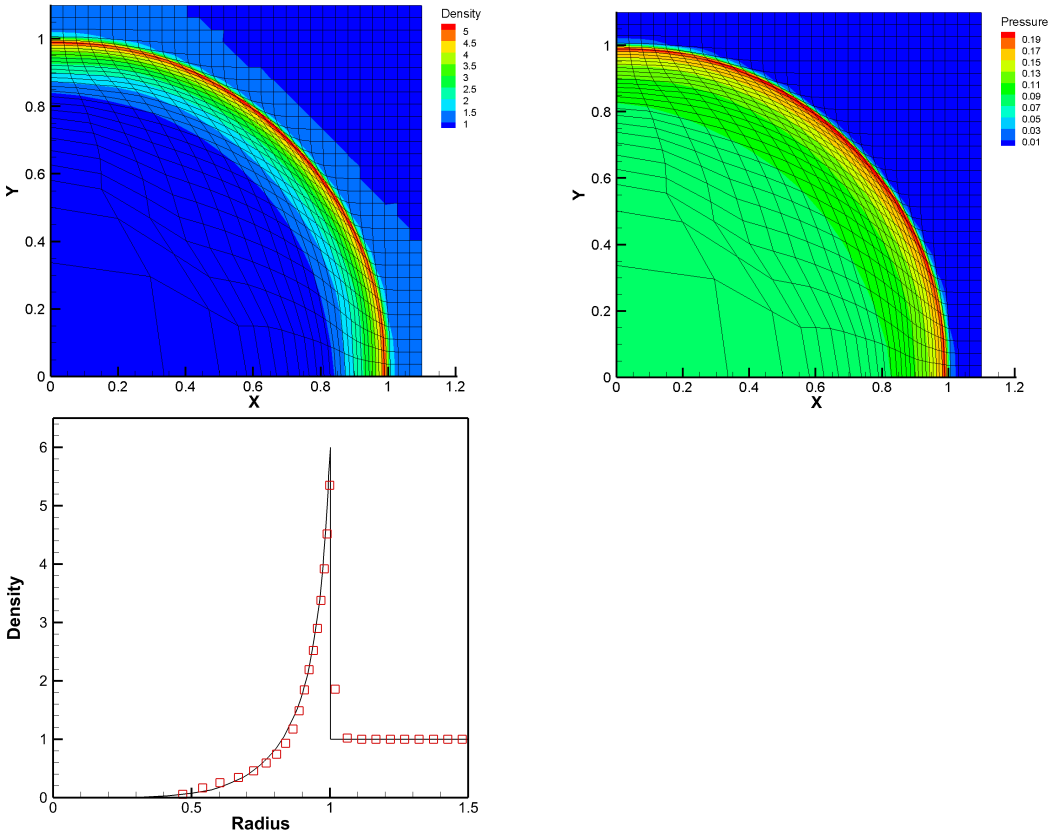

Figure 6. The 2D Sedov problem. T = 1. (**Top left**): mesh and density distributions; (**top right**): mesh and pressure distributions; (**bottom**): density in diagonal cells with respect to the radius.

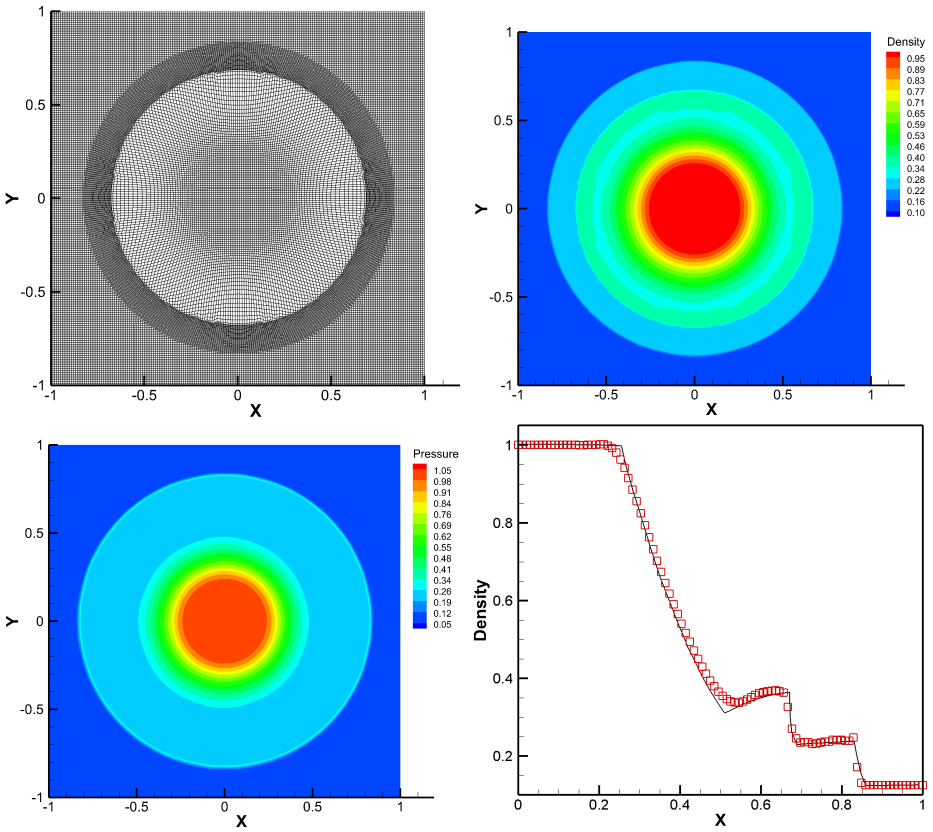

Figure 7. The 2D Sod problem. T = 0.2. (**Top left**): mesh distributions; (**top right**): density distributions; (**bottom left**): pressure distributions; (**bottom right**): density profiles along $y = 0$.

Example 8. *Two-dimensional Saltzman problem [58] is next considered. It describes a piston's prescribed motion impacting fluid in an enclosed space. The computation is carried out in* $[0,1] \times [0,0.1]$ *with* 100×10 *cells. The initial mesh, shown in Figure 8, is represented as*

$$x_{i,j} = i\Delta x + (10 - j)\Delta y sin(\frac{i\pi}{100}), \quad y_{i,j} = j\Delta y, \tag{42}$$

where $\Delta x = \Delta y = 0.01$. *As can be seen, the starting mesh is not consistent with the flow direction to evaluate the robustness of the Lagrangian scheme. The fluid is described with* $\rho = 1$, $u = 0$, $v = 0$, $e = 10^{-4}$, $\gamma = 5/3$. *The left boundary is a moving piston with* $u = 1$, *and other boundaries meet the conditions for a reflective boundary. The CFL number is 0.01 originally, and returns to 0.6 when the computation proceeds to* $t = 0.01$ *[1,2]. According to [14], the shock is supposed to be at* $x = 0.8$ *when* $t = 0.6$, *the analytic value of post-shock is 4.0 for density and 1.333 for pressure. The density and pressure distributions are displayed in Figure 9, and corresponding cut lines are given for comparison. It can be seen that the designed scheme also works well for this mesh distortion problem.*

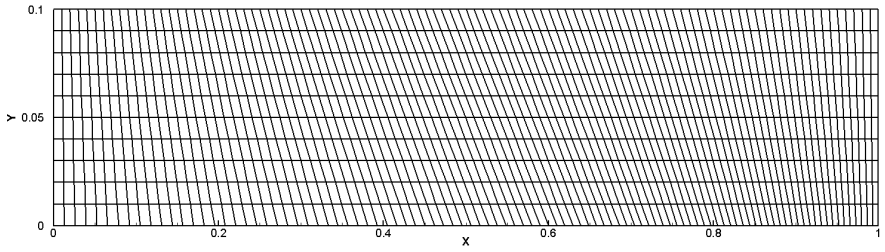

Figure 8. Initial mesh configuration for 2D Saltzman problem.

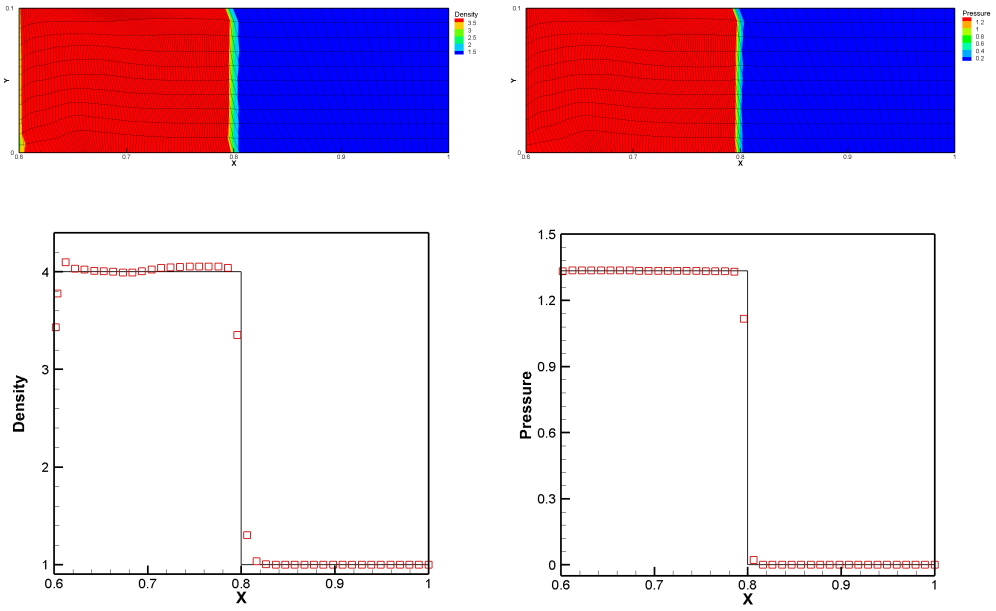

Figure 9. The 2D Saltzman problem. T = 0.6. (**Top left**): mesh and density distributions; (**top right**): mesh and pressure distributions; (**bottom left**): density profiles along $y = 0$ comparison with analytical solution; (**bottom right**): pressure profiles along $y = 0$ comparison with analytical solution.

3.3. Two-Dimensional ALE Tests

Two examples of two-dimensional ALE computations are given in this subsection to further evaluate the capabilities of the proposed US-WENO Lagrangian scheme. Here, we modify the two-dimensional third-order ENO remapping scheme [60] by taking the procedures of US-WENO spatial reconstruction introduced in this paper. The rezone mesh

is obtained by the variational method [61], and the corresponding Euler–Lagrange equation has the following form

$$\begin{aligned} \left(Gx_{\xi}\right)_{\xi} + \left(Gx_{\eta}\right)_{\eta} &= 0, \\ \left(Gy_{\xi}\right)_{\xi} + \left(Gy_{\eta}\right)_{\eta} &= 0, \end{aligned} \tag{43}$$

where (x, y) and (ξ, η) denote the physical and computational coordinates, $G = \omega I$, ω is a monitor function related to the flow variables. In general, the monitor function can take the form: $\omega = \sqrt{1 + \alpha |\nabla \rho|^2}$, and more details can be found in [61]. The two-dimensional Riemann problem [62] and triple-point problem [63–65] are presented to illustrate the superior capability of the developed scheme in two-dimensional ALE simulations.

Example 9. *Two-dimensional Riemann problem [62] with the following initial conditions:*

$$\begin{cases} (\rho_1, u_1, v_1, p_1)^T = (0.5313, 0, 0, 0.4)^T, & x > 1, y > 1, \\ (\rho_2, u_2, v_2, p_2)^T = (1, 0, 0.7276, 1)^T, & x > 1, y < 1, \\ (\rho_3, u_3, v_3, p_3)^T = (1, 0.7276, 0, 1)^T, & x < 1, y > 1, \\ (\rho_4, u_4, v_4, p_4)^T = (0.8, 0, 0, 1)^T, & x < 1, y < 1. \end{cases} \tag{44}$$

The computational domain is $[0, 2] \times [0, 2]$ *with* 200×200 *uniform cells initially. The final computing time is* $t = 0.52$. *In this case, we perform the rezoning and remapping algorithm every five-time Lagrangian steps. As in [61], the monitor function used in this problem is* $\omega = \sqrt{1 + 2000(\rho_{\xi}^2 + \rho_{\eta}^2)}$. *The results of the mesh distribution and density contours are shown in Figure 10. It can be seen that the results with better resolution are obtained due to the local mesh adaptation.*

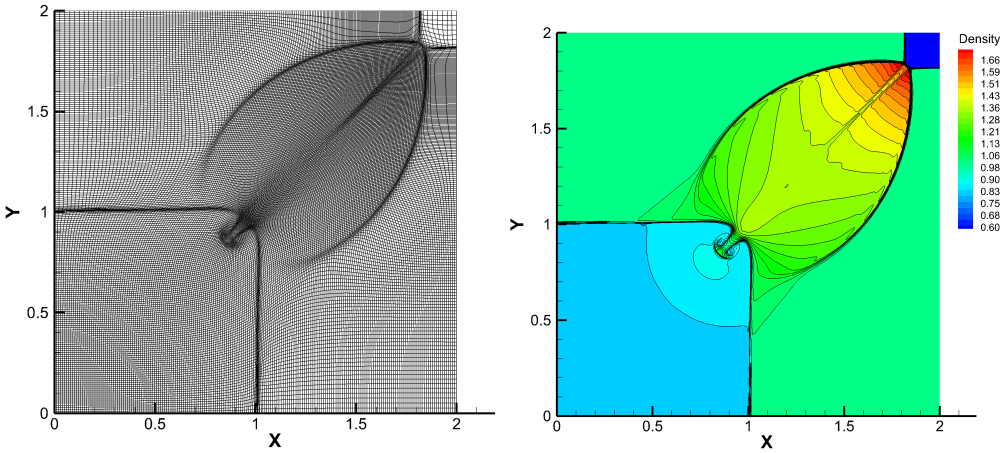

Figure 10. The 2D Riemann problem. T = 0.52. (**Left**): mesh distributions; (**right**): density contours.

Example 10. *Triple-point problem [63–65] is a typical test problem to evaluate the performance of Lagrangian or ALE methods with large mesh deformation. We consider the single material case, and the initial conditions are*

$$(\rho, u, v, p, \gamma)^T = \begin{cases} (1, 0, 0, 1, 1.4)^T, & (x, y) \in [0, 1] \times [0, 3], \\ (0.125, 0, 0, 0.1, 1.4)^T, & (x, y) \in [1, 9] \times [0, 1.5], \\ (1, 0, 0, 0.1, 1.4)^T, & (x, y) \in [1, 9] \times [1.5, 3]. \end{cases} \tag{45}$$

Initially, 180×60 *cells are equally distributed. The final computing time is* $t = 4$. *In this case, we perform the rezoning and remapping algorithm every five-time Lagrangian steps. As in [61], the monitor function used here is* $\omega = \sqrt{1 + 100(\rho_{\xi}^2 + \rho_{\eta}^2)}$. *The results of mesh distributions and density contours are shown in Figure 11. It can be observed that the mesh is performed well and desirable results are obtained by the proposed scheme combined with the rezoning and remapping algorithm.*

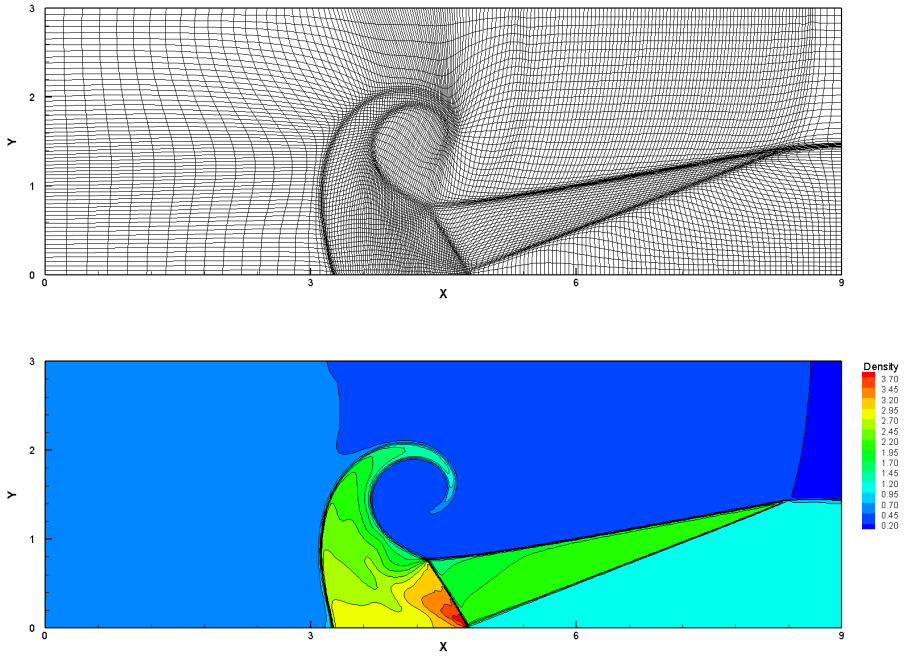

Figure 11. Triple-point problem. T = 4. (**Top**): mesh distributions; (**bottom**): density contours.

3.4. Three-Dimensional Lagrangian Tests

In this subsection, the three-dimensional Sedov problem [18,42,59], Sod problem [57], and Saltzman problem [42] are given to illustrate the superior capability of the designed scheme in high dimensions. In the Lagrangian method, the numerical simulations in three dimensions are more difficult than in two dimensions, mainly because the mesh is more easily distorted in three dimensions. We can calculate these three examples without introducing any further procedures, showing that the designed US-WENO Lagrangian scheme is quite robust in three-dimensional simulations.

Example 11. *The three-dimensional Sedov problem [18,42,59], which describes blast wave energy deposited from a singular point, is a typical test problem for the Lagrangian method. The initial conditions are* $\rho = 1, u = 0, v = 0, w = 0, \gamma = 1.4$, *and the pressure is* 10^{-6} *everywhere except in the cell containing the origin. For the cell containing the origin, the pressure is given by* $p = \frac{(\gamma-1)\varepsilon_0}{V}$, *where* $\varepsilon_0 = 0.106384$ *is the total amount of released energy, and* V *is the cell's volume. The total energy released is given to make the shock arrive at* $r = 1$ *when time* $t = 1$ *[18]. The computation is performed on the domain of* $[0, 1.2] \times [0, 1.2] \times [0, 1.2]$ *with uniform* $30 \times 30 \times 30$ *cells initially. The reflective boundary conditions are performed on* $x = 0$, $y = 0$, *and* $z = 0$, *and other boundaries are non-reflective boundaries. In particular, a smaller CFL number of 0.01 is used first and returns to 0.6 after 30 steps for the singularity of origin. The computation results of the designed scheme are given in Figure 12. The figure shows that the distributions of density and pressure perform well with mesh movement.*

Example 12. *The three-dimensional Sod problem [57]. The initial conditions are*

$$(\rho, u, v, w, p)^T = \begin{cases} (1, 0, 0, 0, 1)^T, & 0 \le \sqrt{x^2 + y^2 + z^2} \le 0.5, \\ (0.125, 0, 0, 0, 0.1)^T, & 0.5 < \sqrt{x^2 + y^2 + z^2} \le 1. \end{cases} \tag{46}$$

The computation is performed on $(x, y, z) \in [0, 1] \times [0, 1] \times [0, 1]$ *with* $50 \times 50 \times 50$ *uniform cells. The symmetric boundary conditions are employed on* $x = 0$, $y = 0$, *and* $z = 0$, *and other boundaries are defined as non-reflective. The computation results of density and pressure distributions and their profiles against radius are presented in Figure 13. The designed scheme also performs well, as before.*

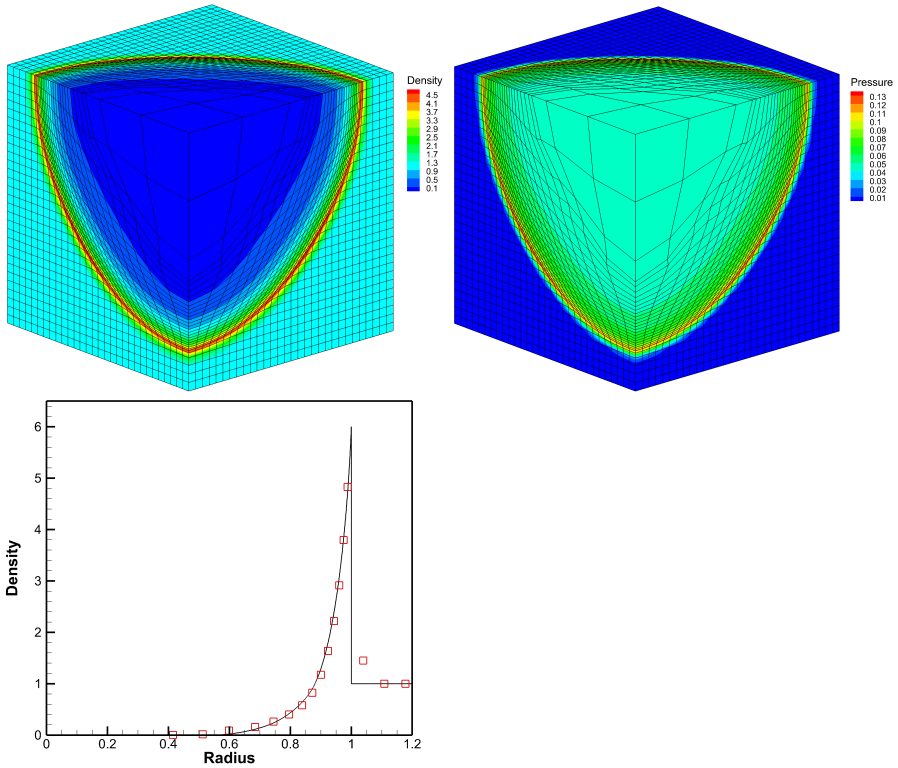

Figure 12. The 3D Sedov problem. T = 1. (**Top left**): mesh and density distributions; (**top right**): mesh and pressure distributions; (**bottom**): comparison between analytical and numerical density profiles along the diagonal cell.

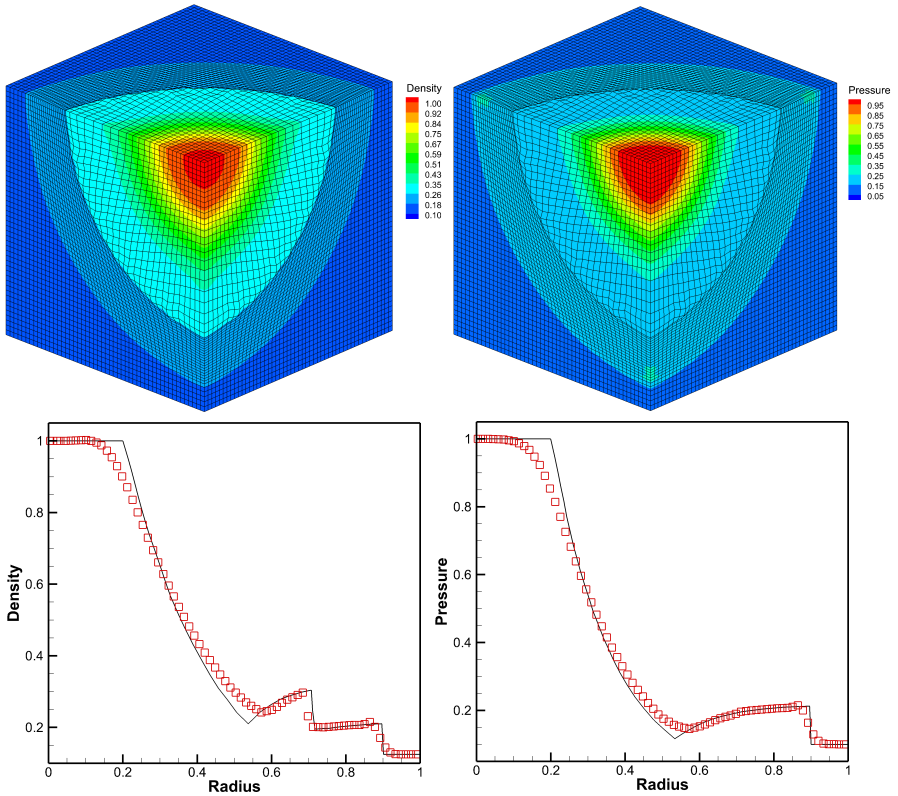

Figure 13. The 3D Sod problem. T = 0.25. (**Top left**): mesh and density distributions; (**top right**): mesh and pressure distributions; (**bottom left**): comparison between analytical and numerical density profiles along the diagonal cell; (**bottom right**): comparison between analytical and numerical pressure profiles along the diagonal cell.

Example 13. *The three-dimensional Saltzman problem [42] is considered to evaluate the robustness of the Lagrangian method, which describes the motion of a planar shock wave on a skewed cartesian grid. The computation is carried out in* $[0,1] \times [0,0.1] \times [0,0.1]$ *with* $100 \times 10 \times 10$ *cells. The initial mesh is displayed in Figure 14 and given by*

$$\begin{aligned} x_{i,j,k} &= \xi_i + \left(0.5\eta_j + \zeta_k - 15\eta_j\zeta_k\right)\sin(\pi\xi_i), \\ y_{i,j,k} &= \eta_j, \\ z_{i,j,k} &= \zeta_k, \end{aligned} \tag{47}$$

where $\xi_i = i\Delta\xi$, $\eta_j = j\Delta\eta$, $\zeta_k = k\Delta\zeta$ *with* $\Delta\xi = \Delta\eta = \Delta\zeta = 0.01$. *The box is filled with the following gas:* $\rho = 1$, $u = 0$, $v = 0$, $w = 0$, $e = 10^{-4}$, $\gamma = 5/3$. *The left boundary behaves like a piston with* $u = 1$, *and the other boundaries are reflected walls. A smaller CFL number of 0.01 is used at first and returns to 0.6 after* $t = 0.01$. *The shock is expected to be located at* $x = 0.8$ *at time* $t = 0.6$, *and the analytic value post-shock are* $\rho = 4$ *and* $p = 1.333$. *Figure 15 gives the density and pressure distributions, the comparison figures of the cut line with the analytical solution are also presented. It can be seen that good results are also obtained for this three-dimensional mesh distortion problem.*

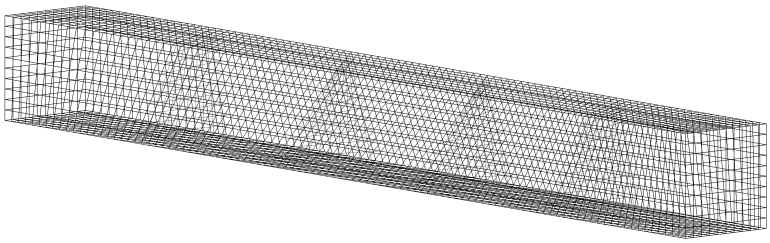

Figure 14. Initial mesh configuration for 3D Saltzman problem.

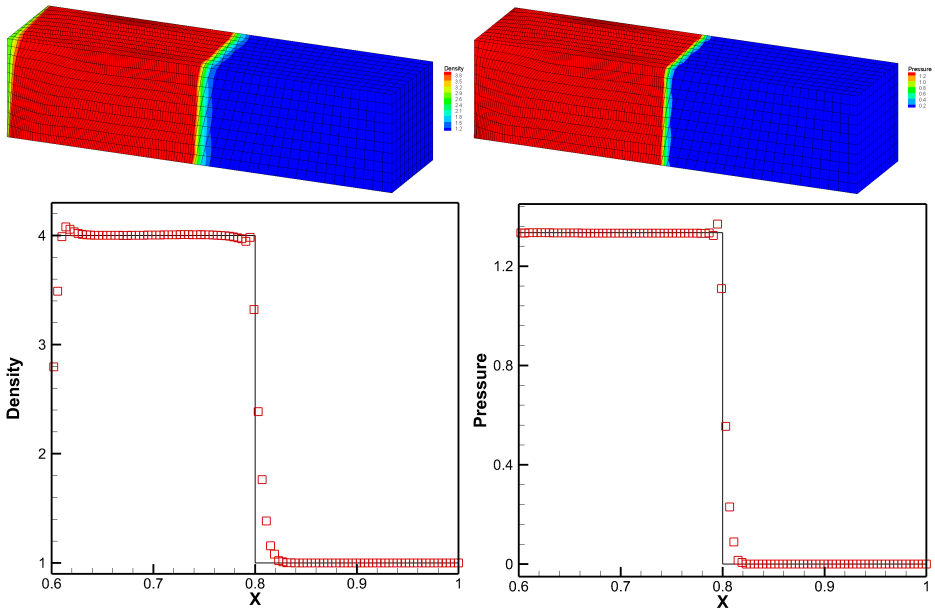

Figure 15. The 3D Saltzman problem. T = 0.6. (**Top left**): mesh and density distributions; (**top right**): mesh and pressure distributions; (**bottom left**): density profiles plotted against the analytical solution; (**bottom right**): pressure profiles plotted against the analytical solution.

4. Concluding Remarks

In this paper, a new finite volume US-WENO Lagrangian scheme is developed for solving Euler equations on quadrilateral and hexahedral meshes. Compared with the high-order ENO Lagrangian scheme [1,2], the most incredible benefits of the developed scheme are its simplicity and robustness. The spatial reconstruction procedures are implemented

by using a convex combination of a quadratic polynomial with several linear polynomials defined on unequal-sized stencils, so the linear weights can be chosen artificially and are not related to the local mesh topology or the location of the quadrature points. Moreover, the linear weights no longer need to change, as the mesh moves with the fluid once they are set initially, avoiding the appearance of negative linear weights. Meanwhile, the designed scheme is quite robust and computationally efficient, so it has great potential for complex multi-dimensional Lagrangian simulations. Our future investigation will focus on developing high-order Lagrangian methods on three-dimensional hybrid grids.

Author Contributions: Conceptualization, Y.T., H.L. and J.Z.; methodology, Y.T., H.L. and J.Z.; software, Y.T.; validation, Y.T.; writing—original draft preparation, Y.T.; writing—review and editing, H.L. and J.Z. All authors have read and agreed to the published version of the manuscript.

Funding: The research is partly supported by National Natural Science Foundation of China (No. 11872210) and Anhui Provincial Key Project of Natural Science in Universities (KJ2021A0965 and 2022AH051588).

Data Availability Statement: No data were used for the research described in the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Cheng, J.; Shu, C.-W. A high order ENO conservative Lagrangian type scheme for the compressible Euler equations. *J. Comput. Phys.* **2007**, *227*, 1567–1596. [CrossRef]
2. Cheng, J.; Shu, C.-W. A third order conservative Lagrangian type scheme on curvilinear meshes for the compressible Euler equations. *Commun. Comput. Phys.* **2008**, *4*, 1008–1024.
3. Hu, C.; Shu, C.-W. Weighted essentially non-oscillatory schemes on triangular meshes. *J. Comput. Phys.* **1999**, *150*, 97–127. [CrossRef]
4. Zhang, Y.T.; Shu, C.-W. Third order WENO scheme on three dimensional tetrahedral meshes. *Commun. Comput. Phys.* **2009**, *5*, 836–848.
5. Luo, H.; Baum, J.D.; Löhner, R. On the computation of multi-material flows using ALE formulation. *J. Comput. Phys.* **2004**, *194*, 304–328. [CrossRef]
6. Barlow, A.J.; Maire, P.-H.; Rider, W.J.; Rieben, R.N.; Shashkov, M.J. Arbitrary Lagrangian Eulerian methods for modeling high-speed compressible multimaterial flows. *J. Comput. Phys.* **2016**, *332*, 603–665. [CrossRef]
7. Burton, D.E. *Exact Conservation of Energy and Momentum in Staggered-Grid Hydrodynamics with Arbitrary Connectivity, Advances in the Free Lagrange Method*; Springer: New York, NY, USA, 1990.
8. Burton, D.E. *Multidimensional Discretization of Conservation Laws for Unstructured Polyhedral Grids*; Technical Report UCRL-JC-118306; Lawrence Livermore National Laboratory: Livermore, CA, USA, 1990.
9. Caramana, E.J.; Rousculp, C.L.; Burton, D.E. A compatible, energy and symmetry preserving lagrangian hydrodynamics algorithm in three-dimensional Cartesian geometry. *J. Comput. Phys.* **2000**, *157*, 89–119. [CrossRef]
10. Caramana, E.J.; Shashkov, M.J.; Whalen, P.P. Formulations of artificial viscosity for multi-dimensional shock wave computations. *J. Comput. Phys.* **1998**, *144*, 70–97. [CrossRef]
11. Loubère, R.; Maire, P.-H.; Vachal, P. 3D staggered Lagrangian hydrodynamics scheme with cell-centered Riemann solver-based artificial viscosity. *Int. J. Numer. Methods Fluids* **2013**, *72*, 22–42. [CrossRef]
12. Cheng, J.; Shu, C.-W. Positivity-preserving Lagrangian scheme for multi-material compressible flow. *J. Comput. Phys.* **2014**, *257*, 143–168. [CrossRef]
13. Georges, G.; Breil, J.; Maire, P.-H. A 3D GCL compatible cell-centered Lagrangian scheme for solving gas dynamics equations. *J. Comput. Phys.* **2016**, *305*, 921–941. [CrossRef]
14. Liu, W.; Cheng, J.; Shu, C.-W. High order conservative Lagrangian schemes with Lax-Wendroff type time discretization for the compressible Euler equations. *J. Comput. Phys.* **2009**, *228*, 8872–8891. [CrossRef]
15. Maire, P.-H. A high-order cell-centered Lagrangian scheme for compressible fluid flows in two-dimensional cylindrical geometry. *J. Comput. Phys.* **2009**, *228*, 6882–6915. [CrossRef]
16. Maire, P.-H.; Abgrall, R.; Breil, J.; Ovadia, J. A cell-centered Lagrangian scheme for two-dimensional compressible flow problems. *SIAM J. Sci. Comput.* **2007**, *29*, 1781–1824. [CrossRef]
17. Maire, P.-H.; Breil, J. A second-order cell-centered Lagrangian scheme for two-dimensional compressible flow problems. *Int. J. Numer. Meth. Fluids* **2008**, *56*, 1417–1423. [CrossRef]
18. Maire, P.-H.; Nkonga, B. Multi-scale Godunov-type method for cell-centered discrete Lagrangian hydrodynamics. *J. Comput. Phys.* **2009**, *228*, 799–821. [CrossRef]
19. Xu, X.; Dai, Z.H.; Gao, Z.M. A 3D cell-centered Lagrangian scheme for the ideal magnetohydrodynamics equations on unstructured meshes. *Comput. Methods Appl. Mech. Eng.* **2018**, *342*, 490–508. [CrossRef]

20. Munz, C.D. On Godunov-type schemes for Lagrangian gas dynamics. *SIAM J. Numer. Anal.* **1994**, *31*, 17–42. [CrossRef]
21. Després, B.; Mazeran, C. Symmetrization of Lagrangian gas dynamic in dimension two and multidimensional solvers. *Comptes Rendus Méc.* **2003**, *331*, 475–480. [CrossRef]
22. Després, B.; Mazeran, C. Lagrangian gas dynamics in two-dimensions and Lagrangian systems. *Arch. Ration. Mech. Anal.* **2005**, *178*, 327–372. [CrossRef]
23. Harten, A.; Engquist, B.; Osher, S.; Chakravathy, S. Uniformly high order accurate essentially non-oscillatory schemes, III. *J. Comput. Phys.* **1987**, *71*, 231–303. [CrossRef]
24. Harten, A.; Osher, S. Uniformly high-order accurate non-oscillatory schemes I. *SIAM J. Numer. Anal.* **1987**, *24*, 279–309. [CrossRef]
25. Jiang, G.; Shu, C.-W. Efficient implementation of weighted ENO schemes. *J. Comput. Phys.* **1996**, *126*, 202–228. [CrossRef]
26. Liu, X.; Osher, S.; Chan, T. Weighted essentially non-oscillatory schemes. *J. Comput. Phys.* **1994**, *115*, 200–212. [CrossRef]
27. Titarev, V.A.; Toro, E.F. Finite-volume WENO schemes for three-dimensional conservation laws. *J. Comput. Phys.* **2004**, *201*, 238–260. [CrossRef]
28. Boscheri, W.; Balsara, D.S.; Dumbser, M. Lagrangian ADER-WENO finite volume schemes on unstructured triangular meshes based on genuinely multidimensional HLL Riemann solvers. *J. Comput. Phys.* **2014**, *267*, 112–138. [CrossRef]
29. Boscheri, W.; Dumbser, M. Arbitrary-Lagrangian-Eulerian one-step WENO finite volume schemes on unstructured triangular meshes. *Commun. Comput. Phys.* **2013**, *14*, 1174–1206. [CrossRef]
30. Boscheri, W.; Dumbser, M. A direct Arbitrary-Lagrangian-Eulerian ADER-WENO finite volume scheme on unstructured tetrahedral meshes for conservative and non-conservative hyperbolic systems in 3D. *J. Comput. Phys.* **2014**, *275*, 484–523. [CrossRef]
31. Boscheri, W.; Dumbser, M.; Balsara, D.S. High order Lagrangian ADER-WENO schemes on unstructured meshes—Application of several node solvers to hydrodynamics and magnetohydrodynamics. *Int. J. Numer. Methods Fluids* **2014**, *76*, 737–778. [CrossRef]
32. Dumbser, M. Arbitrary-Lagrangian-Eulerian ADER-WENO finite volume schemes with time-accurate local time stepping for hyperbolic conservation laws. *Comput. Methods Appl. Mech. Eng.* **2014**, *280*, 57–83. [CrossRef]
33. Dumbser, M.; Käser, M. Arbitrary high order non-oscillatory finite volume schemes on unstructured meshes for linear hyperbolic systems. *J. Comput. Phys.* **2007**, *221*, 693–723. [CrossRef]
34. Dumbser, M.; Käser, M.; Titarev, V.A.; Toro, E.F. Quadrature-free non-oscillatory finite volume schemes on unstructured meshes for nonlinear hyperbolic systems. *J. Comput. Phys.* **2007**, *226*, 204–243. [CrossRef]
35. Anderson, R.W.; Dobrev, V.A.; Kolev, T.V.; Rieben, R.N.; Tomov, V.Z. High-order multi-material ale hydrodynamics. *SIAM J. Sci. Comput.* **2018**, *40*, B32–B58. [CrossRef]
36. Boscheri, W.; Balsara, D.S. High order direct Arbitrary-Lagrangian-Eulerian (ALE) $P_N P_M$ schemes with WENO Adaptive-Order reconstruction on unstructured meshes. *J. Comput. Phys.* **2019**, *398*, 108899. [CrossRef]
37. Dobrev, V.; Kolev, T.; Rieben, R. High-order curvilinear finite element methods for Lagrangian hydrodynamics. *SIAM J. Sci. Comput.* **2012**, *34*, B606–B641. [CrossRef]
38. Gaburro, E. A unified framework for the solution of hyperbolic PDE systems using high order direct Arbitrary-Lagrangian-Eulerian schemes on moving unstructured meshes with topology change. *Arch. Comput. Methods Eng.* **2020**, *28*, 1–73. [CrossRef]
39. Gaburro, E.; Boscheri, W.; Chiocchetti, S.; Klingenberg, C.; Springel, V.; Dumbser, M. High order direct Arbitrary-Lagrangian-Eulerian schemes on moving Voronoi meshes with topology changes. *J. Comput. Phys.* **2020**, *407*, 109167. [CrossRef]
40. Lei, N.; Cheng, J.; Shu, C.-W. A high order positivity-preserving conservative WENO remapping method on 2D quadrilateral meshes. *Comput. Methods Appl. Mech. Eng.* **2021**, *373*, 113497. [CrossRef]
41. Lei, N.; Cheng, J.; Shu, C.-W. A high order positivity-preserving conservative WENO remapping method on 3D tetrahedral meshes. *Comput. Methods Appl. Mech. Eng.* **2022**, *395*, 115037. [CrossRef]
42. Pan, L.; Xu, K. An arbitrary-Lagrangian-Eulerian high-order gas-kinetic scheme for three-dimensional computations. *J. Sci. Comput.* **2021**, *88*, 1–29. [CrossRef]
43. Pan, L.; Zhao, F.X.; Xu, K. High-order ALE gas-kinetic scheme with WENO reconstruction. *J. Comput. Phys.* **2020**, *417*, 109558. [CrossRef]
44. Shi, J.; Hu, C.; Shu, C.-W. A technique of treating negative weights in WENO schemes. *J. Comput. Phys.* **2002**, *175*, 108–127. [CrossRef]
45. Liu, Y.; Zhang, Y.T. A robust reconstruction for unstructured WENO schemes. *J. Sci. Comput.* **2013**, *54*, 603–621. [CrossRef]
46. Zhu, J.; Qiu, J. A new fifth order finite difference WENO scheme for solving hyperbolic conservation laws. *J. Comput. Phys.* **2016**, *318*, 110–121. [CrossRef]
47. Zhu, J.; Qiu, J. New finite volume weighted essentially non-oscillatory scheme on triangular meshes. *SIAM J. Sci. Comput.* **2018**, *40*, 903–928. [CrossRef]
48. Zhu, J.; Qiu, J. A new third order finite volume weighted essentially non-oscillatory scheme on tetrahedral meshes. *J. Comput. Phys.* **2017**, *349*, 220–232. [CrossRef]
49. Batten, P.; Leschziner, M.A.; Goldberg, U.C. Average-state Jacobians and implicit methods for compressible viscous and turbulent flows. *J. Comput. Phys.* **1997**, *137*, 38–78. [CrossRef]
50. Roe, P.L. Approximate Riemann solvers, parameter vectors, and difference schemes. *J. Comput. Phys.* **1981**, *43*, 357–372. [CrossRef]
51. Shu, C.-W.; Osher, S. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *J. Comput. Phys.* **1988**, *77*, 439–471. [CrossRef]

52. Qiu, J.; Shu, C.-W. On the construction, comparison, and local characteristic decomposition for high order central WENO schemes. *J. Comput. Phys.* **2002**, *183*, 187–209. [CrossRef]
53. Sod, G. A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *J. Comput. Phys.* **1978**, *27*, 1–31. [CrossRef]
54. Lax, P.D. Weak solutions of nonlinear hyperbolic equations and their numerical computation. *Commun. Pure Appl. Math.* **1954**, *7*, 159–193. [CrossRef]
55. Shu, C.-W.; Osher, S. Efficient implementation of essentially non-oscillatory shock capturing schemes II. *J. Comput. Phys.* **1989**, *83*, 32–78. [CrossRef]
56. Woodward, P.; Colella, P. The numerical simulation of two-dimensional fluid flow with strong shocks. *J. Comput. Phys.* **1984**, *54*, 115–173. [CrossRef]
57. Toro, E.F. *Riemann Solvers and Numerical Methods for Fluid Dynamics*, 3rd ed.; Springer: New York, NY, USA, 2009.
58. Dukowicz, J.K.; Meltz, B. Vorticity errors in multidimensional Lagrangian codes. *J. Comput. Phys.* **1992**, *99*, 115–134. [CrossRef]
59. Kamm, J.R.; Timmes, F.X. *On Efficient Generation of Numerically Robust Sedov Solutions*; Technical Report LA-UR-07-2849; Los Alamos National Laboratory: Los Alamos, NM, USA, 2007.
60. Cheng, J.; Shu, C.-W. A high order accurate conservative remapping method on staggered meshes. *Appl. Numer. Math.* **2008**, *58*, 1042–1060. [CrossRef]
61. Tang, H.Z.; Tang, T. Moving mesh methods for one- and two-dimensional hyperbolic conservation laws. *SIAM J. Numer. Anal.* **2003**, *41*, 487–515. [CrossRef]
62. Schulz-Rinne, C.W.; Collins, J.P.; Glaz, H.M. Numerical solution of the Riemann problem for two-dimensional gas dynamics. *SIAM J. Sci. Comput.* **1993**, *14*, 1394–1414. [CrossRef]
63. Galera, S.; Maire, P.-H.; Breil, J. A two-dimensional unstructured cell-centered multi-material ALE scheme using VOF interface reconstruction. *J. Comput. Phys.* **2010**, *229*, 5755–5787. [CrossRef]
64. Kucharik, M.; Garimella, R.V.; Schofield, S.P.; Shashkov, M.J. A comparative study of interface reconstruction methods for multi-material ALE simulations. *J. Comput. Phys.* **2010**, *229*, 2432–2452. [CrossRef]
65. Morgan, N.R.; Lipnikov, K.N.; Burton, D.E.; Kenamond, M.A. A Lagrangian staggered grid Godunov-like approach for hydrodynamics. *J. Comput. Phys.* **2014**, *259*, 568–597. [CrossRef]