Science in China Series A: Mathematics
Aug., 2008, Vol. 51, No. 8, 1549–1560
www.scichina.com math.scichina.com
www.springerlink.com

A class of the fourth order finite volume Hermite weighted essentially non-oscillatory schemes

ZHU Jun[1,2] & QIU JianXian[1†]

[1] Department of Mathematics, Nanjing University, Nanjing 210093, China
[2] College of Science, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China
(email: zhujun@nuaa.edu.cn, jxqiu@nju.edu.cn)

Abstract In this paper, we developed a class of the fourth order accurate finite volume Hermite weighted essentially non-oscillatory (HWENO) schemes based on the work (Computers & Fluids, 34: 642–663 (2005)) by Qiu and Shu, with Total Variation Diminishing Runge-Kutta time discretization method for the two-dimensional hyperbolic conservation laws. The key idea of HWENO is to evolve both with the solution and its derivative, which allows for using Hermite interpolation in the reconstruction phase, resulting in a more compact stencil at the expense of the additional work. The main difference between this work and the formal one is the procedure to reconstruct the derivative terms. Comparing with the original HWENO schemes of Qiu and Shu, one major advantage of new HWENO schemes is its robust in computation of problem with strong shocks. Extensive numerical experiments are performed to illustrate the capability of the method.

Keywords: finite volume HWENO scheme, conservation laws, Hermite polynomial, TVD Runge-Kutta time discretization method

MSC(2000): 65M06, 65M99, 35L65

1 Introduction

In recent decades, numerical schemes for solving hyperbolic conservation laws have been developed rapidly. These schemes improve the first-order methods of Godunov[1] to arbitrary order of accuracy. In order to achieve uniform high order accuracy, Harten and Osher[2] gave a weaker version of the Total Variation Diminishing (TVD)[3] criterion, and on which they established the framework for the reconstruction of high order essentially non-oscillatory (ENO) type schemes. Then Harten, et al.[4] developed ENO schemes to solve one dimensional problems. The key idea of ENO schemes is applying the most smooth stencil among all candidate stencils to approximate the variables at cell boundaries to a high order of accuracy and avoid oscillations near discontinuities. In [5], Harten extended the finite volume ENO scheme to the two-dimensional hyperbolic conservation laws. Following the work of Harten, Casper[6] and with Atkins[7] considered the finite volume approach in developing multi-dimensional, high order accurate ENO schemes, and Abgrall[8] extended the method on unstructured meshes.

Received March 12, 2008; accepted May 8, 2008
DOI: 10.1007/s11425-008-0105-0
† Corresponding author
This work was partially supported by the National Natural Science Foundation of China (Grant No. 10671097), the European project ADIGMA on the development of innovative solution algorithms for aerodynamic simulations, Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry and the Natural Science Foundation of Jiangsu Province (Grant No. BK2006511)

In 1994, Liu, Osher and Chan[9] proposed a Weighted ENO (WENO) scheme that was constructed from the r-th order ENO schemes to obtain $(r+1)$-th order accuracy. In 1996, Jiang and Shu[10] proposed the framework to construct finite difference WENO schemes from the r-th order (in L^1 norm sense) ENO schemes to get $(2r-1)$-th order accuracy, gave a new way of measuring the smoothness indicators, and emulated the ideas of minimizing the total variation of the approximation. The construction of finite volume WENO methods on unstructured meshes was presented by Friedrichs[11]. Instead of construction of two-dimensional finite volume WENO schemes used dimensional by dimensional methods, Hu and Shu[12] proposed a full dimensional reconstruction methodology for the third order WENO schemes by using a combination of two-dimensional linear polynomials and the fourth order WENO schemes by using a combination of two dimensional quadratic polynomials, and presented a new way of measuring two dimensional smoothness of numerical solutions. In [13, 14], Qiu and Shu developed the methodology, termed as Hermite WENO (HWENO) schemes and applied them as limiters to Runge-Kutta discontinuous Galerkin methods. The key idea of HWENO is to evolve both with the solution and its derivative. This allows for using Hermite interpolation in the reconstruction phase, resulting in a more compact stencil at the expense of additional work.

In this paper, following the ideas of [14], we construct a kind of two-dimensional finite volume HWENO schemes which has fourth order accuracy, the main difference between this work and [14] is the procedure to reconstruct the derivative terms. Comparing with the original HWENO schemes of Qiu and Shu [14], one major advantage of new HWENO schemes is its robust in the computation of problem with strong shocks. The organization of this paper is as follows: In Section 2 we review and construct two-dimensional finite volume HWENO scheme in details and present extensive numerical results in Section 3 to verify the accuracy and stability of this approach. Concluding remarks are given in Section 4.

2 Reconstruction of HWENO scheme

In this section we consider two-dimensional conservation laws (1):

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

For simplicity of presentation, we assume that the mesh is uniform with the cell size $x_{i+1/2} - x_{i-1/2} = \Delta x$, $y_{j+1/2} - y_{j-1/2} = \Delta y$ and cell centers $(x_i, y_j) = (\frac{1}{2}(x_{i+1/2} + x_{i-1/2}), \frac{1}{2}(y_{j+1/2} + y_{j-1/2}))$. We also denote the cells by $I_{ij} = [x_{i-1/2}, x_{i+1/2}] \times [y_{j-1/2}, y_{j+1/2}]$. Let $v = \frac{\partial u}{\partial x}$, $w = \frac{\partial u}{\partial y}$. Taking the derivatives of (1), we obtain

$$\begin{cases} v_t + h_x + r_y = 0, \\ v(x, y, 0) = \dfrac{\partial u_0(x, y)}{\partial x}, \end{cases} \tag{2}$$

$$\begin{cases} w_t + q_x + s_y = 0, \\ w(x, y, 0) = \dfrac{\partial u_0(x, y)}{\partial y}, \end{cases} \tag{3}$$

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 the equations (1), (2) and (3) on a control volume I_{ij} to obtain the semi-discrete

finite volume scheme as:

$$\frac{d}{dt}\overline{u}_{ij} = -\frac{1}{\Delta x\Delta y}\int_{\partial I_{ij}} F\cdot n ds, \tag{4}$$

$$\frac{d}{dt}\overline{v}_{ij} = -\frac{1}{\Delta y}\int_{\partial I_{ij}} H\cdot n ds, \tag{5}$$

$$\frac{d}{dt}\overline{w}_{ij} = -\frac{1}{\Delta x}\int_{\partial I_{ij}} Q\cdot n ds, \tag{6}$$

where

$$\overline{u}_{ij} = \frac{1}{\Delta x\Delta y}\int_{I_{ij}} u\,dxdy, \quad \overline{v}_{ij} = \frac{1}{\Delta y}\int_{I_{ij}} \frac{\partial u}{\partial x}\,dxdy, \quad \overline{w}_{ij} = \frac{1}{\Delta x}\int_{I_{ij}} \frac{\partial u}{\partial y}\,dxdy$$

and $F = (f, g)^T, H = (h, r)^T, Q = (q, s)^T$.

The line integrals in (4)–(6) are discretized by a q-point Gaussian integration formula

$$\int_{\partial I_{ij}} F\cdot n ds \approx |\partial I_{ij}|\sum_{l=1}^{q}\omega_l F(u(G_l,t))\cdot n, \tag{7}$$

$$\int_{\partial I_{ij}} H\cdot n ds \approx |\partial I_{ij}|\sum_{l=1}^{q}\omega_l H(u(G_l,t),v(G_l,t))\cdot n, \tag{8}$$

$$\int_{\partial I_{ij}} Q\cdot n ds \approx |\partial I_{ij}|\sum_{l=1}^{q}\omega_l Q(u(G_l,t),w(G_l,t))\cdot n. \tag{9}$$

Since we are constructing schemes up to fourth-order accuracy, two-point Gaussian will be used in each line integration, and $F(u(G_l,t))\cdot n$, $H(u(G_l,t),v(G_l,t))\cdot n$, $Q(u(G_l,t),w(G_l,t))\cdot n$ are replaced by numerical fluxes such as the Lax-Friedrichs fluxes:

$$f(u(G_l,t)) \approx \frac{1}{2}[f(u^-(G_l,t)) + f(u^+(G_l,t)) - \alpha(u^+(G_l,t) - u^-(G_l,t))], \tag{10}$$

$$\begin{aligned} h(u(G_l,t),v(G_l,t)) \approx \frac{1}{2}[&h(u^-(G_l,t),v^-(G_l,t)) + h(u^+(G_l,t),v^+(G_l,t)) \\ &- \alpha(v^+(G_l,t) - v^-(G_l,t))], \end{aligned} \tag{11}$$

$$\begin{aligned} q(u(G_l,t),w(G_l,t)) \approx \frac{1}{2}[&q(u^-(G_l,t),w^-(G_l,t)) + q(u^+(G_l,t),w^+(G_l,t)) \\ &- \alpha(w^+(G_l,t) - w^-(G_l,t))], \end{aligned} \tag{12}$$

for $G_l = (x_{i\pm 1/2}, y_{j\pm\sqrt{3}/6})$, $u^\pm(G_l,t)$, $v^\pm(G_l,t)$, $w^\pm(G_l,t)$ are the left and right limits of the solutions u, v, w at the cell interface G_l respectively; and

$$g(u(G_l,t)) \approx \frac{1}{2}[g(u^-(G_l,t)) + g(u^+(G_l,t)) - \alpha(u^+(G_l,t) - u^-(G_l,t))], \tag{13}$$

$$\begin{aligned} r(u(G_l,t),v(G_l,t)) \approx \frac{1}{2}[&r(u^-(G_l,t),v^-(G_l,t)) + r(u^+(G_l,t),v^+(G_l,t)) \\ &- \alpha(v^+(G_l,t) - v^-(G_l,t))], \end{aligned} \tag{14}$$

$$\begin{aligned} s(u(G_l,t),w(G_l,t)) \approx \frac{1}{2}[&s(u^-(G_l,t),w^-(G_l,t)) + s(u^+(G_l,t),w^+(G_l,t)) \\ &- \alpha(w^+(G_l,t) - w^-(G_l,t))], \end{aligned} \tag{15}$$

for $G_l = (x_{i\pm\sqrt{3}/6}, y_{j\pm 1/2})$, $u^\pm(G_l,t)$, $v^\pm(G_l,t)$, $w^\pm(G_l,t)$ are the bottom and top limits of the solutions u, v, w at the cell interface G_l respectively. What we want to do is to reconstruct

$u^{\pm}(G_l,t)$, $v^{\pm}(G_l,t)$, $w^{\pm}(G_l,t)$ from $\{\overline{u}_{ij}, \overline{v}_{ij}, \overline{w}_{ij}\}$. The procedure of reconstruction of $u^{\pm}(G_l,t)$ is the same as that in [14], we do not repeat it here for saving space.

Reconstruction of $v^{-}(G_l,t)$ at $G_l = (x_{i+\frac{1}{2}}, y_{j+\frac{1}{2\sqrt{3}}})$. For simplicity we relabel the cell I_{ij} and its neighboring cells as $I_1,\ldots,I_9$ as shown in Figure 1, where I_{ij} is relabeled as I_5.

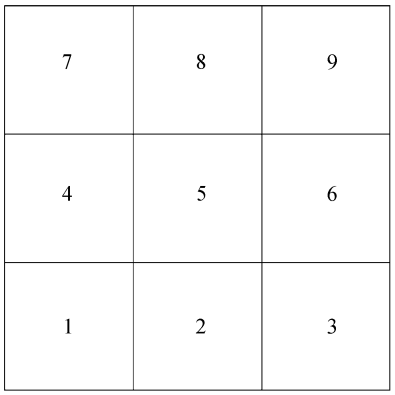

Figure 1 The big stencil

First, we choose the following stencils:

$$S_1=\{I_1,I_2,I_4,I_5\},\quad S_2=\{I_2,I_3,I_5,I_6\},\quad S_3=\{I_4,I_5,I_7,I_8\},$$
$$S_4=\{I_5,I_6,I_8,I_9\},\quad S_5=\{I_1,I_2,I_3,I_4,I_5,I_7\},\quad S_6=\{I_1,I_2,I_3,I_5,I_6,I_9\},$$
$$S_7=\{I_1,I_4,I_5,I_7,I_8,I_9\},\quad S_8=\{I_3,I_5,I_6,I_7,I_8,I_9\}.$$

We construct the polynomials $p_n(x,y)\,(n=1,\ldots,4)$ in the small stencils to approximate function u such that

$$\frac{1}{\Delta x\Delta y}\int_{I_k} p_n(x,y)dxdy=\overline{u}_k,\quad \frac{1}{\Delta y}\int_{I_{k_x}}\frac{\partial p_n(x,y)}{\partial x}dxdy=\overline{v}_{k_x},$$
$$\frac{1}{\Delta x}\int_{I_{k_y}}\frac{\partial p_n(x,y)}{\partial y}dxdy=\overline{w}_{k_y},$$

for

$$n=1,\ k=1,2,4,5,\ k_x=1,4,5,\ k_y=1,2,5;\quad n=2,\ k=2,3,5,6,\ k_x=3,5,6,\ k_y=2,3,5;$$
$$n=3,\ k=4,5,7,8,\ k_x=4,5,7,\ k_y=5,7,8;\quad n=4,\ k=5,6,8,9,\ k_x=5,6,9,\ k_y=5,8,9;$$

and $p_n(x,y)\,(n=5,\ldots,8)$ in the small stencils to approximate v, the first derivative of u, such that $\frac{1}{\Delta y}\int_{I_{k_x}} p_n(x,y)dxdy=\overline{v}_{k_x}$ for

$$n=5,\ k_x=1,2,3,4,5,7;\qquad n=6,\ k_x=1,2,3,5,6,9;$$
$$n=7,\ k_x=1,4,5,7,8,9;\qquad n=8,\ k_x=3,5,6,7,8,9.$$

For simplicity, we rewrite $\overline{u},\overline{v},\overline{w}$ as u,v,w in formulae (16)–(23) and (26)–(33). At point $G_l=(x_{i+\frac{1}{2}},y_{j+\frac{1}{2\sqrt{3}}})$ we have

$$\frac{\partial}{\partial x}p_1(x_{i+\frac{1}{2}},y_{j+\frac{1}{2\sqrt{3}}})=\frac{7}{2}v_5+\frac{3}{2}v_4-4u_5+4u_4+\frac{2}{3}\sqrt{3}u_1-\frac{2}{3}\sqrt{3}u_2-\frac{2}{3}\sqrt{3}u_4+\frac{2}{3}\sqrt{3}u_5$$
$$+\frac{1}{3}\sqrt{3}v_1+\frac{1}{6}\sqrt{3}w_1-\frac{1}{6}\sqrt{3}w_2-\frac{1}{3}\sqrt{3}v_4,\qquad(16)$$

$$\frac{\partial}{\partial x}p_2(x_{i+\frac{1}{2}},y_{j+\frac{1}{2\sqrt{3}}}) = \frac{1}{3}\sqrt{3}u_6 - \frac{1}{6}\sqrt{3}w_3 - \frac{1}{3}\sqrt{3}u_3 - 2u_5 + 2u_6 - \frac{1}{2}v_5 - \frac{1}{2}v_6 + \frac{1}{3}\sqrt{3}u_2 - \frac{1}{3}\sqrt{3}u_5 + \frac{1}{6}\sqrt{3}w_2, \tag{17}$$

$$\frac{\partial}{\partial x}p_3(x_{i+\frac{1}{2}},y_{j+\frac{1}{2\sqrt{3}}}) = -4u_5 + 4u_4 + \frac{7}{2}v_5 + \frac{3}{2}v_4 + \frac{2}{3}\sqrt{3}u_4 - \frac{2}{3}\sqrt{3}u_5 + \frac{1}{3}\sqrt{3}v_4 - \frac{2}{3}\sqrt{3}u_7 + \frac{2}{3}\sqrt{3}u_8 - \frac{1}{3}\sqrt{3}v_7 + \frac{1}{6}\sqrt{3}w_7 - \frac{1}{6}\sqrt{3}w_8, \tag{18}$$

$$\frac{\partial}{\partial x}p_4(x_{i+\frac{1}{2}},y_{j+\frac{1}{2\sqrt{3}}}) = -\frac{1}{3}\sqrt{3}u_6 - 2u_5 + 2u_6 - \frac{1}{2}v_5 - \frac{1}{2}v_6 + \frac{1}{3}\sqrt{3}u_9 - \frac{1}{6}\sqrt{3}w_9 + \frac{1}{3}\sqrt{3}u_5 - \frac{1}{3}\sqrt{3}u_8 + \frac{1}{6}\sqrt{3}w_8, \tag{19}$$

$$p_5(x_{i+\frac{1}{2}},y_{j+\frac{1}{2\sqrt{3}}}) = \frac{1}{3}v_1 - \frac{2}{3}v_2 + \frac{1}{3}v_3 - \frac{1}{2}v_4 + \frac{3}{2}v_5 + \frac{1}{6}\sqrt{3}v_1 - \frac{1}{4}\sqrt{3}v_2 - \frac{1}{4}\sqrt{3}v_4 + \frac{1}{4}\sqrt{3}v_5 + \frac{1}{12}\sqrt{3}v_7, \tag{20}$$

$$p_6(x_{i+\frac{1}{2}},y_{j+\frac{1}{2\sqrt{3}}}) = -\frac{1}{6}v_1 + \frac{1}{3}v_2 - \frac{1}{6}v_3 + \frac{1}{2}v_5 + \frac{1}{2}v_6 - \frac{1}{12}\sqrt{3}v_2 + \frac{1}{12}\sqrt{3}v_5 - \frac{1}{12}\sqrt{3}v_6 + \frac{1}{12}\sqrt{3}v_9, \tag{21}$$

$$p_7(x_{i+\frac{1}{2}},y_{j+\frac{1}{2\sqrt{3}}}) = -\frac{1}{2}v_4 + \frac{3}{2}v_5 + \frac{1}{3}v_7 - \frac{2}{3}v_8 + \frac{1}{3}v_9 - \frac{1}{12}\sqrt{3}v_1 + \frac{1}{4}\sqrt{3}v_4 - \frac{1}{4}\sqrt{3}v_5 - \frac{1}{6}\sqrt{3}v_7 + \frac{1}{4}\sqrt{3}v_8, \tag{22}$$

$$p_8(x_{i+\frac{1}{2}},y_{j+\frac{1}{2\sqrt{3}}}) = \frac{1}{2}v_5 + \frac{1}{2}v_6 - \frac{1}{6}v_7 + \frac{1}{3}v_8 - \frac{1}{6}v_9 - \frac{1}{12}\sqrt{3}v_3 - \frac{1}{12}\sqrt{3}v_5 + \frac{1}{12}\sqrt{3}v_6 + \frac{1}{12}\sqrt{3}v_8. \tag{23}$$

In order to obtain the 4th order approximation to function u, the approximation to the derivatives of u can be only the 3rd order. From definition of approximation polynomials of $p_n(x,y)$, it is easy to see that the $\frac{\partial}{\partial x}p_n(x_{i+\frac{1}{2}},y_{j+\frac{1}{2\sqrt{3}}})\,(n=1,\ldots,4)$ and $p_n(x_{i+\frac{1}{2}},y_{j+\frac{1}{2\sqrt{3}}})\,(n=5,\ldots,8)$ approximates to $v=u_x$ at point G_l up to the 3rd order when u is smooth enough. In this paper, we just choose the linear weights to be $\gamma_l=\frac{1}{8}\,(l=1,\ldots,8)$.

As in [12,15], we use the following smoothness indicator:

$$\beta_l = \sum_{|\alpha|=1}^{2}\int_{I_5}|I_5|^{|\alpha|-1}\left(D^\alpha\frac{\partial}{\partial x}p_l(x,y)\right)^2dxdy, \quad l=1,\ldots,4,$$

$$\beta_l = \sum_{|\alpha|=1}^{2}\int_{I_5}|I_5|^{|\alpha|-1}(D^\alpha p_l(x,y))^2dxdy, \quad l=5,\ldots,8.$$

In the actual numerical implementation the smoothness indicator $\beta_l\,(l=1,\ldots,8)$ can be written explicitly as follows:

$$\begin{aligned}\beta_1 = {} & \frac{16}{3}{w_1}^2 - \frac{32}{3}w_1w_2 + \frac{16}{3}{w_2}^2 + 204u_4v_5 + 18{u_1}^2 - 36u_1u_2 - 36u_1u_4 + 36u_1u_5 \\ & + \frac{46}{3}u_1v_1 - \frac{46}{3}u_1v_4 + 18{u_2}^2 + 36u_2u_4 - 36u_2u_5 - \frac{46}{3}u_2v_1 + \frac{46}{3}u_2v_4 + 210{u_4}^2 \\ & - 420u_4u_5 - \frac{46}{3}u_4v_1 + \frac{586}{3}u_4v_4 + 210{u_5}^2 + \frac{46}{3}u_5v_1 - \frac{586}{3}u_5v_4 - 204u_5v_5 + 55{v_5}^2\end{aligned}$$

$$+94v_5v_4+2v_1w_1-2v_1w_2-2w_1v_4+2w_2v_4+\frac{44}{3}u_5w_1-\frac{44}{3}u_5w_2+\frac{17}{3}{v_1}^2-\frac{34}{3}v_1v_4$$
$$+\frac{146}{3}{v_4}^2+\frac{44}{3}u_1w_1-\frac{44}{3}u_1w_2-\frac{44}{3}u_2w_1+\frac{44}{3}u_2w_2-\frac{44}{3}u_4w_1+\frac{44}{3}u_4w_2, \tag{26}$$

$$\beta_2=210{u_6}^2+\frac{46}{3}u_6v_3+\frac{16}{3}{w_2}^2-420u_5u_6+\frac{586}{3}u_5v_6-\frac{46}{3}u_5v_3+18{u_2}^2-36u_2u_5$$
$$+210{u_5}^2+204u_5v_5+55{v_5}^2-\frac{44}{3}u_5w_2+\frac{44}{3}u_2w_2-36u_2u_3+36u_2u_6+\frac{46}{3}u_2v_3$$
$$-\frac{46}{3}u_2v_6+18{u_3}^2+36u_3u_5-36u_3u_6-\frac{46}{3}u_3v_3+\frac{46}{3}u_3v_6-\frac{586}{3}u_6v_6+\frac{17}{3}{v_3}^2-\frac{34}{3}v_3v_6$$
$$+\frac{146}{3}{v_6}^2-\frac{44}{3}u_2w_3+\frac{44}{3}u_3w_3-\frac{44}{3}u_3w_2+\frac{44}{3}u_5w_3-\frac{44}{3}u_6w_3+\frac{44}{3}u_6w_2+\frac{16}{3}{w_3}^2$$
$$-\frac{32}{3}w_3w_2-204u_6v_5+94v_5v_6-2v_3w_3+2v_3w_2+2w_3v_6-2w_2v_6, \tag{27}$$

$$\beta_3=210{u_4}^2-420u_4u_5-36u_4u_7+36u_4u_8-\frac{46}{3}u_4v_7+\frac{586}{3}u_4v_4+36u_5u_7-36u_5u_8$$
$$+\frac{46}{3}u_5v_7-\frac{586}{3}u_5v_4+18{u_7}^2-36u_7u_8+\frac{46}{3}u_7v_7-\frac{46}{3}u_7v_4+18{u_8}^2-\frac{46}{3}u_8v_7$$
$$+\frac{46}{3}u_8v_4+\frac{17}{3}{v_7}^2-\frac{34}{3}v_7v_4+\frac{146}{3}{v_4}^2+\frac{44}{3}u_4w_7-\frac{44}{3}u_4w_8-\frac{44}{3}u_5w_7+\frac{44}{3}u_5w_8$$
$$-\frac{44}{3}u_7w_7+\frac{44}{3}u_7w_8+\frac{44}{3}u_8w_7-\frac{44}{3}u_8w_8+\frac{16}{3}{w_7}^2-2v_7w_7+2v_7w_8-\frac{32}{3}w_7w_8$$
$$+204u_4v_5+\frac{16}{3}{w_8}^2-2v_4w_8+210{u_5}^2-204u_5v_5+55{v_5}^2+94v_5v_4+2w_7v_4, \tag{28}$$

$$\beta_4=-36u_5u_8+36u_5u_9-\frac{46}{3}u_5v_9+\frac{586}{3}u_5v_6+210{u_6}^2+36u_6u_8-36u_6u_9+\frac{46}{3}u_6v_9$$
$$-\frac{586}{3}u_6v_6+18{u_8}^2-36u_8u_9+\frac{46}{3}u_8v_9-\frac{46}{3}u_8v_6+18{u_9}^2-\frac{46}{3}u_9v_9+\frac{46}{3}u_9v_6$$
$$+\frac{17}{3}{v_9}^2-\frac{34}{3}v_9v_6+210{u_5}^2-420u_5u_6+204u_5v_5-204u_6v_5+55{v_5}^2+94v_5v_6$$
$$+2v_9w_9-2v_9w_8+2v_6w_8-2w_9v_6+\frac{146}{3}{v_6}^2-\frac{44}{3}u_5w_9+\frac{44}{3}u_5w_8+\frac{44}{3}u_6w_9$$
$$-\frac{44}{3}u_6w_8+\frac{44}{3}u_8w_9-\frac{44}{3}u_8w_8-\frac{44}{3}u_9w_9+\frac{44}{3}u_9w_8+\frac{16}{3}{w_9}^2-\frac{32}{3}w_9w_8+\frac{16}{3}{w_8}^2, \tag{29}$$

$$\beta_5=\frac{4}{3}{v_7}^2+v_5v_7-\frac{16}{3}v_4v_7-\frac{19}{3}v_4v_5-\frac{26}{3}v_1v_2-\frac{26}{3}v_1v_4+\frac{13}{3}v_1v_5+\frac{19}{3}v_2v_4$$
$$-\frac{19}{3}v_2v_5+\frac{8}{3}v_1v_7+\frac{8}{3}v_1v_3-\frac{16}{3}v_2v_3+\frac{4}{3}{v_3}^2+\frac{15}{2}{v_4}^2+\frac{19}{6}{v_5}^2+\frac{23}{6}{v_1}^2$$
$$+\frac{15}{2}{v_2}^2-v_2v_7-v_3v_4+v_3v_5, \tag{30}$$

$$\beta_6=-\frac{16}{3}v_1v_2+v_1v_5-\frac{19}{3}v_2v_5+\frac{8}{3}v_1v_3-\frac{26}{3}v_2v_3+\frac{23}{6}{v_3}^2-v_1v_6-v_2v_9+v_5v_9$$
$$+\frac{19}{6}{v_5}^2+\frac{4}{3}{v_1}^2+\frac{15}{2}{v_2}^2+\frac{13}{3}v_3v_5+\frac{19}{3}v_2v_6-\frac{26}{3}v_3v_6+\frac{8}{3}v_3v_9-\frac{19}{3}v_5v_6$$
$$+\frac{15}{2}{v_6}^2-\frac{16}{3}v_6v_9+\frac{4}{3}{v_9}^2, \tag{31}$$

$$\beta_7=\frac{23}{6}{v_7}^2+\frac{13}{3}v_5v_7-\frac{26}{3}v_4v_7-\frac{19}{3}v_4v_5-\frac{16}{3}v_1v_4+v_1v_5+\frac{8}{3}v_1v_7+v_5v_9+\frac{15}{2}{v_4}^2$$
$$+\frac{19}{6}{v_5}^2+\frac{4}{3}{v_1}^2-v_4v_9-v_1v_8+\frac{19}{3}v_4v_8-\frac{19}{3}v_5v_8-\frac{26}{3}v_7v_8+\frac{15}{2}{v_8}^2+\frac{8}{3}v_7v_9$$
$$-\frac{16}{3}v_8v_9+\frac{4}{3}{v_9}^2, \tag{32}$$

$$\begin{aligned}\beta_8 = {} & v_5v_7 - \frac{19}{3}v_5v_6 - \frac{19}{3}v_5v_8 + \frac{13}{3}v_5v_9 + \frac{15}{2}{v_6}^2 + \frac{19}{3}v_6v_8 - \frac{26}{3}v_6v_9 - \frac{26}{3}v_8v_9 \\ & + \frac{15}{2}{v_8}^2 + \frac{23}{6}{v_9}^2 + \frac{4}{3}{v_3}^2 - \frac{16}{3}v_3v_6 + \frac{8}{3}v_3v_9 - \frac{16}{3}v_7v_8 + \frac{8}{3}v_7v_9 + \frac{4}{3}{v_7}^2 \\ & + v_3v_5 - v_3v_8 - v_6v_7 + \frac{19}{6}{v_5}^2. \end{aligned} \tag{33}$$

Then we compute the nonlinear weight based on the linear weight and smoothness indicator[15]:

$$\omega_l = \frac{\overline{\omega}_l}{\sum_{k=1}^{8}\overline{\omega}_k}, \quad \overline{\omega}_l = \frac{\gamma_l}{\sum_{k=1}^{8}(\varepsilon+\beta_k)^2}, \quad l = 1, \ldots, 8, \tag{34}$$

where γ_l is the linear weight determined in the above step, and ε is a small positive number to avoid the denominator to become 0. We use $\varepsilon = 10^{-6}$ in all the computations in this paper. The final approximation is given by

$$v_{i+\frac{1}{2},j+\frac{1}{2\sqrt{3}}} \approx \sum_{l=1}^{4}\omega_l\frac{\partial}{\partial x}p_l(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2\sqrt{3}}}) + \sum_{l=5}^{8}\omega_l p_l(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2\sqrt{3}}}).$$

Remarks. The main difference between the procedure of reconstruction of v and that in [14] is that in this work, more derivative terms are used in the reconstruction, and full 2-degree polynomials are constructed for the derivatives in the small stencils.

The reconstructions to $v^+(G_l)$ and $w^{\pm}(G_l)$ are similar to the above procedure.

For systems of conservation laws, such as the Euler equations of gas dynamics, all of the reconstructions are performed in the local characteristic directions to avoid oscillations.

The semidiscrete scheme (4)–(6), written as $u_t = L(u)$ is then discretized in time by a TVD Runge-Kutta method[16], for example the third order version given by

$$\begin{aligned} u^{(1)} &= u^n + \Delta t L(u^n), \quad u^{(2)} = \frac{3}{4}u^n + \frac{1}{4}u^{(1)} + \frac{1}{4}\Delta t L(u^{(1)}), \\ u^{n+1} &= \frac{1}{3}u^n + \frac{2}{3}u^{(2)} + \frac{2}{3}\Delta t L(u^{(2)}). \end{aligned} \tag{35}$$

3 Numerical tests

In this section we present the results of numerical tests for the scheme in the previous section.

Example 3.1. We solve the following nonlinear scalar Burgers equation in two dimensions:

$$u_t + \left(\frac{u^2}{2}\right)_x + \left(\frac{u^2}{2}\right)_y = 0 \tag{36}$$

with the initial condition $u(x, y, 0) = 0.5 + \sin(\pi(x + y)/2)$ and periodic boundary conditions in two directions. We compute the solution up to $t = 0.5/\pi$. When $t = 0.5/\pi$, the solution is still smooth. The errors and numerical orders of accuracy by the HWENO scheme are shown in Table 1. We can see that the designed order is obtained.

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 two directions. $T = 0.5/\pi$. L^1 and L^∞ errors

cells	L^1 error	L^1 order	L^∞ error	L^∞ order
10×10	1.33e–2		3.79e–2	
20×20	1.56e–3	3.09	9.69e–3	1.97
40×40	1.38e–4	3.50	1.09e–3	3.15
80×80	6.86e–6	4.33	6.21e–5	4.14
160×160	3.43e–7	4.32	3.35e–6	4.21

Example 3.2. We solve the Euler equations

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ \rho u\\ \rho v\\ E\end{pmatrix}+\frac{\partial}{\partial x}\begin{pmatrix}\rho u\\ \rho u^2+p\\ \rho uv\\ u(E+p)\end{pmatrix}+\frac{\partial}{\partial y}\begin{pmatrix}\rho v\\ \rho uv\\ \rho v^2+p\\ v(E+p)\end{pmatrix}=0. \tag{37}$$

In which ρ is density, u is x-direction velocity, v is y-direction velocity, E is total energy, p is pressure. 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 in two directions. We compute the density solution up to $t=2.0$. The exact solution is $\rho(x,y,t)=1+0.2\sin(\pi(x+y-t))$. In Table 2, we show the errors and numerical orders of accuracy by the HWENO scheme and we also can see that the designed order is obtained.

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

cells	L^1 error	L^1 order	L^∞ error	L^∞ order
10×10	1.24e–2		2.13e–2	
20×20	6.65e–4	4.23	1.35e–3	3.98
40×40	2.31e–5	4.84	4.57e–5	4.89
80×80	9.33e–7	4.63	1.59e–6	4.84
160×160	4.55e–8	4.36	7.26e–8	4.46

Example 3.3. We solve the same nonlinear Burgers equation (36) for Riemann problem in a computational domain of $[-1,3]\times[-1,3]$ and set the initial conditions as:

$$(1)\quad u(x,y,0)=\begin{cases}0.1, & x>0,y>0,\\ 2.5, & x<0,y>0,\\ 1.1, & x<0,y<0,\\ 1.5, & x>0,y<0,\end{cases} \tag{38}$$

$$(2)\quad u(x,y,0)=\begin{cases}1.1, & x>0,y>0,\\ 3.1, & x<0,y>0,\\ 2.1, & x<0,y<0,\\ 0.1, & x>0,y<0.\end{cases} \tag{39}$$

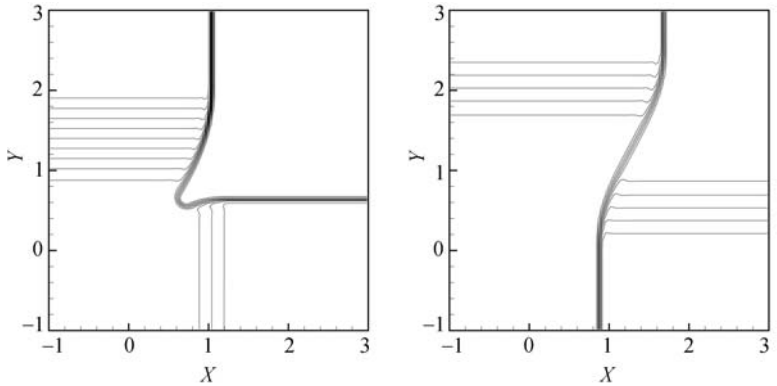

Figure 2 Burgers equation for Riemann problem. $T=0.8$. 120×120 cells. Left: initial condition (1), 15 equally spaced contours from 0.249 to 2.351; right: initial condition (2), 15 equally spaced contours from 0.286 to 2.913.

We plot the results at $t = 0.8$. The solutions are shown in Figure 2. We can see that the scheme gives non-oscillatory shock transitions for this problem.

Example 3.4. 2D Euler equations for Riemann problem[17,18]. We solve the Euler equations (37) in a computational domain of $[0, 1] \times [0, 1]$ and set the initial conditions as:

$$(1)\quad (\rho, u, v, p)^T = \begin{cases} (0.5313, 0, 0, 0.4)^T, & x > 0.5, y > 0.5, \\ (1, 0.7276, 0, 1)^T, & x < 0.5, y > 0.5, \\ (0.8, 0, 0, 1)^T, & x < 0.5, y < 0.5, \\ (1, 0, 0.7276, 1)^T, & x > 0.5, y < 0.5, \end{cases} \tag{40}$$

$$(2)\quad (\rho, u, v, p)^T = \begin{cases} (1.1, 0, 0, 1.1)^T, & x > 0.5, y > 0.5, \\ (0.5065, 0.8939, 0, 0.35)^T, & x < 0.5, y > 0.5, \\ (1.1, 0.8939, 0.8939, 1.1)^T, & x < 0.5, y < 0.5, \\ (0.5065, 0, 0.8939, 0.35)^T, & x > 0.5, y < 0.5, \end{cases} \tag{41}$$

$$(3)\quad (\rho, u, v, p)^T = \begin{cases} (1, 0.1, 0, 1)^T, & x > 0.5, y > 0.5, \\ (0.5313, 0.8276, 0, 0.4)^T, & x < 0.5, y > 0.5, \\ (0.8, 0.1, 0, 0.4)^T, & x < 0.5, y < 0.5, \\ (0.5313, 0.1, 0.7276, 0.4)^T, & x > 0.5, y < 0.5, \end{cases} \tag{42}$$

$$(4)\quad (\rho, u, v, p)^T = \begin{cases} (0.5313, 0.1, 0.1, 0.4)^T, & x > 0.5, y > 0.5, \\ (1.0222, -0.6179, 0.1, 1)^T, & x < 0.5, y > 0.5, \\ (0.8, 0.1, 0.1, 1)^T, & x < 0.5, y < 0.5, \\ (1, 0.1, 0.8276, 1)^T, & x > 0.5, y < 0.5, \end{cases} \tag{43}$$

$$(5)\quad (\rho, u, v, p)^T = \begin{cases} (1, 0.75, -0.5, 1)^T, & x > 0.5, y > 0.5, \\ (2, 0.75, 0.5, 1)^T, & x < 0.5, y > 0.5, \\ (1, -0.75, 0.5, 1)^T, & x < 0.5, y < 0.5, \\ (3, -0.75, -0.5, 1)^T, & x > 0.5, y < 0.5. \end{cases} \tag{44}$$

In Figure 3, we show the computational results for density at (1) $t = 0.25$, (2) $t = 0.25$, (3) $t = 0.3$, (4) $t = 0.2$, (5) $t = 0.3$, respectively.

Example 3.5. Double mach reflection problem. We solve the Euler equations (37) in a computational domain of $[0, 4] \times [0, 1]$. A reflection wall lies at the bottom of the domain starting from $x = \frac{1}{6}$, y=0, making a 60° angle with the x-axis. The reflection boundary condition is used at the wall, which for the rest of the bottom boundary (the part from $x = 0$ to $x = \frac{1}{6}$), the exact post-shock condition is imposed. At the top boundary is the exact motion of the mach 10 shock. The results shown are at $t = 0.2$. We present both the pictures of region $[0, 3] \times [0, 1]$ and the blow-up region around the double mach stems in Figures 4 and 5 respectively. All pictures are the density contours with 30 equal spaced contour lines from 1.5 to 22.7.

Remark. The HWENO schemes developed in [14] failed in this test case.

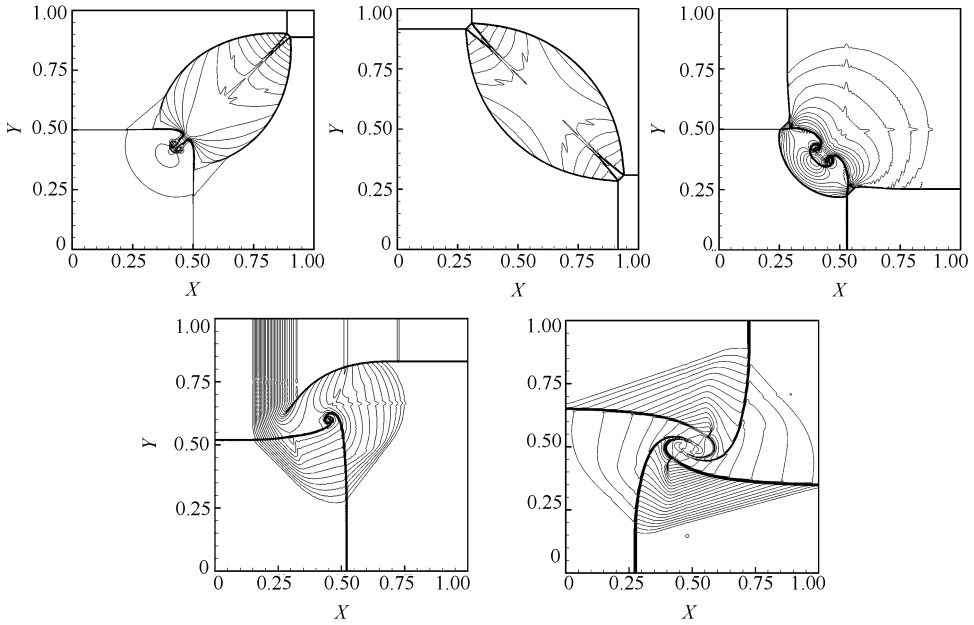

Figure 3 2D Euler equations for Riemann problem with 400×400 cells. From left to right and top to bottom. Initial condition (1) $T = 0.25$. 30 equally spaced density contours from 0.54 to 1.70; Initial condition (2) $T = 0.25$. 30 equally spaced density contours from 0.52 to 1.92; Initial condition (3) $T = 0.3$. 30 equally spaced density from contours 0.55 to 1.21; Initial condition (4) $T = 0.2$. 30 equally spaced density contours from 0.53 to 0.99; Initial condition (5) $T = 0.3$. 30 equally spaced density contours from 0.25 to 3.05.

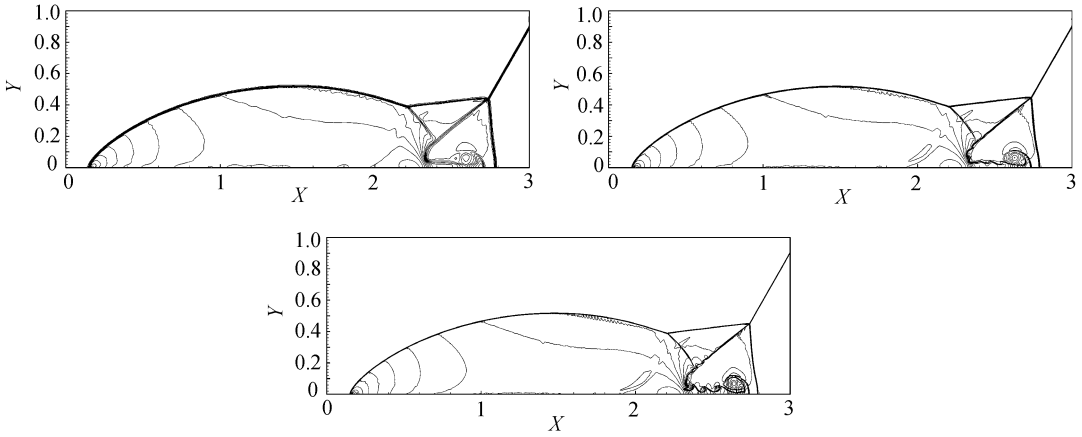

Figure 4 Double Mach refection problem, 30 equally spaced density contours from 1.5 to 22.7. From top to bottom: 800 × 200 cells, 1600 × 400 cells, 2400 × 600 cells.

Example 3.6. A mach 3 wind tunnel with a step. 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 end of the tunnel. Initially, a right going mach 3 flow is used. Reflective boundary conditions are applied alone the walls of the tunnel and in flow and out flow boundary conditions are applied at the entrance and the exit. The results are shown at $t = 4$. We present the pictures of whole region $[0, 3] \times [0, 1]$ in Figure 6. All pictures are the density contours with 30 equal spaced contour lines from 0.32 to 6.15.

Remark. The HWENO schemes developed in [14] failed in this test case.

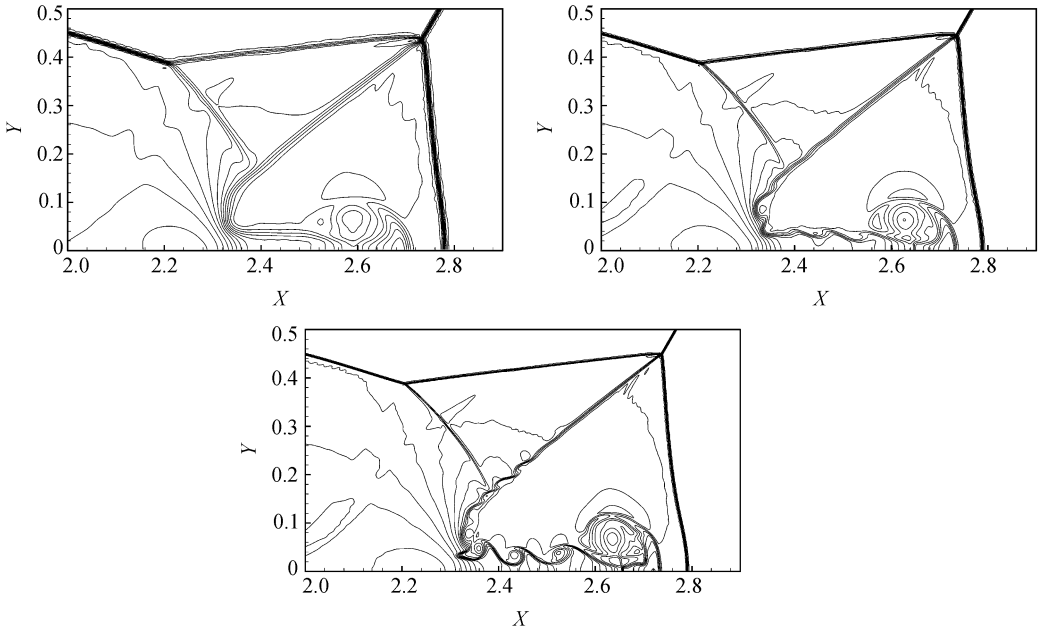

Figure 5 Double Mach refection problem zoom in, 30 equally spaced density contours from 1.5 to 22.7. From left to right and top to bottom: 800 × 200 cells, 1600 × 400 cells, 2400 × 600 cells.

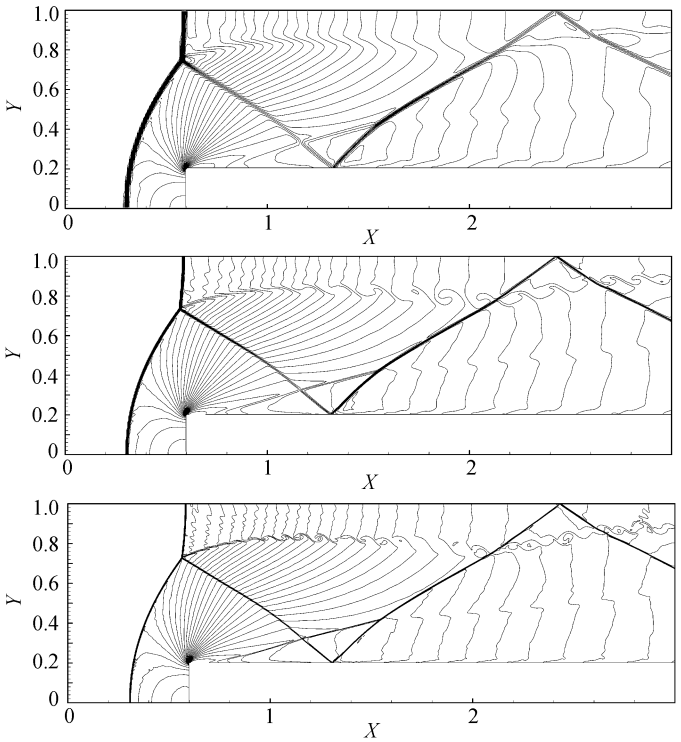

Figure 6 A mach 3 wind tunnel with a step problem, 30 equally spaced density contours from 0.32 to 6.15. From top to bottom: 300 × 100 cells, 600 × 200 cells and 1200 × 400 cells.

4 Concluding remarks

In this paper, we constructed a class of fourth order accurate finite volume HWENO schemes based on the procedure of [14], for two dimensional hyperbolic conservation laws. The key

idea of HWENO is to evolve both with the solution and its derivative, this allows for using Hermite interpolation in the reconstruction phase, resulting in a more compact stencil at the expense of additional work. We use the procedure presented in [14] to reconstruct the values of solution, and the main difference between the procedure of reconstruction of derivatives in this paper and that in [14] is that more derivative terms are used in the reconstruction, and full 2-degree polynomials are constructed for the derivatives in the small stencils in this paper. Comparing with the original HWENO schemes in [14], one major advantage of new HWENO schemes is its robust in computation of problem with strong shocks, such as the double Mach reflection problem and forward step problem. Numerical experiments for two dimensional Burgers' equation and Euler equations of compressible gas dynamics are presented to show the effectiveness of the methods.

References

1 Godunov S K. A finite-difference method for the numerical computation of discontinuous solutions of the equations of fluid dynamics. *Mat Sb*, **47**(3): 271–290 (1959)

2 Harten A, Osher S. Uniformly high-order accurate non-oscillatory schemes. IMRC Technical Summary Rept. 2823, Univ. of Wisconsin, Madison, WI, May, 1985

3 Harten A. High resolution schemes for hyperbolic conservation laws. *J Comput Phys*, **49**(3): 357–393 (1983)

4 Harten A, Engquist B, Osher S, Chakravarthy S. Uniformly high order accurate essentially non-oscillatory schemes III. *J Comput Phys*, **71**(2): 231–323 (1987)

5 Harten A. Preliminary results on the extension of ENO schemes to two-dimensional problems. In: C. Carasso et al. eds. Proceedings of International Conference on Nonlinear Hyperbolic Problems, Saint-Etienne, 1986, Lecture Notes in Mathematics, Berlin: Springer-Verlag, 1987

6 Casper J. Finite-volume implementation of high-order essentially nonoscillatory schemes in two dimensions. *AIAA Journal*, **30**(12): 2829–2835 (1992)

7 Casper J, Atkins H L. A finite-volume high-order ENO scheme for two-dimensional hyperbolic systems. *J Comput Phys*, **106**: 62–76 (1993)

8 Abgrall R. On essentially non-oscillatory schemes on unstructured meshes: Analysis and implementation. *J Comput Phys*, **114**: 45–58 (1994)

9 Liu X D, Osher S, Chan T. Weighted essentially non-oscillatory schemes. *J Comput Phys*, **115**: 200–212 (1994)

10 Jiang G S, Shu C W. Efficient implementation of weighted ENO schemes. *J Comput Phys*, **126**: 202–228 (1996)

11 Friedrichs O. Weighted essentially non-oscillatory schemes for the interpolation of mean values on unstructured grids. *J Comput Phys*, **144**: 194–212 (1998)

12 Hu C Q, Shu C W. Weighted essentially non-oscillatory schemes on triangular meshes. *J Comput Phys*, **150**: 97–127 (1999)

13 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)

14 Qiu J, Shu C W. Hermite WENO schemes and their application as limiters for Runge-Kutta discontiuous Galerkin method II: two dimensional case. *Computers & Fluids*, **34**: 642–663 (2005)

15 Shu C W. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. In: Advanced Numerical Approximation of Nonlinear Hyperbolic Equations. A. Lecture Notes in Mathematics, CIME subseries Berlin-New York: Springer-Verlag, ICASE Report 97–65

16 Shu C W, Osher S. Efficient implementation of essentially non-oscillatory shock capturing schemes. *J Comput Phys*, **77**: 439–471 (1988)

17 Lax P D, Liu X D. Solution of two dimensional Riemann problems of gas dynamics by positive schemes. *SIAM J Sci Comput*, **19**(2): 319–340 (1998)

18 Brio M, Zakharian A R, Webb G M. Two dimensional Riemann solver for Euler equations of gas dynamics. *J Comput Phys*, **167**: 177–195 (2001)