Journal of Scientific Computing (2024) 99:40
https://doi.org/10.1007/s10915-024-02499-0

High-Order Multi-resolution Central Hermite WENO Schemes for Hyperbolic Conservation Laws

Zhanjing Tao[1] · Jinming Zhang[1] · Jun Zhu[2] · Jianxian Qiu[3]

Received: 10 January 2023 / Revised: 10 January 2024 / Accepted: 24 February 2024

Abstract

In this paper, a class of high-order multi-resolution central Hermite WENO (C-HWENO) schemes for solving hyperbolic conservation laws is proposed. Formulated in a central finite volume framework on staggered meshes, the methods adopt the multi-resolution HWENO reconstructions (Li et al. in J Comput Phys 446:110653, 2021; Li et al. in Commun Comput Phys 32(2): 364-400, 2022) in space and the natural continuous extension of Runge–Kutta methods in time. Based on the zeroth-order and first-order moments of the solution defined on a series of hierarchical central spatial stencils, the proposed methods are sixth-order while the C-HWENO methods by Tao et al. (J Comput Phys 318:222-251, 2016) are fifth-order in accuracy. The linear weights of such HWENO reconstructions can be any positive numbers as long as their sum equals one, which leads to much simpler implementation and better cost efficiency than the methods by Tao et al. (J Comput Phys 318:222-251, 2016). The first-order moments are modified and the HWENO reconstructions are applied in the troubled-cells, while the linear reconstructions are used for the rest. Meanwhile, our new methods have compact stencils in the reconstructions and require neither numerical fluxes nor flux splitting. Extensive one- and two-dimensional numerical examples are performed to illustrate the accuracy and high resolution of the new C-HWENO schemes.

✉ Zhanjing Tao
zjtao@jlu.edu.cn

Jinming Zhang
jmzhang21@mails.jlu.edu.cn

Jun Zhu
zhujun@nuaa.edu.cn

Jianxian Qiu
jxqiu@xmu.edu.cn

[1] School of Mathematics, Jilin University, Changchun 130012, Jilin, People's Republic of China

[2] State Key Laboratory of Mechanics and Control for Aerospace Structures, Key Laboratory of Mathematical Modelling and High Performance Computing of Air Vehicles (NUAA), MIIT, Nanjing 210016, Jiangsu, People's Republic of China

[3] School of Mathematical Sciences and Fujian Provincial Key Laboratory of Mathematical Modeling and High-Performance Scientific Computing, Xiamen University, Xiamen 361005, Fujian, People's Republic of China

Keywords Finite volume method · Central scheme · Multi-resolution hermite WENO · Natural continuous extension (NCE) of Runge–Kutta

1 Introduction

In this paper, a class of high-order finite volume multi-resolution central Hermite weighted essentially non-oscillatory (C-HWENO) schemes is designed for solving one- and two-dimensional hyperbolic conservation laws

$$\begin{cases} u_t + \nabla \cdot f(u) = 0, \\ u(x,0) = u_0(x), \end{cases} \tag{1.1}$$

with suitable initial and boundary conditions. Hyperbolic conservation laws arise in a wide range of applications in engineering and science, such as astrophysical modeling, aerodynamics, explosion and blast waves, weather prediction, and multi-phase flow problems. Here (1.1) can be scalar or a system, and it is often nonlinear. Shocks, compound waves, etc., may appear in the solutions of the nonlinear equations regardless of the smoothness of the initial and boundary conditions. When (1.1) is a system, the local characteristic decomposition is applied to control the spurious oscillations.

Finite difference and finite volume WENO (weighted essentially non-oscillatory) schemes have been successfully applied to the simulation of hyperbolic conservation laws with excellent property of the high order accuracy in the smooth region and the non-oscillatory shock transitions near the discontinuities. To improve the accuracy in the smooth region without destroying the non-oscillatory property of the ENO (essentially non-oscillatory) schemes, the first WENO scheme [16] was proposed by Liu, Osher and Chan as a third-order finite volume method for one space dimension. Since then, the methods have been further developed in the design of smoothness indicator, nonlinear weights, higher dimensions, unstructured meshes and better accuracy [1, 6, 7, 11, 24, 25, 34]. Note that higher order accuracy in the WENO scheme relies on enlarging the stencil of the reconstruction. To improve the compactness while maintaining the high order accuracy, Hermite WENO (HWENO) methods were further developed in a finite volume or finite difference framwork [15, 21, 23, 29, 32, 33]. Here, the solution as well as its first-order derivative(s) or moment(s) evolve over time and are used in the Hermite type reconstruction in space.

WENO methods were originally used within the upwind or Godunov framework. Compared with the upwind scheme, the central scheme is an efficient alternative for hyperbolic conservation laws, and it is relatively simpler. The central scheme requires neither numerical flux, that are exact or approximate Riemann solver, nor the flux splitting, and the local characteristic decomposition is not necessary for the low order central schemes. In 1990, Nessyahu and Tadmor first proposed a second-order central scheme [19]. Later, various semi-discrete as well as multi-dimensional versions of the central schemes were developed in [2, 8, 10, 17, 18]. WENO or HWENO methods were also integrated into the central framework [5, 11, 12, 20, 26, 27] to improve the accuracy and resolution. Many central schemes use the natural continuous extension of Runge–Kutta method [30] as time discretization. In addition, the Lax-Wendroff type time discretization was also applied in the C-HWENO methods [26, 27].

The work in this paper is a continuation of the C-HWENO scheme in [27], and they have similar advantages such as the compactness in the reconstruction. Based on the multi-resolution HWENO reconstructions [13, 14], our new schemes are sixth-order while the schemes in [27] are fifth-order in accuracy, and they have better resolution for some examples than the previous schemes. The truly two-dimensional reconstructions are used such that we do not need the information of the mixed-type first-order moment $\overline{vw}_{ij}^{n}$ which is involved in the two-dimensional reconstructions in [27] with a dimension-by-dimension procedure. Furthermore, the same reconstructed polynomials and nonlinear weights are used in our new reconstructions for different quantities, and this leads to much simpler implementation. The new multi-resolution reconstructions were first developed for the WENO schemes [34] where a series of unequal-sized hierarchical central spatial stencils were used in the reconstructions and the linear weights of such schemes can be any positive numbers on the condition that their sum equals one. Following the idea in [31, 32], we also modify the first-order moment(s) when the target cell is identified as a troubled-cell to further improve the stability. Besides, the HWENO reconstructions are only applied in the troubled-cells and the linear reconstructions are used for the rest to save the cost of our schemes.

The structure of this paper is as follows: in Sect. 2, we describe in detail the implementation of the new C-HWENO schemes with the multi-resolution HWENO spatial reconstructions and the natural continuous extension of the Runge–Kutta method as time discretization for one- and two-dimensional hyperbolic conservation laws. In Sect. 3, extensive benchmark examples are presented to verify the accuracy and good performance of the proposed schemes. Concluding remarks are made in Sect. 4.

2 Multi-resolution C-HWENO Schemes with Natural Continuous Extension of Runge–Kutta Time Discretization

In this section, we introduce the finite volume central schemes with sixth-order multi-resolution HWENO reconstructions and the natural continuous extension of Runge–Kutta time discretization for solving hyperbolic conversation laws in one and two dimensions.

2.1 One-Dimensional Case

Consider the one-dimensional scalar conservation law

$$\begin{cases} u_t + f(u)_x = 0, \\ u(x, 0) = u_0(x). \end{cases} \tag{2.1}$$

Let $\{x_i\}_i$ be a uniform partition of the one-dimensional computational domain with the mesh size Δx. Based on $x_{i+1/2} = \frac{1}{2}(x_i + x_{i+1})$, we denote the primal mesh as $I_i = [x_{i-1/2}, x_{i+1/2}]$ with the center x_i and the dual mesh as $I_{i+1/2} = [x_i, x_{i+1}]$ with the center $x_{i+1/2}$. Our method evolves the numerical solutions in a staggered manner with the two meshes.

The first two moments of the solution on primal mesh I_i are denoted as $\{\overline{u}_i, \overline{v}_i\}$, that is

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

The moments of the solution on dual mesh $I_{i+1/2}$ are denoted as $\{\overline{u}_{i+1/2}, \overline{v}_{i+1/2}\}$, that is

$$\overline{u}_{i+1/2} \approx \frac{1}{\Delta x} \int_{I_{i+1/2}} u(x, t) dx, \quad \overline{v}_{i+1/2} \approx \frac{1}{\Delta x} \int_{I_{i+1/2}} u(x, t) \frac{x - x_{i+1/2}}{\Delta x} dx.$$

Suppose at $t = t^n$, the moments of the solution $\{\overline{u}_i^n, \overline{v}_i^n\}_i$ are available on primal mesh. We multiply (2.1) by $\frac{1}{\Delta x}$ and $\frac{x - x_{i+1/2}}{(\Delta x)^2}$, respectively, integrate over $I_{i+1/2} \times [t^n, t^{n+1}]$, apply integration by parts, and obtain

$$\overline{u}_{i+1/2}^{n+1} = \overline{u}_{i+1/2}^{n} - \frac{1}{\Delta x}\int_{t^n}^{t^{n+1}} [f(u(x_{i+1},t)) - f(u(x_i,t))]dt, \tag{2.2}$$

$$\overline{v}_{i+1/2}^{n+1} = \overline{v}_{i+1/2}^{n} - \frac{1}{\Delta x}\int_{t^n}^{t^{n+1}} \left[\frac{1}{2}[f(u(x_{i+1},t)) + f(u(x_i,t))] - \frac{1}{\Delta x}\int_{x_i}^{x_{i+1}} f(u(x,t))dx\right]dt. \tag{2.3}$$

Since the sixth-order HWENO spatial reconstruction will be adopted, we would apply the four-point Gauss-Lobatto quadrature to compute the spatial integral

$$\frac{1}{\Delta x}\int_{x_i}^{x_{i+1}} f(u(x,t))dx \approx \sum_{l=1}^{4} \omega_l f(u(x_l^G,t)). \tag{2.4}$$

where $x_1^G = x_i$, $x_2^G = x_{i+\frac{1}{2}-\frac{\sqrt{5}}{10}}$, $x_3^G = x_{i+\frac{1}{2}+\frac{\sqrt{5}}{10}}$ and $x_4^G = x_{i+1}$ are the quadrature points over the cell $I_{i+1/2}$, and $\omega_1 = \omega_4 = \frac{1}{12}$ and $\omega_2 = \omega_3 = \frac{5}{12}$ are the quadrature weights. This implies that we need the quadrature points

$$x^{G,i} = \{x_{i-\frac{1}{2}+\frac{\sqrt{5}}{10}}, x_i, x_{i+\frac{1}{2}-\frac{\sqrt{5}}{10}}\}. \tag{2.5}$$

in each cell I_i of primal mesh, and we define $x^G = \cup_i x^{G,i}$.

Let's consider the problem (2.1) with $u|_{I_i}$ as a constant at t^n. Suppose we choose a suitable time step $\Delta t \le \frac{C_{cfl}\Delta x}{\max|f'(u)|}$, the discontinuities starting from $x_{i-1/2}$ and $x_{i+1/2}$ at t^n will not affect the solutions at $x_*(x_* \in x^G)$, and therefore $u(x_*,t), t \in (t^n, t^{n+1})$ are smooth. Motivated by this, the three-point Gauss quadrature is used to approximate the temporal integrals in (2.2) and (2.3) as below

$$\int_{t^n}^{t^{n+1}} f(u(x,t))dt \approx \Delta t \sum_{m=1}^{3} \alpha_m f(u(x, t^n + \theta_m \Delta t)), \tag{2.6}$$

where $\theta_1 = \frac{1}{2} - \frac{\sqrt{15}}{10}$, $\theta_2 = \frac{1}{2}$, $\theta_3 = \frac{1}{2} + \frac{\sqrt{15}}{10}$ are the quadrature points, and $\alpha_1 = \alpha_3 = \frac{5}{18}$, $\alpha_2 = \frac{4}{9}$ are the quadrature weights.

Based on (2.2)–(2.6), the proposed central scheme is given as follows.

(a.1) With $\{\overline{u}_i^n, \overline{v}_i^n\}_i$ being available at $t = t^n$ on primal mesh, the first two moments $\{\overline{u}_{i+1/2}^{n+1}, \overline{v}_{i+1/2}^{n+1}\}_i$ of the solution at $t = t^{n+1}$ on dual mesh can be computed as follows,

$$\overline{u}_{i+1/2}^{n+1} = \overline{u}_{i+1/2}^{n} - \frac{\Delta t}{\Delta x}\sum_{m=1}^{3} \alpha_m \left[f(u(x_{i+1},t)) - f(u(x_i,t))\right]\big|_{t=t^n+\theta_m\Delta t}, \tag{2.7}$$

$$\overline{v}_{i+1/2}^{n+1} = \overline{v}_{i+1/2}^{n} - \frac{\Delta t}{\Delta x}\sum_{m=1}^{3} \alpha_m \left[\frac{f(u(x_{i+1},t)) + f(u(x_i,t))}{2} - \sum_{l=1}^{4}\omega_l f(x_l^G,t))\right]\Big|_{t=t^n+\theta_m\Delta t}. \tag{2.8}$$

(a.2) With $\{\overline{u}_{i+1/2}^{n+1}, \overline{v}_{i+1/2}^{n+1}\}_i$ being available at $t = t^{n+1}$ on dual mesh, the first two moments $\{\overline{u}_i^{n+2}, \overline{v}_i^{n+2}\}_i$ at $t = t^{n+2}$ on primal mesh can be computed as follows,

$$\overline{u}_i^{n+2} = \overline{u}_i^{n+1} - \frac{\Delta t}{\Delta x}\sum_{m=1}^{3} \alpha_m \left[f(u(x_{i+1/2},t)) - f(u(x_{i-1/2},t))\right]\big|_{t=t^{n+1}+\theta_m\Delta t},$$

$$\overline{v}_i^{n+2} = \overline{v}_i^{n+1} - \frac{\Delta t}{\Delta x}\sum_{m=1}^{3}\alpha_m\left[\frac{f(u(x_{i+1/2},t)) + f(u(x_{i-1/2},t))}{2} - \sum_{l=1}^{4}\omega_l f(u(\hat{x}_l^G,t))\right]\Big|_{t=t^{n+1}+\theta_m\Delta t},$$

where $\hat{x}_1^G = x_{i-\frac{1}{2}}, \hat{x}_2^G = x_{i-\frac{\sqrt{5}}{10}}, \hat{x}_3^G = x_{i+\frac{\sqrt{5}}{10}}$ and $\hat{x}_4^G = x_{i+\frac{1}{2}}$ are the quadrature points over the cell I_i.

(a.3) Set n to be $n+2$, and return to (a.1).

In order to finalize the proposed scheme, certain quantities in (a.1) and (a.2) need to be reconstructed. Because of its similarity, we only describe (a.1) in detail. To calculate the quantities $\{\overline{u}_{i+1/2}^{n+1}, \overline{v}_{i+1/2}^{n+1}\}_i$, according to (2.7)–(2.8), one will need to obtain accurate approximations of the following quantities

$$\overline{u}_{i+1/2}^n,\ \overline{v}_{i+1/2}^n,\ u(x_*, t^n + \theta_m\Delta t),\ \forall i,\ m = 1,2,3,\ x_* \in x^G, \tag{2.9}$$

with the given data $\{\overline{u}_i^n, \overline{v}_i^n\}_i$. Notice that one would want to get the half-cell averages $\frac{1}{\Delta x}\int_{x_i}^{x_{i+1/2}} u(x,t^n)dx$ and $\frac{1}{\Delta x}\int_{x_{i+1/2}}^{x_{i+1}} u(x,t^n)dx$ to approximate $\overline{u}_{i+1/2}^n$

$$\overline{u}_{i+1/2}^n \approx \frac{1}{\Delta x}\int_{x_i}^{x_{i+1}} u(x,t^n)dx = \frac{1}{\Delta x}\int_{x_i}^{x_{i+1/2}} u(x,t^n)dx + \frac{1}{\Delta x}\int_{x_{i+1/2}}^{x_{i+1}} u(x,t^n)dx.$$

The first-order moment $\overline{v}_{i+1/2}^n$ can be implemented in a similar way

$$\overline{v}_{i+1/2}^n \approx \frac{1}{\Delta x}\int_{x_i}^{x_{i+1/2}} u(x,t^n)\frac{x - x_{i+1/2}}{\Delta x}dx + \frac{1}{\Delta x}\int_{x_{i+1/2}}^{x_{i+1}} u(x,t^n)\frac{x - x_{i+1/2}}{\Delta x}dx.$$

To approximate the point values $u(x_*, t^n + \theta_m\Delta t)(m = 1,2,3, x_* \in x^G)$ in (2.9), we apply the fourth-order natural continuous extension of Runge–Kutta (NCE-RK) time discretization. For more details on this discretization, see [2, 30]. Consider a scalar or a system of ODE problem

$$\begin{cases} y'(t) = H(y(t)), \\ y(t_0) = y_0. \end{cases} \tag{2.10}$$

We start with y^n which is an approximation of $y(t^n)$, and the approximation for $y(t^{n+1})$ can be obtained by the RK method with ν stages

$$y^{n+1} = y^n + \Delta t\sum_{i=1}^{\nu} b_i K^{(i)}, \tag{2.11}$$

where $K^{(i)}$ are RK fluxes defined as

$$K^{(i)} = H(Y^{(i)}), \quad i = 1, \cdots, \nu.$$

Here the intermediate values $Y^{(i)}$ can be computed as below

$$Y^{(i)} = y^n + \Delta t\sum_{j=1}^{\nu} a_{ij}K^{(j)}, \quad i = 1, \cdots, \nu.$$

The vector b and the matrix $A = (a_{ij})$ characterize the RK method. In this paper, we use the explicit fourth-order RK method with

$$A = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1/2 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \quad b = \begin{pmatrix} 1/6 \\ 1/3 \\ 1/3 \\ 1/6 \end{pmatrix}.$$

A natural continuous extension of the RK method (2.10)–(2.11) can further get the approximation of $y(t)$ with the same accuracy when $t \in [t^n, t^{n+1}]$ by

$$w(t)|_{t=t^n+\theta\Delta t} = y^n + \Delta t \sum_{i=1}^{4} B_i(\theta) K^{(i)}, \quad 0 \le \theta \le 1, \tag{2.12}$$

where

$$B_1(\theta) = 2(1-4b_1)\theta^3 + 3(3b_1 - 1)\theta^2 + \theta,$$
$$B_i(\theta) = 4(3c_i - 2)b_i\theta^3 + 3(3-4c_i)b_i\theta^2, \quad i = 2, 3, 4.$$

Here $c_1 = 0, c_2 = c_3 = \frac{1}{2}, c_4 = 1$, and $w(t)$ satisfies the following property

$$\max_{t^n \le t \le t^n+\Delta t} |y(t) - w(t)| = O(\Delta t^4).$$

We apply the fourth-order NCE-RK method (2.12) to assist an auxiliary ODE problem (2.10) starting from $t = t^n$. Here $y(t)$ in (2.10) is the point value $u(x_*, t)$, and $H(y(t))$ is the corresponding $-f_x(u(x_*, t))$. Two ingredients are needed to approximate the point values $u(x_*, t^n + \theta_m \Delta t)$. One part is the calculation of $u(x_*, t^n)$, which can be obtained by multi-resolution HWENO reconstruction based on $\{\overline{u}_i^n, \overline{v}_i^n\}_i$ in step 4. The other is the calculation of $H(y(t)) = -f_x(u(x_*, t))$, which can be calculated using a high-order linear interpolation based on $f(u(x_*, t))$, $x_* \in x^G$ in step 5.

The sixth-order multi-resolution HWENO reconstruction of Li, Shu and Qiu [13, 14] is applied to compute the quantities mentioned above. The multi-resolution HWENO reconstruction has many advantages. First, the linear weights of such reconstruction do not need to be computed like the classical HWENO reconstruction, but can be taken as any positive numbers to avoid the appearance of negative weights which need splitting strategy. Second, this HWENO format uses the same large template as the classical HWENO format and has higher order of accuracy. Since the data is at the same time level during the reconstruction, the superscript n and the dependence on time t will be omitted below.

Step 1 Identify the troubled-cells and modify the first-order moments.

Step 1.1 The troubled-cells indicate that the solutions in those cells may contain discontinuities. In [22], Qiu and Shu compared different troubled-cell indicators for the Runge–Kutta discontinuous Galerkin method, and the KXRCF troubled-cell indicator [9] is used in this paper. First, we divide the boundary of the cell I_i into two parts: inflow boundary ∂I_i^- ($\overrightarrow{v} \cdot \overrightarrow{n} < 0$, where $\overrightarrow{v}$ is the velocity of flow, $\overrightarrow{n}$ is the outer normal vector to ∂I_i) and outflow boundary ∂I_i^+ ($\overrightarrow{v} \cdot \overrightarrow{n} > 0$). The target cell I_i is identified as a troubled-cell if the following criterion is satisfied

$$\frac{\left| \int_{\partial I_i^-} \left(u_i(x) - u_{n_i}(x)\right) ds \right|}{\Delta x^{\frac{l+1}{3}} \left|\partial I_i^-\right| \|u_i(x)\|} > 1 \tag{2.13}$$

where I_{n_i} is the neighbor of I_i on the side of ∂I_i^-. The parameter $l = 5$ is the degree of $u_i(x)$ which is an approximation of $u(x)$, and $u_i(x)$ can be obtained by a sixth-order linear reconstruction

$$\begin{cases} \frac{1}{\Delta x} \int_{I_k} u_i(x) dx = \overline{u}_k, & k = i-1, i, i+1, \\ \frac{1}{\Delta x} \int_{I_{k_x}} u_i(x) \frac{x - x_{k_x}}{\Delta x} dx = \overline{v}_{k_x}, & k_x = i-1, i, i+1. \end{cases} \tag{2.14}$$

$\|u_i(x)\|$ is taken to be L^∞ norm

$$\|u_i(x)\| \approx \max\{|u^+_{i-\frac{1}{2}}|, |u_{i-\frac{1}{2}+\frac{\sqrt{5}}{10}}|, |u_i|, |u_{i+\frac{1}{2}-\frac{\sqrt{5}}{10}}|, |u^-_{i+\frac{1}{2}}|\}$$

where

$$u^+_{i-\frac{1}{2}} = u_i(x_{i-\frac{1}{2}}), u_{i-\frac{1}{2}+\frac{\sqrt{5}}{10}} = u_i(x_{i-\frac{1}{2}+\frac{\sqrt{5}}{10}}), u_i = u_i(x_i),$$
$$u_{i+\frac{1}{2}-\frac{\sqrt{5}}{10}} = u_i(x_{i+\frac{1}{2}-\frac{\sqrt{5}}{10}}), u^-_{i+\frac{1}{2}} = u_i(x_{i+\frac{1}{2}}).$$

The average of line integral in the formula (2.13) can be expressed explicitly

$$\frac{1}{|\partial I_i^-|}\left|\int_{\partial I_i^-}\big(u_i(x) - u_{n_i}(x)\big)\,ds\right|$$
$$= \left|(u^+_{i-1/2} - u^-_{i-1/2}) * s(\overrightarrow{v}_{i-\frac{1}{2}}) + (u^-_{i+1/2} - u^+_{i+1/2}) * s(-\overrightarrow{v}_{i+\frac{1}{2}})\right|$$

where

$$s(x) = \begin{cases} 1, & x > 0, \\ 0, & else. \end{cases} \tag{2.15}$$

For the one-dimensional scalar case, we take $\overrightarrow{v} = f'(u)$. For the one-dimensional Euler equations, we set the density ρ as the indicator variable, and $\overrightarrow{v} = \mu$ is the velocity of the fluid.

Step 1.2 If the target cell I_i is identified to be a troubled-cell, one would like to identify I_{i-1} and I_{i+1} as troubled-cells. Then, we need to modify the first-order moments of all troubled-cells. Take the first-order moment $\overline{v}_i$ as an example. One could reconstruct two quadratic polynomials $p_1(x)$, $p_2(x)$ and a quartic polynomial $p_3(x)$, which satisfy

$$\begin{cases} \frac{1}{\Delta x}\int_{I_k} p_1(x)dx = \overline{u}_k, \quad k = i-1, i, \\ \frac{1}{\Delta x}\int_{I_{i-1}} p_1(x)\frac{x-x_{i-1}}{\Delta x}dx = \overline{v}_{i-1}, \end{cases} \tag{2.16}$$

$$\begin{cases} \frac{1}{\Delta x}\int_{I_k} p_2(x)dx = \overline{u}_k, \quad k = i, i+1, \\ \frac{1}{\Delta x}\int_{I_{i+1}} p_2(x)\frac{x-x_{i+1}}{\Delta x}dx = \overline{v}_{i+1}, \end{cases} \tag{2.17}$$

$$\begin{cases} \frac{1}{\Delta x}\int_{I_k} p_3(x)dx = \overline{u}_k, \quad k = i-1, i, i+1, \\ \frac{1}{\Delta x}\int_{I_{k_x}} p_3(x)\frac{x-x_{k_x}}{\Delta x}dx = \overline{v}_{k_x}, \quad k_x = i-1, i+1. \end{cases} \tag{2.18}$$

We can compute the first-order moment of polynomials $p_l(x), l = 1, 2, 3$ over I_i, and the results can be given explicitly,

$$\begin{cases} \frac{1}{\Delta x}\int_{I_i} p_1(x)\frac{x-x_i}{\Delta x}dx = -\frac{1}{6}\overline{u}_{i-1} + \frac{1}{6}\overline{u}_i - \overline{v}_{i-1}, \\ \frac{1}{\Delta x}\int_{I_i} p_2(x)\frac{x-x_i}{\Delta x}dx = -\frac{1}{6}\overline{u}_i + \frac{1}{6}\overline{u}_{i+1} - \overline{v}_{i+1}, \\ \frac{1}{\Delta x}\int_{I_i} p_3(x)\frac{x-x_i}{\Delta x}dx = -\frac{5}{76}\overline{u}_{i-1} + \frac{5}{76}\overline{u}_{i+1} - \frac{11}{38}\overline{v}_{i-1} - \frac{11}{38}\overline{v}_{i+1}. \end{cases}$$

To compute the nonlinear weights, the smoothness indicators of polynomials $p_k(x)$ can be calculated as below

$$\beta_k = \sum_{\alpha=1}^{r}\int_{I_i}\Delta x^{2\alpha-1}\left(\frac{d^\alpha p_k(x)}{dx^\alpha}\right)^2 dx, \quad k = 1, 2, 3, \tag{2.19}$$

where r is the degree of the polynomials $p_k(x)$. Then we adopt the idea of WENO-Z [3, 4] and define the quantity τ as the absolute difference between smoothness indicators,

$$\tau = \left(\frac{\sum_{k=1}^{2} |\beta_3 - \beta_k|}{2} \right)^2 .$$

The nonlinear weights can be computed

$$\omega_k = \frac{\overline{\omega}_k}{\sum_{l=1}^{3} \overline{\omega}_l}, \quad \overline{\omega}_k = \gamma_k \left(1 + \frac{\tau}{\beta_k + \varepsilon} \right), \quad k = 1, 2, 3, \tag{2.20}$$

here $\varepsilon = 10^{-6}$ and $\gamma_1 = \gamma_2 = 0.002, \gamma_3 = 0.996$ are linear weights which can be any positive numbers provided that their sum equals one.

Then the first-order moment is modified as

$$\begin{aligned} \overline{v}_i^{mod} \approx \frac{\omega_3}{\gamma_3} \left(\frac{1}{\Delta x} \int_{I_i} p_3(x) \frac{x - x_i}{\Delta x} dx - \sum_{k=1}^{2} \gamma_k \frac{1}{\Delta x} \int_{I_i} p_k(x) \frac{x - x_i}{\Delta x} dx \right) \\ + \sum_{k=1}^{2} \omega_k \frac{1}{\Delta x} \int_{I_i} p_k(x) \frac{x - x_i}{\Delta x} dx. \end{aligned}$$

If I_i is not a troubled-cell, we simply set $\overline{v}_i^{mod} = \overline{v}_i$.

Step 2 A reconstruction of $\frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_i} u(x)dx$ and $\frac{1}{\Delta x} \int_{x_i}^{x_{i+1/2}} u(x)dx$ from $\{\overline{u}_{i-1}, \overline{u}_i, \overline{u}_{i+1}, \overline{v}_{i-1}, \overline{v}_i^{mod}, \overline{v}_{i+1}\}$.

If the target cell I_i is identified to be a troubled-cell, we use the following multi-resolution HWENO reconstruction. Based on a series of central spatial stencils, we reconstruct a zeroth degree polynomial $q_1(x)$, a quadratic polynomial $q_2(x)$, a cubic polynomial $q_3(x)$, and a quintic polynomial $q_4(x)$, respectively, and these polynomials satisfy the following conditions

$$\frac{1}{\Delta x} \int_{I_i} q_1(x)dx = \overline{u}_i, \tag{2.21}$$

$$\frac{1}{\Delta x} \int_{I_k} q_2(x)dx = \overline{u}_k, \quad k = i-1, i, i+1, \tag{2.22}$$

$$\begin{cases} \frac{1}{\Delta x} \int_{I_k} q_3(x)dx = \overline{u}_k, \quad k = i-1, i, i+1, \\ \frac{1}{\Delta x} \int_{I_i} q_3(x) \frac{x-x_i}{\Delta x} dx = \overline{v}_i^{mod}, \end{cases} \tag{2.23}$$

$$\begin{cases} \frac{1}{\Delta x} \int_{I_k} q_4(x)dx = \overline{u}_k, \quad k = i-1, i, i+1, \\ \frac{1}{\Delta x} \int_{I_i} q_4(x) \frac{x-x_i}{\Delta x} dx = \overline{v}_i^{mod}, \\ \frac{1}{\Delta x} \int_{I_{k_x}} q_4(x) \frac{x-x_{k_x}}{\Delta x} dx = \overline{v}_{k_x}, \quad k_x = i-1, i+1. \end{cases} \tag{2.24}$$

Then, we express the polynomials $q_l(x), l = 1, 2, 3, 4$ as the linear combination of polynomials $p_k(x), k = 1, 2, 3, 4$ as below

$$\begin{aligned} q_1(x) &= p_1(x), \\ q_2(x) &= \gamma_{1,2} p_1(x) + \gamma_{2,2} p_2(x), \\ q_3(x) &= \gamma_{1,3} p_1(x) + \gamma_{2,3} p_2(x) + \gamma_{3,3} p_3(x), \\ q_4(x) &= \gamma_{1,4} p_1(x) + \gamma_{2,4} p_2(x) + \gamma_{3,4} p_3(x) + \gamma_{4,4} p_4(x), \end{aligned} \tag{2.25}$$

where $\sum_{k=1}^{l} \gamma_{k,l} = 1$, $\gamma_{l,l} \neq 0$, $l = 2, 3, 4$, and $\gamma_{k,l}$ are the linear weights defined as

$$\gamma_{k,l} = \frac{\overline{\gamma}_{k,l}}{\sum_{l_1=1}^{l} \overline{\gamma}_{l_1,l}}, \quad \overline{\gamma}_{k,l} = 10^{k-1}, \quad k = 1, \cdots, l;\ l = 2, 3, 4. \tag{2.26}$$

The choice of $\overline{\gamma}_{k,l}$ is not unique. However, numerical experiments show that the shock jump near the discontinuity will be sharper if the higher degree polynomial is given a larger linear weight. According to (2.26), we have

$$\overline{\gamma}_{1,2} = \overline{\gamma}_{1,3} = \overline{\gamma}_{1,4} = 1, \quad \overline{\gamma}_{2,2} = \overline{\gamma}_{2,3} = \overline{\gamma}_{2,4} = 10, \quad \overline{\gamma}_{3,3} = \overline{\gamma}_{3,4} = 100, \quad \overline{\gamma}_{4,4} = 1000,$$

and the linear weights are

$$\begin{aligned} &\gamma_{1,2} = \frac{1}{11}, \gamma_{2,2} = \frac{10}{11}, \quad \gamma_{1,3} = \frac{1}{111}, \gamma_{2,3} = \frac{10}{111}, \gamma_{3,3} = \frac{100}{111}, \\ &\gamma_{1,4} = \frac{1}{1111}, \gamma_{2,4} = \frac{10}{1111}, \gamma_{3,4} = \frac{100}{1111}, \gamma_{4,4} = \frac{1000}{1111}. \end{aligned}$$

The equivalent polynomials $p_k(x)$, $k = 1, 2, 3, 4$ can be computed as follows

$$\begin{cases} p_1(x) = q_1(x), \\ p_k(x) = \frac{1}{\gamma_{k,k}} q_k(x) - \sum\limits_{l=1}^{k-1} \frac{\gamma_{l,k}}{\gamma_{k,k}} p_l(x), \quad k = 2, 3, 4. \end{cases} \tag{2.27}$$

To compute the nonlinear weights for (2.25), the smoothness indicators of polynomials $p_k(x)$ can be calculated as below

$$\beta_k = \sum_{\alpha=1}^{r} \int_{I_i} \Delta x^{2\alpha-1} \left(\frac{d^\alpha p_k(x)}{dx^\alpha} \right)^2 dx, \quad k = 2, 3, 4, \tag{2.28}$$

where $r = 2, 3, 5$ for $k = 2, 3, 4$, respectively. The only difference is β_1. If β_1 is defined in the same way, it would be zero. Although this does not affect the precision order of the smooth region, it does cause the shock wave transition near the discontinuities to be somewhat fuzzy, especially when the problem involves strong shocks or contact discontinuities. Therefore we would like to scale up β_1 from zero to a positive value in the following approach. One can take two stencils $\{I_{i-1}, I_i\}$ and $\{I_i, I_{i+1}\}$, and obtain the corresponding smoothness indicators

$$\beta_{1L} = (\overline{u}_i - \overline{u}_{i-1})^2, \quad \beta_{1R} = (\overline{u}_{i+1} - \overline{u}_i)^2.$$

Then, we compute the nonlinear weights of the two stencils

$$\omega_{1m} = \frac{\overline{\omega}_{1m}}{\overline{\omega}_{1L} + \overline{\omega}_{1R}}, \quad \overline{\omega}_{1m} = \gamma_{1m} \left(1 + \frac{\tau_1}{\beta_{1m} + \varepsilon} \right), m = L, R,$$

where

$$\gamma_{1L} = \begin{cases} \frac{1}{11}, \beta_{1L} \geq \beta_{1R}, \\ \frac{10}{11}, \beta_{1L} < \beta_{1R}, \end{cases} \quad \gamma_{1R} = 1 - \gamma_{1L}, \ \tau_1 = |\beta_{1R} - \beta_{1L}|^2,$$

and $\varepsilon = 10^{-6}$. Finally, we define

$$\beta_1 = \left(\omega_{1L}(\overline{u}_i - \overline{u}_{i-1}) + \omega_{1R}(\overline{u}_{i+1} - \overline{u}_i) \right)^2.$$

With the smoothness indicators and the linear weights defined above, one can now compute the nonlinear weights. Here we adopt the idea of WENO-Z [3, 4] and define the quantity τ_4 as the absolute difference between smoothness indicators,

$$\tau_4 = \left(\frac{\sum_{k=1}^{3} |\beta_4 - \beta_k|}{3} \right)^2 .$$

The nonlinear weights can be computed

$$\omega_{k,4} = \frac{\overline{\omega}_{k,4}}{\sum_{l=1}^{4} \overline{\omega}_{l,4}}, \quad \overline{\omega}_{k,4} = \gamma_{k,4} \left(1 + \frac{\tau_4}{\beta_k + \varepsilon} \right), \quad k = 1, 2, 3, 4, \tag{2.29}$$

here $\varepsilon = 10^{-6}$. The new final reconstructed polynomial $p(x)$ is given by

$$p(x) = \sum_{k=1}^{4} \omega_{k,4} p_k(x).$$

To compute $\frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_i} u(x)dx$, we further compute the half-cell averages of polynomials $p_k(x)$, $k = 1, 2, 3, 4$ over $[x_{i-1/2}, x_i]$, and the results can be given explicitly,

$$\begin{cases} \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_i} p_1(x)dx = \frac{1}{2}\overline{u}_i, \\ \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_i} p_2(x)dx = \frac{11}{160}\overline{u}_{i-1} + \frac{1}{2}\overline{u}_i - \frac{11}{160}\overline{u}_{i+1}, \\ \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_i} p_3(x)dx = -\frac{119}{14080}\overline{u}_{i-1} + \frac{1}{2}\overline{u}_i + \frac{119}{14080}\overline{u}_{i+1} - \frac{2997}{1760}\overline{v}_i^{mod}, \\ \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_i} p_4(x)dx = -\frac{1431}{225280}\overline{u}_{i-1} + \frac{1}{2}\overline{u}_i + \frac{1431}{225280}\overline{u}_{i+1} - \frac{1111}{51200}\overline{v}_{i-1} \\ -\frac{453109}{281600}\overline{v}_i^{mod} - \frac{1111}{51200}\overline{v}_{i+1}. \end{cases}$$

Finally, a sixth-order HWENO approximation for $\frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_i} u(x)dx$ is given as

$$\frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_i} u(x)dx \approx \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_i} p(x)dx = \sum_{k=1}^{4} \omega_{k,4} \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_i} p_k(x)dx. \tag{2.30}$$

If the cell I_i is not a troubled-cell, $\frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_i} u(x)dx$ can be approximated by the linear reconstruction

$$\begin{aligned} \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_i} u(x)dx \approx \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_i} q_4(x)dx &= -\frac{3}{512}\overline{u}_{i-1} + \frac{1}{2}\overline{u}_i \\ &+ \frac{3}{512}\overline{u}_{i+1} - \frac{5}{256}\overline{v}_{i-1} - \frac{205}{128}\overline{v}_i^{mod} - \frac{5}{256}\overline{v}_{i+1}. \end{aligned}$$

The right half-cell average of u over $[x_i, x_{i+1/2}]$ can be approximated based on the local conservation of u in the cell I_i,

$$\frac{1}{\Delta x} \int_{x_i}^{x_{i+1/2}} u(x)dx = \overline{u}_i - \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_i} u(x)dx.$$

Step 3 A reconstruction of $\frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_i} u(x)\frac{x - x_{i-1/2}}{\Delta x}dx$ and $\frac{1}{\Delta x} \int_{x_i}^{x_{i+1/2}} u(x)\frac{x - x_{i+1/2}}{\Delta x}dx$ from $\{\overline{u}_{i-1}, \overline{u}_i, \overline{u}_{i+1}, \overline{v}_{i-1}, \overline{v}_i^{mod}, \overline{v}_{i+1}\}$. If the target cell I_i is identified to be a troubled-cell, the same stencils as in step 2 are used, together with the same reconstructed polynomials $q_l(x)$, $p_k(x)$, $l, k = 1, 2, 3, 4$ and the nonlinear weights.

To reconstruct $\frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_i} u(x)\frac{x-x_{i-1/2}}{\Delta x}dx$, we can compute the half-cell first-order moment of polynomials $p_k(x)$, $k=1,2,3,4$ over $[x_{i-1/2}, x_i]$, and the results can be given explicitly,

$$\begin{cases} \frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_i} p_1(x)\frac{x-x_{i-1/2}}{\Delta x}dx = \frac{1}{8}\overline{u}_i, \\ \frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_i} p_2(x)\frac{x-x_{i-1/2}}{\Delta x}dx = \frac{11}{1280}\overline{u}_{i-1} + \frac{251}{1920}\overline{u}_i - \frac{11}{768}\overline{u}_{i+1}, \\ \frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_i} p_3(x)\frac{x-x_{i-1/2}}{\Delta x}dx = -\frac{639}{140800}\overline{u}_{i-1} + \frac{25}{192}\overline{u}_i - \frac{283}{422400}\overline{u}_{i+1} - \frac{5217}{17600}\overline{v}_i^{mod}, \\ \frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_i} p_4(x)\frac{x-x_{i-1/2}}{\Delta x}dx = -\frac{59033}{5632000}\overline{u}_{i-1} + \frac{71481}{512000}\overline{u}_i - \frac{11629}{2816000}\overline{u}_{i+1} \\ -\frac{9999}{256000}\overline{v}_{i-1} - \frac{171509}{563200}\overline{v}_i^{mod} + \frac{1111}{64000}\overline{v}_{i+1}. \end{cases}$$

The final reconstruction for $\frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_i} u(x)\frac{x-x_{i-1/2}}{\Delta x}dx$ is given as

$$\frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_i} u(x)\frac{x-x_{i-1/2}}{\Delta x}dx \approx \sum_{k=1}^{4}\omega_{k,4}\frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_i} p_k(x)\frac{x-x_{i-1/2}}{\Delta x}dx, \tag{2.31}$$

with the same nonlinear weights $\omega_{k,4}$, $k=1,2,3,4$ as in (2.29).

The similarity goes to $\frac{1}{\Delta x}\int_{x_i}^{x_{i+1/2}} u(x)\frac{x-x_{i+1/2}}{\Delta x}dx$. One can compute the half-cell first-order moment of polynomials $p_k(x)$, $k=1,2,3,4$ over $[x_i, x_{i+1/2}]$, and the results can be given explicitly,

$$\begin{cases} \frac{1}{\Delta x}\int_{x_i}^{x_{i+1/2}} p_1(x)\frac{x-x_{i+1/2}}{\Delta x}dx = -\frac{1}{8}\overline{u}_i, \\ \frac{1}{\Delta x}\int_{x_i}^{x_{i+1/2}} p_2(x)\frac{x-x_{i+1/2}}{\Delta x}dx = \frac{11}{786}\overline{u}_{i-1} - \frac{251}{1920}\overline{u}_i - \frac{11}{1280}\overline{u}_{i+1}, \\ \frac{1}{\Delta x}\int_{x_i}^{x_{i+1/2}} p_3(x)\frac{x-x_{i+1/2}}{\Delta x}dx = \frac{283}{422400}\overline{u}_{i-1} - \frac{25}{192}\overline{u}_i + \frac{639}{140800}\overline{u}_{i+1} - \frac{5217}{17600}\overline{v}_i^{mod}, \\ \frac{1}{\Delta x}\int_{x_i}^{x_{i+1/2}} p_4(x)\frac{x-x_{i+1/2}}{\Delta x}dx = \frac{11629}{2816000}\overline{u}_{i-1}\frac{71481}{512000}\overline{u}_i + \frac{59033}{5632000}\overline{u}_{i+1} \\ +\frac{1111}{64000}\overline{v}_{i-1} - \frac{171509}{563200}\overline{v}_i^{mod} - \frac{9999}{256000}\overline{v}_{i+1}. \end{cases}$$

The final reconstruction for $\frac{1}{\Delta x}\int_{x_i}^{x_{i+1/2}} u(x)\frac{x-x_{i+1/2}}{\Delta x}dx$ is given as

$$\frac{1}{\Delta x}\int_{x_i}^{x_{i+1/2}} u(x)\frac{x-x_{i+1/2}}{\Delta x}dx \approx \sum_{k=1}^{4}\omega_{k,4}\frac{1}{\Delta x}\int_{x_i}^{x_{i+1/2}} p_k(x)\frac{x-x_{i+1/2}}{\Delta x}dx. \tag{2.32}$$

with the same nonlinear weights $\omega_{k,4}$, $k=1,2,3,4$ as in (2.29).

If the cell I_i is not a troubled-cell, we use the following linear reconstructions

$$\begin{aligned} \frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_i} u(x)\frac{x-x_{i-1/2}}{\Delta x}dx &\approx \frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_i} q_4(x)\frac{x-x_{i-1/2}}{\Delta x}dx \\ &= -\frac{5}{512}\overline{u}_{i-1} + \frac{71}{512}\overline{u}_i - \frac{1}{256}\overline{u}_{i+1} \\ &- \frac{9}{256}\overline{v}_{i-1} - \frac{77}{256}\overline{v}_i^{mod} + \frac{1}{64}\overline{v}_{i+1}, \\ \frac{1}{\Delta x}\int_{x_i}^{x_{i+1/2}} u(x)\frac{x-x_{i+1/2}}{\Delta x}dx &\approx \frac{1}{\Delta x}\int_{x_i}^{x_{i+1/2}} q_4(x)\frac{x-x_{i+1/2}}{\Delta x}dx \\ &= \frac{1}{256}\overline{u}_{i-1} - \frac{71}{512}\overline{u}_i + \frac{5}{512}\overline{u}_{i+1} \\ &+ \frac{1}{64}\overline{v}_{i-1} - \frac{77}{256}\overline{v}_i^{mod} - \frac{9}{256}\overline{v}_{i+1}. \end{aligned}$$

Step 4 A reconstruction of $u(x_*)$, $x_* \in x^{G,i}$ from $\{\overline{u}_{i-1}, \overline{u}_i, \overline{u}_{i+1}, \overline{v}_{i-1}, \overline{v}_i^{mod}, \overline{v}_{i+1}\}$.

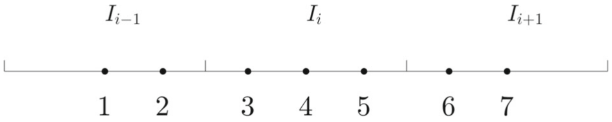


Fig. 1 The labels of Gauss-Lobatto points in the cells $I_{i-1/2}, I_{i+1/2}$

If the target cell I_i is identified to be a troubled-cell, the same stencils as in step 2 are used, together with the same reconstructed polynomials $q_l(x)$, $p_k(x)$, $l, k = 1, 2, 3, 4$ and the nonlinear weights.

Take $u(x_i)$ as an example. We can compute the point value of polynomials $p_k(x), k = 1, 2, 3, 4$ at x_i, and the results can be given explicitly,

$$\begin{cases} p_1(x_i) = \overline{u}_i, \\ p_2(x_i) = -\frac{11}{240}\overline{u}_{i-1} + \frac{131}{120}\overline{u}_i - \frac{11}{240}\overline{u}_{i+1}, \\ p_3(x_i) = -\frac{1}{24}\overline{u}_{i-1} + \frac{13}{12}\overline{u}_i - \frac{1}{24}\overline{u}_{i+1}, \\ p_4(x_i) = -\frac{45997}{384000}\overline{u}_{i-1} + \frac{237997}{192000}\overline{u}_i - \frac{45997}{384000}\overline{u}_{i+1} - \frac{29997}{64000}\overline{v}_{i-1} + \frac{29997}{64000}\overline{v}_{i+1}. \end{cases}$$

The final approximation for $u(x_i)$ is defined as

$$u(x_i) \approx \sum_{k=1}^{4} \omega_{k,4} p_k(x_i). \tag{2.33}$$

If the cell I_i is not a troubled-cell, we use the following linear reconstruction

$$u(x_i) \approx q_4(x_i) = -\frac{43}{384}\overline{u}_{i-1} + \frac{235}{192}\overline{u}_i - \frac{43}{384}\overline{u}_{i+1} - \frac{27}{64}\overline{v}_{i-1} + \frac{27}{64}\overline{v}_{i+1}.$$

The reconstructions of other quadrature points x_*, $x_* \in x^{G,i}$ are similar to the above procedure. The multi-resolution HWENO reconstructions perform better than the HWENO reconstructions in [27]. One advantage is that the same linear weights are used for different reconstructions in multi-resolution HWENO method while the linear weights should be chosen carefully in the HWENO method [27].

Step 5 A linear interpolation of $f_x(u(x_*))$ from $f(u(x_*))$, $x_* \in x^{G,i}$.

The Gauss–Lobatto quadrature points in the cells $I_{i-1/2}, I_{i+1/2}$ are relabeled as $G_1, \cdots, G_7$, namely

$$G_1 = x_{i-1}, \quad G_2 = x_{i-\frac{1}{2}-\frac{\sqrt{5}}{10}}, \quad G_3 = x_{i-\frac{1}{2}+\frac{\sqrt{5}}{10}}, \quad G_4 = x_i,$$
$$G_5 = x_{i+\frac{1}{2}-\frac{\sqrt{5}}{10}}, \quad G_6 = x_{i+\frac{1}{2}+\frac{\sqrt{5}}{10}}, \quad G_7 = x_{i+1}.$$

Note that the points $G_1, \cdots, G_7$ are interior points with respect to the primal mesh, see Fig. 1.

In the target cell I_i, one will need to get the approximations of $f_x(u(x_*, t))$, $x_* = G_3, G_4, G_5$ based on $u(x_*, t)$, $x_* = G_1, \cdots, G_7$. For the brevity of the presentation, we write $f(u(G_k, t)) = f_k, k = 1, \cdots, 7$. We reconstruct a sixth-degree polynomial $Q(x)$ such that

$$Q(x)|_{G_k} = f_k, \quad k = 1, \cdots, 7.$$

The corresponding first derivative of the polynomial $Q(x)$ at $x_* = G_3, G_4, G_5$ are given as follows,

$$\begin{aligned}
Q'(G_3) = \frac{1}{44\Delta x}\Big[&2(-9+5\sqrt{5})f_1 + 4(40-21\sqrt{5})f_2 - (129+13\sqrt{5})f_3 \\
&+44(-1+3\sqrt{5})f_4 - 11(5+\sqrt{5})f_5 + \\
&4(21-8\sqrt{5})f_6 + 2(1-\sqrt{5})f_7\Big], \\
Q'(G_4) = \frac{1}{44\Delta x}\Big[&-2f_1 + 5(-9+5\sqrt{5})f_2 - 5(9+5\sqrt{5})f_3 \\
&+5(9+5\sqrt{5})f_5 + 5(9-5\sqrt{5})f_6 + 2f_7\Big], \\
Q'(G_5) = \frac{1}{44\Delta x}\Big[&-2(1-\sqrt{5})f_1 - 4(21-8\sqrt{5})f_2 + 11(5+\sqrt{5})f_3 \\
&+44(1-3\sqrt{5})f_4 + (129+13\sqrt{5})f_5 - \\
&4(40-21\sqrt{5})f_6 + 2(9-5\sqrt{5})f_7\Big].
\end{aligned}$$

The final reconstructions for $f_x(u(x_*, t))$ are approximated by

$$f_x(u(x_*)) = Q'(x_*),\ x_* = G_3, G_4, G_5. \tag{2.34}$$

Remark 1 For the system case, such as the Euler equation of gas dynamics, the HWENO reconstructions in steps 2–4 can be performed on the unknowns component by component or based on the local characteristic decomposition. The latter is more computationally expensive, but it provides better performance to control the spurious oscillation. Here, the local characteristic decomposition is applied for the HWENO reconstructions in steps 2–4.

2.2 Two-Dimensional Case

Consider the two-dimensional scalar conservation law

$$\begin{cases} u_t + f(u)_x + g(u)_y = 0, \\ u(x, y, 0) = u_0(x, y). \end{cases} \tag{2.35}$$

Let $\{x_i, y_j\}_{ij}$ be a uniform partition of the two-dimensional computational domain with the mesh sizes Δx in the x direction, and Δy in the y direction. With $x_{i+1/2} = \frac{1}{2}(x_i + x_{i+1})$, $y_{j+1/2} = \frac{1}{2}(y_j + y_{j+1})$, we denote the primal mesh as $I_{ij} = [x_{i-1/2}, x_{i+1/2}] \times [y_{j-1/2}, y_{j+1/2}]$ with its center (x_i, y_j), and the dual mesh as $I_{i+1/2,j+1/2} = [x_i, x_{i+1}] \times [y_j, y_{j+1}]$ with its center $(x_{i+1/2}, y_{j+1/2})$. The proposed method evolves the numerical solutions in a staggered manner with the two meshes.

The zeroth-order and first-order moments of the solution on primal mesh I_{ij} are denoted as $\{\overline{p}_{ij}\}$ $(p = u, v, w)$, that is

$$\begin{cases} \overline{u}_{ij} \approx \frac{1}{\Delta x \Delta y}\int_{I_{ij}} u(x, y, t)dxdy, \\ \overline{v}_{ij} \approx \frac{1}{\Delta x \Delta y}\int_{I_{ij}} u(x, y, t)\frac{x-x_i}{\Delta x}dxdy, \\ \overline{w}_{ij} \approx \frac{1}{\Delta x \Delta y}\int_{I_{ij}} u(x, y, t)\frac{y-y_j}{\Delta y}dxdy. \end{cases}$$

The moments of the solution on dual mesh $I_{i+1/2,j+1/2}$ are denoted as $\{\overline{p}_{i+1/2,j+1/2}\}$ $(p = u, v, w)$

$$\begin{cases} \overline{u}_{i+1/2,j+1/2} \approx \frac{1}{\Delta x \Delta y} \int_{I_{i+1/2,j+1/2}} u(x,y,t) dxdy, \\ \overline{v}_{i+1/2,j+1/2} \approx \frac{1}{\Delta x \Delta y} \int_{I_{i+1/2,j+1/2}} u(x,y,t) \frac{x-x_{i+1/2}}{\Delta x} dxdy, \\ \overline{w}_{i+1/2,j+1/2} \approx \frac{1}{\Delta x \Delta y} \int_{I_{i+1/2,j+1/2}} u(x,y,t) \frac{y-y_{j+1/2}}{\Delta y} dxdy. \end{cases}$$

Suppose at $t = t^n$, the approximations for the zeroth-order and first-order moments of the solution $\{\overline{p}^n_{ij}\}_{ij}$ $(p = u, v, w)$ are available on primal mesh. We multiply the equation (2.35) by $\frac{1}{\Delta x \Delta y}$, $\frac{x-x_{i+1/2}}{(\Delta x)^2 \Delta y}$ and $\frac{y-y_{j+1/2}}{\Delta x (\Delta y)^2}$, respectively, integrate over $I_{i+1/2,j+1/2} \times [t^n, t^{n+1}]$, apply integration by parts and get

$$\begin{aligned} \overline{u}^{n+1}_{i+1/2,j+1/2} =& \overline{u}^n_{i+1/2,j+1/2} - \frac{1}{\Delta x \Delta y} \int_{t^n}^{t^{n+1}} \int_{y_j}^{y_{j+1}} \left[f(u(x_{i+1},y,t)) - f(u(x_i,y,t)) \right] dydt \\ & - \frac{1}{\Delta x \Delta y} \int_{t^n}^{t^{n+1}} \int_{x_i}^{x_{i+1}} \left[g(u(x,y_{j+1},t)) - g(u(x,y_j,t)) \right] dxdt, \end{aligned} \tag{2.36}$$

$$\begin{aligned} \overline{v}^{n+1}_{i+1/2,j+1/2} =& \overline{v}^n_{i+1/2,j+1/2} + \frac{1}{\Delta x^2 \Delta y} \int_{t^n}^{t^{n+1}} \int_{x_i}^{x_{i+1}} \int_{y_j}^{y_{j+1}} f(u(x,y,t)) dxdydt \\ & - \frac{1}{2 \Delta x \Delta y} \int_{t^n}^{t^{n+1}} \int_{y_j}^{y_{j+1}} \left[f(u(x_{i+1},y,t)) + f(u(x_i,y,t)) \right] dydt \\ & - \frac{1}{\Delta x \Delta y} \int_{t^n}^{t^{n+1}} \int_{x_i}^{x_{i+1}} \left[g(u(x,y_{j+1},t)) - g(u(x,y_j,t)) \right] \frac{x - x_{i+1/2}}{\Delta x} dxdt, \end{aligned} \tag{2.37}$$

$$\begin{aligned} \overline{w}^{n+1}_{i+1/2,j+1/2} =& \overline{w}^n_{i+1/2,j+1/2} - \frac{1}{\Delta x \Delta y} \int_{t^n}^{t^{n+1}} \int_{y_j}^{y_{j+1}} \left[f(u(x_{i+1},y,t)) - f(u(x_i,y,t)) \right] \\ & \frac{y - y_{j+1/2}}{\Delta y} dydt \\ & + \frac{1}{\Delta x \Delta y^2} \int_{t^n}^{t^{n+1}} \int_{x_i}^{x_{i+1}} \int_{y_j}^{y_{j+1}} g(u(x,y,t)) dxdydt \\ & - \frac{1}{2 \Delta x \Delta y} \int_{t^n}^{t^{n+1}} \int_{x_i}^{x_{i+1}} \left[g(u(x,y_{j+1},t)) + g(u(x,y_j,t)) \right] dxdt. \end{aligned} \tag{2.38}$$

Similarly as in one-dimensional case, the spatial integrals in (2.36)–(2.38) can be approximated by one-dimensional four-point Gauss-Lobatto quadrature, or its tensor-version in two dimensions. If we choose a small enough time step Δt, the temporal integrals in (2.36)–(2.38) can be approximated by three-point Gauss quadrature according to (2.6). Then, the proposed central scheme is given as follows.

(b.1) With $\{\overline{p}^n_{ij}\}_{ij}$ $(p = u, v, w)$ being available at $t = t^n$ on primal mesh, the zeroth-order and first-order moments $\{\overline{p}^{n+1}_{i+1/2,j+1/2}\}_{ij}$ $(p = u, v, w)$ of the solution at $t = t^{n+1}$ on dual mesh can be computed as follows,

$$
\begin{aligned}
\overline{u}^{n+1}_{i+1/2,j+1/2} =& \overline{u}^{n}_{i+1/2,j+1/2} - \frac{\Delta t}{\Delta x}\sum_{m=1}^{3}\sum_{l=1}^{4}\alpha_m\omega_l \\
&\left[f(u(x_{i+1},y_l^G,t)) - f(u(x_i,y_l^G,t))\right]|_{t=t^n+\theta_m\Delta t} \\
&- \frac{\Delta t}{\Delta y}\sum_{m=1}^{3}\sum_{l=1}^{4}\alpha_m\omega_l\left[g(u(x_l^G,y_{j+1},t)) - g(u(x_l^G,y_j,t))\right]|_{t=t^n+\theta_m\Delta t},
\end{aligned} \tag{2.39}
$$

$$
\begin{aligned}
\overline{v}^{n+1}_{i+1/2,j+1/2} =& \overline{v}^{n}_{i+1/2,j+1/2} + \frac{\Delta t}{\Delta x}\sum_{m=1}^{3}\sum_{l_1=1}^{4}\sum_{l_2=1}^{4}\alpha_m\omega_{l_1}\omega_{l_2} f(u(x_{l_1}^G,y_{l_2}^G,t))\big|_{t=t^n+\theta_m\Delta t} \\
&- \frac{\Delta t}{2\Delta x}\sum_{m=1}^{3}\sum_{l=1}^{4}\alpha_m\omega_l\left[f(u(x_{i+1},y_l^G,t)) + f(u(x_i,y_l^G,t))\right]\big|_{t=t^n+\theta_m\Delta t} \\
&- \frac{\Delta t}{\Delta y}\sum_{m=1}^{3}\sum_{l=1}^{4}\alpha_m\omega_l\left[\left[g(u(x_l^G,y_{j+1},t)) - g(u(x_l^G,y_j,t))\right]\frac{x_l^G - x_{i+1/2}}{\Delta x}\right]\big|_{t=t^n+\theta_m\Delta t},
\end{aligned} \tag{2.40}
$$

$$
\begin{aligned}
\overline{w}^{n+1}_{i+1/2,j+1/2} =& \overline{w}^{n}_{i+1/2,j+1/2} - \frac{\Delta t}{\Delta x}\sum_{m=1}^{3}\sum_{l=1}^{4}\alpha_m\omega_l \\
&\left[\left[f(u(x_{i+1},y_l^G,t)) - f(u(x_i,y_l^G,t))\right]\frac{y_l^G - y_{j+1/2}}{\Delta y}\right]\big|_{t=t^n+\theta_m\Delta t} \\
&+ \frac{\Delta t}{\Delta y}\sum_{m=1}^{3}\sum_{l_1=1}^{4}\sum_{l_2=1}^{4}\alpha_m\omega_{l_1}\omega_{l_2} g(u(x_{l_1}^G,y_{l_2}^G,t))\big|_{t=t^n+\theta_m\Delta t} \\
&- \frac{\Delta t}{2\Delta y}\sum_{m=1}^{3}\sum_{l=1}^{4}\alpha_m\omega_l\left[g(u(x_l^G,y_{j+1},t)) + g(u(x_l^G,y_j,t))\right]\big|_{t=t^n+\theta_m\Delta t},
\end{aligned} \tag{2.41}
$$

where $y_1^G = y_j, y_2^G = y_{j+\frac{1}{2}-\frac{\sqrt{5}}{10}}, y_3^G = y_{j+\frac{1}{2}+\frac{\sqrt{5}}{10}}$ and $y_4^G = y_{j+1}$ are the Gauss-Lobatto quadrature point on the y axis over cell $I_{j+1/2}$. We define $y^G = \cup_j y^{G,j} = \cup_j\{y_{j-\frac{1}{2}+\frac{\sqrt{5}}{10}}, y_j, y_{j+\frac{1}{2}-\frac{\sqrt{5}}{10}}\}$.

(b.2) With $\{\overline{p}^{n+1}_{i+1/2,j+1/2}\}_{ij} (p = u, v, w)$ being available at $t = t^{n+1}$ on dual mesh, the zeroth-order and first-order moments $\{\overline{p}^{n+2}_{ij}\}_{ij}$ $(p = u, v, w)$ at $t = t^{n+2}$ on primal mesh can be computed similarly as in (b.1).

(b.3) Set n to be $n + 2$, and return to (b.1).

In order to finalize the proposed scheme, certain quantities in (b.1) need to be reconstructed. According to (2.39)–(2.41), one would need to calculate accurate approximations for the following quantities

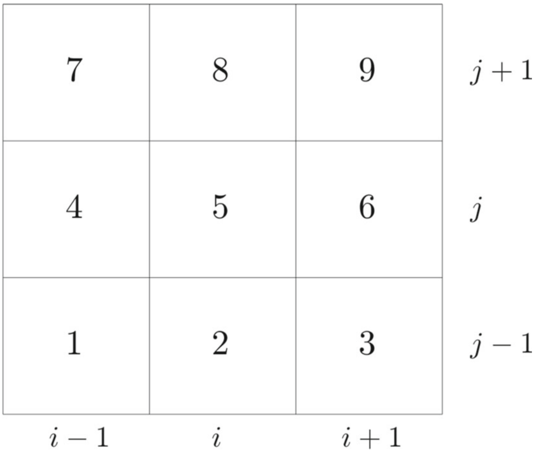
$$
\begin{aligned}
&\overline{p}^n_{i+1/2,j+1/2},\ p = u, v, w,\ \forall i, j,\ \ u(x_*, y_*, t^n + \theta_m\Delta t),\ (x_*, y_*) \in (x, y)^G \\
&\quad = x^G \times y^G,\ m = 1, 2, 3.
\end{aligned}
$$

To obtain the cell average $\overline{u}^n_{i+1/2,j+1/2}$, one would like to approximate four quarter-cell averages as follows,

$$
\begin{aligned}
&\frac{1}{\Delta x\Delta y}\int_{x_i}^{x_{i+1/2}}\int_{y_j}^{y_{j+1/2}} u(x,y,t^n)dxdy, \quad \frac{1}{\Delta x\Delta y}\int_{x_{i+1/2}}^{x_{i+1}}\int_{y_j}^{y_{j+1/2}} u(x,y,t^n)dxdy, \\
&\frac{1}{\Delta x\Delta y}\int_{x_i}^{x_{i+1/2}}\int_{y_{j+1/2}}^{y_{j+1}} u(x,y,t^n)dxdy, \quad \frac{1}{\Delta x\Delta y}\int_{x_{i+1/2}}^{x_{i+1}}\int_{y_{j+1/2}}^{y_{j+1}} u(x,y,t^n)dxdy.
\end{aligned}
$$

The first-order moments $\overline{p}^n_{i+1/2,j+1/2},\ p = v, w$ can be obtained similarly.

Fig. 2 The labels for two-dimensional cell I_{ij} and its neighboring cells



To approximate the point value $u(x_*, y_*, t^n + \theta_m \Delta t), (x_*, y_*) \in (x, y)^G, m = 1, 2, 3$, similar to one-dimensional case, we will use the fourth-order NCE-RK method to assist an auxiliary ODE problem (2.10). Here $y(t)$ in (2.10) is the point value $u(x_*, y_*, t)$, and $H(y(t))$ is the corresponding $-f_x(u(x_*, y_*, t)) - g_y(u(x_*, y_*, t))$. Two ingredients are needed to approximate the point value $u(x_*, y_*, t^n + \theta_m \Delta t)$. One part is the calculation of $u(x_*, y_*, t^n)$, which can be obtained by a sixth-order multi-resolution HWENO reconstruction based on $\overline{p}^n_{ij}$ $(p = u, v, w)$. The other is the calculation of $-f_x(u(x_*, y_*, t)) - g_y(u(x_*, y_*, t))$. The values of $f_x(u(x_*, y_*, t))$ and $g_y(u(x_*, y_*, t))$ are approximated using a linear interpolation based on $f(u(x_*, y_*, t))$ and $g(u(x_*, y_*, t))$, respectively.

Details are given in the following steps. The superscript n and the dependence on time t will be omitted below. For the brevity of the presentation, we relabel the cell I_{ij} and its neighboring cells as $I_1, \cdots, I_9$, see Fig. 2. The cell averages are relabeled as $\overline{u}_1, \cdots, \overline{u}_9$, namely

$$\begin{aligned}
&\overline{u}_1 = \overline{u}_{i-1,j-1}, \quad &&\overline{u}_2 = \overline{u}_{i,j-1}, \quad &&\overline{u}_3 = \overline{u}_{i+1,j-1}, \\
&\overline{u}_4 = \overline{u}_{i-1,j}, &&\overline{u}_5 = \overline{u}_{ij}, &&\overline{u}_6 = \overline{u}_{i+1,j}, \\
&\overline{u}_7 = \overline{u}_{i-1,j+1}, &&\overline{u}_8 = \overline{u}_{i,j+1}, &&\overline{u}_9 = \overline{u}_{i+1,j+1}.
\end{aligned}$$

The first-order moments are relabeled in a similar way.

Step 1 Identify the troubled-cells and modify the first-order moments.

Step 1.1. We choose the KXRCF troubled-cell indicator to identify the troubled-cells. Again, we divide the boundary of the cell I_{ij} into two parts: inflow boundary ∂I^-_{ij} ($\overrightarrow{v} \cdot \overrightarrow{n} < 0$, where $\overrightarrow{v}$ is the velocity of flow, $\overrightarrow{n}$ is the outer normal vector to ∂I_{ij}) and outflow boundary ∂I^+_{ij} ($\overrightarrow{v} \cdot \overrightarrow{n} > 0$). The target cell I_{ij} is identified as a troubled-cell if the following criterion is satisfied

$$\frac{\left| \int_{\partial I^-_{ij}} \left(u_{ij}(x, y) - u_{n_{ij}}(x, y) \right) ds \right|}{h_{ij}^{\frac{l+1}{3}} \left| \partial I^-_{ij} \right| \| u_{ij}(x, y) \|} > 1 \tag{2.42}$$

where $I_{n_{ij}}$ is the neighbor of I_{ij} on the side of ∂I^-_{ij}, h_{ij} is the length of I_{ij}. The parameter $l = 5$ is the degree of $u_{ij}(x, y)$ which is a sixth-order linear approximation of $u(x, y)$, and

$u_{ij}(x,y)$ satisfies the following conditions

$$\begin{cases} \frac{1}{\Delta x\Delta y}\int_{I_k} u_{ij}(x,y)dxdy = \overline{u}_k, \quad k=1,\cdots,9, \\ \frac{1}{\Delta x\Delta y}\int_{I_{k_x}} u_{ij}(x,y)\frac{x-x_{k_x}}{\Delta x}dxdy = \overline{v}_{k_x}, \quad k_x = 1,3,4,5,6,7,9, \\ \frac{1}{\Delta x\Delta y}\int_{I_{k_y}} u_{ij}(x,y)\frac{y-y_{k_y}}{\Delta y}dxdy = \overline{w}_{k_y}, \quad k_y = 1,2,3,5,7,8,9. \end{cases} \tag{2.43}$$

The polynomial can be obtained by ensuring that $u_{ij}(x,y)$ has the same average of $u(x,y)$ in the cell I_{ij}, and match the other conditions in a least square method.

$\|u_{ij}(x,y)\|$ is taken to be L^∞ norm

$$\|u_{ij}(x,y)\| \approx \max\left\{|u^{\pm}_{i\mp\frac{1}{2},j_l}|, |u^{\mp}_{i_l,j\pm\frac{1}{2}}|, l=1,2,3,4; |u_{i_l,j_m}|, l=2,3, m=2,3\right\}$$

where

$$\begin{aligned} i_1 &= i-\frac{1}{2}, i_2 = i-\frac{\sqrt{5}}{10}, i_3 = i+\frac{\sqrt{5}}{10}, i_4 = i+\frac{1}{2};\ j_1 = j-\frac{1}{2}, j_2 \\ &= j-\frac{\sqrt{5}}{10}, j_3 = j+\frac{\sqrt{5}}{10}, j_4 = j+\frac{1}{2}, \end{aligned}$$

are the quadrature points.

The average of line integral in the formula (2.42) can be expressed explicitly

$$\begin{aligned} &\frac{1}{|\partial I_{ij}^-|}\left|\int_{\partial I_{ij}^-}\big(u_{ij}(x,y)-u_{n_{ij}}(x,y)\big)\,ds\right| \\ &= \frac{1}{|\partial I_{ij}^-|}\left|\sum_{m=1}^{4}\Delta y\,\omega_m\left[(u^+_{i-1/2,j_m}-u^-_{i-1/2,j_m})*s(\overrightarrow{v}_{i-\frac{1}{2},j})\right.\right. \\ &\quad \left.+(u^-_{i+1/2,j_m}-u^+_{i+1/2,j_m})*s(-\overrightarrow{v}_{i+\frac{1}{2},j})\right] \\ &\quad +\sum_{m=1}^{4}\Delta x\,\omega_m\left[(u^+_{i_m,j-1/2}-u^-_{i_m,j-1/2})*s(\overrightarrow{v}_{i,j-\frac{1}{2}})\right. \\ &\quad \left.\left.+(u^-_{i_m,j+1/2}-u^+_{i_m,j+1/2})*s(-\overrightarrow{v}_{i,j+\frac{1}{2}})\right]\right| \end{aligned}$$

where $\omega_1=\omega_4=\frac{1}{12}$ and $\omega_2=\omega_3=\frac{5}{12}$ are the quadrature weights, and $s(x)$ is defined in (2.15).

For the two-dimensional scalar case, we take $\overrightarrow{v}=f'(u)$ in the x direction and $\overrightarrow{v}=g'(u)$ in the y direction. For the two-dimensional Euler equations, we set the density ρ as the indicator variable, and $\overrightarrow{v}=\mu$ is the velocity of the fluid in the x direction and $\overrightarrow{v}=\nu$ is the velocity of the fluid in the y direction.

Step 1.2 If the target cell I_{ij} is identified to be a troubled-cell, one would like to identify the eight neighboring cells of I_{ij} as troubled-cells and modify the first-order moments $\overline{v}_{ij}$ and $\overline{w}_{ij}$ as below,

- Based on $\{\overline{u}_{i-1,j},\overline{u}_{ij},\overline{u}_{i+1,j},\overline{v}_{i-1,j},\overline{v}_{i+1,j}\}$, we apply step 1.2 in Sect. 2.1 along the x direction to obtain $\{\overline{v}^{mod}_{ij}\}$.
- Based on $\{\overline{u}_{i,j-1},\overline{u}_{ij},\overline{u}_{i,j+1},\overline{w}_{i,j-1},\overline{w}_{i,j+1}\}$, we apply step 1.2 in Sect. 2.1 along the y direction to obtain $\{\overline{w}^{mod}_{ij}\}$.

If the cell I_{ij} is not a troubled-cell, we simply set $\overline{v}^{mod}_{ij}=\overline{v}_{ij}$, $\overline{w}^{mod}_{ij}=\overline{w}_{ij}$.

Step 2 A reconstruction of quarter-cell averages in the cell I_{ij}.

If the cell I_{ij} is identified as a troubled-cell in step 1, we will apply the following multi-resolution HWENO reconstruction. With a series of central spatial stencils, we reconstruct a zeroth degree polynomial $q_1(x, y)$, a quadratic polynomial $q_2(x, y)$, a cubic polynomial $q_3(x, y)$, and a quintic polynomial $q_4(x, y)$, respectively, and these polynomials satisfy the following conditions

$$\frac{1}{\Delta x \Delta y}\int_{I_5} q_1(x, y)dxdy = \overline{u}_5, \tag{2.44}$$

$$\frac{1}{\Delta x \Delta y}\int_{I_k} q_2(x, y)dxdy = \overline{u}_k, \quad k = 1, \cdots, 9, \tag{2.45}$$

$$\begin{cases} \frac{1}{\Delta x \Delta y}\int_{I_k} q_3(x, y)dxdy = \overline{u}_k, \quad k = 1, \cdots, 9, \\ \frac{1}{\Delta x \Delta y}\int_{I_5} q_3(x, y)\frac{x-x_i}{\Delta x}dxdy = \overline{v}_5^{mod}, \\ \frac{1}{\Delta x \Delta y}\int_{I_5} q_3(x, y)\frac{y-y_j}{\Delta y}dxdy = \overline{w}_5^{mod}, \end{cases} \tag{2.46}$$

$$\begin{cases} \frac{1}{\Delta x \Delta y}\int_{I_k} q_4(x, y)dxdy = \overline{u}_k, \quad k = 1, \cdots, 9, \\ \frac{1}{\Delta x \Delta y}\int_{I_5} q_4(x, y)\frac{x-x_i}{\Delta x}dxdy = \overline{v}_5^{mod}, \\ \frac{1}{\Delta x \Delta y}\int_{I_5} q_4(x, y)\frac{y-y_j}{\Delta y}dxdy = \overline{w}_5^{mod}, \\ \frac{1}{\Delta x \Delta y}\int_{I_{k_x}} q_4(x, y)\frac{x-x_{k_x}}{\Delta x}dxdy = \overline{v}_{k_x}, \quad k_x = 1, 3, 4, 6, 7, 9, \\ \frac{1}{\Delta x \Delta y}\int_{I_{k_y}} q_4(x, y)\frac{y-y_{k_y}}{\Delta y}dxdy = \overline{w}_{k_y}, \quad k_y = 1, 2, 3, 7, 8, 9. \end{cases} \tag{2.47}$$

Note that the number of equations is greater than the number of unknowns for the polynomials $q_l(x, y), l = 2, 3, 4$. These polynomials must have the same average of u in the cell I_{ij}, and we can get them by requiring that they match the other conditions in a least square method.

Then, we express the polynomial $q_4(x, y)$ as the linear combination of polynomials $p_k(x, y), k = 1, 2, 3, 4$ as below

$$q_4(x, y) = \gamma_{1,4}p_1(x, y) + \gamma_{2,4}p_2(x, y) + \gamma_{3,4}p_3(x, y) + \gamma_{4,4}p_4(x, y). \tag{2.48}$$

The equivalent polynomials $p_k(x, y), k = 1, 2, 3, 4$ are computed as follows

$$\begin{cases} p_1(x, y) = q_1(x, y), \\ p_k(x, y) = \frac{1}{\gamma_{k,k}}q_k(x, y) - \sum\limits_{l=1}^{k-1} \frac{\gamma_{l,k}}{\gamma_{k,k}} p_l(x, y), \quad k = 2, 3, 4, \end{cases} \tag{2.49}$$

where the linear weights $\gamma_{l,k}, l = 1, \cdots, k, k = 2, 3, 4$ are defined in (2.26).

To compute the nonlinear weights for (2.48), the smoothness indicators of polynomials $p_k(x, y)$ can be calculated as below

$$\beta_k = \sum_{|\alpha|=1}^{r} \int_{I_{ij}} (\Delta x \Delta y)^{|\alpha|-1} \left(\frac{\partial^{|\alpha|}}{\partial x^{\alpha_1}\partial y^{\alpha_2}} p_k(x, y)\right)^2 dxdy, \quad k = 2, 3, 4,$$

where $\alpha = (\alpha_1, \alpha_2)$, $|\alpha| = \alpha_1 + \alpha_2$ and $r = 2, 3, 5$ for $k = 2, 3, 4$, respectively. The only difference is β_1. One would like to define a new polynomial $q_1^{new}(x, y)$. First, we reconstruct four linear polynomials as follows

$$\frac{1}{\Delta x \Delta y}\int_{I_k} q_{11}(x, y)dxdy = \overline{u}_k, \quad k = 2, 4, 5,$$

$$\frac{1}{\Delta x \Delta y}\int_{I_k} q_{12}(x, y)dxdy = \overline{u}_k, \quad k = 2, 5, 6,$$

$$\frac{1}{\Delta x\Delta y}\int_{I_k} q_{13}(x,y)dxdy = \overline{u}_k, \quad k = 4,5,8,$$
$$\frac{1}{\Delta x\Delta y}\int_{I_k} q_{14}(x,y)dxdy = \overline{u}_k, \quad k = 5,6,8,$$

and the corresponding smoothness indicators are

$$\begin{aligned}
\beta_{11} &= (\overline{u}_5-\overline{u}_2)^2+(\overline{u}_5-\overline{u}_4)^2,\\
\beta_{12} &= (\overline{u}_5-\overline{u}_2)^2+(\overline{u}_5-\overline{u}_6)^2,\\
\beta_{13} &= (\overline{u}_5-\overline{u}_8)^2+(\overline{u}_5-\overline{u}_4)^2,\\
\beta_{14} &= (\overline{u}_5-\overline{u}_8)^2+(\overline{u}_5-\overline{u}_6)^2.
\end{aligned}$$

We define the quantity τ_1 as the absolute difference between smoothness indicators,

$$\tau_1 = \left(\frac{\sum_{k\neq l}|\beta_{1l}-\beta_{1k}|}{6}\right)^2,$$

and the nonlinear weights of $q_{1l}(x,y), l = 1,2,3,4$ can be computed as

$$\omega_{1l} = \frac{\overline{\omega}_{1l}}{\sum_{m=1}^4 \overline{\omega}_{1m}}, \quad \overline{\omega}_{1l} = \frac{1}{4}\left(1+\frac{\tau_1}{\beta_{1l}+\varepsilon}\right), \quad l = 1,2,3,4,$$

here $\varepsilon = 10^{-6}$. The polynomial $q_1^{new}(x,y)$ is given by

$$q_1^{new}(x,y) = \sum_{l=1}^4 \omega_{1l}q_{1l}(x,y),$$

and the nonlinear weights β_1 can be computed as

$$\beta_1 = \sum_{|\alpha|=1}(\Delta x\Delta y)^{|\alpha|}\left(\frac{\partial^{|\alpha|}}{\partial x^{\alpha_1}\partial y^{\alpha_2}}q_1^{new}(x,y)\right)^2.$$

Based on the smoothness indicators, the nonlinear weights $\omega_{k,4}$ of $p_k(x,y), k = 1,2,3,4$ can be computed as in (2.29), and the final reconstructed polynomial $p(x,y)$ is given by

$$p(x,y) = \sum_{k=1}^4 \omega_{k,4}p_k(x,y).$$

Finally, a sixth-order HWENO approximation for a quarter-cell average is given by

$$\begin{aligned}
\frac{1}{\Delta x\Delta y}\int_{x_{i-1/2}}^{x_i}\int_{y_{j-1/2}}^{y_j} u(x,y)dxdy &\approx \frac{1}{\Delta x\Delta y}\int_{x_{i-1/2}}^{x_i}\int_{y_{j-1/2}}^{y_j} p(x,y)dxdy\\
&= \sum_{k=1}^4 \omega_{k,4}\frac{1}{\Delta x\Delta y}\int_{x_{i-1/2}}^{x_i}\int_{y_{j-1/2}}^{y_j} p_k(x,y)dxdy.
\end{aligned} \tag{2.50}$$

If the cell I_{ij} is not a troubled-cell, it can be approximated by the linear reconstruction

$$\frac{1}{\Delta x\Delta y}\int_{x_{i-1/2}}^{x_i}\int_{y_{j-1/2}}^{y_j} u(x,y)dxdy \approx \frac{1}{\Delta x\Delta y}\int_{x_{i-1/2}}^{x_i}\int_{y_{j-1/2}}^{y_j} q_4(x,y)dxdy.$$

We can obtain the approximations for other quarter-cell averages with similar reconstruction. Once three quarter-cell averages are computed, one can obtain the fourth one based on the local conservation of u [26].

Step 3 A reconstruction of quarter-cell first-order moments in the cell I_{ij}.

If the target cell I_{ij} is identified to be a troubled-cell, the same stencils as in step 2 are used, together with the same reconstructed polynomials $q_l(x, y)$, $p_k(x, y)$, $l, k = 1, 2, 3, 4$ and the nonlinear weights.

A sixth-order HWENO approximation for a quarter-cell first-order moment is given as

$$\begin{aligned}
&\frac{1}{\Delta x \Delta y}\int_{x_{i-1/2}}^{x_i}\int_{y_{j-1/2}}^{y_j} u(x, y)\frac{x - x_{i-1/2}}{\Delta x}dxdy \\
&\approx \frac{1}{\Delta x \Delta y}\int_{x_{i-1/2}}^{x_i}\int_{y_{j-1/2}}^{y_j} p(x, y)\frac{x - x_{i-1/2}}{\Delta x}dxdy \\
&= \sum_{k=1}^{4}\omega_{k,4}\frac{1}{\Delta x \Delta y}\int_{x_{i-1/2}}^{x_i}\int_{y_{j-1/2}}^{y_j} p_k(x, y)\frac{x - x_{i-1/2}}{\Delta x}dxdy.
\end{aligned}$$

If the cell I_{ij} is not a troubled-cell, we use the following linear reconstruction

$$\frac{1}{\Delta x \Delta y}\int_{x_{i-1/2}}^{x_i}\int_{y_{j-1/2}}^{y_j} u(x, y)\frac{x - x_{i-1/2}}{\Delta x}dxdy \approx \frac{1}{\Delta x \Delta y}\int_{x_{i-1/2}}^{x_i}\int_{y_{j-1/2}}^{y_j} q_4(x, y)\frac{x - x_{i-1/2}}{\Delta x}dxdy.$$

The similarity goes to the other quarter-cell first-order moments.

Step 4 A reconstruction of $u(x_*, y_*)$, $(x_*, y_*) \in (x, y)^G$.

If the target cell I_{ij} is identified to be a troubled-cell, the same stencils as in step 2 are used, together with the same reconstructed polynomials $q_l(x, y)$, $p_k(x, y)$, $l, k = 1, 2, 3, 4$ and the nonlinear weights.

The final approximation for $u(x_*, y_*)$ is given as

$$u(x_*, y_*) \approx \sum_{k=1}^{4}\omega_{k,4}p_k(x_*, y_*).$$

If the cell I_{ij} is not a troubled-cell, we use the following linear reconstruction

$$u(x_*, y_*) \approx q_4(x_*, y_*).$$

Step 5. Reconstruct $f_x(u(x_*, y_*))$ and $g_y(u(x_*, y_*))$ from $u(x_*, y_*)$, $(x_*, y_*) \in (x, y)^G$.

The Gauss-Lobatto quadrature points (x_*, y_*) within cell I_{ij} are relabeled as $G^1_{ij}, \cdots, G^9_{ij}$, namely

$$\begin{aligned}
&G^1_{ij} = (x_{i-\frac{1}{2}+\frac{\sqrt{5}}{10}}, y_{j-\frac{1}{2}+\frac{\sqrt{5}}{10}}), \quad G^2_{ij} = (x_i, y_{j-\frac{1}{2}+\frac{\sqrt{5}}{10}}), \quad G^3_{ij} = (x_{i+\frac{1}{2}-\frac{\sqrt{5}}{10}}, y_{j-\frac{1}{2}+\frac{\sqrt{5}}{10}}),\\
&G^4_{ij} = (x_{i-\frac{1}{2}+\frac{\sqrt{5}}{10}}, y_j), \quad G^5_{ij} = (x_i, y_j), \quad G^6_{ij} = (x_{i+\frac{1}{2}-\frac{\sqrt{5}}{10}}, y_j),\\
&G^7_{ij} = (x_{i-\frac{1}{2}+\frac{\sqrt{5}}{10}}, y_{j+\frac{1}{2}-\frac{\sqrt{5}}{10}}), \quad G^8_{ij} = (x_i, y_{j+\frac{1}{2}-\frac{\sqrt{5}}{10}}), \quad G^9_{ij} = (x_{i+\frac{1}{2}-\frac{\sqrt{5}}{10}}, y_{j+\frac{1}{2}-\frac{\sqrt{5}}{10}}).
\end{aligned}$$

Note that the points $G^1_{ij}, \cdots, G^9_{ij}$ are interior points with respect to the primal mesh, see Fig. 3.

According to [27], we apply a dimension-by-dimension approach and adopt the procedure in step 5 of Sect. 2.1 in each direction. Along the x direction,

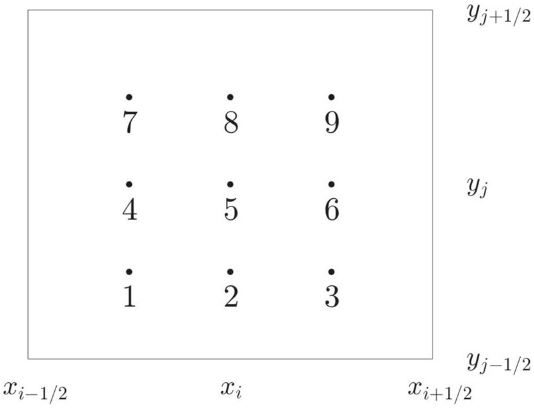


Fig. 3 The labels of two-dimensional Gauss-Lobatto points within I_{ij}

- We reconstruct $f_x(u(G_{ij}^k)),\ k = 1, 2, 3$ from

$$f(u(Q)), \quad Q = G_{i-1,j}^2, G_{i-1,j}^3, G_{ij}^1, G_{ij}^2, G_{ij}^3, G_{i+1,j}^1, G_{i+1,j}^2.$$

- We reconstruct $f_x(u(G_{ij}^k)), k = 4, 5, 6$ from

$$f(u(Q)), \quad Q = G_{i-1,j}^5, G_{i-1,j}^6, G_{ij}^4, G_{ij}^5, G_{ij}^6, G_{i+1,j}^4, G_{i+1,j}^5.$$

- We reconstruct $f_x(u(G_{ij}^k)), k = 7, 8, 9$ from

$$f(u(Q)), \quad Q = G_{i-1,j}^8, G_{i-1,j}^9, G_{ij}^7, G_{ij}^8, G_{ij}^9, G_{i+1,j}^7, G_{i+1,j}^8.$$

Similarly, along the y direction,

- We reconstruct $g_y(u(G_{ij}^k)), k = 1, 4, 7$ from

$$g(u(Q)), \quad Q = G_{i,j-1}^4, G_{i,j-1}^7, G_{ij}^1, G_{ij}^4, G_{ij}^7, G_{i,j+1}^1, G_{i,j+1}^4.$$

- We reconstruct $g_y(u(G_{ij}^k)), k = 2, 5, 8$ from

$$g(u(Q)), \quad Q = G_{i,j-1}^5, G_{i,j-1}^8, G_{ij}^2, G_{ij}^5, G_{ij}^8, G_{i,j+1}^2, G_{i,j+1}^5.$$

- We reconstruct $g_y(u(G_{ij}^k)), k = 3, 6, 9$ from

$$g(u(Q)), \quad Q = G_{i,j-1}^6, G_{i,j-1}^9, G_{ij}^3, G_{ij}^6, G_{ij}^9, G_{i,j+1}^3, G_{i,j+1}^6.$$

Remark 2 For the system, the HWENO reconstructions in steps 2–4 are performed in the local characteristic directions to avoid the oscillations near discontinuities. In the two-dimensional case, we implement the HWENO reconstructions along the two characteristic directions provided by $f(u)$ and $g(u)$, respectively. Then we take the average of these two reconstructions as our final approximation. For more details about this procedure, one can refer to [26].

3 Numerical Examples

In this section, we present some typical experiments to test the performance of finite volume central schemes with the fifth-order HWENO reconstructions and sixth-order multi-resolution HWENO reconstructions which are termed as C-HWENO5 [27] and C-HWENO6, respectively. For the smooth problems, we also show the results of C-HWENO6-M5 which means the first-order moments of all cells are modified by the fifth-order HWENO method. The fourth-order NCE-RK method is adopted as time discretization. The CFL number C_{cfl} is taken as 0.1 and 0.2 for the smooth and non-smooth problems, respectively. For the smooth problems, we choose a smaller CFL number to ensure the spatial errors dominate. The time step Δt is taken as

$$\Delta t = \frac{C_{cfl}\Delta x}{\max|f'(u)|}$$

and

$$\Delta t = \frac{C_{cfl}}{\max|f'(u)|/\Delta x + \max|g'(u)|/\Delta y}$$

in one- and two-dimensional scalar case, respectively. For the system case, $f'(u)$ and $g'(u)$ are replaced by the eigenvalue of the Jacobian of $f(u)$ and $g(u)$, with the largest absolute value.

3.1 Smooth Problems

Example 3.1 We solve the one-dimensional Burgers' equation,

$$u_t + \left(\frac{u^2}{2}\right)_x = 0. \tag{3.1}$$

The initial condition is $u(x,0) = 0.5 + \sin(\pi x)$ with a 2-periodic boundary condition. When final time is $T = 0.5/\pi$, the solution is still smooth, and the corresponding errors and numerical orders of accuracy by C-HWENO5, C-HWENO6-M5 and C-HWENO6 schemes are listed in Table 1. We can observe that all schemes achieve their designed orders of accuracy. Compared with the results of C-HWENO5 scheme proposed in [27], our new scheme C-HWENO6 has smaller errors and higher orders of accuracy with the same computational grid.

Example 3.2 We solve the one-dimensional nonlinear system of Euler equations,

$$U_t + f(U)_x = 0, \tag{3.2}$$

where

$$U = (\rho, \rho\mu, E)^T, \quad f(U) = (\rho\mu, \rho\mu^2 + p, \mu(E+p))^T.$$

Here ρ is the density, μ is the velocity, E is the total energy and p is the pressure related to the total energy by $E = \frac{p}{\gamma-1} + \frac{1}{2}\rho\mu^2$. We consider a non-physical accuracy test with $\gamma = 3$. The initial conditions are set to be

$$\rho(x,0) = \frac{1+0.2\sin(x)}{2\sqrt{3}}, \quad \mu(x,0) = \sqrt{3}\rho(x,0), \quad p(x,0) = \rho^3(x,0),$$

with a 2π-periodic boundary condition. Under the above conditions, $2\sqrt{3}\rho(x,t)$ can be proved to be the exact solution of Burgers' equation,

$$\begin{cases} u_t + \left(\frac{u^2}{2}\right)_x = 0, \\ u(x,0) = 1 + 0.2\sin(x), \end{cases}$$

Table 1 $u_t + \left(\frac{u^2}{2}\right)_x = 0$, with $u(x, 0) = 0.5 + \sin(\pi x)$ and a 2-periodic boundary condition.

	C-HWENO5				C-HWENO6-M5			
N	L_1 error	Order	L_∞ error	Order	L_1 error	Order	L_∞ error	Order
60	1.59e−05		1.47e−04		5.49e−06		6.34e−05	
80	4.29e−06	4.55	3.67e−05	4.82	1.48e−06	4.56	1.74e−05	4.49
100	1.57e−06	4.50	1.22e−05	4.94	5.20e−07	4.69	6.22e−06	4.61
120	6.73e−07	4.65	5.18e−06	4.70	2.16e−07	4.82	2.74e−06	4.50
140	3.26e−07	4.70	2.94e−06	3.67	1.03e−07	4.80	1.34e−06	4.64
160	1.74e−07	4.70	1.73e−06	3.97	5.44e−08	4.78	7.06e−07	4.80
180	9.86e−08	4.82	1.03e−06	4.40	3.08e−08	4.83	3.99e−07	4.84
200	5.87e−08	4.92	6.96e−07	3.72	1.84e−08	4.89	2.38e−07	4.90

	C-HWENO6			
N	L_1 error	Order	L_∞ error	Order
60	2.86e−08		4.84e−07	
80	4.72e−09	6.26	7.45e−08	6.50
100	1.10e−09	6.53	1.81e−08	6.34
120	3.29e−10	6.62	5.22e−09	6.82
140	1.16e−10	6.76	1.88e−09	6.62
160	4.86e−11	6.52	7.93e−10	6.46
180	2.22e−11	6.65	3.43e−10	7.12
200	1.12e−11	6.49	1.70e−10	6.66

$T = 0.5/\pi$. C-HWENO5, C-HWENO6-M5 and C-HWENO6 schemes. L^1 and L^∞ errors and orders of accuracy

and the corresponding velocity $\mu(x, t)$ and pressure $p(x, t)$ satisfy

$$\mu(x, t) = \sqrt{3}\rho(x, t), \quad p(x, t) = \rho^3(x, t).$$

We compute the solution up to $T = 3$. For C-HWENO5, C-HWENO6-M5 and C-HWENO6 schemes, the L_1 and L_∞ errors and numerical orders of accuracy of density ρ are listed in Table 2. All schemes achieve their designed orders of accuracy. Again, the numerical solutions of C-HWENO6 scheme have smaller errors and higher orders of accuracy than that of C-HWENO5 scheme on the same computational mesh.

Example 3.3 We solve the two-dimensional Burgers' equation,

$$u_t + \left(\frac{u^2}{2}\right)_x + \left(\frac{u^2}{2}\right)_y = 0. \tag{3.3}$$

The initial condition is $u(x, y, 0) = 0.5 + \sin(\pi(x + y)/2)$ with a 4-periodic boundary condition in each direction. When $T = 0.5/\pi$, the solution is still smooth, and the corresponding errors and numerical orders of accuracy by C-HWENO5, C-HWENO6-M5 and C-HWENO6 schemes are listed in Table 3. The results are similar to one-dimensional case and our new scheme performs better than C-HWENO5 scheme on the same computational mesh.

Table 2 Euler equations, with $\rho(x,0)=\frac{1+0.2\sin(x)}{2\sqrt{3}}$, $\mu(x,0)=\sqrt{3}\rho(x,0)$, $p(x,0)=\rho^3(x,0)$, and a 2π-periodic boundary condition.

	C-HWENO5				C-HWENO6-M5			
N	L_1 error	Order	L_∞ error	Order	L_1 error	Order	L_∞ error	Order
60	1.88e−05		2.52e−04		6.92e−06		1.09e−04	
80	3.85e−06	5.51	5.94e−05	5.02	1.93e−06	4.44	3.24e−05	4.22
100	1.09e−06	5.66	1.75e−05	5.48	6.98e−07	4.56	1.31e−05	4.06
120	4.25e−07	5.17	8.00e−06	4.29	2.85e−07	4.91	5.92e−06	4.36
140	2.01e−07	4.86	4.16e−06	4.24	1.41e−07	4.57	2.84e−06	4.77
160	1.07e−07	4.72	2.19e−06	4.80	7.22e−08	5.01	1.51e−06	4.73
180	5.93e−08	5.01	1.23e−06	4.90	4.10e−08	4.80	8.77e−07	4.61
200	3.52e−08	4.95	7.67e−07	4.48	2.42e−08	5.00	5.22e−07	4.92

	C-HWENO6			
N	L_1 error	Order	L_∞ error	Order
60	7.64e−08		1.66e−06	
80	1.57e−08	5.50	2.54e−07	6.53
100	3.36e−09	6.91	8.15e−08	5.09
120	1.05e−09	6.38	2.55e−08	6.37
140	3.72e−10	6.73	8.89e−09	6.84
160	1.53e−10	6.65	3.67e−09	6.63
180	6.85e−11	6.82	1.75e−09	6.29
200	3.34e−11	6.82	8.60e−10	6.74

$T=3$. C-HWENO5, C-HWENO6-M5 and C-HWENO6 schemes. L^1 and L^∞ errors and orders of accuracy of density ρ

Example 3.4 We solve the two-dimensional nonlinear system of Euler equations,

$$U_t+f(U)_x+g(U)_y=0, \tag{3.4}$$

where

$$\begin{aligned}&U=(\rho,\rho\mu,\rho\nu,E)^T,\quad f(U)=(\rho\mu,\rho\mu^2+p,\rho\mu\nu,\mu(E+p))^T,\\&g(U)=(\rho\nu,\rho\mu\nu,\rho\nu^2+p,\nu(E+p))^T.\end{aligned}$$

Here ρ is the density, $(\mu,\nu)^T$ is the velocity, E is the total energy and p is the pressure related to the total energy by $E=\frac{p}{\gamma-1}+\frac{1}{2}\rho(\mu^2+\nu^2)$. Here we take $\gamma=3$. The initial conditions are set to be

$$\begin{aligned}&\rho(x,y,0)=\frac{1+0.2\sin(\frac{x+y}{2})}{\sqrt{6}},\quad \mu(x,y,0)=\nu(x,y,0)=\sqrt{\frac{3}{2}}\rho(x,y,0),\\&p(x,y,0)=\rho^3(x,y,0),\end{aligned}$$

Table 3 $u_t + \left(\frac{u^2}{2}\right)_x + \left(\frac{u^2}{2}\right)_y = 0$, with $u(x, y, 0) = 0.5 + \sin(\pi(x + y)/2)$ and a 4-periodic boundary condition.

	C-HWENO5				C-HWENO6-M5			
$N_x \times N_y$	L_1 error	Order	L_∞ error	Order	L_1 error	Order	L_∞ error	Order
60×60	3.09e−05		2.77e−04		1.25e−05		1.45e−04	
80×80	8.29e−06	4.57	7.48e−05	4.55	3.18e−06	4.76	4.00e−05	4.48
100×100	2.94e−06	4.65	2.56e−05	4.81	1.08e−06	4.84	1.42e−05	4.64
120×120	1.27e−06	4.60	1.04e−05	4.94	4.55e−07	4.74	5.90e−06	4.82
140×140	6.16e−07	4.69	4.82e−06	4.99	2.17e−07	4.80	2.78e−06	4.88
160×160	3.23e−07	4.83	2.64e−06	4.51	1.13e−07	4.89	1.45e−06	4.87
180×180	1.78e−07	5.06	1.55e−06	4.52	6.30e−08	4.96	8.30e−07	4.74
200×200	1.06e−07	4.92	9.57e−07	4.58	3.77e−08	4.87	4.99e−07	4.83

	C-HWENO6			
$N_x \times N_y$	L_1 error	Order	L_∞ error	Order
60×60	1.51e−06		1.90e−05	
80×80	2.56e−07	6.17	3.89e−06	5.51
100×100	5.87e−08	6.60	8.58e−07	6.77
120×120	1.75e−08	6.64	2.93e−07	5.89
140×140	6.29e−09	6.64	1.02e−07	6.85
160×160	2.54e−09	6.79	4.33e−08	6.42
180×180	1.15e−09	6.73	1.95e−08	6.77
200×200	5.58e−10	6.86	9.62e−09	6.71

$T = 0.5/\pi$. C-HWENO5, C-HWENO6-M5 and C-HWENO6 schemes. L^1 and L^∞ errors and orders of accuracy

with a 4π-periodic boundary condition in each direction. Under the above conditions, $\sqrt{6}\rho(x, y, t)$ can be proved to be the exact solution of Burgers' equation,

$$\begin{cases} u_t + \left(\frac{u^2}{2}\right)_x + \left(\frac{u^2}{2}\right)_y = 0, \\ u(x, y, 0) = 1 + 0.2\sin(\frac{x+y}{2}), \end{cases}$$

and the corresponding velocity $\mu(x, y, t)$, $\nu(x, y, t)$ and pressure $p(x, y, t)$ satisfy

$$\mu(x, y, t) = \nu(x, y, t) = \sqrt{\frac{3}{2}}\rho(x, y, t), \quad p(x, y, t) = \rho^3(x, y, t).$$

The solutions are computed up to $T = 3$. For C-HWENO5, C-HWENO6-M5 and C-HWENO6 schemes, the L_1 and L_∞ error and numerical orders of accuracy of density ρ are listed in Table 4. Again, The results are similar to one-dimensional case and our new scheme performs better than C-HWENO5 scheme on the same computational mesh.

We also compare the CPU time of the WENO and HWENO schemes when they are applied to Examples 3.1–3.4. On the same grid, the HWENO scheme has double the DoFs in 1D and triple the DoFs in 2D than the WENO scheme. Therefore, we calculate the CPU time of the HWENO scheme, and the WENO scheme with double mesh elements in 1D and triple mesh elements in 2D ($\sqrt{3}$ times in each direction), respectively. The WENO-MR and

Table 4 Euler equations, with $\rho(x, y, 0) = (1+0.2\sin(\frac{x+y}{2}))/\sqrt{6}$, $\mu(x, y, 0) = \nu(x, y, 0) = \sqrt{\frac{3}{2}}\rho(x, y, 0)$, $p(x, y, 0) = \rho^3(x, y, 0)$, and a 4π-periodic boundary condition in each direction.

	C-HWENO5				C-HWENO6-M5			
$N_x \times N_y$	L_1 error	Order	L_∞ error	Order	L_1 error	Order	L_∞ error	Order
60×60	4.92e−05		6.74e−04		2.58e−05		3.32e−04	
80×80	1.19e−05	4.63	1.74e−04	4.71	7.32e−06	4.38	1.21e−04	3.51
100×100	3.56e−06	5.41	6.03e−05	4.75	2.60e−06	4.64	4.98e−05	3.98
120×120	1.36e−06	5.28	2.58e−05	4.66	1.11e−06	4.67	2.12e−05	4.68
140×140	6.61e−07	4.68	1.26e−05	4.65	5.23e−07	4.88	1.09e−05	4.32
160×160	3.40e−07	4.98	7.09e−06	4.30	2.76e−07	4.79	5.89e−06	4.61
180×180	1.93e−07	4.81	4.15e−06	4.55	1.52e−07	5.06	3.28e−06	4.97
200×200	1.14e−07	5.00	2.47e−06	4.92	9.13e−08	4.84	1.95e−06	4.94

	C-HWENO6			
$N_x \times N_y$	L_1 error	Order	L_∞ error	Order
60×60	5.15e−06		5.73e−05	
80×80	1.01e−06	5.66	1.91e−05	3.82
100×100	2.79e−07	5.77	5.60e−06	5.50
120×120	8.35e−08	6.62	1.79e−06	6.26
140×140	3.25e−08	6.12	7.87e−07	5.33
160×160	1.29e−08	6.92	3.33e−07	6.44
180×180	6.10e−09	6.36	1.45e−07	7.06
200×200	2.95e−09	6.90	7.74e−08	5.96

$T = 3$. C-HWENO5, C-HWENO6-M5 and C-HWENO6 schemes. L^1 and L^∞ errors and orders of accuracy of density ρ

HWENO-MR reconstructions are applied in our comparison. The results are presented in Tables 5, 6, 7, 8. We can observe that CPU time of the HWENO schemes is less than that of WENO schemes with the same DoFs, and the central schemes are more efficient than the upwind schemes.

In addition, we also show the plots of numerical errors vs. CPU time by the WENO and HWENO schemes in Figs. 4, 5, 6, 7. One can observe from the results that the HWENO schemes have higher efficiency than the WENO schemes, and the central schemes outperform the upwind schemes. In summary, our central HWENO scheme is more effective.

3.2 Non-smooth Problems

We now test the resolution and non-oscillatory property of the proposed methods when solving non-smooth problems which contain shock, rarefaction or contact discontinuity.

Example 3.5 We consider the same one-dimensional nonlinear Burgers' Eq. (3.1), with the same initial and boundary conditions in example 3.1. We compute the solution up to $T = 1.5/\pi$ after a shock forms. The solutions of C-HWENO5 and C-HWENO6 schemes with $N = 80$ mesh elements, as well as the exact solution are shown in Fig. 8. We can observe that both solutions are non-oscillatory near the shock.

Table 5 CPU time (in seconds) of the Up-WENO5, C-WENO5, Up-HWENO6 and C-HWENO6 schemes for Example 3.1

N	Up-WENO5	C-WENO5	N	Up-HWENO6	C-HWENO6
120	0.13	0.05	60	0.08	0.04
160	0.24	0.09	80	0.17	0.07
200	0.41	0.15	100	0.31	0.12
240	0.60	0.22	120	0.53	0.19
280	0.85	0.35	140	0.87	0.28
320	1.14	0.50	160	1.13	0.38
360	1.54	0.61	180	1.30	0.52
400	2.15	0.82	200	1.69	0.68

Table 6 CPU time (in seconds) of the Up-WENO5, C-WENO5, Up-HWENO6 and C-HWENO6 schemes for Example 3.2

N	Up-WENO5	C-WENO5	N	Up-HWENO6	C-HWENO6
120	0.95	0.23	60	0.30	0.12
160	1.87	0.42	80	0.56	0.21
200	2.78	0.68	100	0.99	0.36
240	4.08	0.98	120	1.47	0.57
280	5.68	1.37	140	2.17	0.78
320	7.80	1.85	160	3.08	1.09
360	9.96	2.37	180	4.03	1.46
400	12.25	2.92	200	4.88	1.88

Table 7 CPU time (in seconds) of the Up-WENO5, C-WENO5, Up-HWENO6 and C-HWENO6 schemes for Example 3.3

N	Up-WENO5	C-WENO5	N	Up-HWENO6	C-HWENO6
$60\sqrt{3} \times 60\sqrt{3}$	10.77	6.03	60×60	6.00	2.26
$80\sqrt{3} \times 80\sqrt{3}$	26.69	14.14	80×80	15.28	5.26
$100\sqrt{3} \times 100\sqrt{3}$	61.45	28.98	100×100	32.72	11.28
$120\sqrt{3} \times 120\sqrt{3}$	120.97	52.59	120×120	81.77	21.21
$140\sqrt{3} \times 140\sqrt{3}$	192.00	98.54	140×140	115.97	39.41
$160\sqrt{3} \times 160\sqrt{3}$	308.05	163.35	160×160	180.75	67.49
$180\sqrt{3} \times 180\sqrt{3}$	448.29	219.21	180×180	282.43	102.00
$200\sqrt{3} \times 200\sqrt{3}$	650.23	308.36	200×200	389.47	146.61

Example 3.6 We consider the one-dimensional nonlinear non-convex scalar Buckley-Leverett problem

$$u_t + \left(\frac{4u^2}{4u^2 + (1-u)^2}\right)_x = 0,$$

Table 8 CPU time (in seconds) of the Up-WENO5, C-WENO5, Up-HWENO6 and C-HWENO6 schemes for Example 3.4

N	Up-WENO5	C-WENO5	N	Up-HWENO6	C-HWENO6
$60\sqrt{3} \times 60\sqrt{3}$	87.74	36.70	60×60	62.17	22.45
$80\sqrt{3} \times 80\sqrt{3}$	241.78	94.04	80×80	169.00	59.83
$100\sqrt{3} \times 100\sqrt{3}$	537.52	205.46	100×100	370.72	129.43
$120\sqrt{3} \times 120\sqrt{3}$	972.42	401.89	120×120	713.43	227.21
$140\sqrt{3} \times 140\sqrt{3}$	1653.34	666.14	140×140	1226.73	402.95
$160\sqrt{3} \times 160\sqrt{3}$	2695.56	1056.65	160×160	2169.54	606.44
$180\sqrt{3} \times 180\sqrt{3}$	3876.92	1555.10	180×180	3402.54	928.92
$200\sqrt{3} \times 200\sqrt{3}$	5631.28	2264.86	200×200	4951.52	1342.99

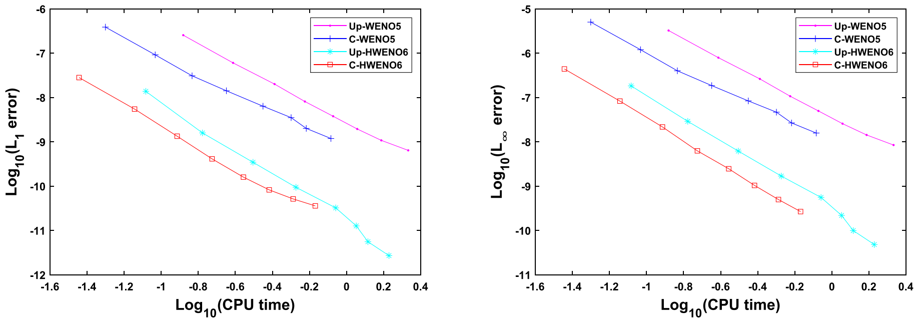


Fig. 4 Numerical errors vs. CPU time of the WENO and HWENO schemes for Example 3.1

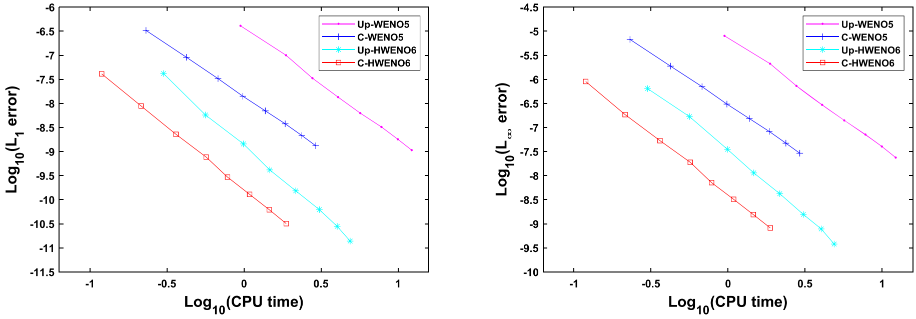


Fig. 5 Numerical errors vs. CPU time of the WENO and HWENO schemes for Example 3.2

with the initial condition: $u = 1$ when $-\frac{1}{2} \leq x \leq 0$, and $u = 0$ elsewhere, and constant boundary conditions. The computational domain is $[-1, 1]$, and the final time is $T = 0.4$. The exact solution includes rarefaction, shock wave and contact discontinuity. Note that we can not obtain the correct entropy solution of this problem with some high-order methods. In Fig. 9, we show the solutions of C-HWENO5 and C-HWENO6 schemes with a $N = 80$ mesh, together with the exact solution. One can see that both schemes can capture the major features of the entropy solution well with comparable resolution.

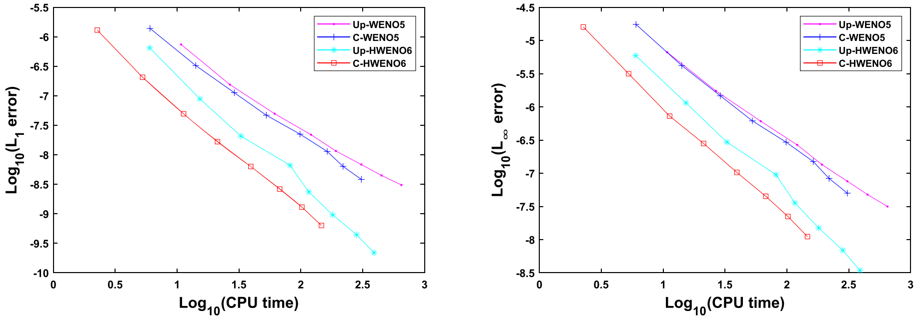


Fig. 6 Numerical errors vs. CPU time of the WENO and HWENO schemes for Example 3.3

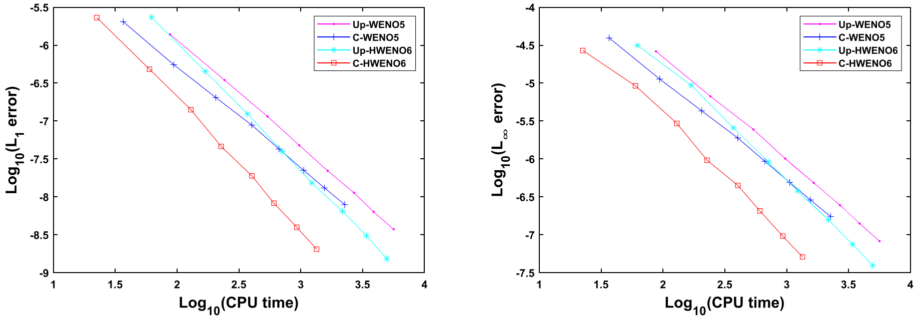


Fig. 7 Numerical errors vs. CPU time of the WENO and HWENO schemes for Example 3.4

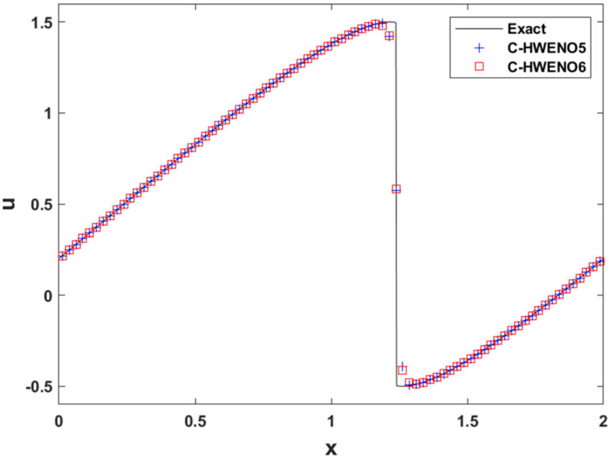
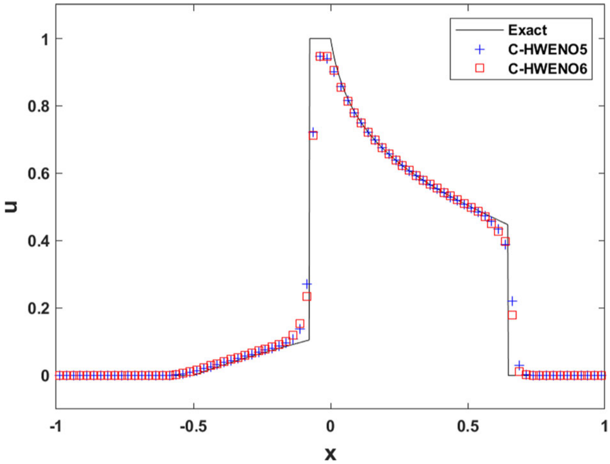


Fig. 8 Burgers' equation in one dimension. $T = 1.5/\pi$ and $N = 80$

Fig. 9 The Buckley-Leverett problem. $T = 0.4$ and $N = 80$



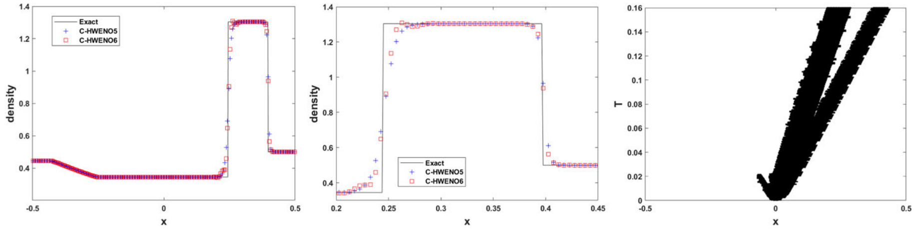


Fig. 10 The Lax problem. $T = 0.16$ and $N = 200$. Density ρ (left), density zoomed in (middle), time history of the troubled-cells in C-HWENO6 (right)

Example 3.7 We solve the one-dimensional Euler equations (3.2) with a Riemann initial condition for the Lax problem

$$\begin{aligned} (\rho, \mu, p) &= (0.445, 0.698, 3.528), \quad & x \le 0, \\ (\rho, \mu, p) &= (0.5, 0, 0.571), \quad & x > 0. \end{aligned}$$

The inflow/outflow boundary conditions are applied to the left/right ends. The computational domain is $[-0.5, 0.5]$, with the final time $T = 0.16$. In Fig. 10, we plot the numerical density ρ by C-HWENO5 and C-HWENO6 schemes on a $N = 200$ mesh, together with the exact solution. We further show the zoom-in picture of the density and the time history of the troubled-cells in C-HWENO6 scheme. One can see that our new scheme gives better resolution near the discontinuities than C-HWENO5 scheme. There are some oscillations around $x = 0.25$, this may due to the modification of the first-order moments which can not control oscillation very well. We will further study the strategy of modification in future research and expect a better method.

Example 3.8 We solve the shock density wave interaction problem, also known as Shu-Osher problem, which describes the interaction between shock and entropy waves. This example is modeled by the one-dimensional Euler equations (3.2) with a moving Mach-3 shock interacting with sine waves in density, and the initial condition is set to be

$$(\rho, \mu, p) = (3.857143, 2.629369, 10.333333), \quad x < -4,$$

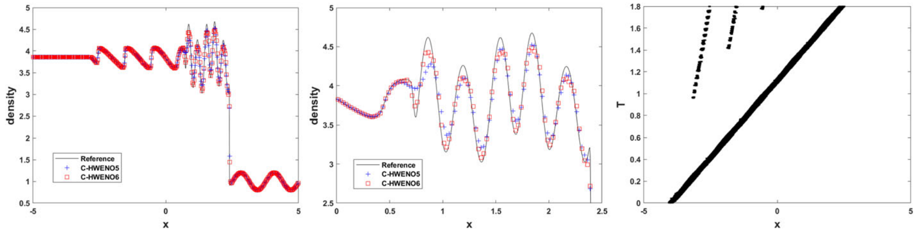


Fig. 11 The shock density wave interaction problem. $T = 1.8$ and $N = 400$. Density ρ (left), density zoomed in (middle), time history of the troubled-cells in C-HWENO6 (right)

$$(\rho, \mu, p) = (1 + \tau \sin 5x, 0, 1), \qquad\qquad x \geq -4,$$

where $\tau = 0.2$, and the boundary conditions are taken to be the same as the initial data. The computational domain is $[-5, 5]$, with the final time $T = 1.8$. In Fig. 11, we plot the numerical density ρ by C-HWENO5 and C-HWENO6 schemes on a $N = 400$ mesh, as well as a reference solution obtained by the fifth-order finite difference WENO scheme [7] with 16000 grid points. The solution contains both shocks and complex smooth structures, therefore we further show the zoom-in picture of the oscillatory region of the density and the time history of the troubled-cells in C-HWENO6 scheme. We can observe that both schemes have non-oscillatory property to capture discontinuities and C-HWENO6 scheme has better resolution to resolve smooth features especially when x is between 0.5 and 1.4.

Example 3.9 We solve the interaction of blast waves, which is modeled by the one-dimensional Euler equations (3.2) with the initial condition,

$$\begin{aligned}
(\rho, \mu, p) &= (1, 0, 1000), \quad 0 \leq x < 0.1,\\
(\rho, \mu, p) &= (1, 0, 0.01), \quad 0.1 \leq x < 0.9,\\
(\rho, \mu, p) &= (1, 0, 100), \quad 0.9 \leq x \leq 1,
\end{aligned}$$

and reflecting boundary conditions applied to both ends. The computational domain is $[0, 1]$, with the final time $T = 0.038$. In Fig. 12, we report the numerical density ρ by C-HWENO5 and C-HWENO6 schemes on a $N = 800$ mesh, as well as a reference solution obtained by the fifth-order finite difference WENO scheme [7] with 16000 grid points. We further show the zoom-in picture of the density and the time history of the troubled-cells in C-HWENO6 scheme. One can see that both schemes give good resolution to resolve the complicated structure of this problem.

There is a zeroth degree polynomial in the reconstruction procedure. For the problems with peak, the nonlinear weights of zeroth degree polynomial is relatively high, and high degree polynomials are hardly used. Therefore, the reconstructed polynomials tends to the low degree polynomials, and the scheme has less amplitude near peak for the Shu-Osher problem and the blast waves interaction problem.

Example 3.10 We solve the same two-dimensional Burgers' equation (3.3) with the same initial and boundary conditions in example 3.3. We compute up to $T = 1.5/\pi$ after a shock forms. In Fig. 13, we show a slice of the numerical solutions at $x = y$ by C-HWENO5 and C-HWENO6 schemes with 80×80 mesh elements, as well as the exact solution. The surfaces of solutions are also given. For this problem, both schemes give equally good non-oscillatory shock transitions.

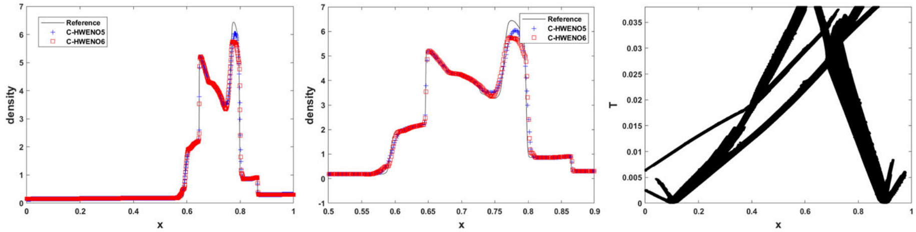


Fig. 12 The interaction of blast waves problem. $T = 0.038$ and $N = 800$. Density ρ (left), density zoomed in (middle), time history of the troubled-cells in C-HWENO6 (right)

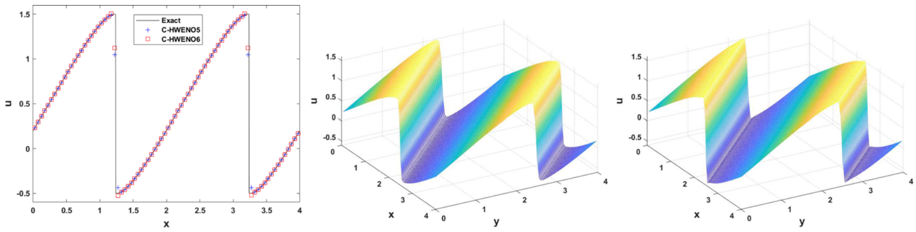


Fig. 13 Burgers' equation in two dimensions. $T = 1.5/\pi$ and $N_x \times N_y = 80 \times 80$. A slice of the solutions at $x = y$ (left), the surfaces of solutions computed by C-HWENO5 (middle) and C-HWENO6 (right)

Example 3.11 We solve the example of double Mach reflection, which is modeled by the two-dimensional Euler equations (3.4). The computational domain is $[0, 4] \times [0, 1]$ with a reflecting wall lying at the bottom, starting from $x = \frac{1}{6}$. Initially a right-moving Mach-10 shock is positioned at $x = \frac{1}{6}$, $y = 0$, and makes a 60° angle with x-axis. For the bottom boundary, the exact post-shock condition is used for the part from $x = 0$ to $x = \frac{1}{6}$, and the reflective boundary condition is imposed for the rest. For the top boundary, the flow values are set to describe the exact motion of the Mach-10 shock. Post-shock/pre-shock conditions are applied to the left/right boundaries. The solutions are computed up to $T = 0.2$. The density contour plots in the region $[0, 3] \times [0, 1]$ by C-HWENO5 and C-HWENO6 schemes with 1920×480 mesh elements are presented in Fig. 14. Each contour plot has 30 contour lines with density ranging from 1.5 to 22.7. We further show the locations of the troubled-cells at the final time in C-HWENO6 scheme. One can see that our new scheme gives better resolution to capture the fine local structures than C-HWENO5 scheme.

Example 3.12 We solve the problem for a Mach-3 wind tunnel with a step, which is modeled by the two-dimensional Euler equations (3.4). The length of the wind tunnel is 3 units and the width is 1 unit, i.e. the computational domain is $[0, 3] \times [0, 1]$. The height of the step is 0.2 units, and is located 0.6 length units from the left 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, and inflow/outflow boundary conditions are applied at the entrance/exit. The corner of the step is a singular point and is treated in the same way as in [28] based on the assumption of a nearly steady flow in the region around the corner. The schemes are run up to $T = 4$. The density contour plots by C-HWENO5 and C-HWENO6 schemes with 960×320 mesh elements are shown in Fig. 15. Each contour plot has 30 contour lines with density ranging from 0.32 to 6.15. We further show the locations of the troubled-cells at the final time in C-HWENO6 scheme. One can see that our new scheme gives better resolution to capture the fine local structures than C-HWENO5 scheme for this problem.

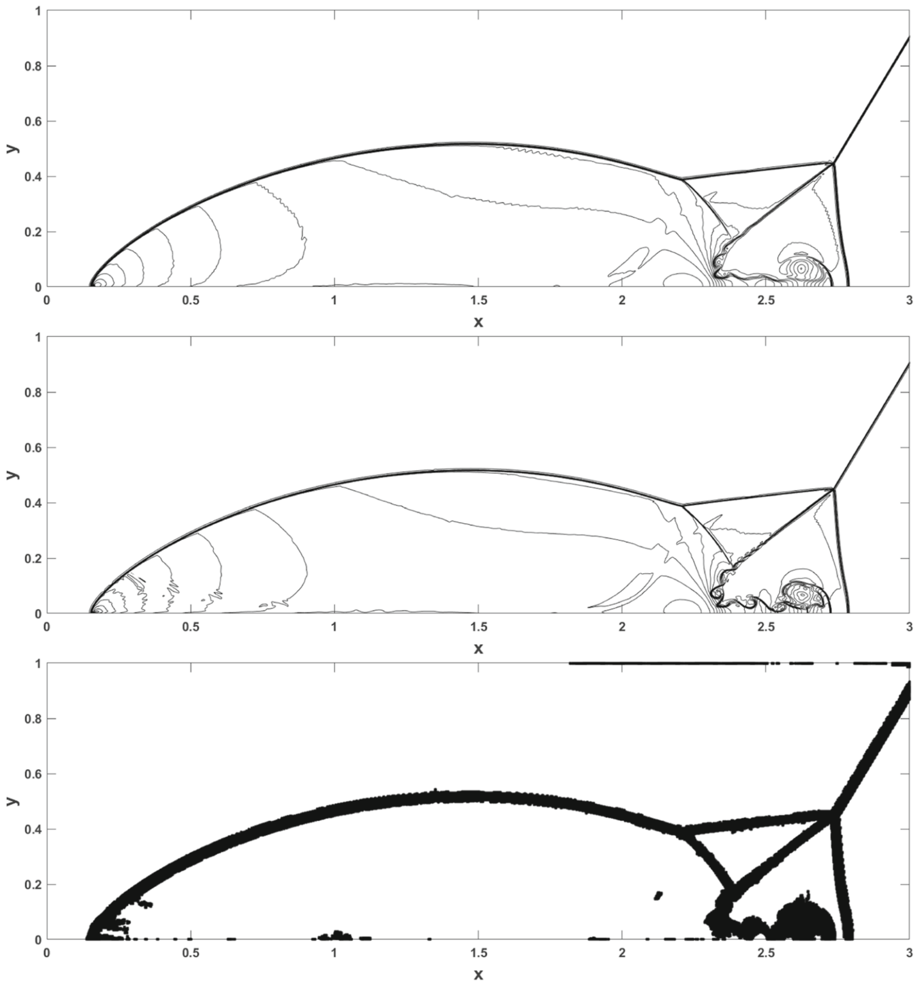


Fig. 14 Double Mach reflection problem. $T = 0.2$ and $N_x \times N_y = 1920 \times 480$. C-HWENO5 (top), C-HWENO6 (middle), the locations of the troubled-cells at the final time in C-HWENO6 (bottom). 30 equally spaced density contours from 1.5 to 22.7

Finally, we would like to compare the performance of the WENO and HWENO schemes in the simulations of the double Mach reflection problem and forward step problem. The WENO-MR, WENO-Z and HWENO-MR reconstructions are applied in our comparison. In order to maintain the same degree of freedom, we show the CPU time and contour plots by the HWENO schemes and the WENO schemes with triple mesh elements ($\sqrt{3}$ times in each direction). For the WENO methods, we use 1920×480 and 960×320 mesh elements for the double Mach reflection problem and forward step problem, respectively. For the HWENO methods, we use 1108×277 and 555×185 mesh elements for the double Mach reflection problem and forward step problem, respectively. For the sake of fairness, all the methods are implemented without the troubled-cell indicator, and the LF flux is used in all upwind methods. We apply the fourth-order RK method and NCE-RK method for the upwind

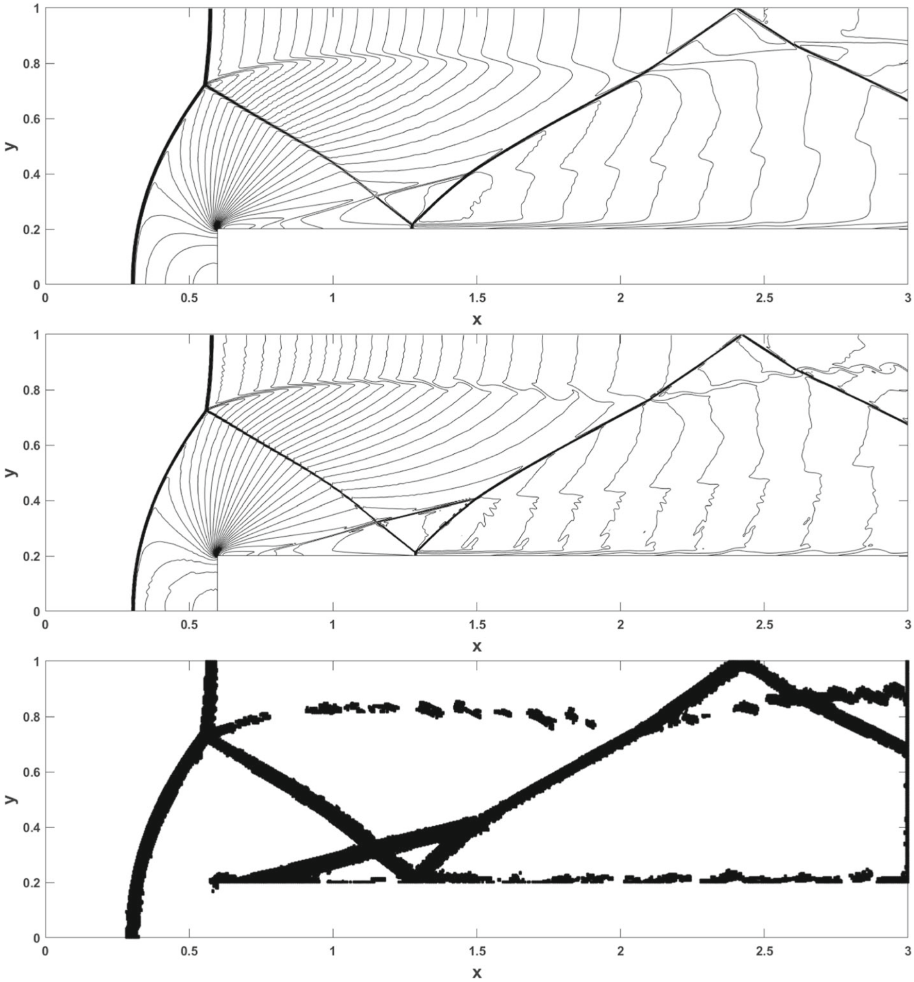


Fig. 15 Forward step problem. $T = 4$ and $N_x \times N_y = 960 \times 320$. C-HWENO5 (top), C-HWENO6 (middle), the locations of the troubled-cells at the final time in C-HWENO6 (bottom). 30 equally spaced density contours from 0.32 to 6.15

and central scheme, respectively. For the upwind HWENO scheme, we use the HWENO procedure at page 4 in [32] to modify the first-order moments.

The CPU time of the WENO and HWENO schemes are presented in Table 9, and the CFL number for the central and upwind schemes is 0.2 and 0.6, respectively. Although the CFL number for the upwind scheme is three times of that for the central scheme, the CPU time of the central WENO scheme is only slightly bigger than that of the upwind WENO scheme. This means that each single step of the central WENO scheme is much cheaper than the upwind WENO scheme in CPU time. For the HWENO scheme, the central scheme even has less CPU time than the upwind scheme due to the fact that less characteristic decompositions are used in the central scheme.

The contour plots are shown in Figs. 16, 17, 18, 19. We can observe that the WENO schemes (with more mesh elements) have better resolution than the HWENO schemes. For

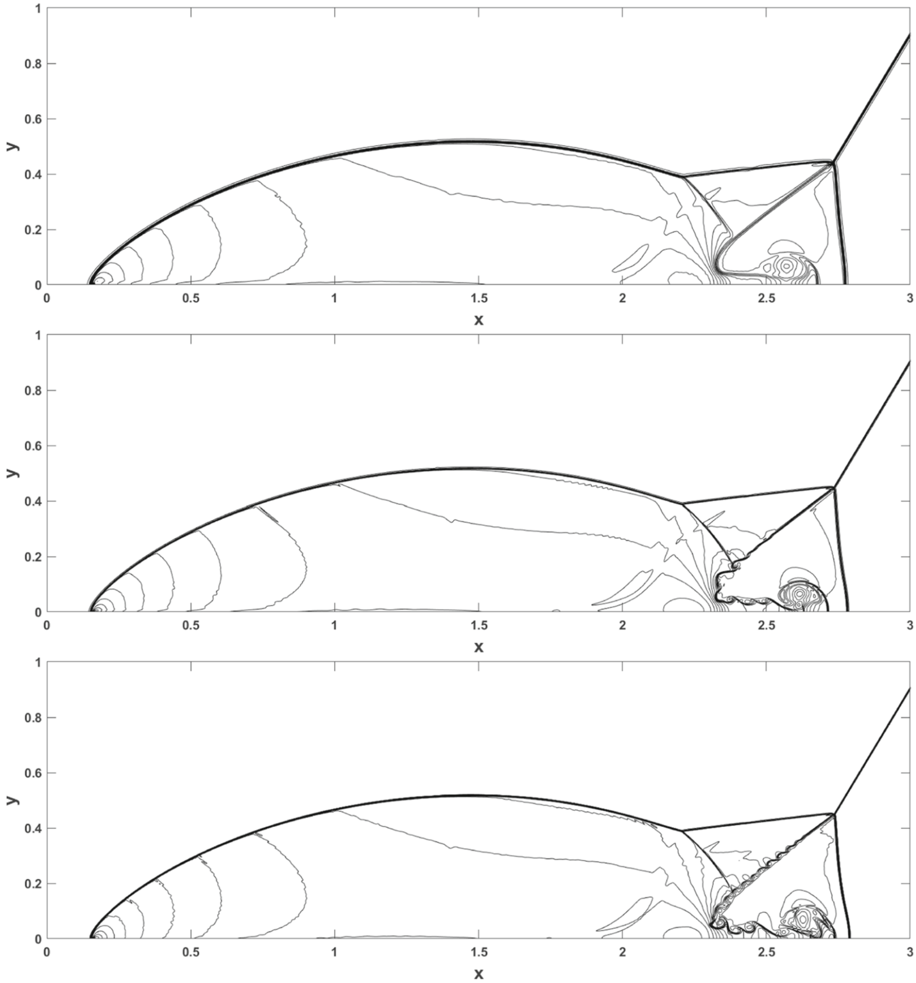


Fig. 16 Double Mach reflection problem. 1108×277 and 1920×480 mesh elements for the HWENO and WENO schemes, respectively. $T = 0.2$. From top to bottom: Up-HWENO6, Up-WENO5-MR, Up-WENO5-Z. 30 equally spaced density contours from 1.5 to 22.7

the double Mach reflection problem, the WENO-Z schemes can capture more fine local structures. For the forward step problem, the results of the WENO-MR schemes are better. However, the HWENO schemes are more efficient in CPU time. The CPU time of the central HWENO scheme is about 30% of that by the central WENO scheme, and the CPU time of the upwind HWENO scheme is about 60% of that by the upwind WENO scheme.

4 Concluding Remarks

In this paper, a class of new high-order central Hermite WENO schemes is designed to solve the hyperbolic conservation laws in one and two dimensions. The multi-resolution HWENO reconstructions [13, 14] based on the zeroth-order and first-order moments of the solution

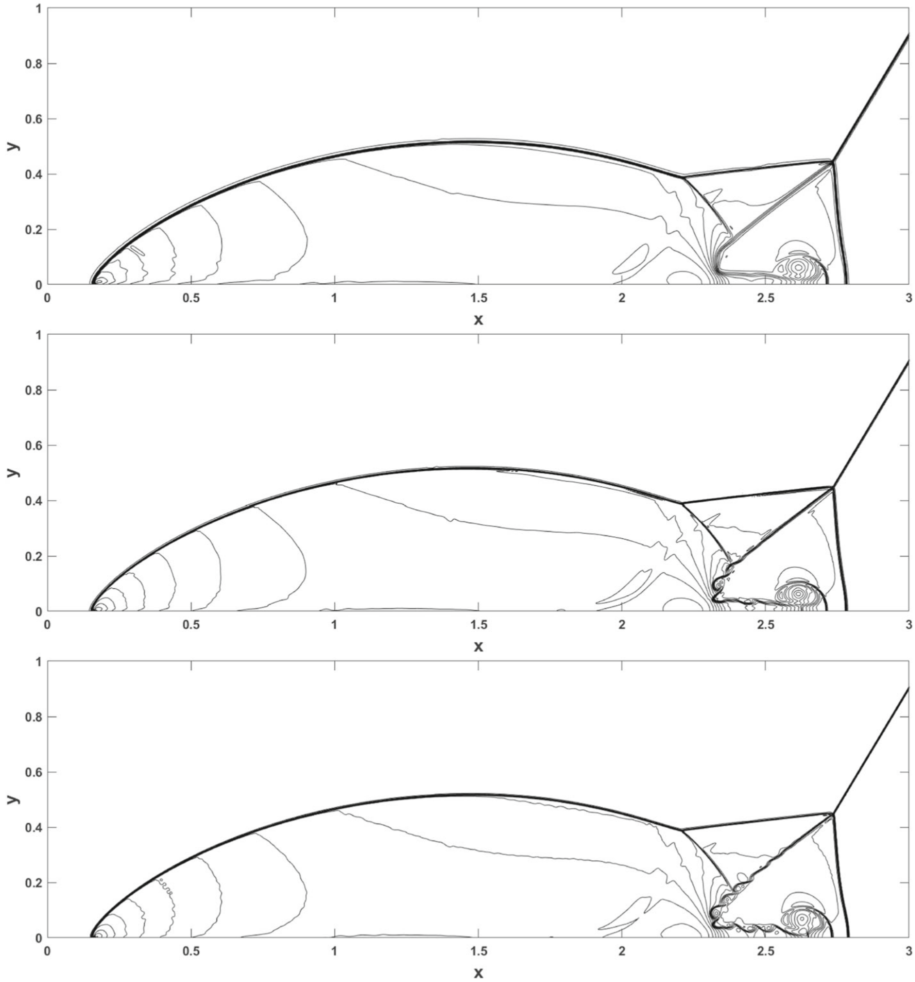


Fig. 17 Double Mach reflection problem. 1108 × 277 and 1920 × 480 mesh elements for the HWENO and WENO schemes, respectively. $T = 0.2$. From top to bottom: C-HWENO6, C-WENO5-MR, C-WENO5-Z. 30 equally spaced density contours from 1.5 to 22.7

Table 9 CPU time (in hours). WENO and HWENO schemes

CPU time	Double mach reflection problem	Forward step problem
C-HWENO6	25.4466	25.4794
Up-HWENO6	46.3500	47.0914
C-WENO5-MR	87.9331	90.3750
Up-WENO5-MR	76.7532	77.1261
C-WENO5-Z	87.1033	87.9883
Up-WENO5-Z	76.6949	76.8953

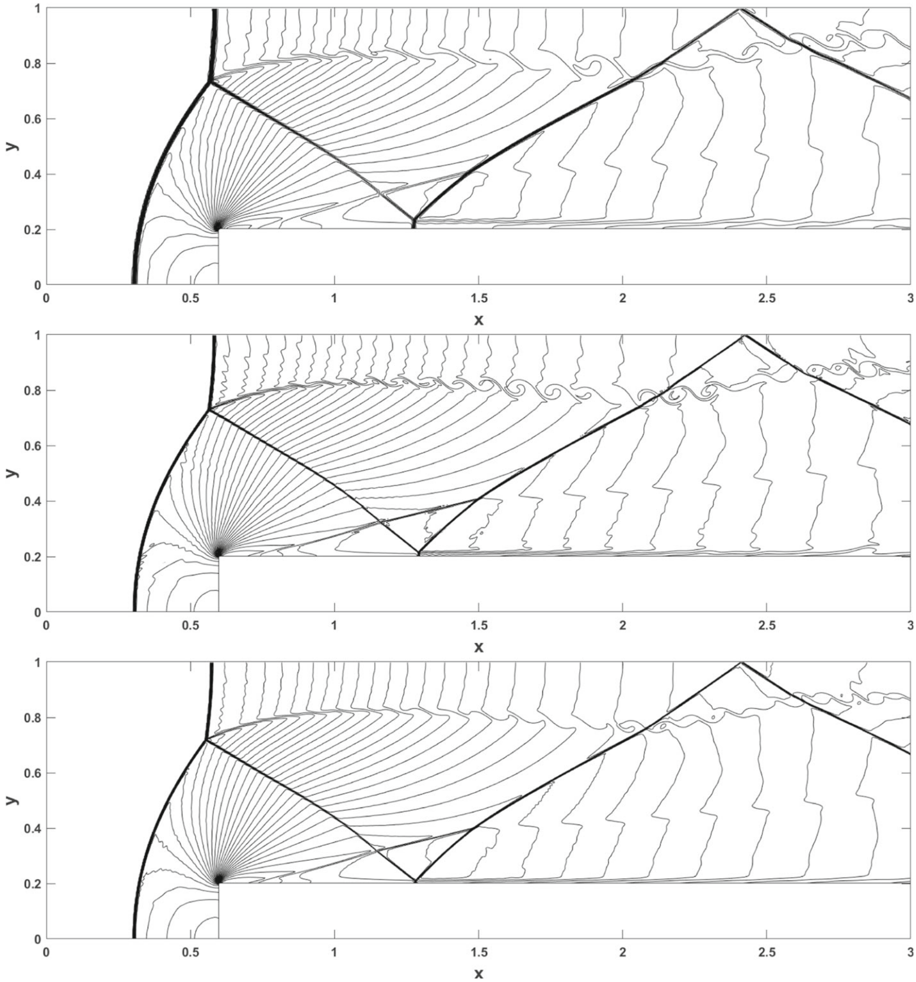


Fig. 18 Forward step problem. 555 × 185 and 960 × 320 mesh elements for the HWENO and WENO schemes, respectively. $T = 4$. From top to bottom: Up-HWENO6, Up-WENO5-MR, Up-WENO5-Z. 30 equally spaced density contours from 0.32 to 6.15

are used for the spatial discretization and the natural continuous extension of Runge–Kutta method is used as the time discretization, in a central finite volume framework on staggered meshes. our new schemes require neither numerical fluxes nor flux splitting.

The compact reconstructions exploit a series of hierarchical central spatial stencils, and this leads to better accuracy and resolution than the HWENO reconstructions in [27] with the same size of the stencils. In such new reconstructions, one can artificially set the positive linear weights as long as their sum equals one to avoid the splitting treatment for the negative weights. Meanwhile, the truly two-dimensional HWENO reconstructions are used, and the mixed-type first-order moment $\overline{vw}_{ij}^{n}$ (which is included for the two-dimensional reconstructions with a dimension-by-dimension procedure in [27]) is not needed. For the troubled-cells,

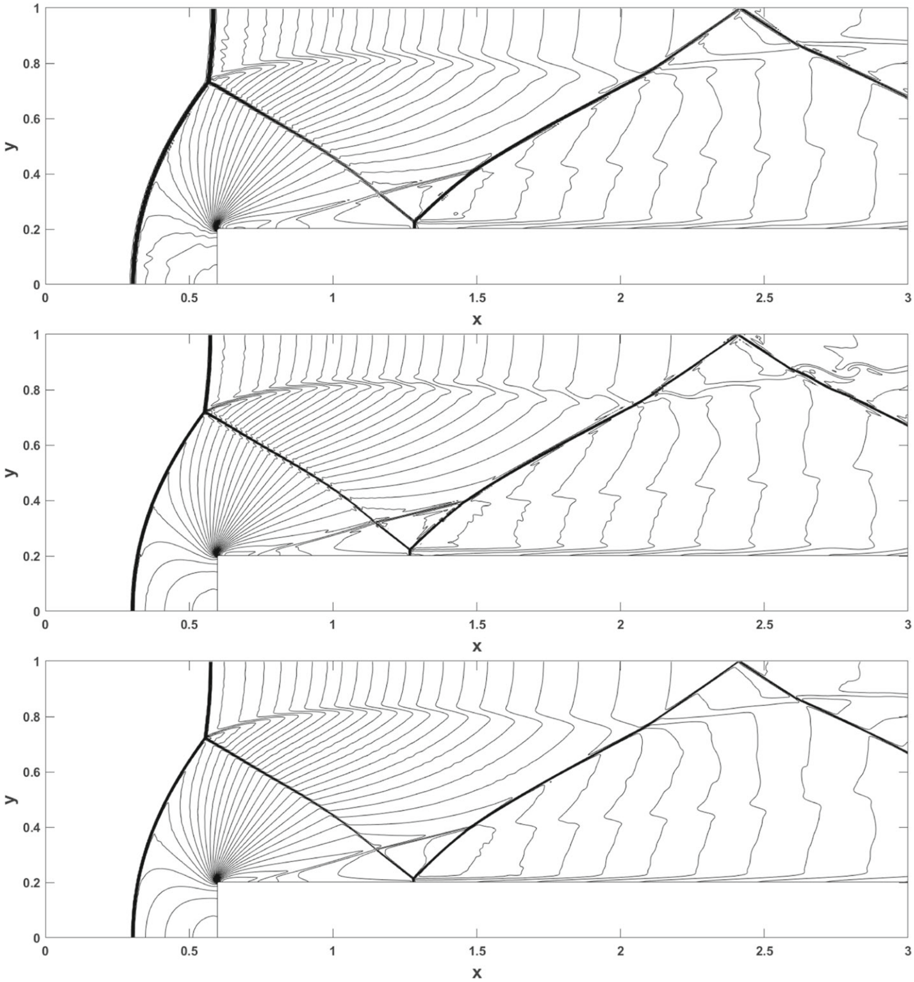


Fig. 19 Forward step problem. 555×185 and 960×320 mesh elements for the HWENO and WENO schemes, respectively. $T = 4$. From top to bottom: C-HWENO6, C-WENO5-MR, C-WENO5-Z. 30 equally spaced density contours from 0.32 to 6.15

we modify the first-order moments and use the HWENO reconstructions. For the cells which are not troubled-cells, we simply use the linear approximations.

Funding Zhanjing Tao: Research is supported by NSFC Grant 12001231 and Fundamental Research Funds for the Central Universities. Jun Zhu: Research is supported by NSFC Grant 11872210. Jianxian Qiu: Research is supported by NSFC Grant 12071392.

Data Availability Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

Declarations

Conflict of interest The authors have no relevant financial or non-financial interests to disclose.

References

1. Balsara, D.S., Shu, C.-W.: Monotonicity preserving weighted essentially non-oscillatory schemes with increasingly high order of accuracy. J. Comput. Phys. **160**(2), 405–452 (2000)
2. Bianco, F., Puppo, G., Russo, G.: High-order central schemes for hyperbolic systems of conservation laws. SIAM J. Sci. Comput. **21**(1), 294–322 (1999)
3. Borges, R., Carmona, M., Costa, B., Don, W.S.: An improved weighted essentially non-oscillatory scheme for hyperbolic conservation laws. J. Comput. Phys. **227**(6), 3191–3211 (2008)
4. Castro, M., Costa, B., Don, W.S.: High order weighted essentially non-oscillatory WENO-Z schemes for hyperbolic conservation laws. J. Comput. Phys. **230**(5), 1766–1792 (2011)
5. Dumbser, M., Boscheri, W., Semplice, M., Russo, G.: Central WENO schemes for hyperbolic conservation laws on fixed and moving unstructured meshes. SIAM J. Sci. Comput. **39**(6), 2564 (2016)
6. Hu, C., Shu, C.-W.: Weighted essentially non-oscillatory schemes on triangular meshes. J. Comput. Phys. **150**(1), 97–127 (1999)
7. Jiang, G.S., Shu, C.-W.: Efficient implementation of weighted ENO schemes. J. Comput. Phys. **126**(1), 202–228 (1996)
8. Jiang, G.-S., Tadmor, E.: Nonoscillatory central schemes for multidimensional hyperbolic conservation laws. SIAM J. Sci. Comput. **19**(6), 1892–1917 (1998)
9. Krivodonova, L., Xin, J., Remacle, J.F., Chevaugeon, N., Flaherty, J.E.: Shock detection and limiting with discontinuous Galerkin methods for hyperbolic conservation laws. Appl. Numer. Math. **48**(3), 323–338 (2004)
10. Kurganov, A., Levy, D.: A third-order semidiscrete central scheme for conservation laws and convection–diffusion equations. SIAM J. Sci. Comput. **22**(4), 1461–1488 (2000)
11. Levy, D., Puppo, G., Russo, G.: Central WENO schemes for hyperbolic systems of conservation laws. Esaim Math. Model. Numer. Anal. **33**(3), 547–571 (1999)
12. Levy, D., Puppo, G., Russo, G.: A fourth-order central WENO scheme for multidimensional hyperbolic systems of conservation laws. SIAM J. Sci. Comput. **24**(2), 480–506 (2002)
13. Li, J., Shu, C.-W., Qiu, J.: Multi-resolution HWENO schemes for hyperbolic conservation laws. J. Comput. Phys. **446**, 110653 (2021)
14. Li, J., Shu, C.-W., Qiu, J.: Moment-based multi-resolution HWENO scheme for hyperbolic conservation laws. Commun. Comput. Phys. **32**(2), 364–400 (2022)
15. Liu, H., Qiu, J.: Finite difference Hermite WENO schemes for conservation laws. J. Sci. Comput. **63**, 548–572 (2015)
16. Liu, X.D., Osher, S., Chan, T.: Weighted essentially non-oscillatory schemes. J. Comput. Phys. **115**(1), 200–212 (1994)
17. Liu, X.-D., Tadmor, E.: Third order nonoscillatory central scheme for hyperbolic conservation laws. Numer. Math. **79**(3), 397–425 (1998)
18. Liu, Y.: Central schemes on overlapping cells. J. Comput. Phys. **209**(1), 82–104 (2005)
19. Nessyahu, H., Tadmor, E.: Non-oscillatory central differencing for hyperbolic conservation laws. J. Comput. Phys. **87**, 408–463 (1990)
20. Qiu, J., Shu, C.-W.: On the construction, comparison, and local characteristic decomposition for high-order central WENO schemes. J. Comput. Phys. **183**(1), 187–209 (2002)
21. 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)
22. Qiu, J., Shu, C.-W.: A comparison of troubled-cell indicators for Runge–Kutta discontinuous Galerkin methods using weighted essentially nonoscillatory limiters. SIAM J. Sci. Comput. **27**(3), 995–1013 (2005)
23. 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**, 642–663 (2005)
24. Shi, J., Hu, C., Shu, C.-W.: A technique of treating negative weights in WENO schemes. J. Comput. Phys. **175**(1), 108–127 (2002)
25. Shu, C.-W.: High order weighted essentially nonoscillatory schemes for convection dominated problems. SIAM Rev. **51**(1), 82–126 (2009)
26. Tao, Z., Li, F., Qiu, J.: High-order central Hermite WENO schemes on staggered meshes for hyperbolic conservation laws. J. Comput. Phys. **281**, 148–176 (2015)
27. 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)

28. Woodward, P., Colella, P.: The numerical simulation of two-dimensional fluid flow with strong shocks. J. Comput. Phys. **54**, 115–173 (1984)
29. Zahran, Y.H., Abdalla, A.H.: Seventh order Hermite WENO scheme for hyperbolic conservation laws. Comput. Fluids **131**, 66–80 (2016)
30. Zennaro, M.: Natural continuous extensions of Runge–Kutta methods. Math. Comput. **46**, 119–133 (1986)
31. Zhao, Z., Chen, Y., Qiu, J.: A hybrid Hermite WENO scheme for hyperbolic conservation laws. J. Comput. Phys. **405**, 109175 (2020)
32. Zhao, Z., Qiu, J.: A Hermite WENO scheme with artificial linear weights for hyperbolic conservation laws. J. Comput. Phys. **417**, 109583 (2020)
33. Zhu, J., Qiu, J.: A class of the fourth order finite volume Hermite weighted essentially non-oscillatory schemes. Sci. China **51**(08), 1549–1560 (2008)
34. Zhu, J., Shu, C.-W.: A new type of multi-resolution WENO schemes with increasingly higher order of accuracy. J. Comput. Phys. **375**, 659–683 (2018)