J Sci Comput (2009) 39: 293–321
DOI 10.1007/s10915-009-9271-7

Hermite WENO Schemes and Their Application as Limiters for Runge-Kutta Discontinuous Galerkin Method, III: Unstructured Meshes

Jun Zhu · Jianxian Qiu

Received: 31 August 2008 / Accepted: 9 January 2009 / Published online: 30 January 2009

Abstract In [J. Comput. Phys. 193:115–135, 2004] and [Comput. Fluids 34:642–663, 2005], Qiu and Shu developed a class of high order weighted essentially non-oscillatory (WENO) schemes based on Hermite polynomials, termed HWENO (Hermite WENO) schemes, for solving nonlinear hyperbolic conservation law systems, and applied them as limiters for the Runge-Kutta discontinuous Galerkin (RKDG) methods on structured meshes. In this continuation paper, we extend the method to solve two dimensional problems on unstructured meshes. The emphasis is again on the application of such HWENO finite volume methodology as limiters for RKDG methods to maintain compactness of RKDG methods. Numerical experiments for two dimensional Burgers' equation and Euler equations of compressible gas dynamics are presented to show the effectiveness of these methods.

Keywords Runge-Kutta discontinuous Galerkin method · Limiters · HWENO finite volume scheme · High order accuracy

1 Introduction

In [19, 20], a class of high order weighted essentially non-oscillatory (WENO) schemes [10, 12, 13, 16, 17] based on Hermite polynomials, termed HWENO (Hermite WENO)

The research was partially supported by the European project ADIGMA on the development of innovative solution algorithms for aerodynamic simulations, NSFC grant 10671091 and JSNSF BK2006511.

J. Zhu · J. Qiu (✉)
Department of Mathematics, Nanjing University, Nanjing, Jiangsu 210093, China
e-mail: jxqiu@nju.edu.cn

J. Zhu
e-mail: zhujun@nuaa.edu.cn

J. Zhu
College of Science, Nanjing University of Aeronautics and Astronautics, Nanjing, Jiangsu 210016, China

schemes, for solving nonlinear hyperbolic conservation laws, were developed and applied as limiters for the Runge-Kutta discontinuous Galerkin (RKDG) methods [3–8] on structured meshes by Qiu and Shu. In this continuation paper, we extend the method to solve nonlinear hyperbolic conservation laws:

$$\begin{cases} u_t + f(u)_x + g(u)_y = 0, \\ u(x, y, 0) = u_0(x, y), \end{cases} \tag{1.1}$$

on two dimensional unstructured meshes.

Essentially non-oscillatory (ENO) and WENO are finite difference or finite volume schemes. ENO schemes were designed by Harten et al. in 1987 [11, 26, 27]. The first WENO scheme was constructed in 1994 by Liu, Osher and Chan for a third order version [16]. In 1996, third and fifth order finite difference WENO schemes in multi-space dimensions were constructed by Jiang and Shu [13], with a general framework for the design of smoothness indicators and nonlinear weights. Finite volume WENO schemes on unstructured and structured meshes also were developed in [10, 12, 15, 21, 24]. A key idea in WENO schemes is a linear combination of lower order fluxes or reconstruction to obtain a higher order approximation. Both ENO and WENO schemes use the idea of adaptive stencils to automatically achieve high order accuracy and non-oscillatory property near discontinuities. For the system case, WENO schemes based on local characteristic decompositions and flux splitting to avoid spurious oscillatory.

The main difference between the Hermite WENO scheme designed in [19, 20], see also related earlier work in [9, 18, 28], and the traditional WENO schemes is that the former has a more compact stencil than the latter for the same order of accuracy. This compactness is achieved by evolving both the function and its first derivative values in time and they are both used in the reconstruction in HWENO schemes. As a result, a fifth order one dimensional HWENO reconstruction uses only three points, while a fifth order one dimensional WENO reconstruction would need to use five points.

One major emphasis of the HWENO methodology in [19, 20] is its application as limiters for the RKDG (Runge-Kutta discontinuous Galerkin) methods. The discontinuous Galerkin (DG) method is a finite element method which evolves k degrees of freedom (in one dimension) per cell for a k-th order accurate scheme, thus no reconstruction is needed. We refer to, e.g. [4, 7, 8, 23] for the detailed description of the RKDG methods. An important component of RKDG methods for solving conservation laws (1.1) with strong shocks in the solutions is a nonlinear limiter, which is applied to control spurious oscillation. Although many limiters exist in the literature, e.g. [1–7], they tend to degenerate accuracy when mistakenly used in smooth regions of the solution. In [22, 30], Qiu et al. studied using WENO methodology as limiters for RKDG methods on structured and unstructured meshes. The idea is to first identify "troubled cells", namely those cells where limiting might be needed, then to abandon all moments in those cells except the cell averages and reconstruct those moments from the information of neighboring cells using a WENO methodology. This technique works quite well in one and two dimensional test problems. In [19, 20], this approach is further improved by using HWENO rather than WENO methodology in the limiter so that a more compact stencil is used on structured mesh. More recently, Luo et al. [17], following [19, 20], developed a Hermite WENO-based limiter for the second order RKDG method on unstructured meshes.

In this continuation paper, we extend the HWENO methodology in [19, 20] to solve two dimensional problems on unstructured meshes. Both the third and the forth order schemes are considered. The emphasis is again on the application of such HWENO finite volume

methodology as the limiters for RKDG methods to maintain compactness of RKDG methods. We describe the construction of the third and the forth order Hermite WENO schemes in Sect. 2 and the details of the procedure for HWENO used as limiters for the second and third order DG methods in Sect. 3. Numerical experiments for two dimensional Burgers' equation and Euler equations of compressible gas dynamics are presented to show the effectiveness of these methods in Sect. 4. Concluding remarks are given in Sect. 5.

2 The Construction of Hermite WENO Schemes on Unstructured Meshes

We consider two dimensional conservation laws (1.1). Let $v = \frac{\partial u}{\partial x}$ and $w = \frac{\partial u}{\partial y}$. Taking the derivatives of (1.1), we can get:

$$\begin{cases} v_t + h_x + r_y = 0, \\ v(x, y, 0) = \frac{\partial u_0(x,y)}{\partial x}, \end{cases} \tag{2.1}$$

$$\begin{cases} w_t + q_x + s_y = 0, \\ w(x, y, 0) = \frac{\partial u_0(x,y)}{\partial y}, \end{cases} \tag{2.2}$$

where $h(u, v) = f'(u)v$, $r(u, v) = g'(u)v$, $q(u, w) = f'(u)w$, $s(u, w) = g'(u)w$. We integrate (1.1), (2.1), (2.2) on the target cell Δ_0 to obtain the semi-discrete finite volume scheme as:

$$\frac{du(t)}{dt} = -\frac{1}{|\Delta_0|}\int_{\partial\Delta_0} F \cdot n ds, \tag{2.3}$$

$$\frac{dv(t)}{dt} = -\frac{1}{\sqrt{|\Delta_0|}}\int_{\partial\Delta_0} H \cdot n ds, \tag{2.4}$$

$$\frac{dw(t)}{dt} = -\frac{1}{\sqrt{|\Delta_0|}}\int_{\partial\Delta_0} Q \cdot n ds, \tag{2.5}$$

where $u(t) = \frac{1}{|\Delta_0|}\int_{\partial\Delta_0} u(x, y, t)dxdy$, $v(t) = \frac{1}{\sqrt{|\Delta_0|}}\int_{\Delta_0} \frac{\partial u(x,y,t)}{\partial x}dxdy$, $w(t) = \frac{1}{\sqrt{|\Delta_0|}}\int_{\Delta_0} \frac{\partial u(x,y,t)}{\partial y}dxdy$ and $F = (f, g)^T$, $H = (h, r)^T$, $Q = (q, s)^T$, n denotes the outward unit normal to the edge of the target cell. The line integrals in (2.3), (2.4), (2.5) are discretized by a 2-point Gaussian integration formula on every edge:

$$\int_{\partial\Delta_0} F \cdot n ds \approx \sum_{ll=1}^{3} |\partial\Delta_{0_{ll}}| \sum_{l=1}^{2} \omega_l F(u(G_{ll_l}, t)) \cdot n_{ll}, \tag{2.6}$$

$$\int_{\partial\Delta_0} H \cdot n ds \approx \sum_{ll=1}^{3} |\partial\Delta_{0_{ll}}| \sum_{l=1}^{2} \omega_l H(u(G_{ll_l}, t), v(G_{ll_l}, t)) \cdot n_{ll}, \tag{2.7}$$

$$\int_{\partial\Delta_0} Q \cdot n ds \approx \sum_{ll=1}^{3} |\partial\Delta_{0_{ll}}| \sum_{l=1}^{2} \omega_l Q(u(G_{ll_l}, t), w(G_{ll_l}, t)) \cdot n_{ll}. \tag{2.8}$$

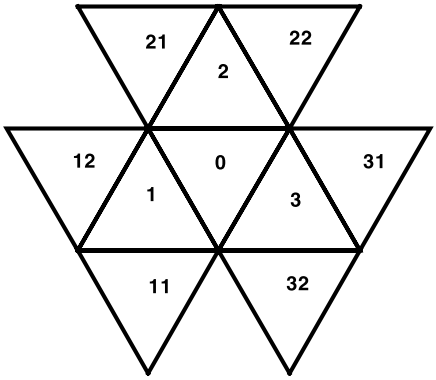

Fig. 1 The big stencil

And $F(u(G_{ll_l},t))\cdot n_{ll}$, $H(u(G_{ll_l},t),v(G_{ll_l},t))\cdot n_{ll}$, $Q(u(G_{ll_l},t),w(G_{ll_l},t))\cdot n_{ll}$ are replaced by numerical fluxes such as the Lax-Friedrichs fluxes.

Then we use third order version TVD Runge-Kutta time discrete method [26]:

$$\begin{cases} u^{(1)} = u^n + \Delta t L(u^n), \\ u^{(2)} = \frac{3}{4}u^n + \frac{1}{4}u^{(1)} + \frac{1}{4}\Delta t L(u^{(1)}), \\ u^{n+1} = \frac{1}{3}u^n + \frac{2}{3}u^{(2)} + \frac{2}{3}\Delta t L(u^{(2)}), \end{cases} \tag{2.9}$$

to obtain fully discrete scheme both in space and time.

2.1 The Construction of HWENO3 Scheme

For the simplicity, we use the reconstruction stencils which are shown in Fig. 1.

1. The reconstruction of function u at Gauss quadrature points (x_{G_l}, y_{G_l}) in the boundaries of target cell Δ_0 (as shown in the figure).

Step 2.1.1. We select the big stencil as $S = \{\Delta_0, \Delta_1, \Delta_2, \Delta_3\}$. Then we construct a quadratic polynomial $P(x, y)$ to obtain a third order approximation of u by requiring that it has the same cell average as u on the target cell Δ_0, and matches the cell averages of u, v, or w on the other triangles in the set $S \setminus \{\Delta_0\}$ in a least square sense [12].

Step 2.1.2. We then construct three linear polynomials $q_i(x, y)$, $i = 1, 2, 3$ and the other three linear polynomials $q_i(x, y)$, $i = 4, 5, 6$ (which satisfy the cell averages of variable u on the target cell Δ_0 and its neighbor triangle, then match the cell averages of variables v and w on the other triangles in a least square sense):

$$\frac{1}{|\Delta_\ell|}\int_{\Delta_\ell} q_i(x,y)dxdy = \bar{u}_\ell, \tag{2.10}$$

$$\frac{1}{\sqrt{|\Delta_{\ell_x}|}}\int_{\Delta_{\ell_x}} \frac{\partial q_i(x,y)}{\partial x}dxdy = \bar{v}_{\ell_x}, \tag{2.11}$$

$$\frac{1}{\sqrt{|\Delta_{\ell_y}|}}\int_{\Delta_{\ell_y}} \frac{\partial q_i(x,y)}{\partial y}dxdy = \bar{w}_{\ell_y}. \tag{2.12}$$

For

$$i = 1, \quad \ell = 0, 1, 2; \qquad i = 2, \quad \ell = 0, 2, 3; \qquad i = 3, \quad \ell = 0, 3, 1;$$

$$i=4,\quad \ell=0,1,\quad \ell_x=1,\quad \ell_y=1;\qquad i=5,\quad \ell=0,2,\quad \ell_x=2,\quad \ell_y=2;$$
$$i=6,\quad \ell=0,3,\quad \ell_x=3,\quad \ell_y=3.$$

Step 2.1.3. We find the combination coefficients, also called linear weights, denoted by $\gamma_1,\ldots,\gamma_6$, satisfying:

$$P(x_{G_l},y_{G_l})=\sum_{i=1}^{6}\gamma_i q_i(x_{G_l},y_{G_l}) \tag{2.13}$$

for the quadratic polynomial $P(x,y)$ defined in step 1. Then we can get a third order approximation of u at the point G_l for all smooth u. We know that (2.13) holds for any polynomial u of degree at most 1, if $\sum_{i=1}^{6}\gamma_i=1$. This is because every $q_i(x,y)$ reconstructs linear polynomial exactly. These are three other constraints on the linear weights from requiring (2.13) to hold for $u=x^2,xy,y^2$. This leaves 2 free parameters in determining the linear weights. And these free parameters are determined by asking for

$$\min\left(\sum_{i=1}^{6}\gamma_i^2\right) \tag{2.14}$$

subject to the constraints listed above. By doing so, we can get the linear weights uniquely but can not maintain them positively all the time, we can use the methods that presented in [12, 24] to overcome this drawback.

Step 2.1.4. We compute the smoothness indicators, denote by β_i, $i=1,\ldots,6$, which measure how smooth the functions $q_i(x,y)$, $i=1,\ldots,6$, are in the target cell Δ_0. The smaller these smoothness indicators, the smoother the functions are in the target cell. We use the same recipe for the smoothness indicators as in [12, 13]:

$$\beta_i=\sum_{|\ell|=1}|\Delta_0|^{|\ell|-1}\int_{\Delta_0}\left(\frac{\partial^{|\ell|}}{\partial x^{\ell_1}\partial y^{\ell_2}}q_i(x,y)\right)^2 dxdy,\ i=1,\ldots,6 \tag{2.15}$$

where $\ell=(\ell_1,\ell_2)$.

Step 2.1.5. We compute the non-linear weights based on the smoothness indicators:

$$\omega_i=\frac{\bar{\omega}_i}{\sum_{\ell=1}^{6}\bar{\omega}_\ell},\qquad \bar{\omega}_\ell=\frac{\gamma_\ell}{(\varepsilon+\beta_\ell)^2}. \tag{2.16}$$

Here ε is a small positive number to avoid the denominator to become zero. We take $\varepsilon=10^{-6}$ in our computation.

The final approximations are then given by: $u(x_{G_l},y_{G_l})\approx\sum_{i=1}^{6}\omega_i q_i(x_{G_l},y_{G_l})$.

2. The reconstruction of function v at Gauss quadrature points (x_{G_l},y_{G_l}) in the boundaries of target cell Δ_0 (as shown in the figure).

Step 2.2.1. We select the big stencil as $S=\{\Delta_0,\Delta_1,\Delta_2,\Delta_3\}$. Then we construct a quadratic polynomial $P(x,y)$ to obtain a third order approximation of v by requiring that it has the same cell average as v on the target cell and matches the cell averages of u, v, or w on the other triangles in a least square sense [12].

Step 2.2.2. We then construct three linear polynomials $q_i(x,y)$, $i=1,2,3$.

$$\frac{1}{\sqrt{|\Delta_{\ell_x}|}}\int_{\Delta_{\ell_x}}q_i(x,y)dxdy=\bar{v}_{\ell_x}. \tag{2.17}$$

For

$$i=1,\quad \ell_x=0,1,2;\qquad i=2,\quad \ell_x=0,2,3;\qquad i=3,\quad \ell_x=0,3,1.$$

And construct three quadratic polynomials $q_i(x,y)$, $i=4,5,6$.

$$\frac{1}{|\Delta_\ell|}\int_{\Delta_\ell} q_i(x,y)dxdy=\bar{u}_\ell, \tag{2.18}$$

$$\frac{1}{\sqrt{|\Delta_{\ell_x}|}}\int_{\Delta_{\ell_x}} \frac{\partial q_i(x,y)}{\partial x}dxdy=\bar{v}_{\ell_x}, \tag{2.19}$$

$$\frac{1}{\sqrt{|\Delta_{\ell_y}|}}\int_{\Delta_{\ell_y}} \frac{\partial q_i(x,y)}{\partial y}dxdy=\bar{w}_{\ell_y}. \tag{2.20}$$

For

$$\begin{aligned} &i=4,\quad \ell=0,1,\quad \ell_x=0,1,\quad \ell_y=0,1;\\ &i=5,\quad \ell=0,2,\quad \ell_x=0,2,\quad \ell_y=0,2;\\ &i=6,\quad \ell=0,3,\quad \ell_x=0,3,\quad \ell_y=0,3. \end{aligned}$$

Step 2.2.3. We use the same methodology as Step 2.1.3 to find the linear weights γ_i, $i=1,\ldots,6$.

Step 2.2.4. We compute the smoothness indicators, denote by β_i, $i=1,\ldots,6$:

$$\beta_i=\sum_{|\ell|=1}|\Delta_0|^{|\ell|-1}\int_{\Delta_0}\left(\frac{\partial^{|\ell|}}{\partial x^{\ell_1}\partial y^{\ell_2}}q_i(x,y)\right)^2 dxdy,\quad i=1,2,3, \tag{2.21}$$

$$\beta_i=\sum_{|\ell|=1}|\Delta_0|^{|\ell|-1}\int_{\Delta_0}\left(\frac{\partial^{|\ell|}}{\partial x^{\ell_1}\partial y^{\ell_2}}\left(\frac{\partial}{\partial x}q_i(x,y)\right)\right)^2 dxdy,\quad i=4,5,6, \tag{2.22}$$

where $\ell=(\ell_1,\ell_2)$.

Step 2.2.5. With the linear weights and smoothness indicators we can get nonlinear weights by (2.16).

The final approximations are then given by:

$$v(x_{G_l},y_{G_l})\approx\sum_{i=1}^{3}\omega_i q_i(x_{G_l},y_{G_l})+\sum_{i=4}^{6}\omega_i\frac{\partial}{\partial x}q_i(x_{G_l},y_{G_l}).$$

The procedure of reconstruction of w is similar to that of v.

Remark When some triangles merge in the stencils, we can always use the next layer of triangles to overcome this situation.

2.2 The Construction of HWENO4 Scheme

1. Reconstruction of function u at the boundaries of target cell Δ_0 of Gauss quadrature points (x_{G_l},y_{G_l}).

Step 2.3.1. We select the big stencil as $S = \{\Delta_0, \Delta_1, \Delta_2, \Delta_3, \Delta_{11}, \Delta_{12}, \Delta_{21}, \Delta_{22}, \Delta_{31}, \Delta_{32}\}$. Then we construct a third degree polynomial $Q(x, y)$ to obtain a fourth order approximation of u by requiring that it has the same cell average as u on the target cell Δ_0 and matches the cell averages of u, v, or w on the other triangles in the set $S \setminus \{\Delta_0\}$ in a least square sense.

Step 2.3.2. We can then construct nine quadratic polynomials $q_i(x, y)$, $i = 1, \ldots, 9$, which satisfy the following conditions:

$$\frac{1}{|\Delta_\ell|}\int_{\Delta_\ell} q_i(x, y)dxdy = \bar{u}_\ell, \tag{2.23}$$

$$\frac{1}{\sqrt{|\Delta_{\ell_x}|}}\int_{\Delta_{\ell_x}} \frac{\partial q_i(x, y)}{\partial x}dxdy = \bar{v}_{\ell_x}, \tag{2.24}$$

$$\frac{1}{\sqrt{|\Delta_{\ell_y}|}}\int_{\Delta_{\ell_y}} \frac{\partial q_i(x, y)}{\partial y}dxdy = \bar{w}_{\ell_y}. \tag{2.25}$$

For

$$\begin{aligned}
&i = 1, \quad \ell = 0, 1, 11, 12, 3, 32; && i = 2, \quad \ell = 0, 1, 11, 12, 2, 21;\\
&i = 3, \quad \ell = 0, 2, 21, 22, 1, 12; && i = 4, \quad \ell = 0, 2, 21, 22, 3, 31;\\
&i = 5, \quad \ell = 0, 3, 31, 32, 2, 22; && i = 6, \quad \ell = 0, 3, 31, 32, 1, 11;\\
&i = 7, \quad \ell = 0, 1, 11, 12, \quad \ell_x = 1, \quad \ell_y = 1;\\
&i = 8, \quad \ell = 0, 2, 21, 22, \quad \ell_x = 2, \quad \ell_y = 2;\\
&i = 9, \quad \ell = 0, 3, 31, 32, \quad \ell_x = 3, \quad \ell_y = 3.
\end{aligned}$$

The remaining steps are the same as those for Sect. 2.1. The final approximations are then given by: $u(x_{G_l}, y_{G_l}) \approx \sum_{i=1}^{9} \omega_i q_i(x_{G_l}, y_{G_l})$.

2. Reconstruct function v at the boundaries of target cell Δ_0 of Gauss quadrature points (x_{G_l}, y_{G_l}).

Step 2.4.1. We select the big stencil as $S = \{\Delta_0, \Delta_1, \Delta_2, \Delta_3, \Delta_{11}, \Delta_{12}, \Delta_{21}, \Delta_{22}, \Delta_{31}, \Delta_{32}\}$. Then we construct a third degree polynomial $Q(x, y)$ to obtain a fourth order approximation of v by requiring that it has the same cell average as v on the target cell and matches the cell averages of u, v, or w on the other triangles in a least square sense.

Step 2.4.2. We can then construct six quadratic polynomials $q_i(x, y)$, $i = 1, \ldots, 6$.

$$\frac{1}{\sqrt{|\Delta_{\ell_x}|}}\int_{\Delta_{\ell_x}} q_i(x, y)dxdy = \bar{v}_{\ell_x}. \tag{2.26}$$

For

$$\begin{aligned}
&i = 1, \quad \ell_x = 0, 1, 11, 12, 3, 32; && i = 2, \quad \ell_x = 0, 1, 11, 12, 2, 21;\\
&i = 3, \quad \ell_x = 0, 2, 21, 22, 1, 12; && i = 4, \quad \ell_x = 0, 2, 21, 22, 3, 31;\\
&i = 5, \quad \ell_x = 0, 3, 31, 32, 2, 22; && i = 6, \quad \ell_x = 0, 3, 31, 32, 1, 11.
\end{aligned}$$

And construct three cubic polynomials $q_i(x, y)$, $i = 7, 8, 9$.

$$\frac{1}{|\Delta_\ell|}\int_{\Delta_\ell} q_i(x, y)dxdy = \bar{u}_\ell, \tag{2.27}$$

$$\frac{1}{\sqrt{|\Delta_{\ell_x}|}}\int_{\Delta_{\ell_x}}\frac{\partial q_i(x,y)}{\partial x}dxdy=\bar{v}_{\ell_x}, \tag{2.28}$$

$$\frac{1}{\sqrt{|\Delta_{\ell_y}|}}\int_{\Delta_{\ell_y}}\frac{\partial q_i(x,y)}{\partial y}dxdy=\bar{w}_{\ell_y}. \tag{2.29}$$

For

$$\begin{aligned}
&i=7,\quad \ell=0,1,11,12,\quad \ell_x=0,11,12,\quad \ell_y=0,11,12;\\
&i=8,\quad \ell=0,2,21,22,\quad \ell_x=0,21,22,\quad \ell_y=0,21,22;\\
&i=9,\quad \ell=0,3,31,32,\quad \ell_x=0,31,32,\quad \ell_y=0,31,32.
\end{aligned}$$

Then do the remaining steps. The final approximations are then given by: $v(x_{G_l},y_{G_l})\approx\sum_{i=1}^{6}\omega_i q_i(x_{G_l},y_{G_l})+\sum_{i=7}^{9}\omega_i\frac{\partial}{\partial x}q_i(x_{G_l},y_{G_l})$.

The procedure of reconstruction of function w at the boundaries of target cell Δ_0 of Gauss quadrature points (x_{G_l},y_{G_l}) is similar to that of v.

3 HWENO Reconstruction as a Limiter to the RKDG Method on Unstructured Meshes

In this section we give the details of the procedure using the HWENO reconstruction as a limiter for the RKDG method.

Given a triangulation consisting of cells Δ_j, $\mathbb{P}^k(\Delta_j)$ denotes the set of polynomials of degree at most k defined on Δ_j. Here k could actually change from cell to cell, but for simplicity we assume it is a constant over the whole triangulation. In the DG method, the solution as well as the test function space is given by $V_h^k=\{v(x,y):v(x,y)|_{\Delta_j}\in\mathbb{P}^k(\Delta_j)\}$. We emphasize that the procedure described below does not depend on the specific basis chosen for the polynomials. We adopt a local orthogonal basis over a target cell, such as Δ_0: $\{v_l^{(0)}(x,y),l=0,\ldots,K;K=(k+1)(k+2)/2-1\}$:

$$\begin{aligned}
v_0^{(0)}(x,y)&=1,\\
v_1^{(0)}(x,y)&=\frac{x-x_0}{\sqrt{|\Delta_0|}},\\
v_2^{(0)}(x,y)&=a_{21}\frac{x-x_0}{\sqrt{|\Delta_0|}}+\frac{y-y_0}{\sqrt{|\Delta_0|}}+a_{22},\\
v_3^{(0)}(x,y)&=\frac{(x-x_0)^2}{|\Delta_0|}+a_{31}\frac{x-x_0}{\sqrt{|\Delta_0|}}+a_{32}\frac{y-y_0}{\sqrt{|\Delta_0|}}+a_{33},\\
v_4^{(0)}(x,y)&=a_{41}\frac{(x-x_0)^2}{|\Delta_0|}+\frac{(x-x_0)(y-y_0)}{|\Delta_0|}+a_{42}\frac{x-x_0}{\sqrt{|\Delta_0|}}+a_{43}\frac{y-y_0}{\sqrt{|\Delta_0|}}+a_{44},\\
v_5^{(0)}(x,y)&=a_{51}\frac{(x-x_0)^2}{|\Delta_0|}+a_{52}\frac{(x-x_0)(y-y_0)}{|\Delta_0|}+\frac{(y-y_0)^2}{|\Delta_0|}\\
&\quad+a_{53}\frac{x-x_0}{\sqrt{|\Delta_0|}}+a_{54}\frac{y-y_0}{\sqrt{\Delta_0}}+a_{55},\quad\ldots
\end{aligned}$$

where (x_0, y_0) and $|\Delta_0|$ are the barycenter and the area of the target cell Δ_0, respectively. Then we would need to solve a linear system to obtain the values of $a_{\ell m}$ by the orthogonality property:

$$\int_{\Delta_0} v_i^{(0)}(x,y)v_j^{(0)}(x,y)dxdy = w_i\delta_{ij} \tag{3.1}$$

with $w_i = \int_{\Delta_0}(v_i^{(0)}(x,y))^2dxdy$.

The numerical solution $u^h(x,y,t)$ in the space V_h^k can be written as:

$$u^h(x,y,t) = \sum_{l=0}^{K} u_0^{(l)}(t)v_l^{(0)}(x,y), \quad \text{for } (x,y) \in \Delta_0$$

and the degrees of freedom $u_0^{(l)}(t)$ are the moments defined by

$$u_0^{(l)}(t) = \frac{1}{w_l}\int_{\Delta_0} u^h(x,y,t)v_l^{(0)}(x,y)dxdy, \quad l = 0,\ldots,K.$$

In order to determine the approximate solution, we evolve the degrees of freedom $u_0^{(l)}(t)$:

$$\begin{aligned}\frac{d}{dt}u_0^{(l)}(t) = \frac{1}{w_l}\Bigg(&\int_{\Delta_0}\left(f(u^h(x,y,t))\frac{\partial}{\partial x}v_l^{(0)}(x,y) + g(u^h(x,y,t))\frac{\partial}{\partial y}v_l^{(0)}(x,y)\right)dxdy\\ &-\int_{\partial\Delta_0}(f(u^h(x,y,t)),g(u^h(x,y,t)))^T\cdot n v_l^{(0)}(x,y)ds\Bigg),\\ l = 0,\ldots,K.&\end{aligned} \tag{3.2}$$

In (3.2) the integral terms can be computed either exactly or by suitable numerical quadratures which are exact for polynomials of degree up to $2k$ for the element integral and up to $2k+1$ for the edge integral. In this paper, we use A_G Gaussian points ($A_G = 6$ for $k = 1$ and $A_G = 7$ for $k = 2$) for the element quadrature and E_G Gaussian points ($E_G = 2$ for $k = 1$ and $E_G = 3$ for $k = 2$) for the edge quadrature:

$$\begin{aligned}&\int_{\Delta_0}\left(f(u^h(x,y,t))\frac{\partial}{\partial x}v_l^{(0)}(x,y) + g(u^h(x,y,t))\frac{\partial}{\partial y}v_l^{(0)}(x,y)\right)dxdy\\ &\quad\approx|\Delta_0|\sum_G\sigma_G\Bigg(f(u^h(x_G,y_G,t))\frac{\partial}{\partial x}v_l^{(0)}(x_G,y_G)\\ &\qquad+g(u^h(x_G,y_G,t))\frac{\partial}{\partial y}v_l^{(0)}(x_G,y_G)\Bigg),\end{aligned} \tag{3.3}$$

$$\begin{aligned}&\int_{\partial\Delta_0}(f(u^h(x,y,t)),g(u^h(x,y,t)))^T\cdot n v_l^{(0)}(x,y)ds\\ &\quad\approx\sum_{ll=1}^{3}|\partial\Delta_{0_{ll}}|\sum_G\bar{\sigma}_G\big((f(u^h(\bar{x}_{ll_G},\bar{y}_{ll_G},t)),\\ &\qquad\qquad g(u^h(\bar{x}_{ll_G},\bar{y}_{ll_G},t)))^T\cdot n_{ll}v_l^{(0)}(\bar{x}_{ll_G},\bar{y}_{ll_G})\big),\end{aligned} \tag{3.4}$$

where $(x_G, y_G) \in \Delta_0$ and $(\bar{x}_{ll_G}, \bar{y}_{ll_G}) \in \partial\Delta_{0ll}$ are the Gaussian quadrature points, and σ_G and $\bar{\sigma}_G$ are the Gaussian quadrature weights. Since the edge integral is on element boundaries where the numerical solution can be discontinuous, the flux $(f(u^h(x, y, t)), g(u^h(x, y, t)))^T \cdot n$ is replaced by a monotone numerical flux. The simple Lax-Friedrichs flux is used in all of our numerical tests. The semi-discrete scheme (3.2) is discretized in time by a non-linear stable Runge-Kutta time discretization.

The method described above can compute solutions of (1.1), which are either smooth or have weak discontinuities, without further modification. If the discontinuities are strong, however, the scheme will generate significant oscillations and even nonlinear instability. To avoid such difficulties, we use limiter technique after each Runge-Kutta inner stage (or after the complete Runge-Kutta time step) to control the spurious oscillations of the numerical solution.

The limiter procedure can be divided into two steps:

(1) Identify the "troubled cells", namely those cells which might need the limiting procedure.
(2) Replace the solution polynomials in those troubled cells by reconstructed polynomials using the HWENO methodology which maintain the original cell averages (conservation), have the same orders of accuracy as before, but are less oscillatory.

In this paper, we will use two indicators to detect "troubled cells".

One is TVB indicator that adopted in [7]. The main procedure is as follows: We use (x_{m_ℓ}, y_{m_ℓ}), $\ell = 1, 2, 3$, to denote the midpoints of the edges on the boundary of the target cell Δ_0, and (x_{b_i}, y_{b_i}), $i = 1, 2, 3$, to denote the barycenters of the neighboring triangles Δ_i, $i = 1, 2, 3$, as shown in Fig. 2.

We then have

$$\begin{aligned} x_{m_1} - x_{b_0} &= \alpha_1(x_{b_1} - x_{b_0}) + \alpha_2(x_{b_3} - x_{b_0}), \\ y_{m_1} - y_{b_0} &= \alpha_1(y_{b_1} - y_{b_0}) + \alpha_2(y_{b_3} - y_{b_0}) \end{aligned} \tag{3.5}$$

with nonnegative $\alpha_1,\ \alpha_2$, which depend only on (x_{m_1}, y_{m_1}) and the geometry. We then define

$$\tilde{u}^h(x_{m_1}, y_{m_1}, t) \equiv u^h(x_{m_1}, y_{m_1}, t) - u_0^{(0)}(t), \tag{3.6}$$

$$\Delta u(x_{m_1}, y_{m_1}, t) \equiv \alpha_1(u_1^{(0)}(t) - u_0^{(0)}(t)) + \alpha_2(u_3^{(0)}(t) - u_0^{(0)}(t)), \tag{3.7}$$

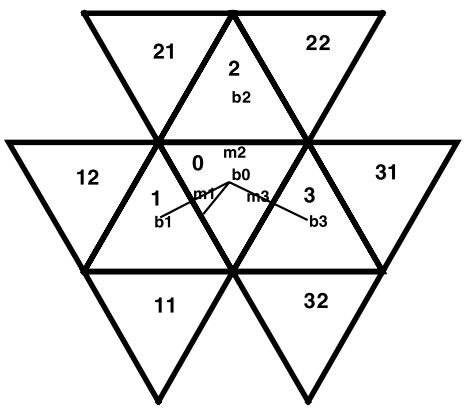

Fig. 2 The limiting diagram

where $u_l^{(0)}(t)$ denotes the cell average on Δ_l $(l = 0, 1, 2, 3)$.

Using the TVB modified *minmod* function [25] defined as

$$\tilde{m}(a_1, a_2) = \begin{cases} a_1 & \text{if } |a_1| \le M|\Delta_0|, \\ \begin{cases} s\min(|a_1|, |a_2|) & \text{if } s = \text{sign}(a_1) = \text{sign}(a_2) \\ 0 & \text{otherwise} \end{cases} & \text{otherwise} \end{cases} \tag{3.8}$$

where $M > 0$ is the TVB constant whose choice is problem dependent, we can compute the quantity

$$\tilde{u}^{mod} = \tilde{m}(\tilde{u}^h(x_{m_1}, y_{m_1}, t), \gamma\Delta u(x_{m_1}, y_{m_1}, t)) \tag{3.9}$$

with $\gamma > 1$ (we take $\gamma = 1.5$ in our numerical tests). If $\tilde{u}^{mod} \neq \tilde{u}^h(x_{m_1}, y_{m_1}, t)$, Δ_0 is marked as a "troubled cell" for further reconstruction. This procedure is repeated for the other two edges of Δ_0 as well. Since the HWENO reconstruction maintains high order accuracy in the troubled cells, it is less crucial to choose an accurate M.

The other indicator is the usage of the shock detect technique adopted in [14], we termed it as KXRCF indicator. We divide the boundary of the target cell Δ_0 into two parts: $\partial\Delta_0^+$ and $\partial\Delta_0^-$, where the flow is into $(v \cdot n < 0)$ and out of $(v \cdot n > 0)$ Δ_0 respectively. Then the target cell Δ_0 is identified as "troubled cells" when

$$\frac{|\int_{\partial\Delta_0^-}(u^h(x, y, t)|_{\Delta_0} - u^h(x, y, t)|_{\Delta_i})ds|}{h^{\frac{k+1}{2}}|\partial\Delta_0^-| \cdot \|u^h(x, y, t)|_{\Delta_0}\|} > 1, \tag{3.10}$$

where h is the radius of the circumscribed circle in Δ_0, Δ_i is neighbor of Δ_0 on side of $\partial\Delta_0^-$.

For the troubled cells, we reconstruct the polynomial solutions while retaining their cell averages. In other words, we reconstruct the degrees of freedom $u_0^{(l)}(t)$, $l = 1, \ldots, K$ and retain only the cell average $u_0^{(0)}(t)$.

For the $k = 1$ case, we summarize the procedure to reconstruct the first order moments $u_0^{(1)}(t)$ and $u_0^{(2)}(t)$ in the troubled cell Δ_0 using the HWENO reconstruction procedure. For the simplicity, we rewrite $u^{(*)}(t)$ to be $u^{(*)}$ if it will not cause confusion.

Step 3.1.1. We select the big stencil as $S = \{\Delta_0, \Delta_1, \Delta_2, \Delta_3\}$. Then we construct polynomial $P(x, y)$ to approximate u by requiring that it has the same cell average as $u^{(0)}$ on the target cell Δ_0, and matches the cell averages of $u^{(0)}$, $u^{(1)}$ or $u^{(2)}$ on the other triangles in the set $S \setminus \{\Delta_0\}$ in a least square sense.

Step 3.1.2. We then construct six linear polynomials $q_i(x, y)$, $i = 1, \ldots, 6$, satisfying:

$$\frac{1}{|\Delta_\ell|}\int_{\Delta_\ell} q_i(x, y)dxdy = u_\ell^{(0)}, \tag{3.11}$$

$$\frac{1}{\int_{\Delta_{\ell_x}}(v_1^{(\ell_x)}(x, y))^2dxdy}\int_{\Delta_{\ell_x}} q_i(x, y)v_1^{(\ell_x)}(x, y)dxdy = u_{\ell_x}^{(1)}, \tag{3.12}$$

$$\frac{1}{\int_{\Delta_{\ell_y}}(v_2^{(\ell_y)}(x, y))^2dxdy}\int_{\Delta_{\ell_y}} q_i(x, y)v_2^{(\ell_y)}(x, y)dxdy = u_{\ell_y}^{(2)}. \tag{3.13}$$

For

$$i = 1, \quad \ell = 0, 1, 2; \qquad i = 2, \quad \ell = 0, 2, 3;$$

$$i=3,\quad \ell=0,3,1;\qquad i=4,\quad \ell=0,\quad \ell_x=1,\quad \ell_y=1;$$
$$i=5,\quad \ell=0,\quad \ell_x=2,\quad \ell_y=2;\qquad i=6,\quad \ell=0,\quad \ell_x=3,\quad \ell_y=3.$$

Step 3.1.3. We find the combination coefficients, also called linear weights, denoted by $\gamma_1^{(l)},\ldots,\gamma_6^{(l)}$, $l=1,2$, satisfying:

$$\int_{\Delta_0} P(x,y)v_l^{(0)}(x,y)dxdy=\sum_{i=1}^{6}\gamma_i^{(l)}\int_{\Delta_0} q_i(x,y)v_l^{(0)}(x,y)dxdy,$$
$$l=1,2. \tag{3.14}$$

The linear weights are achieved by asking for

$$\min\left(\sum_{i=1}^{6}(\gamma_i^{(l)})^2\right),\quad l=1,2. \tag{3.15}$$

By doing so, we can get the linear weights uniquely but can not maintain them positively all the time, we can use the methods that produced in [12, 24] to overcome this drawback.

Step 3.1.4. We compute the smoothness indicators, denote by β_i, $i=1,\ldots,6$, for the smaller stencils S_i, $i=1,\ldots,6$, which measure how smooth the functions $q_i(x,y)$, $i=1,\ldots,6$ are in the target cell Δ_0. The smaller these smoothness indicators, the smoother the functions are in the target cell. We use the same recipe for the smoothness indicators as in [12, 13]:

$$\beta_i=\sum_{|\ell|=1}^{k}|\Delta_0|^{|\ell|-1}\int_{\Delta_0}\left(\frac{\partial^{|\ell|}}{\partial x^{\ell_1}\partial y^{\ell_2}}q_i(x,y)\right)^2 dxdy \tag{3.16}$$

where $\ell=(\ell_1,\ell_2)$.

Step 3.1.5. We compute the non-linear weights based on the smoothness indicators:

$$\omega_i=\frac{\bar{\omega}_i}{\sum_{\ell=1}^{6}\bar{\omega}_\ell},\qquad \bar{\omega}_\ell=\frac{\gamma_\ell}{(\varepsilon+\beta_\ell)^2}. \tag{3.17}$$

Here ε is a small positive number to avoid the denominator to become zero. We take $\varepsilon=10^{-6}$ in our computation.

The moments of the reconstructed polynomial are then given by:

$$u_0^{(l)}(t)=\frac{1}{\int_{\Delta_0}(v_l^{(0)}(x,y))^2dxdy}\sum_{i=1}^{6}\omega_i^{(l)}\int_{\Delta_0} q_i(x,y)v_l^{(0)}(x,y)dxdy,$$
$$l=1,2. \tag{3.18}$$

For the $k=2$ case, the procedure to reconstruct the first and second order moments $u_0^{(1)}(t)$, $u_0^{(2)}(t)$, $u_0^{(3)}(t)$, $u_0^{(4)}(t)$ and $u_0^{(5)}(t)$ in the troubled cell Δ_0 is analogous to that for the $k=1$ case. The troubled cell and its neighboring cells are shown in Fig. 2.

Step 3.2.1. We select the big stencil as $S=\{\Delta_0,\Delta_1,\Delta_2,\Delta_3,\ \Delta_{11},\Delta_{12},\Delta_{21},\ \Delta_{22},\ \Delta_{31},\ \Delta_{32}\}$. Then we construct polynomial $Q(x,y)$ to approximate u by requiring that it has the same cell average as $u^{(0)}$ on the target cell Δ_0 and matches the cell averages of $u^{(0)}$, $u^{(1)}$ or $u^{(2)}$ on the other triangles in the set $S\setminus\{\Delta_0\}$ in a least square sense.

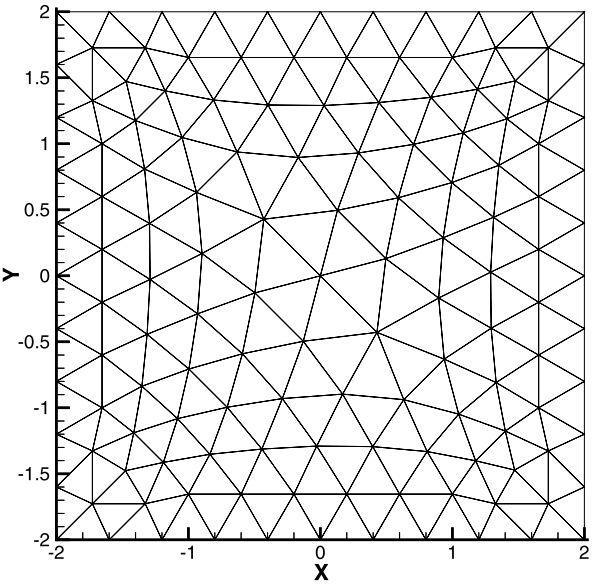

Fig. 3 Burgers equation. Sample mesh. The mesh points on the boundary are uniformly distributed with cell length $h = 4/10$

Table 1 $u_t + (\frac{u^2}{2})_x + (\frac{u^2}{2})_y = 0$. $u(x, y, 0) = 0.5 + \sin(\pi(x + y)/2)$. Periodic boundary conditions in both directions. $t = 0.5/\pi$. L^1 and L^∞ errors. HWENO schemes

h	HWENO3				HWENO4			
	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
4/10	1.67E–2		7.73E–2		1.64E–3		9.10E–3	
4/20	3.61E–3	2.21	2.04E–2	1.92	1.51E–4	3.44	1.57E–3	2.53
4/40	5.39E–4	2.74	3.18E–3	2.68	1.19E–5	3.67	1.64E–4	3.26
4/80	7.33E–5	2.88	4.62E–4	2.78	7.87E–7	3.92	1.03E–5	3.98
4/160	9.83E–6	2.90	6.89E–5	2.74	5.05E–8	3.96	7.96E–7	3.71

Step 3.2.2. We can then construct quadratic polynomials $q_i(x, y)$, $i = 1, \ldots, 9$, which satisfy the following conditions:

$$\frac{1}{|\Delta_\ell|} \int_{\Delta_\ell} q_i(x, y) dx dy = u_\ell^{(0)}, \tag{3.19}$$

$$\frac{1}{\int_{\Delta_{\ell_x}} (v_1^{(\ell_x)}(x, y))^2 dx dy} \int_{\Delta_{\ell_x}} q_i(x, y) v_1^{(\ell_x)}(x, y) dx dy = u_{\ell_x}^{(1)}, \tag{3.20}$$

$$\frac{1}{\int_{\Delta_{\ell_y}} (v_2^{(\ell_y)}(x, y))^2 dx dy} \int_{\Delta_{\ell_y}} q_i(x, y) v_2^{(\ell_y)}(x, y) dx dy = u_{\ell_y}^{(2)}. \tag{3.21}$$

Table 2 $u_t + (\frac{u^2}{2})_x + (\frac{u^2}{2})_y = 0$. $u(x, y, 0) = 0.5 + \sin(\pi(x + y)/2)$. Periodic boundary conditions in both directions. $t = 0.5/\pi$. L^1 and L^∞ errors. RKDG with the HWENO limiter (TVB indicator, constant $M = 0.01$) compared to RKDG without limiter

	h	DG with HWENO limiter				DG without limiter			
		L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
P^1	4/10	5.77E–2		7.30E–1		2.41E–2		2.56E–1	
	4/20	1.45E–2	1.99	2.55E–1	1.51	6.07E–3	1.99	7.54E–2	1.77
	4/40	3.57E–3	2.02	6.40E–2	2.00	1.53E–3	1.98	2.14E–2	1.81
P^2	4/80	8.80E–4	2.02	1.83E–2	1.80	3.91E–4	1.97	5.71E–3	1.91
	4/160	1.74E–4	2.34	4.24E–3	2.12	9.87E–5	1.99	1.55E–3	1.88
	4/10	5.61E–3		1.03E–1		1.70E–3		5.28E–2	
	4/20	4.99E–4	3.49	9.15E–3	3.49	2.45E–4	2.79	8.19E–3	2.69
	4/40	5.69E–5	3.13	1.56E–3	2.55	3.17E–5	2.95	1.55E–3	2.39
	4/80	6.88E–6	3.05	2.37E–4	2.71	4.01E–6	2.98	2.37E–4	2.71
	4/160	8.59E–7	3.00	3.20E–5	2.89	5.03E–7	3.00	3.20E–5	2.89

Table 3 $u_t + (\frac{u^2}{2})_x + (\frac{u^2}{2})_y = 0$. $u(x, y, 0) = 0.5 + \sin(\pi(x + y)/2)$. Periodic boundary conditions in both directions. $t = 0.5/\pi$. Time (seconds)

HWENO3	HWENO4	DG with HWENO limiter		DG without limiter	
649	854	P^1	562	P^1	198
		P^2	697	P^2	356

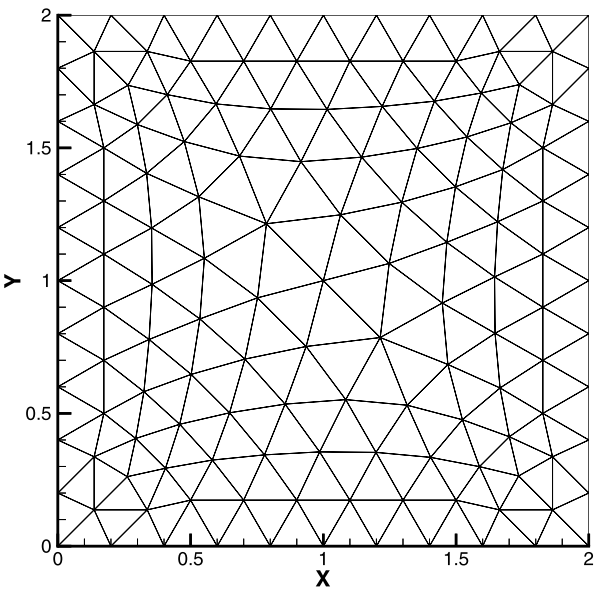

Fig. 4 2D-Euler equations. Sample mesh. The mesh points on the boundary are uniformly distributed with cell length $h = 2/10$

Table 4 2D-Euler equations: initial data $\rho(x, y, 0) = 1 + 0.2\sin(\pi(x + y))$, $u(x, y, 0) = 0.7$, $v(x, y, 0) = 0.3$, and $p(x, y, 0) = 1$. Periodic boundary conditions in both directions. $t = 2.0$. L^1 and L^∞ errors. HWENO schemes

h	HWENO3				HWENO4			
	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
2/10	3.84E–2		7.18E–2		6.28E–3		1.32E–2	
2/20	6.39E–3	2.59	1.23E–2	2.54	2.53E–4	4.63	6.49E–4	4.35
2/40	8.17E–4	2.97	1.88E–3	2.71	8.44E–6	4.90	2.44E–5	4.73
2/80	9.07E–5	3.17	2.17E–4	3.12	3.06E–7	4.78	9.67E–7	4.66
2/160	1.02E–5	3.15	3.59E–5	2.60	1.58E–8	4.28	5.45E–8	4.15

Table 5 2D-Euler equations: initial data $\rho(x, y, 0) = 1 + 0.2\sin(\pi(x + y))$, $u(x, y, 0) = 0.7$, $v(x, y, 0) = 0.3$, and $p(x, y, 0) = 1$. Periodic boundary conditions in both directions. $t = 2.0$. L^1 and L^∞ errors. RKDG with the HWENO limiter (TVB indicator, constant $M = 0.01$) compared to RKDG without limiter

	h	DG with HWENO limiter				DG without limiter			
		L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
P^1	2/10	5.29E–2		1.11E–1		4.39E–3		2.23E–2	
	2/20	2.31E–2	1.19	6.07E–2	0.88	1.03E–3	2.08	5.42E–3	2.04
	2/40	7.63E–3	1.60	2.31E–2	1.39	2.54E–4	2.02	1.29E–3	2.06
	2/80	2.02E–3	1.91	7.89E–3	1.55	6.38E–5	1.99	3.27E–4	1.98
	2/160	3.83E–4	2.40	2.48E–3	1.67	1.62E–5	1.97	8.48E–5	1.95
P^2	2/10	2.30E–3		1.33E–2		4.48E–4		5.94E–3	
	2/20	3.29E–4	2.81	1.69E–3	2.98	6.17E–5	2.86	1.14E–3	2.38
	2/40	4.45E–5	2.89	2.78E–4	2.60	7.05E–6	3.12	1.94E–4	2.56
	2/80	5.51E–6	3.01	4.17E–5	2.74	7.76E–7	3.18	2.87E–5	2.76
	2/160	6.95E–7	2.99	5.17E–6	3.00	1.10E–7	2.81	3.62E–6	2.99

Table 6 2D-Euler equations: initial data $\rho(x, y, 0) = 1 + 0.2\sin(\pi(x + y))$, $u(x, y, 0) = 0.7$, $v(x, y, 0) = 0.3$, and $p(x, y, 0) = 1$. Periodic boundary conditions in both directions. $t = 2.0$. Time (seconds)

HWENO3	HWENO4	DG with HWENO limiter		DG without limiter	
43890	122118	P^1	42085	P^1	12024
		P^2	113475	P^2	40527

Table 7 Double Mach refection problem. The maximum and average percentages of troubled cells subject to the HWENO limiting

Double Mach refection problem. Troubled cells			
Indicator type		TVB, $M = 100$	KXRCF
P^1	maximum percentage	8.56	2.73
	average percentage	2.87	2.04
P^2	maximum percentage	10.11	4.12
	average percentage	4.20	3.01

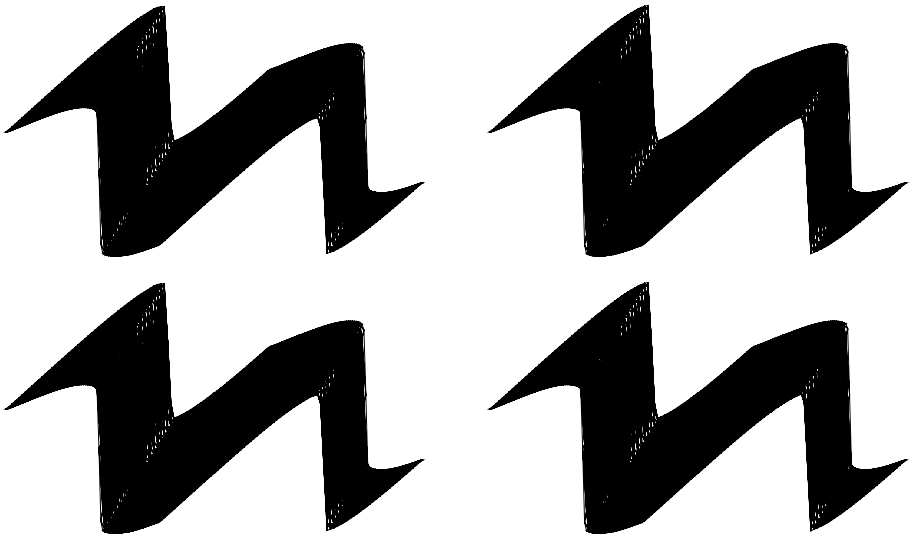

Fig. 5 Burgers' equation. $t = 1.5/\pi$. The surface of the solution. *Left*: second order ($k = 1$); *right*: third order ($k = 2$) RKDG with the HWENO limiter. *Top*: TVB indicator, constant $M = 0.01$; *bottom*: KXRCF indicator. The mesh points on the boundary are uniformly distributed with cell length $h = 4/80$

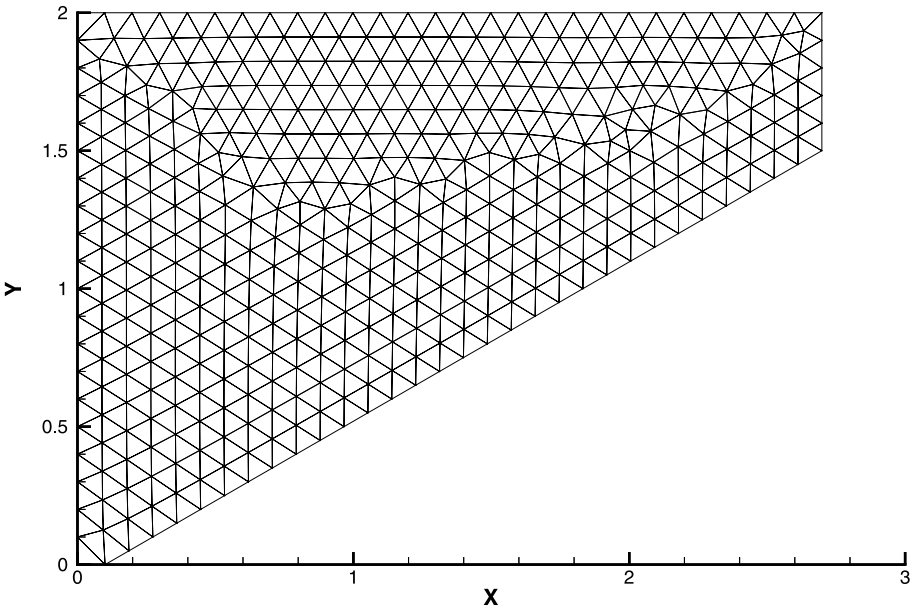

Fig. 6 Double Mach refection problem. Sample mesh. The mesh points on the boundary are uniformly distributed with cell length $h = 1/10$

For

$$i = 1, \quad \ell = 0, 1, 11, 12, 3, 32; \qquad i = 2, \quad \ell = 0, 1, 11, 12, 2, 21;$$

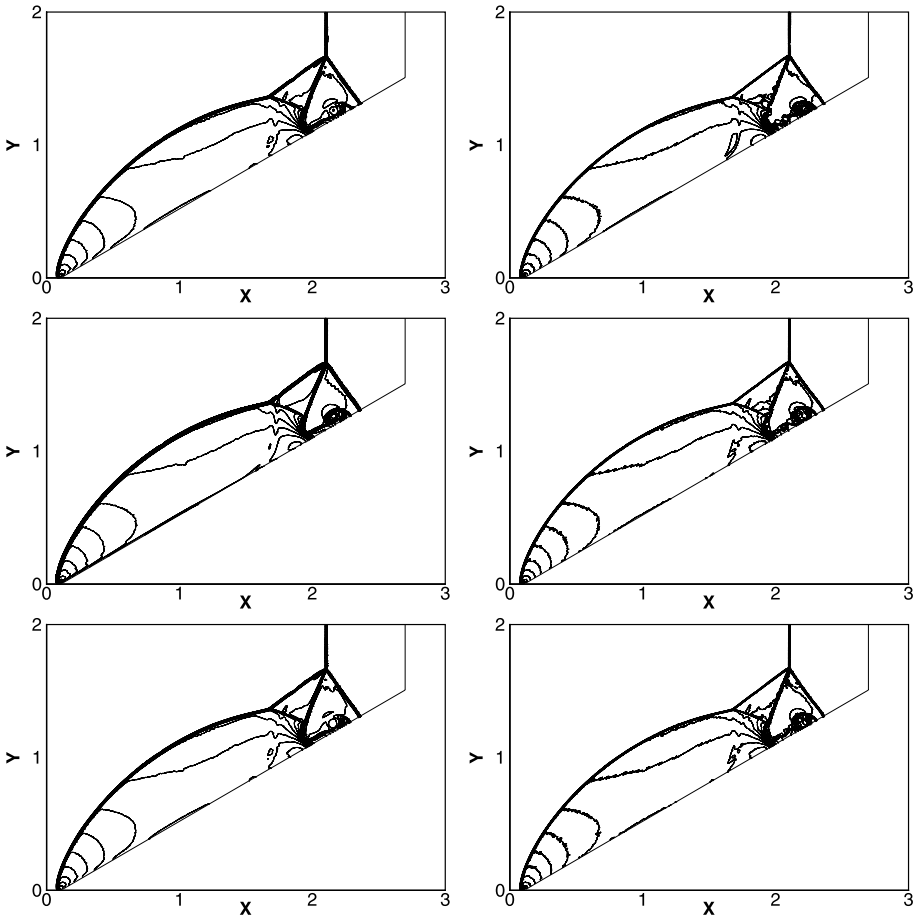

Fig. 7 Double Mach refection problem. 30 equally spaced density contours from 1.5 to 22.7. *Left*: second order ($k = 1$); *right*: third order ($k = 2$) RKDG with the HWENO limiter. *Top*: TVB indicator, constant $M = 100$; *middle*: KXRCF indicator; *bottom*: *left*: HWENO3 scheme, *right*: HWENO4 scheme. The mesh points on the boundary are uniformly distributed with cell length $h = 1/300$

Table 8 Forward step problem. The maximum and average percentages of troubled cells subject to the HWENO limiting

Forward step problem. Troubled cells			
Indicator type		TVB, $M = 100$	KXRCF
P^1	maximum percentage	3.13	3.26
	average percentage	2.27	1.19
P^2	maximum percentage	4.56	5.09
	average percentage	3.75	3.18

$$i = 3, \quad \ell = 0, 2, 21, 22, 1, 12; \qquad i = 4, \quad \ell = 0, 2, 21, 22, 3, 31;$$
$$i = 5, \quad \ell = 0, 3, 31, 32, 2, 22; \qquad i = 6, \quad \ell = 0, 3, 31, 32, 1, 11;$$

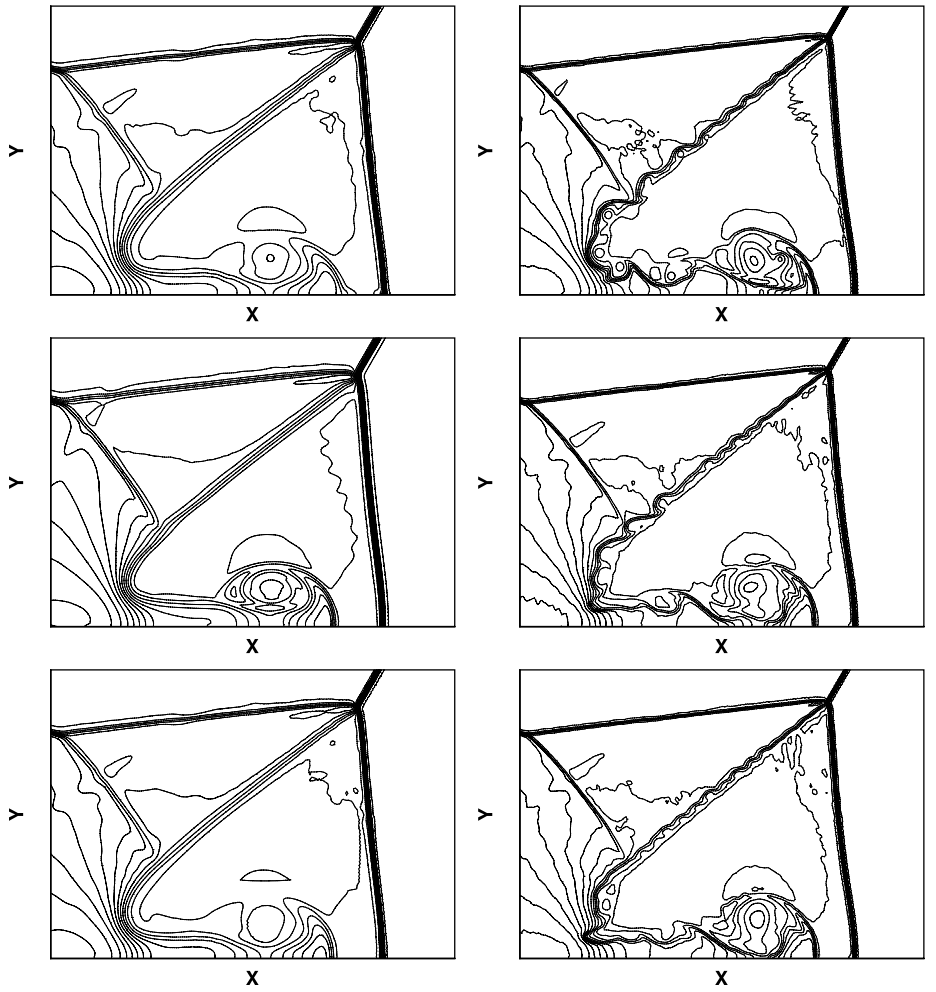

Fig. 8 Double Mach refection problem. Zoom-in pictures around the Mach stem. 30 equally spaced density contours from 1.5 to 22.7. *Left*: second order ($k = 1$); *right*: third order ($k = 2$) RKDG with the HWENO limiter. *Top*: TVB indicator, constant $M = 100$; *middle*: KXRCF indicator; *bottom*: *left*: HWENO3 scheme, *right*: HWENO4 scheme. The mesh points on the boundary are uniformly distributed with cell length $h = 1/30$

$$
\begin{aligned}
&i = 7, \quad \ell = 0, 1, 11, 12, \quad \ell_x = 1, \quad \ell_y = 1;\\
&i = 8, \quad \ell = 0, 2, 21, 22, \quad \ell_x = 2, \quad \ell_y = 2;\\
&i = 9, \quad \ell = 0, 3, 31, 32, \quad \ell_x = 3, \quad \ell_y = 3.
\end{aligned}
$$

The remaining steps are the same as those for the $k = 1$ case. Finally, the moments of the reconstructed polynomial are given by:

$$
u_0^{(l)}(t) = \frac{1}{\int_{\Delta_0} (v_l^{(0)}(x, y))^2 dxdy} \sum_{i=1}^{9} \omega_i^{(l)} \int_{\Delta_0} q_i(x, y) v_l^{(0)}(x, y) dxdy,
$$

$$
l = 1, 2, 3, 4, 5. \tag{3.22}
$$

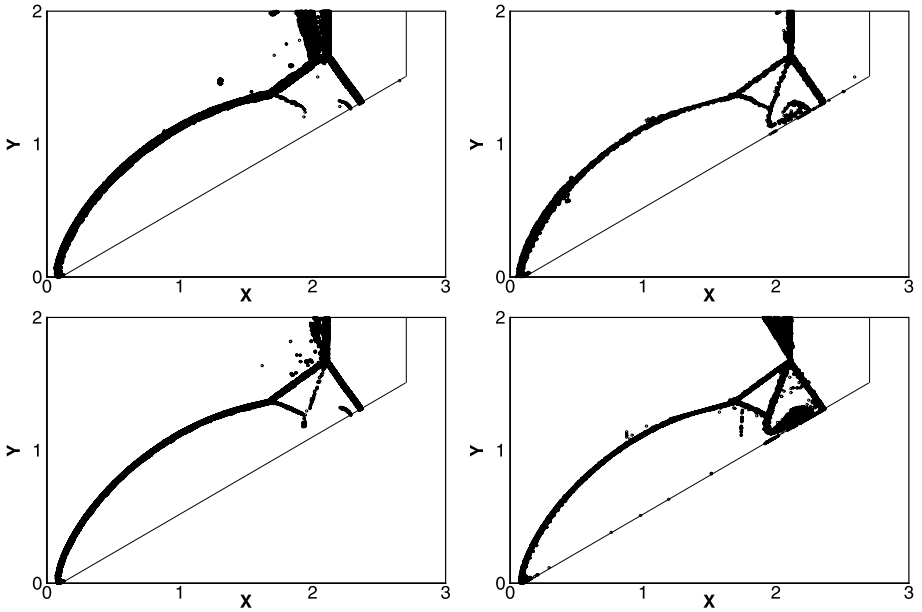

Fig. 9 Double Mach refection problem. Troubled cells. *Circles* denote triangles which are identified as "troubled cells" subject to the HWENO limiting at the final computational step. *Left*: second order ($k = 1$); *right*: third order ($k = 2$) RKDG with the HWENO limiter. *Top*: TVB indicator, constant $M = 100$; *bottom*: KXRCF indicator. The mesh points on the boundary are uniformly distributed with cell length $h = 1/300$

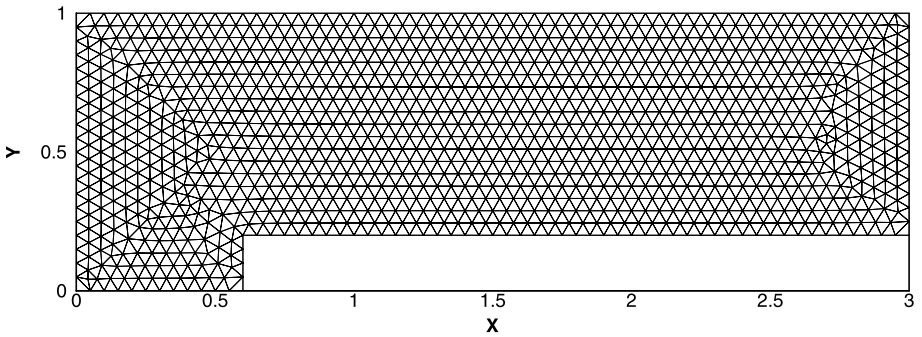

Fig. 10 Forward step problem. Sample mesh. The mesh points on the boundary are uniformly distributed with cell length $h = 1/20$

4 Numerical Results

In this section we provide numerical results to demonstrate the performance of the HWENO schemes and as limiters for the RKDG methods on unstructured meshes described in Sect. 2 and Sect. 3.

We first test the accuracy of the schemes in two dimensional problems.

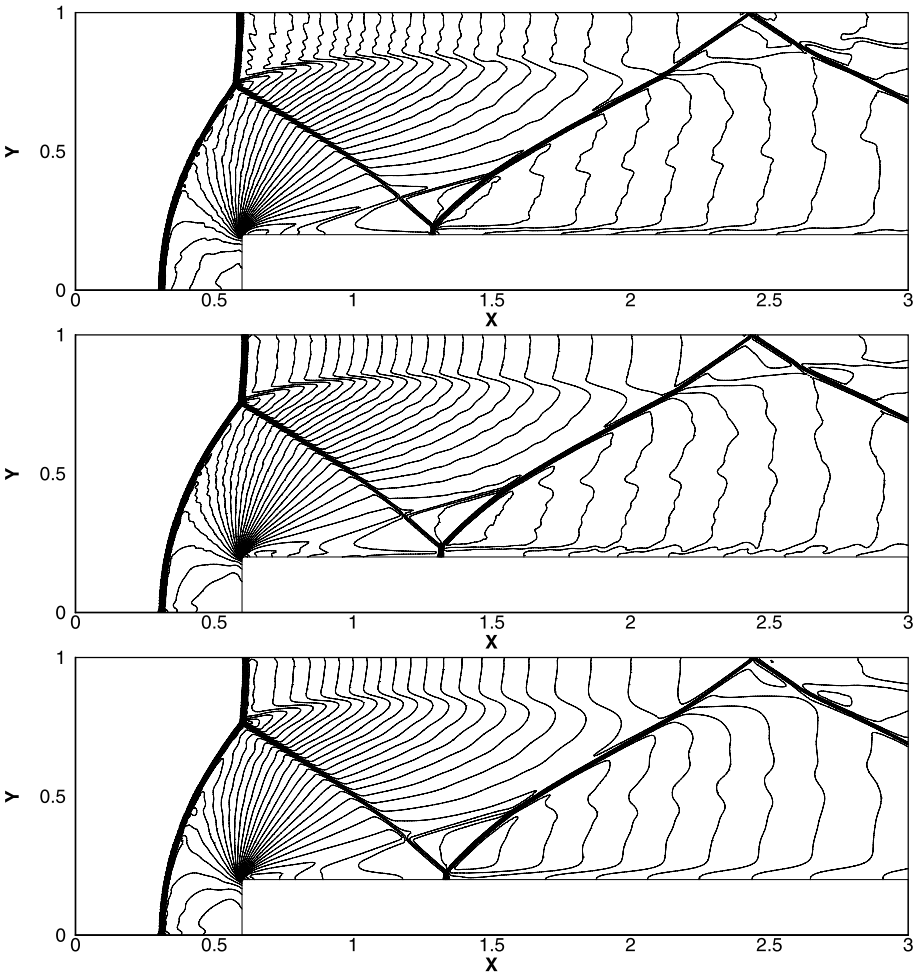

Fig. 11 Forward step problem. 30 equally spaced density contours from 0.32 to 6.15. Second order ($k = 1$) RKDG with the HWENO limiter. *Top*: TVB indicator, constant $M = 100$; *middle*: KXRCF indicator; *bottom*: HWENO3 scheme. The mesh points on the boundary are uniformly distributed with cell length $h = 1/160$

Example 1 We solve the following nonlinear scalar Burgers' equation in two dimensions:

$$u_t + \left(\frac{u^2}{2}\right)_x + \left(\frac{u^2}{2}\right)_y = 0 \tag{4.1}$$

with the initial condition $u(x, y, 0) = 0.5 + \sin(\pi(x + y)/2)$ and periodic boundary conditions in both directions. We compute the solution up to $t = 0.5/\pi$, when the solution is still smooth. For this test case the mesh we used is shown in Fig. 3. The errors and numerical orders of accuracy for HWENO schemes are shown in Table 1 and for the RKDG method with the HWENO limiter comparing with the original RKDG method without limiter are shown in Table 2 and the computing time in Table 3 with the PC environment of Microsoft Windows 2000, 4 CPU 2.6 GHz, 1 GB RAM. We can see that the HWENO schemes and RKDG methods can keep the designed order of accuracy.

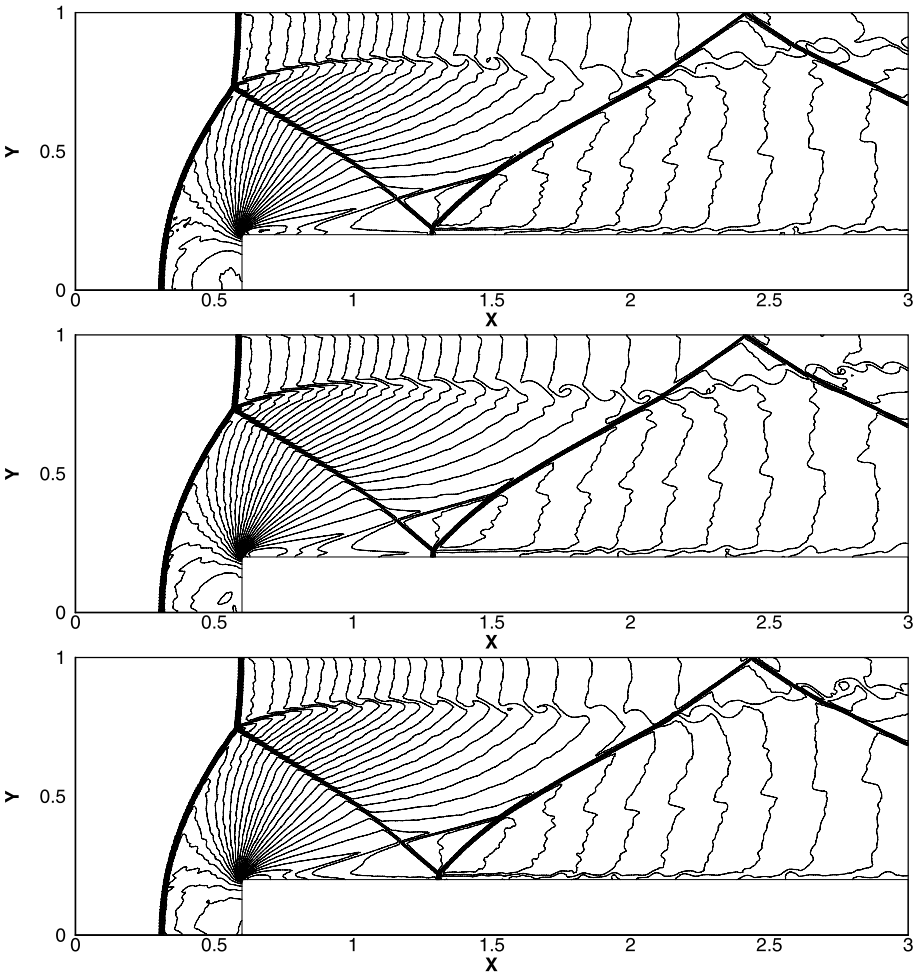

Fig. 12 Forward step problem. 30 equally spaced density contours from 0.32 to 6.15. Third order ($k = 2$) RKDG with the HWENO limiter. *Top*: TVB indicator, constant $M = 100$; *middle*: KXRCF indicator; *bottom*: HWENO4 scheme. The mesh points on the boundary are uniformly distributed with cell length $h = 1/160$

Example 2 We solve the Euler equations

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ \rho u\\ \rho v\\ E\end{pmatrix}+\frac{\partial}{\partial x}\begin{pmatrix}\rho u\\ \rho u^2+p\\ \rho uv\\ u(E+p)\end{pmatrix}+\frac{\partial}{\partial y}\begin{pmatrix}\rho v\\ \rho uv\\ \rho v^2+p\\ v(E+p)\end{pmatrix}=0 \tag{4.2}$$

in which ρ is the density, u is the x-direction velocity, v is the y-direction velocity, E is the total energy, and $p = \frac{E}{\gamma-1} - \frac{1}{2}\rho(u^2+v^2)$ is the pressure, with $\gamma = 1.4$. The initial conditions are: $\rho(x, y, 0) = 1 + 0.2\sin(\pi(x + y))$, $u(x, y, 0) = 0.7$, $v(x, y, 0) = 0.3$, $p(x, y, 0) = 1$. Periodic boundary conditions are applied in both directions. The exact solution is $\rho(x, y, t) = 1 + 0.2\sin(\pi(x + y - t))$. We compute the solution up to $t = 2$. For this test case the mesh we used is shown in Fig. 4. The errors and numerical orders of accuracy for HWENO schemes are shown in Table 4 and for the RKDG method with the HWENO

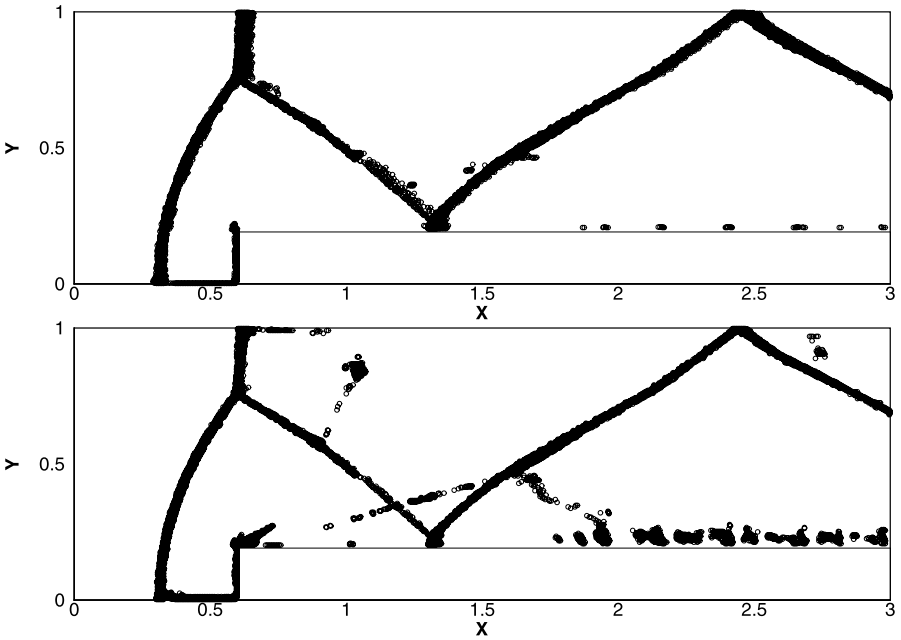

Fig. 13 Forward step problem. Troubled cells. Circles denote triangles which are identified as "troubled cells" subject to the HWENO limiting at the final computational step. Second order ($k = 1$) RKDG with the HWENO limiter. *Top*: TVB indicator, constant $M = 100$; $bottom$: KXRCF indicator. The mesh points on the boundary are uniformly distributed with cell length $h = 1/160$

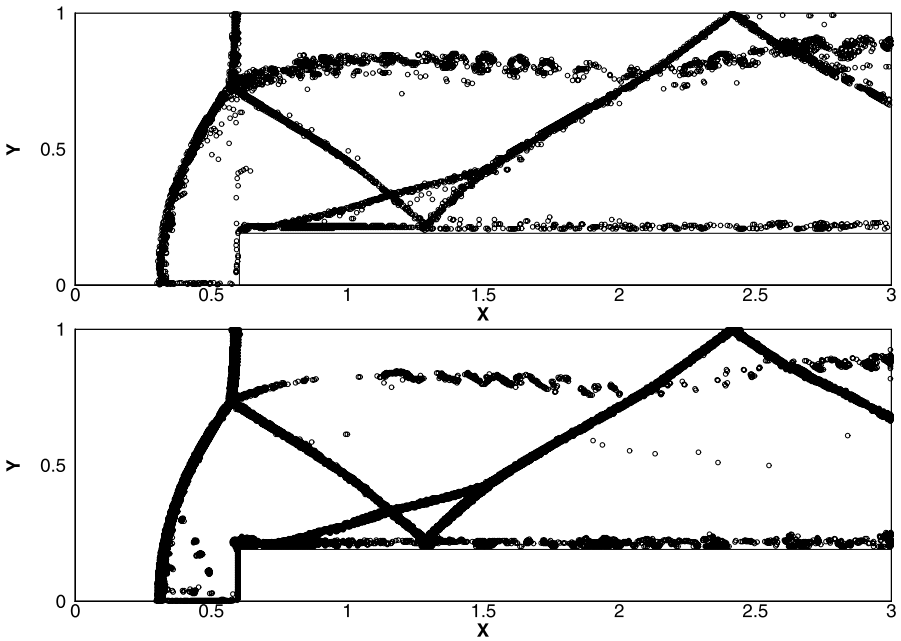

Fig. 14 Forward step problem. Troubled cells. Circles denote triangles which are identified as "troubled cells" subject to the HWENO limiting at the final computational step. Third order ($k = 2$) RKDG with the HWENO limiter. *Top*: TVB indicator, constant $M = 100$; *bottom*: KXRCF indicator. The mesh points on the boundary are uniformly distributed with cell length $h = 1/160$

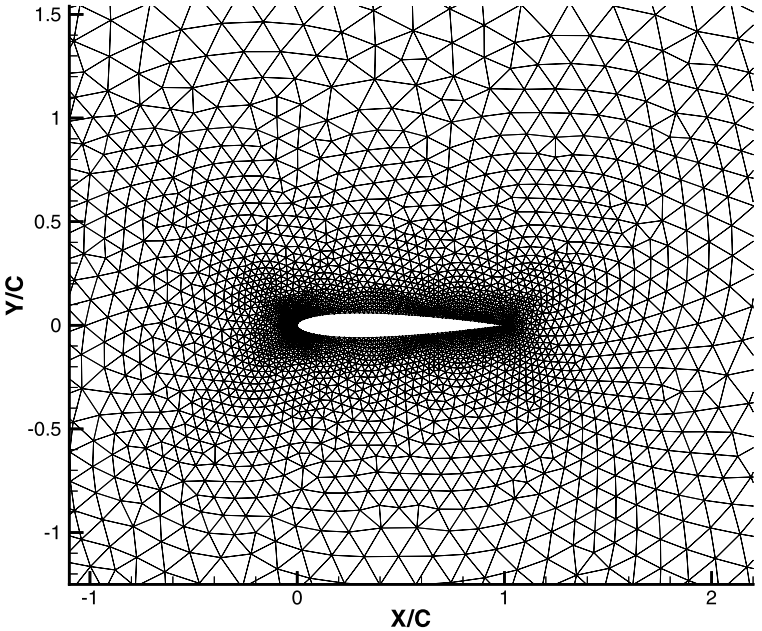

Fig. 15 NACA0012 airfoil mesh zoom in

Table 9 NACA0012 airfoil problem. The maximum and average percentages of troubled cells subject to the HWENO limiting

$M_\infty = 0.8$, angle of attack $\alpha = 1.25°$			
	Indicator type	TVB, $M = 100$	KXRCF
P^1	maximum percentage	8.60	13.8
	average percentage	4.61	6.38
P^2	maximum percentage	14.5	13.2
	average percentage	4.66	6.14
$M_\infty = 0.85$, angle of attack $\alpha = 1°$			
	Indicator type	TVB, $M = 100$	KXRCF
P^1	maximum percentage	12.7	14.4
	average percentage	4.90	7.74
P^2	maximum percentage	14.7	13.5
	average percentage	7.95	7.13

limiter comparing with the original RKDG method without limiter are shown in Table 5 and the computing time in Table 6 with the PC environment of Microsoft Windows 2000, 4 CPU 2.6 GHz, 1 GB RAM. Similar to the previous example, we can see that these schemes again keep the designed order of accuracy.

Remark From these examples, we can see the TVB indicator needs

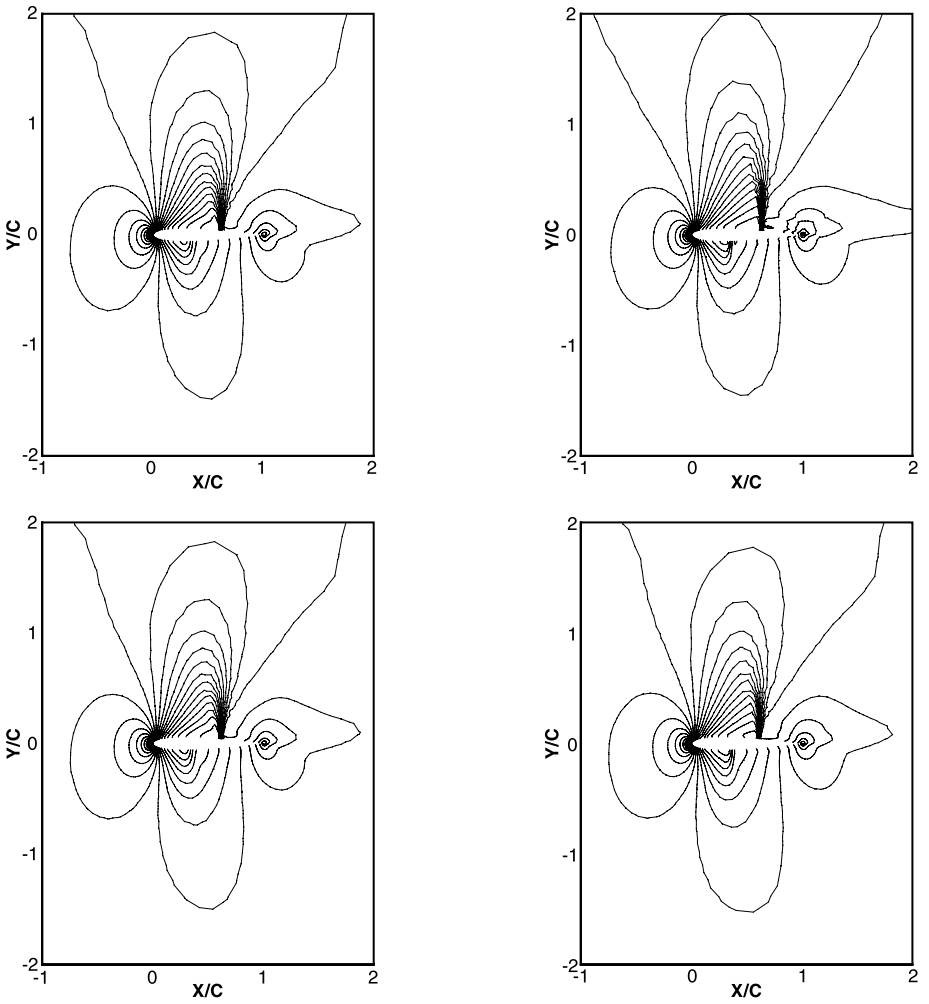

Fig. 16 NACA0012 airfoil. Mach number. $M_\infty = 0.8$, angle of attack $\alpha = 1.25°$, 30 equally spaced Mach number contours from 0.172 to 1.325. *Left*: second order ($k = 1$); *right*: third order ($k = 2$) RKDG with the HWENO limiter. *Top*: TVB indicator, constant $M = 100$; *bottom*: KXRCF indicator

We now test the performance of the HWENO schemes and the RKDG methods with the HWENO limiters for problems containing shocks.

Example 3 We solve the same nonlinear Burgers' equation (4.1) with the same initial condition $u(x, y, 0) = 0.5 + \sin(\pi(x + y)/2)$, except that we plot the results at $t = 1.5/\pi$ when a shock has already appeared in the solution. The solutions are shown in Fig. 5. We can see that the schemes give non-oscillatory shock transitions for this problem.

Example 4 Double Mach reflection problem. This model problem is originally from [29]. We solve the Euler equations (4.2) in a computational domain of a tube which contains a wedge (30°). The shock moves with a Mach number of 10, the undisturbed air ahead

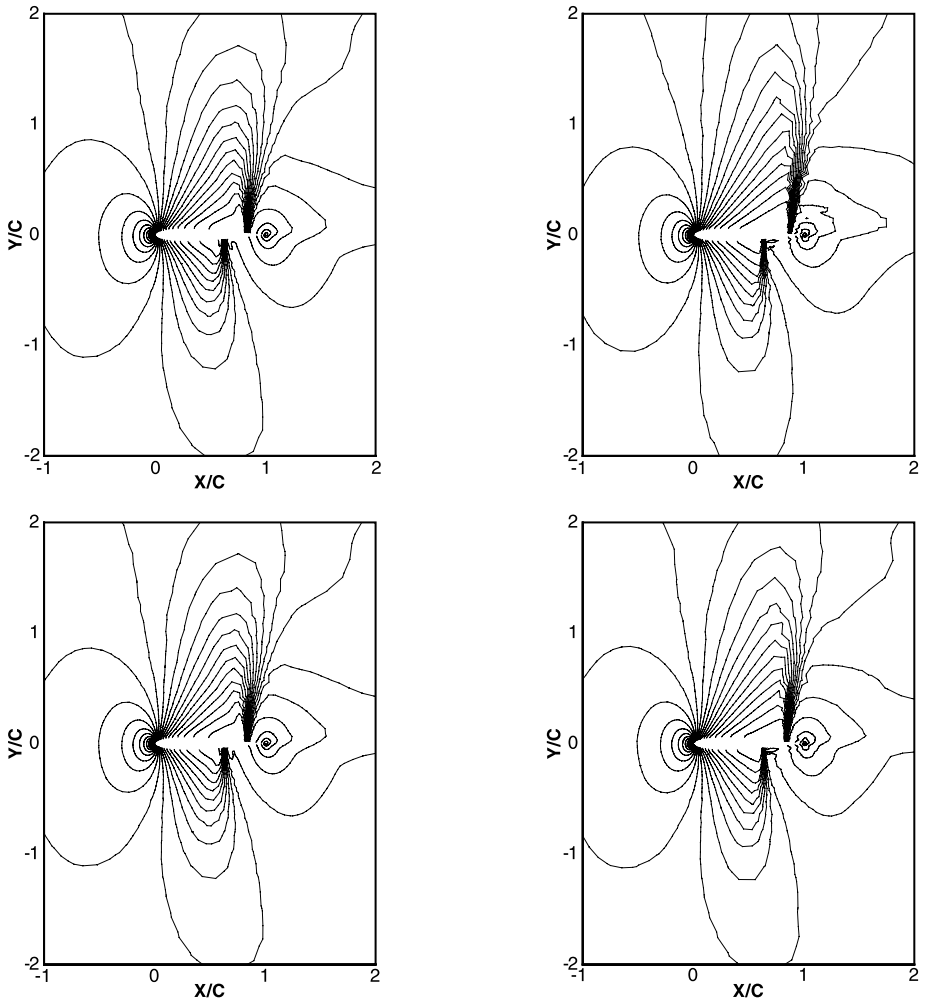

Fig. 17 NACA0012 airfoil. Mach number. $M_\infty = 0.85$, angle of attack $\alpha = 1°$, 30 equally spaced Mach number contours from 0.158 to 1.357. *Left*: second order ($k = 1$); *right*: third order ($k = 2$) RKDG with the HWENO limiter. *Top*: TVB indicator, constant $M = 100$; *bottom*: KXRCF indicator

the shock has a density of 1.4 and a pressure of 1 and the left hand side of the shock has a density of 8, velocity of 8.25 and pressure of 116.5. The results are shown at $t = 0.2$. Two different orders of accuracy for the RKDG with HWENO limiters, $k = 1$ and $k = 2$ (second and third order) and the HWENO schemes are used in the numerical experiments. In Table 7 we document the percentage of cells declared to be "troubled cells" for different orders of accuracy. A mesh coarser than what is used in the actual computation is shown in Fig. 6. The simulation results on the mesh with different orders of accuracy, the "zoomed-in" pictures around the double Mach stem to show more details and associated "troubled cells" distribution at finial computational step are shown in Fig. 7, Fig. 8 and Fig. 9 respectively.

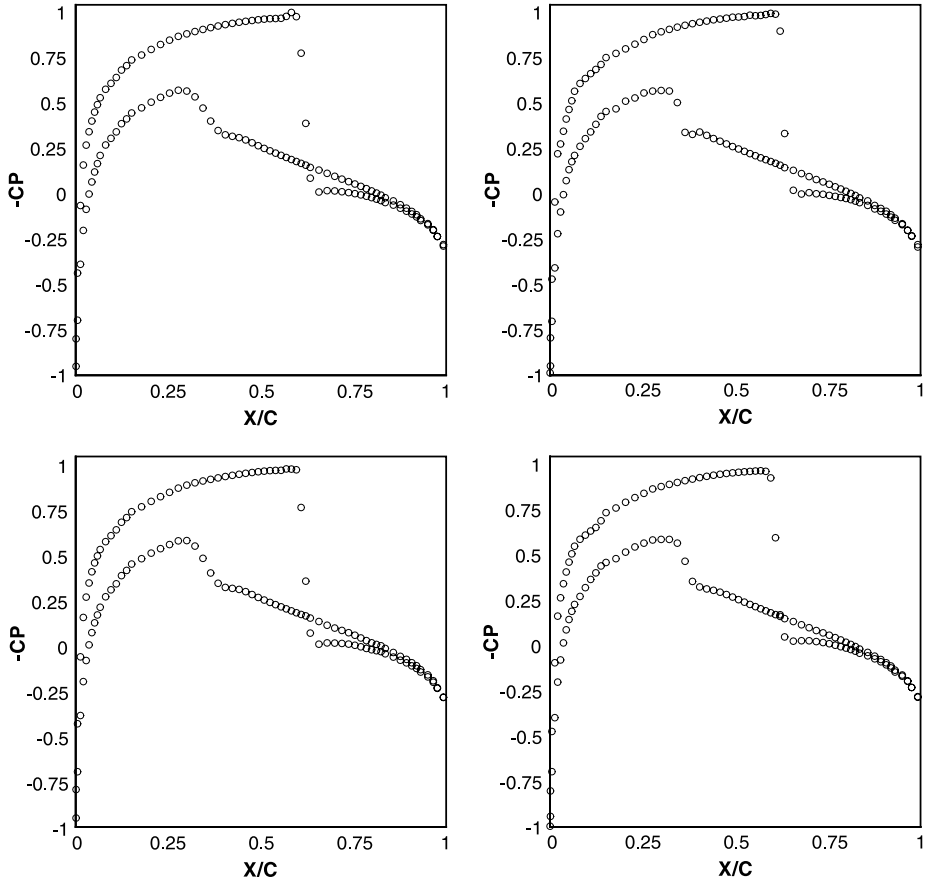

Fig. 18 NACA0012 airfoil. Pressure distribution. $M_\infty = 0.8$, angle of attack $\alpha = 1.25°$. *Left*: second order ($k = 1$); *right*: third order ($k = 2$) RKDG with the HWENO limiter. *Top*: TVB indicator, constant $M = 100$; *bottom*: KXRCF indicator

Example 5 A Mach 3 wind tunnel with a step. This model problem is also originally from [29]. The setup of the problem is as follows. The wind tunnel is 1 length unit wide and 3 length units long. The step is 0.2 length units high and is located 0.6 length units from the left-hand end of the tunnel. The problem is initialized by a right-going Mach 3 flow. Reflective boundary conditions are applied along the wall of the tunnel and inflow/outflow boundary conditions are applied at the entrance/exit. At the corner of the step, there is a singularity. However we do not modify our schemes or refine the mesh near the corner, in order to test the performance of our schemes for such singularity. The results are shown at $t = 4$. In Table 8 we document the percentage of cells declared to be "troubled cells" for different orders of accuracy to identify troubled cells. We present the triangulation of the whole region $[0, 3] \times [0, 1]$ in Fig. 10. And in Fig. 11 and Fig. 12, we show 30 equally spaced density contours from 0.32 to 6.15 computed by the second and third order RKDG schemes with the HWENO limiters and HWENO schemes. We can clearly observe that the third order scheme gives better resolution than the second order scheme, especially for the resolution of the physical instability and roll-up of the contact line. Then associated "troubled cells" distribution at the final computational step are shown in Fig. 13 and Fig. 14.

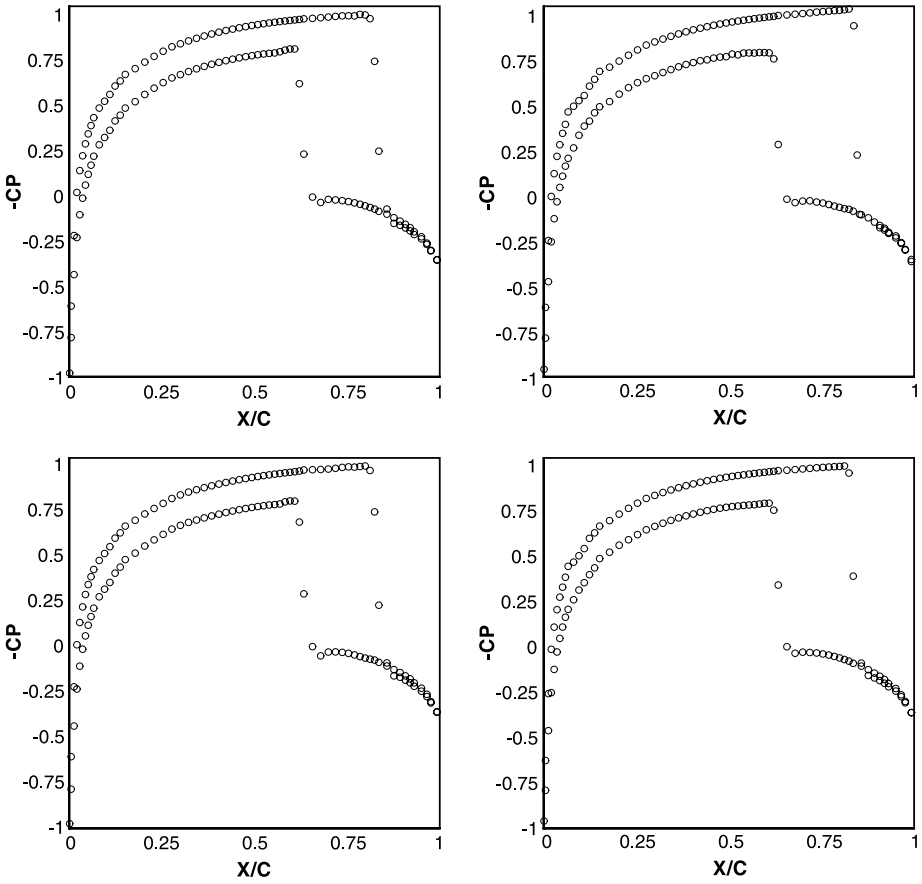

Fig. 19 NACA0012 airfoil. Pressure distribution. $M_\infty = 0.85$, angle of attack $\alpha = 1°$. *Left*: second order ($k = 1$); *right*: third order ($k = 2$) RKDG with the HWENO limiter. *Top*: TVB indicator, constant $M = 100$; *bottom*: KXRCF indicator

Example 6 We consider inviscid Euler transonic flow past a single NACA0012 airfoil configuration with Mach number $M_\infty = 0.8$, angle of attack $\alpha = 1.25°$ and with $M_\infty = 0.85$, angle of attack $\alpha = 1°$. The computational domain is $[-15, 15] \times [-15, 15]$. The mesh used in the computation is shown in Fig. 15, consisting of 9340 elements with the maximum diameter of the circumcircle being 1.4188 and the minimum diameter being 0.0031 near the airfoil. And we use curved cells near the airfoil for computing. The second and third order RKDG scheme with the HWENO limiter are used in the numerical experiments. In Table 9 we document the percentage of cells declared to be "troubled cells" for different orders of accuracy to identify troubled cells. We can see that only a small percentage of cells are declared as "troubled cells". Mach number and pressure distributions are shown in Fig. 16, Fig. 17, Fig. 18 and Fig. 19 respectively. We can see that the third order scheme has better resolution than the second order one.

to choose the parameter M which depends on the solution of the problem. For scalar problems it is possible to estimate M by the initial condition; but, it is more difficult to estimate M for the system case. It is hard to accumulate M properly. And the KXRCF

indicator is based on a strong super-convergence at the outflow boundary of each element in smooth regions for the RKDG method to detect discontinuities and to lower the order of accuracy in the approximation there to avoid spurious oscillations near such discontinuities. This indicator does not need to define parameter M and can detect strong shocks well.

5 Concluding Remarks

We have developed a class of finite volume high order Hermite WENO schemes. And then use HWENO reconstructions as limiters for the RKDG methods to solve hyperbolic conservation laws on unstructured meshes. The main idea is to first identify troubled cells subject to the HWENO limiting, then reconstruct the polynomial solution inside the troubled cells by the HWENO reconstruction using the cell averages and cell derivative averages of neighboring cells, while maintaining the original cell averages of the troubled cells. Numerical results are provided to show that the method is stable, accurate, and robust in maintaining accuracy.

References

1. Biswas, R., Devine, K.D., Flaherty, J.: Parallel, adaptive finite element methods for conservation laws. Appl. Numer. Math. **14**, 255–283 (1994)
2. Burbeau, A., Sagaut, P., Bruneau, C.H.: A problem-independent limiter for high-order Runge-Kutta discontinuous Galerkin methods. J. Comput. Phys. **169**, 111–150 (2001)
3. Cockburn, B., Shu, C.-W.: The Runge-Kutta local projection P1-discontinuous Galerkin finite element method for scalar conservation laws. Math. Model. Numer. Anal. **25**, 337–361 (1991)
4. Cockburn, B., Shu, C.-W.: TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws, II: general framework. Math. Comput. **52**, 411–435 (1989)
5. Cockburn, B., Lin, S.-Y., Shu, C.-W.: TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws, III: one dimensional systems. J. Comput. Phys. **84**, 90–113 (1989)
6. Cockburn, B., Hou, S., Shu, C.-W.: The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws, IV: the multidimensional case. Math. Comput. **54**, 545–581 (1990)
7. Cockburn, B., Shu, C.-W.: The Runge-Kutta discontinuous Galerkin method for conservation laws, V: multidimensional systems. J. Comput. Phys. **141**, 199–224 (1998)
8. Cockburn, B., Shu, C.-W.: Runge-Kutta discontinuous Galerkin method for convection-dominated problems. J. Sci. Comput. **16**, 173–261 (2001)
9. Dougherty, R.L., Edelman, A.S., Hyman, J.M.: Nonnegativity-monotonicity-, or convexity-preserving cubic and quintic Hermite interpolation. Math. Comput. **52**, 471–494 (1989)
10. Friedrichs, O.: Weighted essentially non-oscillatory schemes for the interpolation of mean values on unstructured grids. J. Comput. Phys. **144**, 194–212 (1998)
11. Harten, A., Engquist, B., Osher, S., Chakravathy, S.: Uniformly high order accurate essentially non-oscillatory schemes, III. J. Comput. Phys. **71**, 231–303 (1987)
12. Hu, C., Shu, C.-W.: Weighted essentially non-oscillatory schemes on triangular meshes. J. Comput. Phys. **150**, 97–127 (1999)
13. Jiang, G., Shu, C.-W.: Efficient implementation of weighted ENO schemes. J. Comput. Phys. **126**, 202–228 (1996)
14. 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**, 323–338 (2004)
15. Levy, D., Puppo, G., Russo, G.: Central WENO schemes for hyperbolic systems of conservation laws. Math. Model. Numer. Anal. **33**, 547–571 (1999)
16. Liu, X., Osher, S., Chan, T.: Weighted essentially non-oscillatory schemes. J. Comput. Phys. **115**, 200–212 (1994)
17. Luo, H., Baum, J.D., Lohner, R.: A Hermite WENO-based limiter for discontinuous Galerkin method on unstructured grids. J. Comput. Phys. (in press)

18. Nakamura, T., Tanaka, R., Yabe, T., Takizawa, K.: Exactly conservative semi-Lagrangian schemes for multi-dimensional hyperbolic equations with directional splitting technique. J. Comput. Phys. **174**, 171–207 (2001)
19. 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**, 115–135 (2004)
20. 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)
21. Qiu, J., Shu, C.-W.: On the construction, comparison, and local characteristic decomposition for high order central WENO schemes. J. Comput. Phys. **183**, 187–209 (2002)
22. Qiu, J., Shu, C.-W.: Runge-Kutta discontinuous Galerkin method using WENO limiters. SIAM J. Sci. Comput. **26**, 907–929 (2005)
23. Reed, W.H., Hill, T.R.: Triangular mesh methods for neutron transport equation. Tech. Report LA-UR-73-479, Los Alamos Scientific Laboratory (1973)
24. Shi, J., Hu, C., Shu, C.-W.: A technique of treating negative weights in WENO schemes. J. Comput. Phys. **175**, 108–127 (2002)
25. Shu, C.-W.: TVB uniformly high-order schemes for conservation laws. Math. Comput. **49**, 105–121 (1987)
26. Shu, C.-W., Osher, S.: Efficient implementation of essentially non-oscillatory shock-capturing schemes. J. Comput. Phys. **77**, 439–471 (1988)
27. Shu, C.-W., Osher, S.: Efficient implementation of essentially non-oscillatory shock capturing schemes II. J. Comput. Phys. **83**, 32–78 (1989)
28. Takewaki, H., Nishiguchi, A., Yabe, T.: Cubic interpolated pseudoparticle method (CIP) for solving hyperbolic type equations. J. Comput. Phys. **61**, 261–268 (1985)
29. Woodward, P., Colella, P.: The numerical simulation of two-dimensional fluid flow with strong shocks. J. Comput. Phys. **54**, 115–173 (1984)
30. Zhu, J., Qiu, J., Shu, C.-W., Dumbser, M.: Runge-Kutta discontinuous Galerkin method using WENO limiters, II: unstructured meshes. J. Comput. Phys. **227**, 4330–4353 (2008)