



Dimension by Dimension Finite Volume HWENO Method for Hyperbolic Conservation Laws

Feng Zheng¹ · Jianxian Qiu²

Received: 23 August 2022 / Revised: 9 March 2023 / Accepted: 7 April 2023 /
Published online: 11 July 2023
© Shanghai University 2023

Abstract

In this paper, we propose a finite volume Hermite weighted essentially non-oscillatory (HWENO) method based on the dimension by dimension framework to solve hyperbolic conservation laws. It can maintain the high accuracy in the smooth region and obtain the high resolution solution when the discontinuity appears, and it is compact which will be good for giving the numerical boundary conditions. Furthermore, it avoids complicated least square procedure when we implement the genuine two dimensional (2D) finite volume HWENO reconstruction, and it can be regarded as a generalization of the one dimensional (1D) HWENO method. Extensive numerical tests are performed to verify the high resolution and high accuracy of the scheme.

Keywords Finite volume · Dimension by dimension · HWENO · Hyperbolic conservation laws

Mathematics Subject Classification 65M60 · 35L65

1 Introduction

Based on the work in [31], we propose a dimension by dimension finite volume Hermite weighted essentially non-oscillatory (HWENO) method for solving two dimensional (2D) hyperbolic conservation laws. The HWENO method comes from the weighted essentially non-oscillatory (WENO) method. In 1994, the finite volume WENO method was first

The research of Feng Zheng is partially supported by the NSFC grant 12101128.
The research of Jianxian Qiu is partially supported by the NSFC grant 12071392.

✉ Jianxian Qiu
jxqiu@xmu.edu.cn

Feng Zheng
fzbz200808-31@163.com

¹ College of Mathematics and Statistics, Fujian Normal University, Fuzhou 350117, Fujian, China

² School of Mathematical Sciences and Fujian Provincial Key Laboratory of Mathematical Modeling & High-Performance Scientific Computing, Xiamen University, Xiamen 361005, Fujian, China

proposed by Liu et al. [14]. Then in 1996, Jiang and Shu introduced a finite difference WENO to solve hyperbolic conservation laws [10]. The WENO reconstruction is a convex combination of several low order reconstructions, and the coefficients of the low order reconstructions are specially chosen such that they can achieve the optimal accuracy in the smooth area and achieve high resolution when the discontinuity appears. It achieves great success, and has been applied in many areas, such as the 2D finite volume WENO method [7], the three dimensional (3D) finite volume WENO method [23, 30], shallow water equations [28], relativistic hydrodynamics [26], the fast sweeping method [29], multi-fluids [8], detonation wave [24], and Hamilton-Jacobi equations [9]. The HWENO method is a generalization of the WENO method. The main difference between them is that the HWENO method is more compact than the WENO method under the same order of accuracy. Two different types of information on each cell are needed to implement the HWENO method, while only one information on each cell is used in the WENO method. So, the compactness is achieved. The HWENO method is first proposed in [17] by Qiu and Shu, in which, the HWENO method is used to solve the hyperbolic conservation laws and as a limiter in the discontinuous Galerkin (DG) method. After that, the HWENO method is booming, for example, [1, 2, 13, 16, 18, 22]. In 2008, a unified framework termed as the $P_N P_M$ method was proposed by Dumbser et al. [5] to define the numerical scheme extended by the DG method. These HWENO schemes can be seen as the $P_1 P_M$ method.

However, both the WENO and the HWENO methods mentioned above have the following disadvantages. First, we need to use the least square method to implement the HWENO method. The HWENO method above is a convex combination of the reconstructed polynomials. All of them are genuine 2D reconstructions on the structured meshes. For the sake of stability, we need to require the reconstructed polynomial to be equal to the value at the targeted cell and require the polynomial to be equal to values of other cells in a least-square sense. It is not an easy task to figure out the polynomial. Usually, we need to resort to the mathematical softwares to write out the polynomials explicitly. Furthermore, if we want to require the reconstructed polynomial to be identical to more given information [31], it becomes a quadratic programming problem which is more difficult. Second, it is not guaranteed that these optimal coefficients are all positive. The negative coefficients would lead to serious oscillation, instability, or even blowing up of the numerical solutions. Although we can partly resolve this problem by using the technique proposed in [19], it still does not work in some extreme cases, see [15] by using the WENO method. Third, in some particular cases, the linear weights would not even exist.

As to the first disadvantage, we want to use the dimension by dimension approach. As we know, the dimension by dimension method exploits the tensor product nature of the reconstruction. Therefore, in the higher dimensional case, we do not need to design the reconstruction method additionally, we can obtain satisfying results by simply performing the same routine in different dimensions. So, it is easy to generalize the method to the high dimensional case. In [3], the authors proposed the dimension by dimension finite volume ENO method in the 2D case. They do not design a set of small stencils for the 2D interpolation. Instead, they just repeatedly use the same 1D ENO procedure in different directions. Therefore, they avoid the complicated least square procedure. The WENO method in [19, 35] belongs to this class. In [22, 32], the authors also use the HWENO or central HWENO method in the dimension by dimension way. Therefore, we will follow this idea to solve the hyperbolic conservation laws. As to the second and third disadvantages, in [34], Zhu and Qiu proposed a new type of WENO method. They use the combination of a fourth degree polynomial and two linear polynomials. The new type of WENO method contains the same ingredients as the classical WENO method, such as linear weights, smoothness indicators,

and nonlinear weights. The difference is that the linear weights can be artificial as long as their sum is equal to one. Therefore, we can overcome both negative linear weights and non-existing linear weights which may exist in the classical WENO method. Moreover, this new type of WENO method does not degrade the order of accuracy by carefully designing the methodology of computing the nonlinear weights. In [31], the authors have extended the idea to the HWENO framework. Therefore, we follow the idea to avoid the issues coming from the linear weights.

In this work, we proposed the dimension by dimension finite volume HWENO method to solve hyperbolic conservation laws. It is a hybrid method. After revising the moments for troubled-cell identified by the KXRCF limiter, we will simply use an efficient linear approximation in the smooth region and use the more expensive HWENO method when discontinuity appears. The implementation is in the dimension by dimension way. Both the cell averages and moments are evolved in the end. This method can avoid the complicated least square procedure in the genuine 2D finite volume HWENO reconstruction, and it is stable by setting the linear weights artificially. The implementation is simple. We also perform extensive numerical experiments to verify the high resolution and high accuracy of the scheme.

The organization of this paper is as follows: in Sect. 2, we review the finite volume HWENO scheme in the 1D case. In Sect. 3, we describe the detailed steps of the finite volume dimension by dimension HWENO method. In Sect. 4, we present some classical numerical tests to verify the numerical accuracy and efficiency of the scheme. In Sect. 5, we give a conclusion.

2 Review the Finite Volume HWENO Method in the 1D Case

2.1 The Framework of the Finite Volume Scheme for Solving 1D Hyperbolic Conservation Laws

One dimensional scalar hyperbolic conservation laws can be written as

$$\begin{cases} u_t + f(u)_x = 0, \\ u(x, 0) = u_0(x). \end{cases} \tag{1}$$

For simplicity, the computational domain is equally divided: $a = x_{1/2} < x_{3/2} < \dots < x_{N+1/2} = b$. The cell is denoted as $I_i = [x_{i-1/2}, x_{i+1/2}]$ with the mesh size being denoted as $\Delta x = x_{i+1/2} - x_{i-1/2}$. Furthermore, we define

$$\bar{u}_i = \frac{1}{\Delta x} \int_{I_i} u(x, t) dx, \quad \bar{v}_i = \frac{1}{\Delta x} \int_{I_i} u(x, t) \frac{x - x_i}{\Delta x} dx.$$

We multiply (1) by functions 1 and $\frac{x-x_i}{\Delta x}$, use the integration by parts formula, then we obtain

$$\begin{cases} \frac{d\bar{u}_i}{dt} = -\frac{1}{\Delta x} [f(u(x_{i+1/2})) - f(u(x_{i-1/2}))], \\ \frac{d\bar{v}_i}{dt} = -\frac{1}{2\Delta x} [f(u(x_{i+1/2})) + f(u(x_{i-1/2}))] + \frac{1}{\Delta x^2} \int_{I_i} f(u) dx. \end{cases}$$

After replacing the fluxes and integration with numerical fluxes and numerical integration, we have the following scheme:

$$\begin{cases} \frac{d\bar{u}_i}{dt} = -\frac{1}{\Delta x} \left[\hat{f}(u_{i+1/2}^-, u_{i+1/2}^+) - \hat{f}(u_{i-1/2}^-, u_{i-1/2}^+) \right], \\ \frac{d\bar{v}_i}{dt} = -\frac{1}{2\Delta x} \left[\hat{f}(u_{i+1/2}^-, u_{i+1/2}^+) + \hat{f}(u_{i-1/2}^-, u_{i-1/2}^+) \right] + \frac{1}{\Delta x} \sum_l \omega_l f(u_{i+G_l}), \end{cases}$$

where \hat{f} is the numerical flux, G_l and ω_l are the Gauss-Lobatto quadrature points and coefficients:

$$G_1 = \frac{1}{2}, \quad G_2 = \frac{\sqrt{5}}{10}, \quad G_3 = -\frac{\sqrt{5}}{10}, \quad G_4 = -\frac{1}{2},$$

$$\omega_1 = \frac{1}{12}, \quad \omega_2 = \frac{5}{12}, \quad \omega_3 = \frac{5}{12}, \quad \omega_4 = \frac{1}{12}.$$

The $u_{i\pm 1/2}^\pm$ and u_{i+G_l} are obtained by the HWENO method. Then, we can rewrite the semi-discrete system (3.2) as $U_i = \mathcal{L}(U)$, where \mathcal{L} denotes the operator of the spatial discretization, and the third-order total variation diminishing (TVD) Runge-Kutta time discretization [21] is utilized to solve the semi-discrete form (3.2):

$$\begin{cases} U^{(1)} = U^n + \Delta t \mathcal{L}(U^n), \\ U^{(2)} = \frac{3}{4}U^n + \frac{1}{4}(U^{(1)} + \Delta t \mathcal{L}(U^{(1)})), \\ U^{n+1} = \frac{1}{3}U^n + \frac{2}{3}(U^{(2)} + \Delta t \mathcal{L}(U^{(2)})). \end{cases}$$

2.2 HWENO Reconstruction

Here, we review the HWENO reconstruction method for hyperbolic conservation laws. The steps are similar to [31].

Procedure 1 Identify the troubled-cell

A cell is marked as a troubled cell, which means that the solution in the cell may contain discontinuity. Therefore, we can design a hybrid scheme by using the efficient linear reconstruction in the smooth area and the expensive HWENO method in the discontinuous region. Here, we choose the KXRCF troubled-cell indicator [11] and follow the suggestion in [31] to identify troubled-cell automatically. It is noted that, if cell I_i is marked as a troubled-cell through the KXRCF indicator, then its neighbours I_{i-1} and I_{i+1} will also be marked as a troubled-cell [31].

Procedure 2 Modify $\{\bar{v}_i\}$ from the values $\{\bar{u}_i\}$ and $\{\bar{v}_i\}$.

In order for better performance, we will modify the first moment \bar{v}_i when cell I_i is marked as a troubled-cell. The steps are as follows.

Step 1. Given a big stencil $S_0 = \{I_{i-1}, I_i, I_{i+1}\}$ and small stencils $S_1 = \{I_{i-1}, I_i\}$, $S_2 = \{I_i, I_{i+1}\}$, we need to construct a fifth order polynomial $p_0(x)$ and second order polynomials $p_1(x)$, $p_2(x)$, such that

$$\begin{cases} \frac{1}{\Delta x} \int_{I_{i+l}} p_0(x) dx = \bar{u}_{i+l}, & l = -1, 0, 1, \\ \frac{1}{\Delta x} \int_{I_{i\pm 1}} p_0(x) \frac{x - x_{i\pm 1}}{\Delta x} dx = \bar{v}_{i\pm 1}, \\ \frac{1}{\Delta x} \int_{I_{i+l}} p_1(x) dx = \bar{u}_{i+l}, & l = -1, 0, \\ \frac{1}{\Delta x} \int_{I_{i+l}} p_2(x) dx = \bar{u}_{i+l}, & l = 0, 1. \end{cases}$$

Then, we obtain the first moment of these polynomials:

$$\bar{v}_{i,p_m} = \frac{1}{\Delta x} \int_{I_i} p_m(x) \frac{x - x_i}{\Delta x} dx, \quad m = 0, 1, 2.$$

Step 2. Based on the formula in [10], we compute the smoothness indicators, denoted as $\beta_0, \beta_1,$ and $\beta_2,$ respectively. For each polynomials:

$$\beta_m = \frac{1}{\Delta x} \sum_{k=1}^r \int_{I_i} \left(\Delta x^k \frac{\partial^k}{\partial x^k} p_m(x) \right)^2 dx, \quad m = 0, 1, 2,$$

where $r = 4$ for $p_0(x), r = 1$ for $p_1(x)$ and $p_2(x)$. Then, we can compute

$$\tau = \frac{(\beta_0 - \beta_1)^2 + (\beta_0 - \beta_2)^2}{4}.$$

Step 3. We take the linear weights as

$$r_0 = 0.98, \quad r_1 = 0.01, \quad r_2 = 0.01.$$

Then, we have

$$\bar{v}_{i,p_0} = r_0 \left(\frac{1}{r_0} \bar{v}_{i,p_0} - \frac{r_1}{r_0} \bar{v}_{i,p_1} - \frac{r_2}{r_0} \bar{v}_{i,p_2} \right) + r_1 \bar{v}_{i,p_1} + r_2 \bar{v}_{i,p_2}.$$

Therefore, \bar{v}_{i,p_0} will be fifth order accuracy modification to \bar{v}_i .

Step 4. We compute the nonlinear weights

$$\omega_k = \frac{\bar{\omega}_k}{\bar{\omega}_0 + \bar{\omega}_1 + \bar{\omega}_2}, \quad \bar{\omega}_k = 1 + \frac{\tau}{\beta_k + \varepsilon}, \quad k = 0, 1, 2,$$

where $\varepsilon = 10^{-14}$ to avoid division by zero. Then, by replacing the linear weights with non-linear weights, we have

$$\bar{v}_i^{\text{mod}} = \omega_0 \left(\frac{1}{r_0} \bar{v}_{i,p_0} - \frac{r_1}{r_0} \bar{v}_{i,p_1} - \frac{r_2}{r_0} \bar{v}_{i,p_2} \right) + \omega_1 \bar{v}_{i,p_1} + \omega_2 \bar{v}_{i,p_2}.$$

Procedure 3 Reconstruct $\{u_{i+G_i}\}$ from the values $\{\bar{u}_i, \bar{v}_i^{\text{mod}}\}$.

If cell I_i is marked as a troubled-cell, then we will reconstruct u_{i+G_i} by the HWENO method. The steps are as follows.

Step 1. Given a big stencil $S_0 = \{I_{i-1}, I_i, I_{i+1}\}$ and small stencils $S_1 = \{I_{i-1}, I_i\}$, $S_2 = \{I_i, I_{i+1}\}$, we need to construct polynomials $p_0(x)$, $p_1(x)$, and $p_2(x)$, such that

$$\begin{aligned} \frac{1}{\Delta x} \int_{I_{i+l}} p_0(x) dx &= \bar{u}_{i+l}, & \frac{1}{\Delta x} \int_{I_{i+l}} p_0(x) \frac{x - x_{i+l}}{\Delta x} dx &= \bar{v}_{i+l}^{\text{mod}}, \quad l = -1, 0, 1, \\ \frac{1}{\Delta x} \int_{I_{i+l}} p_1(x) dx &= \bar{u}_{i+l}, & \frac{1}{\Delta x} \int_{I_i} p_1(x) \frac{x - x_i}{\Delta x} dx &= \bar{v}_i^{\text{mod}}, \quad l = -1, 0, \\ \frac{1}{\Delta x} \int_{I_{i+l}} p_2(x) dx &= \bar{u}_{i+l}, & \frac{1}{\Delta x} \int_{I_i} p_2(x) \frac{x - x_i}{\Delta x} dx &= \bar{v}_i^{\text{mod}}, \quad l = 0, 1. \end{aligned}$$

Step 2. We compute the smoothness indicators, denoted as β_0, β_1 , and β_2 , respectively, for each polynomials based on the formula in [10]:

$$\beta_m = \frac{1}{\Delta x} \sum_{k=1}^r \int_{I_i} \left(\Delta x^k \frac{\partial^k}{\partial x^k} p_m(x) \right)^2 dx, \quad m = 0, 1, 2,$$

where $r = 5$ for $p_0(x)$, $r = 2$ for $p_1(x)$ and $p_2(x)$. Then, we can compute

$$\tau = \frac{(\beta_0 - \beta_1)^2 + (\beta_0 - \beta_2)^2}{4}.$$

Step 3. We take the linear weights as

$$r_0 = 0.98, \quad r_1 = 0.01, \quad r_2 = 0.01.$$

Then, we have

$$p_0(x_{i+G_i}) = r_0 \left(\frac{1}{r_0} p_0(x_{i+G_i}) - \frac{r_1}{r_0} p_1(x_{i+G_i}) - \frac{r_2}{r_0} p_2(x_{i+G_i}) \right) + r_1 p_1(x_{i+G_i}) + r_2 p_2(x_{i+G_i}).$$

Step 4. We compute the nonlinear weights

$$\omega_k = \frac{\bar{\omega}_k}{\bar{\omega}_0 + \bar{\omega}_1 + \bar{\omega}_2}, \quad \bar{\omega}_k = 1 + \frac{\tau}{\beta_k + \varepsilon}, \quad k = 0, 1, 2,$$

where $\varepsilon = 10^{-14}$ to avoid division by zero. Then, we have

$$u_{i+G_i} = \omega_0 \left(\frac{1}{r_0} p_0(x_{i+G_i}) - \frac{r_1}{r_0} p_1(x_{i+G_i}) - \frac{r_2}{r_0} p_2(x_{i+G_i}) \right) + \omega_1 p_1(x_{i+G_i}) + \omega_2 p_2(x_{i+G_i}).$$

However, if cell I_i is good, we will simply take $u_{i+G_i} = p_0(x_{i+G_i})$, where $p_0(x)$ is the polynomial obtained from the above steps.

3 The HWENO Method in 2D

3.1 The Framework of the Finite Volume Scheme for Solving 2D Euler Equations

Two dimensional hyperbolic conservation laws can be written as follows:

$$\begin{cases} u_t + f(u)_x + g(u)_y = 0, \\ u(x, y, 0) = u_0(x, y). \end{cases} \tag{2}$$

For simplicity, the computational domain is equally divided: $a = x_{1/2} < x_{3/2} < \dots < x_{N_x+1/2} = b$, $c = y_{1/2} < y_{3/2} < \dots < y_{N_y+1/2} = d$. The cell is denoted as $I_{i,j} = [x_{i-1/2}, x_{i+1/2}] \times [y_{j-1/2}, y_{j+1/2}]$, where the cell center is denoted as (x_i, y_j) and the mesh size is denoted as $\Delta x = x_{i+1/2} - x_{i-1/2}$, $\Delta y = y_{j+1/2} - y_{j-1/2}$. Furthermore, we define

$$\begin{aligned} \bar{u}_{i,j} &= \frac{1}{\Delta x \Delta y} \iint_{I_{i,j}} u(x, y, t) dx dy, & \bar{v}_{i,j} &= \frac{1}{\Delta x \Delta y} \iint_{I_{i,j}} u(x, y, t) \frac{x - x_i}{\Delta x} dx dy, \\ \bar{w}_{i,j} &= \frac{1}{\Delta x \Delta y} \iint_{I_{i,j}} u(x, y, t) \frac{y - y_j}{\Delta y} dx dy, & \bar{s}_{i,j} &= \frac{1}{\Delta x \Delta y} \iint_{I_{i,j}} u(x, y, t) \frac{(x - x_i)(y - y_j)}{\Delta x \Delta y} dx dy. \end{aligned}$$

We multiply (2) by functions 1 , $\frac{x-x_i}{\Delta x}$, $\frac{y-y_j}{\Delta y}$, and $\frac{(x-x_i)(y-y_j)}{\Delta x \Delta y}$. Use the integration by parts formula. Then we obtain

$$\left\{ \begin{aligned} \frac{d\bar{u}_{i,j}}{dt} &= -\frac{1}{\Delta x \Delta y} \int_{y_{j-1/2}}^{y_{j+1/2}} [f(u(x_{i+1/2}, y)) - f(u(x_{i-1/2}, y))] dy \\ &\quad - \frac{1}{\Delta x \Delta y} \int_{x_{i-1/2}}^{x_{i+1/2}} [g(u(x, y_{j+1/2})) - g(u(x, y_{j-1/2}))] dx, \\ \frac{d\bar{v}_{i,j}}{dt} &= -\frac{1}{2\Delta x \Delta y} \int_{y_{j-1/2}}^{y_{j+1/2}} [f(u(x_{i+1/2}, y)) + f(u(x_{i-1/2}, y))] dy + \frac{1}{\Delta x^2 \Delta y} \iint_{I_{i,j}} f(u) dx dy \\ &\quad - \frac{1}{\Delta x \Delta y} \int_{x_{i-1/2}}^{x_{i+1/2}} [g(u(x, y_{j+1/2})) - g(u(x, y_{j-1/2}))] \frac{x - x_i}{\Delta x} dx, \\ \frac{d\bar{w}_{i,j}}{dt} &= -\frac{1}{\Delta x \Delta y} \int_{y_{j-1/2}}^{y_{j+1/2}} [f(u(x_{i+1/2}, y)) - f(u(x_{i-1/2}, y))] \frac{y - y_j}{\Delta y} dy \\ &\quad - \frac{1}{2\Delta x \Delta y} \int_{x_{i-1/2}}^{x_{i+1/2}} [g(u(x, y_{j+1/2})) + g(u(x, y_{j-1/2}))] dx + \frac{1}{\Delta x \Delta y^2} \iint_{I_{i,j}} g(u) dx dy, \\ \frac{d\bar{s}_{i,j}}{dt} &= -\frac{1}{2\Delta x \Delta y} \int_{y_{j-1/2}}^{y_{j+1/2}} [f(u(x_{i+1/2}, y)) + f(u(x_{i-1/2}, y))] \frac{y - y_j}{\Delta y} dy + \frac{1}{\Delta x^2 \Delta y} \iint_{I_{i,j}} f(u) \frac{y - y_j}{\Delta y} dx dy \\ &\quad - \frac{1}{2\Delta x \Delta y} \int_{x_{i-1/2}}^{x_{i+1/2}} [g(u(x, y_{j+1/2})) + g(u(x, y_{j-1/2}))] \frac{x - x_i}{\Delta x} dx + \frac{1}{\Delta x \Delta y^2} \iint_{I_{i,j}} g(u) \frac{x - x_i}{\Delta x} dx dy. \end{aligned} \right.$$

After replacing the fluxes with numerical fluxes and integration with numerical integration, we have the following scheme:

$$\begin{aligned}
\frac{d\bar{u}_{ij}}{dt} &= -\frac{1}{\Delta x} \sum_l \omega_l \left[\hat{f} \left(u_{i+1/2,j+G_l}^-, u_{i+1/2,j+G_l}^+ \right) - \hat{f} \left(u_{i-1/2,j+G_l}^-, u_{i-1/2,j+G_l}^+ \right) \right] \\
&\quad - \frac{1}{\Delta y} \sum_l \omega_l \left[\hat{g} \left(u_{i+G_l,j+1/2}^-, u_{i+G_l,j+1/2}^+ \right) - \hat{g} \left(u_{i+G_l,j-1/2}^-, u_{i+G_l,j-1/2}^+ \right) \right], \\
\frac{d\bar{v}_{ij}}{dt} &= -\frac{1}{2\Delta x} \sum_l \omega_l \left[\hat{f} \left(u_{i+1/2,j+G_l}^-, u_{i+1/2,j+G_l}^+ \right) + \hat{f} \left(u_{i-1/2,j+G_l}^-, u_{i-1/2,j+G_l}^+ \right) \right] \\
&\quad + \frac{1}{\Delta x} \sum_{l_1} \sum_{l_2} \omega_{l_1} \omega_{l_2} f \left(u_{i+G_{l_1},j+G_{l_2}} \right) \\
&\quad - \frac{1}{\Delta x} \sum_l \omega_l G_l \left[\hat{g} \left(u_{i+G_l,j+1/2}^-, u_{i+G_l,j+1/2}^+ \right) - \hat{g} \left(u_{i+G_l,j-1/2}^-, u_{i+G_l,j-1/2}^+ \right) \right], \\
\frac{d\bar{w}_{ij}}{dt} &= -\frac{1}{\Delta x} \sum_l \omega_l G_l \left[\hat{f} \left(u_{i+1/2,j+G_l}^-, u_{i+1/2,j+G_l}^+ \right) - \hat{f} \left(u_{i-1/2,j+G_l}^-, u_{i-1/2,j+G_l}^+ \right) \right] \\
&\quad - \frac{1}{2\Delta y} \sum_l \omega_l \left[\hat{g} \left(u_{i+G_l,j+1/2}^-, u_{i+G_l,j+1/2}^+ \right) + \hat{g} \left(u_{i+G_l,j-1/2}^-, u_{i+G_l,j-1/2}^+ \right) \right] \\
&\quad + \frac{1}{\Delta y} \sum_{l_1} \sum_{l_2} \omega_{l_1} \omega_{l_2} g \left(u_{i+G_{l_1},j+G_{l_2}} \right), \\
\frac{d\bar{s}_{ij}}{dt} &= -\frac{1}{2\Delta x} \sum_l \omega_l G_l \left[\hat{f} \left(u_{i+1/2,j+G_l}^-, u_{i+1/2,j+G_l}^+ \right) + \hat{f} \left(u_{i-1/2,j+G_l}^-, u_{i-1/2,j+G_l}^+ \right) \right] \\
&\quad - \frac{1}{2\Delta y} \sum_l \omega_l G_l \left[\hat{g} \left(u_{i+G_l,j+1/2}^-, u_{i+G_l,j+1/2}^+ \right) + \hat{g} \left(u_{i+G_l,j-1/2}^-, u_{i+G_l,j-1/2}^+ \right) \right] \\
&\quad + \frac{1}{\Delta x} \sum_{l_1} \sum_{l_2} \omega_{l_1} \omega_{l_2} G_{l_2} f \left(u_{i+G_{l_1},j+G_{l_2}} \right) + \frac{1}{\Delta y} \sum_{l_1} \sum_{l_2} \omega_{l_1} \omega_{l_2} G_{l_1} g \left(u_{i+G_{l_1},j+G_{l_2}} \right).
\end{aligned}$$

The $u_{i\pm 1/2,j+G_l}^\pm$, $u_{i+G_l,j\pm 1/2}^\pm$, $u_{i+G_{l_1},j+G_{l_2}}$ are obtained by the HWENO method, which will be introduced in later section. Then, we can rewrite the semi-discrete system (3.2) as $U_t = \mathcal{L}(U)$ and use the third-order TVD Runge-Kutta time discretization [21] to solve the semi-discrete form.

3.2 Dimension by Dimension HWENO Reconstruction

Dimension by dimension HWENO reconstruction, which can maintain both high order accuracy in the smooth case and keep high resolution when discontinuity appears, is the key component for our scheme. It can be regarded as a generalization of the 1D case. Before describing the dimension by dimension HWENO reconstruction algorithm, we need to introduce **Procedure 4**.

Procedure 4 Modify $\{\bar{s}_{ij}\}$ from the values $\{\bar{w}_{ij}\}$.

If cell I_{ij} is a troubled-cell, then we need to modify value \bar{s}_{ij} . The steps are as follows.

Step 1. Given a big stencil $S_0 = \{I_{i-2,j}, I_{i-1,j}, I_{ij}, I_{i+1,j}, I_{i+2,j}\}$ and small stencils $S_1 = \{I_{i-1,j}, I_{ij}\}$, $S_2 = \{I_{ij}, I_{i+1,j}\}$, we need to construct polynomials $p_0(x)$, $p_1(x)$, and $p_2(x)$, such that

$$\begin{aligned} \frac{1}{\Delta x} \int_{I_{i+l}} p_0(x) dx &= \bar{w}_{i+l,j}, \quad l = -2, -1, 0, 1, 2, \\ \frac{1}{\Delta x} \int_{I_{i+l}} p_1(x) dx &= \bar{w}_{i+l,j}, \quad l = -1, 0, \\ \frac{1}{\Delta x} \int_{I_{i+l}} p_2(x) dx &= \bar{w}_{i+l,j}, \quad l = 0, 1. \end{aligned}$$

Then, we can obtain the first moment of these polynomials:

$$\bar{s}_{i,j,p_m} = \frac{1}{\Delta x} \int_{I_i} p_m(x) \frac{x - x_i}{\Delta x} dx, \quad m = 0, 1, 2.$$

It is noted that \bar{s}_{i,j,p_0} , \bar{s}_{i,j,p_1} , and \bar{s}_{i,j,p_2} are all the approximations to the value $\bar{s}_{i,j}$ in the smooth region.

Step 2. We compute the smoothness indicators, denoted as β_0 , β_1 , and β_2 , respectively. For each polynomial based on the formula in [10]:

$$\beta_m = \frac{1}{\Delta x} \sum_{k=1}^r \int_{x_{i-1/2}}^{x_{i+1/2}} \left(\Delta x^k \frac{\partial^k p_m(x)}{\partial x^k} \right)^2 dx, \quad m = 0, 1, 2,$$

where $r = 4$ for $p_0(x)$, $r = 1$ for $p_1(x)$ and $p_2(x)$. Then, we can compute

$$\tau = \frac{(\beta_0 - \beta_1)^2 + (\beta_0 - \beta_2)^2}{4}.$$

Step 3. We take the linear weights as

$$r_0 = 0.98, \quad r_1 = 0.01, \quad r_2 = 0.01.$$

Then, we have

$$\bar{s}_{i,j,p_0} = r_0 \left(\frac{1}{r_0} \bar{s}_{i,j,p_0} - \frac{r_1}{r_0} \bar{s}_{i,j,p_1} - \frac{r_2}{r_0} \bar{s}_{i,j,p_2} \right) + r_1 \bar{s}_{i,j,p_1} + r_2 \bar{s}_{i,j,p_2}.$$

Step 4. We compute the nonlinear weights

$$\omega_k = \frac{\bar{\omega}_k}{\bar{\omega}_0 + \bar{\omega}_1 + \bar{\omega}_2}, \quad \bar{\omega}_k = 1 + \frac{\tau}{\beta_k + \varepsilon}, \quad k = 0, 1, 2,$$

where $\varepsilon = 10^{-14}$ to avoid division by zero. Then, we have

$$\bar{s}_{i,j}^{\text{mod}} = \omega_0 \left(\frac{1}{r_0} \bar{s}_{i,j,p_0} - \frac{r_1}{r_0} \bar{s}_{i,j,p_1} - \frac{r_2}{r_0} \bar{s}_{i,j,p_2} \right) + \omega_1 \bar{s}_{i,j,p_1} + \omega_2 \bar{s}_{i,j,p_2}.$$

Remark 1 To conclude, we input the variable $\{\bar{w}_{i,j}\}$ and implement the above procedure in the x direction. Then we obtain the variable $\{\bar{s}_{i,j}^{\text{mod}}\}$. If we input the variable $\{\bar{v}_{i,j}\}$ and implement the above procedure in the y direction, we can also obtain the variable $\{\bar{s}_{i,j}^{\text{mod}}\}$.

Remark 2 In the 2D case, we will implement the KXRFC indicator in the dimension by dimension way. For the 2D Euler equations, the density and energy will be the indicator

variables. What's more, if cell $I_{i,j}$ is marked as a troubled-cell, then its eight neighbours will also be marked as troubled-cells.

Remark 3 For the system cases, we will perform HWENO reconstruction procedures on the local characteristic field if the target cell is marked as a troubled cell, otherwise we will simply perform the linear approximation procedures component by component.

Algorithm 1 Dimension by dimension HWENO method.

Input: $\{\bar{u}_{i,j}, \bar{v}_{i,j}, \bar{w}_{i,j}, \bar{s}_{i,j}\}$

Output: $u_{i+G_{l_1}, j+G_{l_2}}$

- i) Apply Procedure 1 in the dimension by dimension way to identify the troubled-cell
 - ii) Apply Procedure 2 in the x direction, input $\{\bar{u}_{i,j}, \bar{v}_{i,j}\}$, output $\{\bar{v}_{i,j}^{\text{mod}}\}$
 - iii) Apply Procedure 3 in the y direction, input $\{\bar{u}_{i,j}, \bar{w}_{i,j}\}$, output $\{\bar{w}_{i,j}^{\text{mod}}\}$
 - iv) Apply Procedure 4 in the x direction, input $\{\bar{w}_{i,j}^{\text{mod}}\}$, output $\{\bar{s}_{i,j}^{(1)}\}$. Apply Procedure 4 in the y direction, input $\{\bar{v}_{i,j}^{\text{mod}}\}$, output $\{\bar{s}_{i,j}^{(2)}\}$. Take $\bar{s}_{i,j}^{\text{mod}} = (\bar{s}_{i,j}^{(1)} + \bar{s}_{i,j}^{(2)})/2$
 - v) Apply Procedure 3 in the x direction, input $\{\bar{u}_{i,j}, \bar{v}_{i,j}^{\text{mod}}\}$, output $\{\bar{u}_{i+G_{l_1}, \bullet}\}$
 - vi) Apply Procedure 3 in the x direction, input $\{\bar{w}_{i,j}^{\text{mod}}, \bar{s}_{i,j}^{\text{mod}}\}$, output $\{\bar{w}_{i+G_{l_1}, \bullet}\}$
 - vii) Apply Procedure 3 in the y direction, input $\{\bar{u}_{i+G_{l_1}, \bullet}, \bar{w}_{i+G_{l_1}, \bullet}\}$, output $\{u_{i+G_{l_1}, j+G_{l_2}}\}$
-

4 Numerical Test

In this section, we will show the numerical results of the dimension by dimension HWENO method. We take the CFL number as 0.6.

Example 1 Accuracy test for the Burgers equation.

We solve the following nonlinear Burgers equation:

$$u_t + \left(\frac{u^2}{2}\right)_x + \left(\frac{u^2}{2}\right)_y = 0$$

with the initial condition $u(x, y, 0) = 0.5 + \sin(\pi(x+y)/2)$ and periodic boundary condition. The computational domain is $[0, 4] \times [0, 4]$. When $t = 0.5/\pi$, the solution is still smooth. Table 1 shows the numerical error. From the table, we can see that our scheme can achieve the designed precision, and the numerical errors are smaller than the one obtained by the new hybrid HWENO method [31]. The numerical error against CPU time graphs of our method and WENO method are in Fig. 1. We can see that our method can achieve smaller numerical error at the cost of more CPU time than the WENO method in this numerical test.

Example 2 Accuracy test for the Euler system.

We consider the 2D Euler system accuracy test. The initial conditions are

$$\rho(x, y, 0) = 1 + 0.2 \sin(\pi(x+y)), \quad u(x, y, 0) = v(x, y, 0) = 1, \quad p(x, y, 0) = 1.$$

Table 1 Accuracy test for Burgers equation

Mesh size	Dimension by dimension HWENO				New hybrid HWENO			
	L_∞ error	Order	L_1 error	Order	L_∞ error	Order	L_1 error	Order
40× 40	3.85E−06		3.58E−07		2.49E−05		2.70E−06	
80× 80	4.19E−08	6.52	7.45E−09	5.59	8.91E−07	4.81	5.01E−08	5.75
120× 120	6.32E−09	4.67	9.86E−10	4.99	7.81E−08	6.00	4.15E−09	6.14
160× 160	1.63E−09	4.71	2.42E−10	4.89	1.27E−08	6.32	7.00E−10	6.18
200× 200	5.58E−10	4.81	8.07E−11	4.92	3.26E−09	6.09	1.94E−10	5.74
240× 240	2.30E−10	4.87	3.27E−11	4.94	1.17E−09	5.63	7.65E−11	5.12

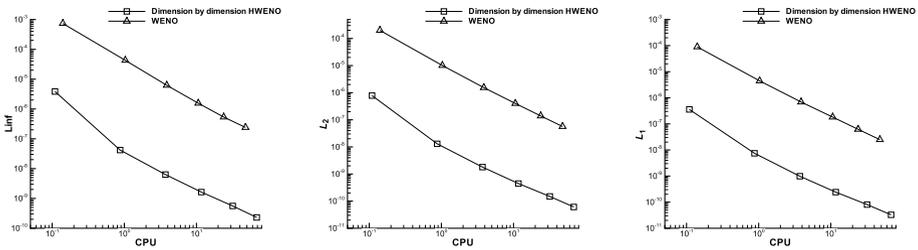


Fig. 1 Computing CPU time and error for accuracy test of Burgers equation. Scales of CPU time and error are logarithmic

Table 2 Accuracy test for the Euler system

Mesh size	Dimension by dimension HWENO				New hybrid HWENO			
	L_∞ error	Order	L_1 error	Order	L_∞ error	Order	L_1 error	Order
30× 30	5.59E−08		3.57E−08		4.87E−07		3.11E−07	
60× 60	1.74E−09	5.01	1.10E−09	5.01	7.14E−09	6.09	4.55E−09	6.09
90× 90	2.28E−10	5.01	1.45E−10	5.00	6.20E−10	6.03	3.95E−10	6.03
120× 120	5.41E−11	5.00	3.44E−11	5.00	1.10E−10	6.00	7.01E−11	6.01
150× 150	1.77E−11	5.00	1.13E−11	5.00	3.00E−11	5.84	1.84E−11	5.99
180× 180	7.12E−12	5.00	4.53E−12	5.00	9.71E−12	5.98	6.19E−12	5.98

The computational domain is $[0, 2] \times [0, 2]$. Periodic boundary conditions are used in this test. The exact solution of ρ is

$$\rho(x, y, t) = 1 + 0.2 \sin(\pi(x + y - 2t)).$$

We set the final time $t = 2$. We list the error in Table 2. We can also see that our method achieves the designed fifth order accuracy, and the numerical errors are smaller than those obtained by the new hybrid HWENO method [31]. The numerical error against CPU time graphs of our method and WENO method are in Fig. 2. We can also see that our method performs better than the WENO method in this numerical test.

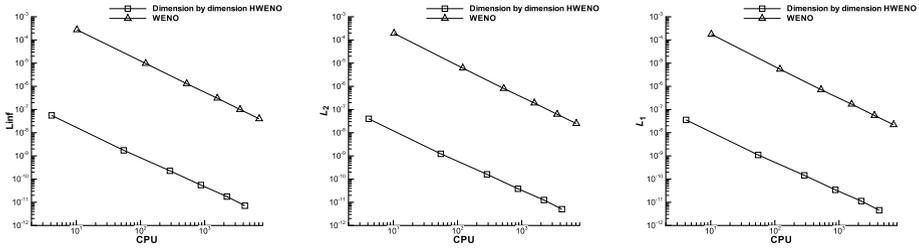


Fig. 2 Computing CPU time and error for accuracy test of the Euler system. Scales of CPU time and error are logarithmic

Example 3 Two-dimensional vortex evolution problem for the Euler equations.

We consider the accuracy test for the compressible Euler equations of gas dynamics. The mean flow is $\rho = 1, p = 1,$ and $(u, v) = (1, 1)$. We add an isentropic vortex to the mean flow (perturbations in (u, v) and the temperature $T = \frac{p}{\rho}$, no perturbation in the entropy $S = \frac{p}{\rho^\gamma}$):

$$(\delta u, \delta v) = \frac{\epsilon}{2\pi} e^{0.5(1-r^2)(-\bar{y}, \bar{x})},$$

where $(\bar{x}, \bar{y}) = (x - 5, y - 5), r^2 = x^2 + y^2,$ and the vortex strength $\epsilon = 5$. The computational domain is taken as $[0, 10] \times [0, 10],$ and periodic boundary conditions are used. The exact solution of the Euler equation with the above initial and boundary conditions is just the passive convection of the vortex with the mean velocity. We compute the solution to the time $T = 2.0$ for the accuracy, see Table 3. We can see that our method can achieve the designed order.

Example 4 Discontinuity test for the Burgers equation.

We solve the following nonlinear Burgers equation:

$$u_t + \left(\frac{u^2}{2}\right)_x + \left(\frac{u^2}{2}\right)_y = 0$$

with the initial condition $u(x, y, 0) = 0.5 + \sin(\pi(x + y)/2)$ and periodic boundary condition. The computational domain is $[0, 4] \times [0, 4].$ We take the final time as $t = 1.5/\pi.$ Now, the discontinuity appears. From Fig. 3, we can see that our scheme can obtain the high resolution solution.

Example 5 The 2D Riemann problem.

The 2D Riemann problems have been studied in [12, 27, 33]. Now they have become benchmarks to test the resolution of the scheme. The computational domain is $[0, 1] \times [0, 1],$ and the initial data are given by

Table 3 Two-dimensional vortex evolution problem for the Euler equations

Mesh size	L_∞ error	Order	L_2 error	Order	L_1 error	Order
10× 10	2.34E−01		4.86E−02		2.30E−02	
20× 20	2.24E−01	0.07	2.97E−02	0.71	1.25E−02	0.88
40× 40	2.02E−02	3.47	2.44E−03	3.61	1.21E−03	3.36
80× 80	7.42E−05	8.09	1.24E−05	7.62	4.32E−06	8.13
160× 160	2.44E−06	4.92	4.06E−07	4.93	1.40E−07	4.95
320× 320	7.56E−08	5.01	1.27E−08	4.99	4.69E−09	4.90

$$\begin{aligned}
 \text{(a) } (\rho, u, v, p) &= \begin{cases} (1.5, 0, 0, 1.5), & x > 0.5, y > 0.5, \\ (0.5323, 1.206, 0, 0.3), & x \leq 0.5, y > 0.5, \\ (0.5323, 0, 1.206, 0.3), & x > 0.5, y \leq 0.5, \\ (0.138, 1.206, 1.206, 0.029), & x \leq 0.5, y \leq 0.5, \end{cases} \\
 \text{(b) } (\rho, u, v, p) &= \begin{cases} (0.5313, 0, 0, 0.4), & x > 0.5, y > 0.5, \\ (1, 0.7276, 0, 1), & x \leq 0.5, y > 0.5, \\ (1, 0, 0.7276, 1), & x > 0.5, y \leq 0.5, \\ (0.8, 0, 0, 1), & x \leq 0.5, y \leq 0.5, \end{cases} \\
 \text{(c) } (\rho, u, v, p) &= \begin{cases} (1, 0.1, 0, 1), & x > 0.5, y > 0.5, \\ (0.5313, 0.8276, 0, 0.4), & x \leq 0.5, y > 0.5, \\ (0.5313, 0.1, 0.7276, 0.4), & x > 0.5, y \leq 0.5, \\ (0.8, 0.1, 0, 0.4), & x \leq 0.5, y \leq 0.5, \end{cases} \\
 \text{(d) } (\rho, u, v, p) &= \begin{cases} (1.1, 0, 0, 1.1), & x > 0.5, y > 0.5, \\ (0.5065, 0.8939, 0, 0.35), & x \leq 0.5, y > 0.5, \\ (0.5065, 0, 0.8939, 0.35), & x > 0.5, y \leq 0.5, \\ (1.1, 0.8939, 0.8939, 1.1), & x \leq 0.5, y \leq 0.5. \end{cases}
 \end{aligned}$$

We use 400×400 cells to solve the problem. Figure 4 shows the density contours with 30 equally spaced contour lines. We can see that our scheme can correctly capture the complicated flow structure. The computational results are comparable to those in the literature.

Example 6 Double Mach reflection.

Double Mach reflection is a standard test case for high resolution schemes [20]. High order schemes are good for resolving the flow below the Mach stem. The computational domain for this problem is $[0, 4] \times [0, 1]$. The wall lies at the bottom boundary from $x = 1/6$. A right-moving $Mach = 10$ shock is located at $x = 1/6, y = 0$, making a 60° angle with the x -axis. The reflective boundary condition is used at the wall, while the exact post-shock condition is used at the rest of the bottom boundary. At the top boundary, the flow values are set to describe the exact motion of the Mach 10 shock. Figure 5 shows the

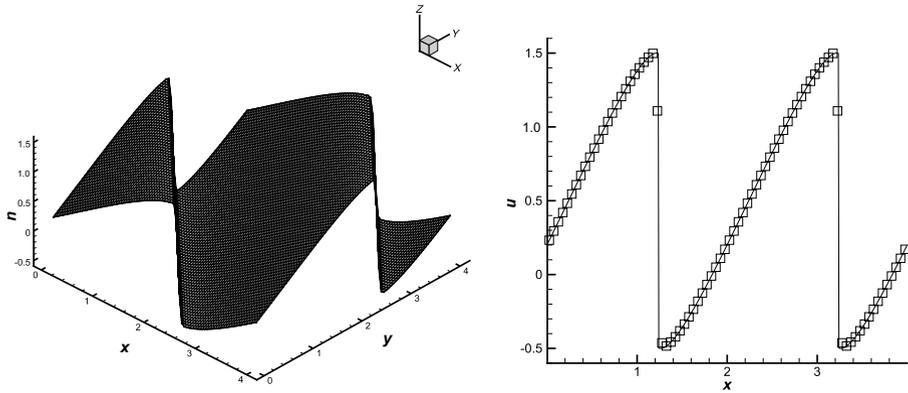


Fig. 3 Burgers equation. 80×80 cells. Left: the surface of the solution; right: solution along the cut line $y = x$

computational result at time $t = 0.2$ with 1920×480 cells. From the figure, we can see that the scheme can capture the complicated flow structure well.

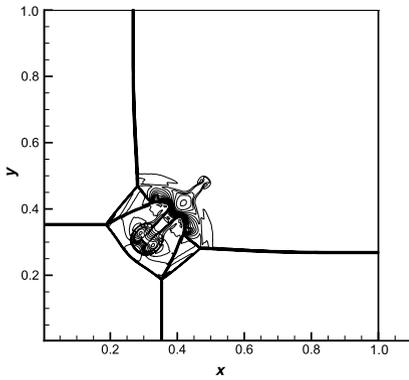
Example 7 A Mach 3 wind tunnel with a step.

It is also a standard test case for high resolution schemes. The computational domain is $[0, 3] \times [0, 1]$ except $[0.6, 3] \times [0, 0.2]$ which simulates a 1 length unit wide and 3 length units long wind tunnel with a 0.2 length unites high step located at 0.6 length units from the left-hand end of the tunnel. The problem is initialized by a right-going Mach 3 flow. Reflective boundary conditions are applied along the walls of the tunnel. The inflow and outflow boundary conditions are applied to the left-hand end and right-hand end. The corner of the step is a singularity point. [25] used an assumption of nearly steady flow in the region near the corner to fix the singularity. In this paper, we do not modify our method near the corner. Figure 6 shows the contour picture for the density at time $t = 4$ with 960×320 cells. We can see that the scheme can solve the problem well.

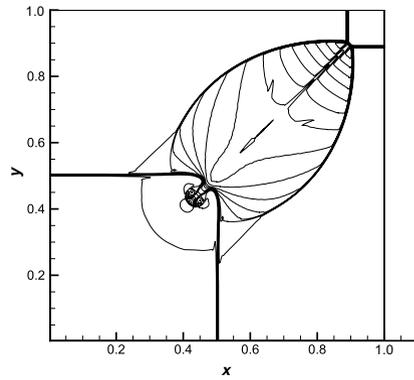
Example 8 2D shock vortex interactions.

The model problem shows the interaction between a stationary shock and a vortex. The computational domain is taken as $[0, 2] \times [0, 1]$. A stationary $Mach = 1.1$ shock is positioned at $x = 0.5$. The left state is $(\bar{\rho}, \bar{u}, \bar{v}, \bar{p}) = (1, 1.1\sqrt{\gamma}, 0, 1)$ and the right state can be obtained through the Rankine-Hugoniot condition. An isentropic vortex is at $(0.25, 0.5)$. The perturbation to the velocity (u, v) , the temperature $(T = p/\rho)$, and the entropy $(S = \ln(p/\rho^\gamma))$ are as follows:

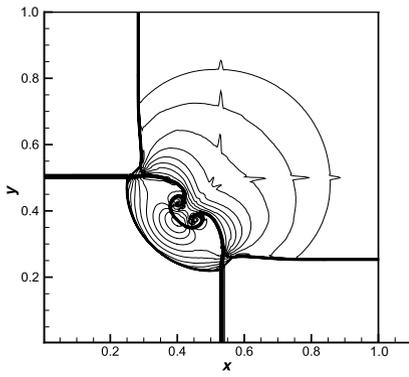
$$\begin{cases} (\delta u, \delta v) = \epsilon \tau e^{\alpha(1-\tau^2)} (\sin \theta, -\cos \theta), \\ \delta T = -\frac{(\gamma - 1)\epsilon^2 e^{2\alpha(1-\tau^2)}}{4\alpha\gamma}, \\ \delta S = 0, \end{cases}$$



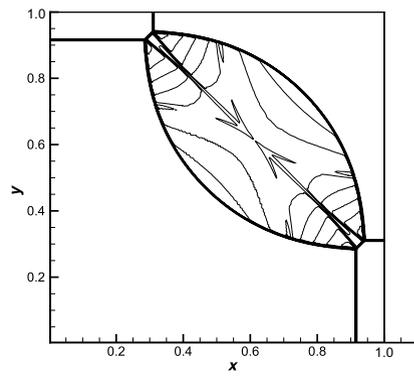
(a) Density contour from 0.1 to 1.8, $T=0.35$



(b) Density contour from 0.5 to 1.9, $T=0.25$



(c) Density contour from 0.47 to 1.3, $T=0.3$



(d) Density contour from 0.5 to 2.0, $T=0.25$

Fig. 4 The 2D Riemann Problem

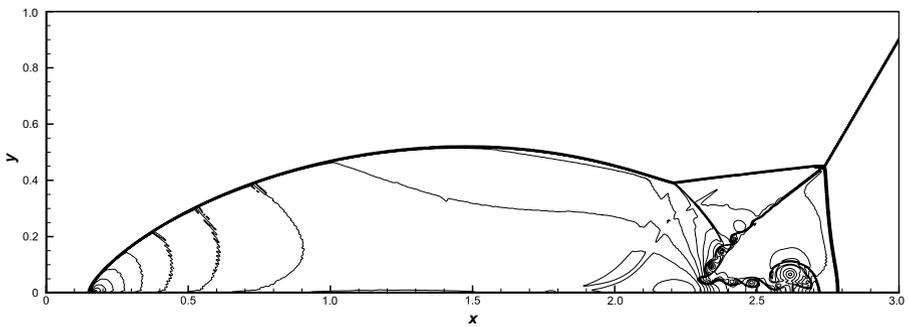


Fig. 5 Double Mach reflection. Density, 30 equally spaced contour lines from 1.5 to 22.7

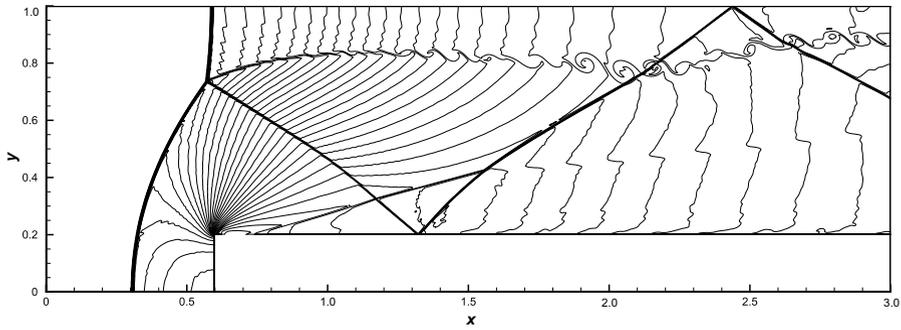


Fig. 6 A Mach 3 wind tunnel with a step. Density, 30 equally spaced contour lines from 0.32 to 6.15

where

$$\tau = r/r_c, R = \sqrt{(x - x_c)^2 + (y - y_c)^2}, \theta = \arctan \left(\frac{y - y_c}{x - x_c} \right).$$

Here, r_c , which is the radius of the vortex, is taken as $r_c = 0.05$; α , which describes the decay rate of the vortex, is taken as $\alpha = 0.204$; ϵ , which indicates the strength of the vortex, is taken as $\epsilon = 0.3$. Therefore, the initial values for the post-shock field are

$$(\rho_l, u_l, v_l, p_l) = \left((\bar{T} + \delta T)^{\frac{1}{\gamma-1}}, \bar{u} + \delta u, \bar{v} + \delta v, (\bar{T} + \delta T)^{\frac{\gamma}{\gamma-1}} \right),$$

where $\bar{T} = \bar{p}/\bar{\rho}$. We use 250×100 cells. The upper and bottom boundaries are set to be reflective. Figure 7 shows the pressure contour lines at time $t = 0.05, t = 0.2, t = 0.35, t = 0.6,$ and $t = 0.8,$ respectively. We can see that our scheme can capture the shock and vortex well.

Example 9 Shock passing a backward facing corner (diffraction).

The shock diffraction is a very common phenomenon. The problem has been tested in [4]. It has a different structure according to different Mach numbers and different shapes of backward facing corners. Here, we consider Shock diffraction around a 90 degree corner. The computational domain is $[0, 1] \times [6, 11]$ and $[1, 13] \times [0, 11]$. The initial condition is a pure right moving shock of *Mach* = 5.09 located at $x = 0.5$ and $6 \leq y \leq 11$, moving into undisturbed air with the density $\rho = 1.4$, the velocity $u = v = 0$, and the pressure $p = 1$. The initial inflow boundary condition is specified at $x = 0, 6 \leq y \leq 11$, and the initial outflow boundary condition is specified at $x = 13, 0 \leq y \leq 11$. The reflective boundary conditions is at $0 \leq x \leq 1, y = 6$ and $x = 1, 0 \leq y \leq 6$. The Neumann boundary condition is at $1 \leq x \leq 13, y = 0$ and $0 \leq x \leq 13, y = 11$. The final time is $t = 2.3$. Due to the expansion of the flow, the density and pressure will suffer a sudden drop around the corner. Many schemes fail to solve this problem because of the negative density or pressure during the computation. Here, we use the dimension by dimension HWEMO method to solve this problem with 1040×880 cells. Although the method does not contain any positivity-preserving techniques, it maintains the positivity of the

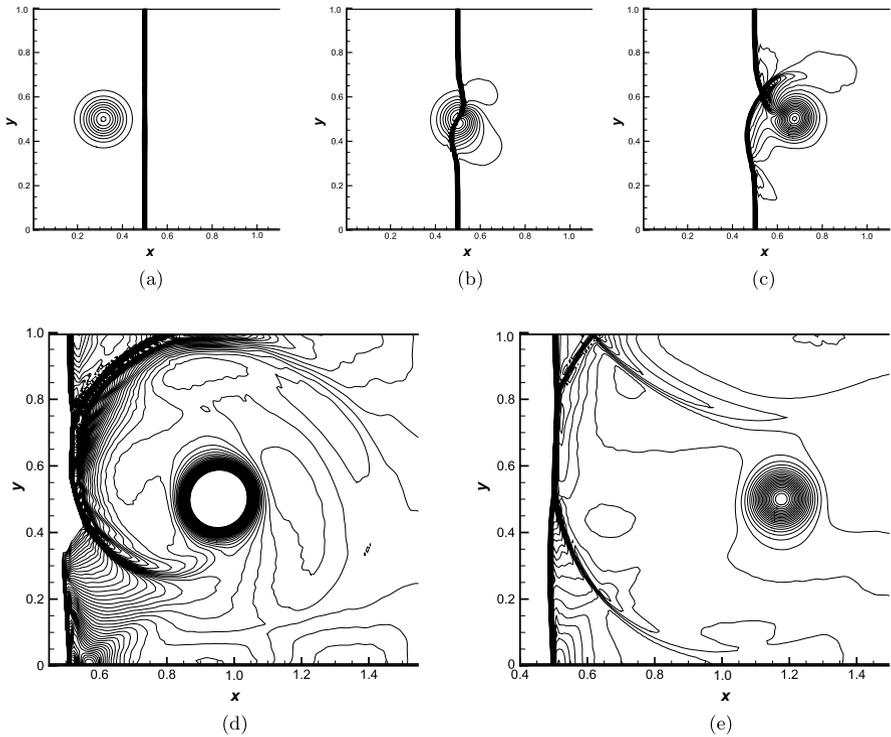


Fig. 7 The 2D shock vortex interaction. Pressure. (a) $t=0.05$; (b) $t=0.2$; (c) $t=0.35$; (e) $t=0.6$. (d) $t=0.8$, 90 contours from 1.19 to 1.37

density and pressure during the computation. From Fig. 8, we can see that the results are comparable with that in [4].

Example 10 High Mach number astrophysical jets.

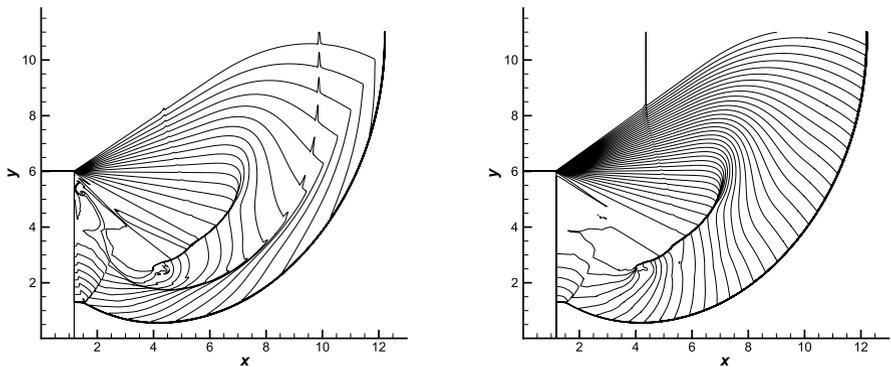


Fig. 8 Shock diffraction problem. Left: density, 20 equally spaced contour lines from 0.066 227 to 7.066 8; right: pressure, 40 equally spaced contour lines from 0.091 to 31

We consider two high Mach number astrophysical jets without the radiative cooling [6]. In these examples, we take $\gamma = 5/3$. The first example is Mach 80 problem. The computational domain is $[0, 2] \times [-0.5, 0.5]$, which is initialized by setting $(\rho, u, v, p) = (0.5, 0, 0, 0.4127)$. The right, top, and bottom boundaries are outflows. For the left boundary, $(\rho, u, v, p) = (5, 30, 0, 0.4127)$ if $y \in [-0.05, 0.05]$ and $(\rho, u, v, p) = (0.5, 0, 0, 0.4127)$ otherwise. The numerical result for the density, pressure, and temperature with 448×224 grid points at time $t = 0.07$ are shown in Fig. 9. The second example is Mach 2 000 problem. The computational domain is $[0, 1] \times [-0.25, 0.25]$, which is full of gas with $(\rho, u, v, p) = (0.5, 0, 0, 0.4127)$. The right, top, and bottom boundary are outflows. For the left boundary, $(\rho, u, v, p) = (5, 800, 0, 0.4127)$ if $y \in [-0.05, 0.05]$ and $(\rho, u, v, p) = (0.5, 0, 0, 0.4127)$ otherwise. The numerical result for the density, pressure, and temperature with 640×320 grid points at time $t = 0.001$ are shown in Fig. 9. The scales are logarithmic. From the figures, we can see that our results match well with those in [6].

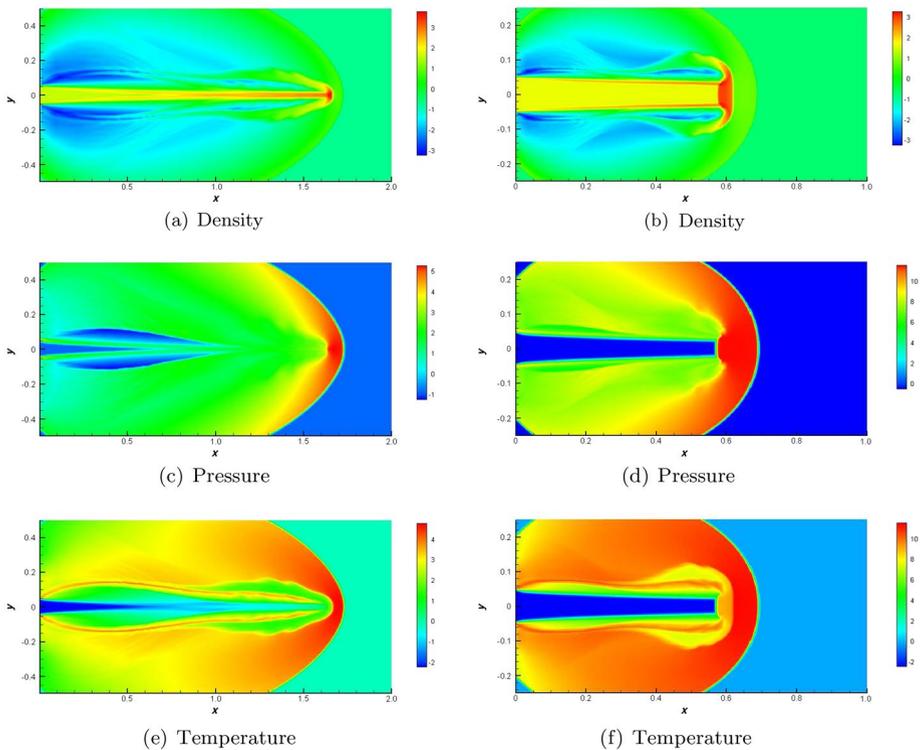


Fig. 9 High Mach number astrophysical jet. Left: the density, pressure, and temperature for Mach 80 at $t = 0.07$ with mesh 448×224 . Right: the density, pressure, and temperature for Mach 2 000 at $t = 0.001$ with mesh 640×320 . Scales are logarithmic

5 Conclusion

In this paper, we propose the dimension-by-dimension HWENO method to solve hyperbolic conservation laws. It is the generalization of the 1D HWENO method, which leads to an easier 2D simulation. Extensive numerical tests are performed to verify the high resolution and high accuracy of the scheme.

Acknowledgements The authors would like to thank Ph.D. candidate Jiayin Li in Xiamen University and anonymous referees for their comments on this paper.

Compliance with Ethical Standards

Conflict of Interest Jianxian Qiu is an editorial board member for *Communications on Applied Mathematics and Computation* and was not involved in the editorial review or the decision to publish this article. On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

1. Balsara, D.S., Altmann, C., Munz, C.-D., Dumbser, M.: A sub-cell based indicator for troubled zones in RKDG schemes and a novel class of hybrid RKDG plus HWENO schemes. *J. Comput. Phys.* **226**(1), 586–620 (2007)
2. Capdeville, G.: A Hermite upwind WENO scheme for solving hyperbolic conservation laws. *J. Comput. Phys.* **227**(4), 2430–2454 (2008)
3. Casper, J., Atkins, H.L.: A finite-volume high-order ENO scheme for two-dimensional hyperbolic systems. *J. Comput. Phys.* **106**(1), 62–76 (1993)
4. Cockburn, B., Shu, C.-W.: The Runge-Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems. *J. Comput. Phys.* **141**(2), 199–224 (1998)
5. Dumbser, M., Balsara, D.S., Toro, E.F., Munz, C.-D.: A unified framework for the construction of one-step finite volume and discontinuous Galerkin schemes on unstructured meshes. *J. Comput. Phys.* **227**(18), 8209–8253 (2008)
6. Ha, Y., Gardner, C.L., Gelb, A., Shu, C.-W.: Numerical simulation of high Mach number astrophysical jets with radiative cooling. *J. Sci. Comput.* **24**(1), 597–612 (2005)
7. Hu, C.Q., Shu, C.-W.: Weighted essentially non-oscillatory schemes on triangular meshes. *J. Comput. Phys.* **150**(1), 97–127 (1999)
8. Hu, X.Y., Khoo, B.C.: An interface interaction method for compressible multifluids. *J. Comput. Phys.* **198**(1), 35–64 (2004)
9. Jiang, G.-S., Peng, D.: Weighted ENO schemes for Hamilton-Jacobi equations. *SIAM J. Sci. Comput.* **21**(6), 2126–2143 (2000)
10. Jiang, G.-S., Shu, C.-W.: Efficient implementation of weighted ENO schemes. *J. Comput. Phys.* **126**(1), 202–228 (1995)
11. Krivodonova, L., Xin, J., Remacle, J.F., Chevaugneon, N., Flaherty, J.E.: Shock detection and limiting with discontinuous Galerkin methods for hyperbolic conservation laws. *Appl. Numer. Math.* **48**(3), 323–338 (2004)
12. Lax, P.D., Liu, X.-D.: Solution of two-dimensional Riemann problems of gas dynamics by positive schemes. *SIAM J. Sci. Comput.* **19**(2), 319–340 (1998)
13. Liu, H., Qiu, J.: Finite difference Hermite WENO schemes for conservation laws, II: an alternative approach. *J. Sci. Comput.* **66**(2), 598–624 (2016)
14. Liu, X.-D., Osher, S., Chan, T.: Weighted essentially non-oscillatory schemes. *J. Comput. Phys.* **115**(1), 200–212 (1994)
15. Liu, Y., Zhang, Y.-T.: A robust reconstruction for unstructured WENO schemes. *J. Sci. Comput.* **54**(2/3), 603–621 (2013)
16. Luo, D., Huang, W., Qiu, J.: A hybrid LDG-HWENO scheme for KdV-type equations. *J. Comput. Phys.* **313**, 754–774 (2016)
17. Qiu, J., Shu, C.-W.: Hermite WENO schemes and their application as limiters for Runge-Kutta discontinuous Galerkin method: one-dimensional case. *J. Comput. Phys.* **193**(1), 115–135 (2004)

18. Qiu, J., Shu, C.-W.: Hermite WENO schemes and their application as limiters for Runge-Kutta discontinuous Galerkin method II: two-dimensional case. *Comput. Fluids* **34**(6), 642–663 (2005)
19. Shi, J., Hu, C., Shu, C.-W.: A technique of treating negative weights in WENO schemes. *J. Comput. Phys.* **175**(1), 108–127 (2002)
20. Shu, C.-W.: Essentially non-oscillatory and weighted essentially non-oscillatory schemes. *Acta Numer.* **29**, 701–762 (2020)
21. Shu, C.-W., Osher, S.: Efficient implementation of essentially non-oscillatory shock-capturing schemes. *J. Comput. Phys.* **77**(2), 439–471 (1988)
22. Tao, Z., Li, F., Qiu, J.: High-order central Hermite WENO schemes: dimension-by-dimension moment-based reconstructions. *J. Comput. Phys.* **318**, 222–251 (2016)
23. Titarev, V.A., Toro, E.F.: Finite-volume WENO schemes for three-dimensional conservation laws. *J. Comput. Phys.* **201**(1), 238–260 (2004)
24. Wang, C., Shu, C.-W., Han, W., Ning, J.: High resolution WENO simulation of 3D detonation waves. *Combust. Flame* **160**(2), 447–462 (2013)
25. Woodward, P., Colella, P.: The numerical simulation of two-dimensional fluid flow with strong shocks. *J. Comput. Phys.* **54**(1), 115–173 (1984)
26. Wu, K., Tang, H.: High-order accurate physical-constraints-preserving finite difference WENO schemes for special relativistic hydrodynamics. *J. Comput. Phys.* **298**, 539–564 (2015)
27. Wu, K., Yang, Z., Tang, H.: A third-order accurate direct Eulerian GRP scheme for the Euler equations in gas dynamics. *J. Comput. Phys.* **264**(1), 177–208 (2014)
28. Xing, Y.L., Shu, C.-W.: High order finite difference WENO schemes with the exact conservation property for the shallow water equations. *J. Comput. Phys.* **208**(1), 206–227 (2005)
29. Xiong, T., Zhang, M., Zhang, Y.-T., Shu, C.-W.: Fast sweeping fifth order WENO scheme for static Hamilton-Jacobi equations with accurate boundary treatment. *J. Sci. Comput.* **45**(1/2/3), 514–536 (2010)
30. Zhang, Y.-T., Shu, C.-W.: Third order WENO scheme on three dimensional tetrahedral meshes. *Communicat. Comput. Phys.* **5**(2/3/4), 836–848 (2009)
31. Zhao, Z., Qiu, J.: A Hermite WENO scheme with artificial linear weights for hyperbolic conservation laws. *J. Comput. Phys.* **417**, 109583 (2020)
32. Zheng, F., Qiu, J.: Directly solving the Hamilton-Jacobi equations by Hermite WENO schemes. *J. Comput. Phys.* **307**, 423–445 (2016)
33. Zhu, J., Qiu, J.: A class of the fourth order finite volume Hermite weighted essentially non-oscillatory schemes. *Sci. China Ser. A* **51**(08), 1549–1560 (2008)
34. Zhu, J., Qiu, J.: A new fifth order finite difference WENO scheme for solving hyperbolic conservation laws. *J. Comput. Phys.* **318**, 110–121 (2016)
35. Zhu, J., Qiu, J.: A new type of finite volume WENO schemes for hyperbolic conservation laws. *J. Sci. Comput.* **73**(2/3), 1338–1359 (2017)

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.