Available online at www.sciencedirect.com

SCIENCE DIRECT®

Journal of Computational Physics 204 (2005) 82–99

JOURNAL OF COMPUTATIONAL PHYSICS

www.elsevier.com/locate/jcp

Hermite WENO schemes for Hamilton–Jacobi equations

Jianxian Qiu [a,1], Chi-Wang Shu [b,*,2]

[a] *Department of Mechanical Engineering, National University of Singapore, Singapore 119260, Singapore*
[b] *Division of Applied Mathematics, Brown University, Providence, RI 02912, USA*

Received 23 August 2004; accepted 4 October 2004
Available online 11 November 2004

Abstract

In this paper, a class of weighted essentially non-oscillatory (WENO) schemes based on Hermite polynomials, termed HWENO (Hermite WENO) schemes, for solving Hamilton–Jacobi equations is presented. The idea of the reconstruction in the HWENO schemes comes from the original WENO schemes, however both the function and its first derivative values are evolved in time and used in the reconstruction, while only the function values are evolved and used in the original WENO schemes. Comparing with the original WENO schemes of Jiang and Peng [Weighted ENO schemes for Hamilton–Jacobi equations, SIAM Journal on Scientific Computing 21 (2000) 2126] for Hamilton–Jacobi equations, one major advantage of HWENO schemes is its compactness in the reconstruction. Extensive numerical experiments are performed to illustrate the capability of the method.

MSC: 65M06; 65M99; 70H20
Keywords: WENO scheme; Hamilton–Jacobi equation; Hermite interpolation; High order accuracy

1. Introduction

In this paper, we construct a class of fifth order WENO schemes based on Hermite polynomials, termed HWENO schemes, for solving the Hamilton–Jacobi (HJ) equations:

* Corresponding author. Tel.: +1 401 863 2549; fax: +1 401 863 1355.
E-mail addresses: mpeqjx@nus.edu.sg (J. Qiu), shu@dam.brown.edu (C.-W. Shu).
[1] Research partially supported by NUS Research Project R-265-000-118-112 and NNSFC grant 10371118.
[2] Research partially supported by the Chinese Academy of Sciences while the author was in residence at the University of Science and Technology of China (grant 2004-1-8) and at the Institute of Computational Mathematics and Scientific/Engineering Computing. Additional support is provided by ARO grant W911NF-04-1-0291, NSF grant DMS-0207451 and AFOSR grant F49620-02-1-0113.

0021-9991/$ - see front matter
doi:10.1016/j.jcp.2004.10.003

$$\begin{cases} \phi_t + H(\nabla_x \phi) = 0, \\ \phi(x,0) = \phi_0(x), \end{cases} \tag{1.1}$$

where $x = (x_1, \ldots, x_d)$ are d-spatial variables. The HJ equations appear often in applications, such as in control theory, differential games, geometric optics and image processing. The solutions to (1.1) typically are continuous but with discontinuous derivatives, even if the initial condition $\phi_0(x) \in C^\infty$. It is well known that the HJ equations are closely related to conservation laws, hence successful numerical methods for conservation laws can be adapted for solving the HJ equations. Along this line we mention the early work of Osher and Sethian [20] and Osher and Shu [21] in constructing high order ENO (essentially non-oscillatory) schemes for solving the HJ equations. These ENO schemes for solving the HJ equations were based on ENO schemes for solving hyperbolic conservation laws in [9,26,27]. ENO schemes for solving the HJ equations on unstructured meshes were constructed in [15]. More recently, ENO schemes based on radial basis functions were constructed in [6]. Central high resolution schemes were developed in [3,4,14]. Finite element methods suitable for arbitrary triangulations were developed in [1,5,10,16]. Finally, most relevant to our work, we mention the WENO schemes for solving the HJ equations [12] by Jiang and Peng, based on the WENO schemes for solving conservation laws [18,13]. Zhang and Shu [29] further developed high order WENO schemes on unstructured meshes for solving two dimensional HJ equations.

We now review WENO schemes in more detail as they are most relevant to our work in this paper. WENO schemes have been designed in recent years as a class of high order finite volume or finite difference schemes to solve hyperbolic conservation laws and Hamilton–Jacobi equations with the property of maintaining both uniform high order accuracy and an essentially non-oscillatory shock transition. The first WENO scheme is constructed in [18] for a third order finite volume version in one space dimension. In [13], third and fifth order finite difference WENO schemes in multi space dimensions are constructed, with a general framework for the design of the smoothness indicators and nonlinear weights. Finite volume WENO schemes on unstructured and structured meshes are designed in, e.g. [8,10,17,22]. WENO schemes are designed based on the successful ENO schemes in [9,26,27]. Both ENO and WENO schemes use the idea of adaptive stencils in the reconstruction procedure based on the local smoothness of the numerical solution to automatically achieve high order accuracy and a non-oscillatory property near discontinuities. For a detailed review of ENO and WENO schemes, we refer to the lecture notes [25].

The framework of the finite volume and finite difference WENO schemes is to evolve only one degree of freedom per cell, namely the cell average for the finite volume version or the point value at the center of the cell for the finite difference version. In [23,24] we presented a class of WENO schemes based on Hermite polynomials, termed HWENO schemes, for solving one and two dimensional nonlinear hyperbolic conservation law systems. The main difference between the Hermite WENO scheme designed in [23,24], see also related earlier work in [2,7,19,28], 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. Numerical examples in [23,24] demonstrate that HWENO schemes work well for solving hyperbolic conservation laws.

In this paper, based on the HWENO methodology for conservation laws in [23,24], we develop HWENO schemes to solve the HJ equations. We evolve both the solution ϕ at the grid points and the cell averages of the x and y derivatives of the solution ϕ. Both the point values of the solution and the cell averages of its derivatives are used to reconstruct the point values of the derivatives at the grid points and at the interfaces of the cells. Comparing with the original WENO schemes of Jiang and Peng [12], one major advantage of

HWENO schemes is its compactness in the reconstruction. For example, in the one dimensional case, six points are needed in the stencil for a fifth order WENO reconstruction, while only four points are needed for a fifth order HWENO reconstruction. We do remark that HWENO schemes are more costly in computation and storage than the regular WENO schemes on the same mesh, since both the solution and its derivatives must be stored and evolved in time. It is however our experience that HWENO schemes are more accurate than WENO schemes on the same mesh for most problems.

The organization of this paper is as follows. In Section 2, we describe in detail the construction and implementation of the HWENO schemes with Runge–Kutta time discretizations, for one and two dimensional Hamilton–Jacobi equations (1.1). In Section 3 we provide extensive numerical examples to demonstrate the behavior of the HWENO schemes. Concluding remarks are given in Section 4.

2. The construction of Hermite WENO schemes for the Hamilton–Jacobi equations

In this section we will present the details of the construction of Hermite WENO schemes for both one and two dimensional Hamilton–Jacobi equations.

2.1. One dimensional case

We first consider the one dimensional Hamilton–Jacobi equation (1.1). For simplicity, we assume that the grid points $\{x_i\}$ are uniformly distributed with the cell size $x_{i+1} - x_i = \Delta x$ and cell centers $x_{i+\frac{1}{2}} = \frac{1}{2}(x_i + x_{i+1})$. We also denote the cells by $I_i = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$. This assumption is not essential: the method can be easily defined for non-uniform meshes.

Let $u = \phi_x$, and taking the x derivative of (1.1), we obtain the conservation law:

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

We denote $\phi_i(t) = \phi(x_i,t)$ and the cell averages of u as $\overline{u}_i(t) = \frac{1}{\Delta x}\int_{I_i} u(x,t)\,\mathrm{d}x$. Integrating (2.1) over the cell I_i we obtain an equivalent form:

$$\begin{cases} \frac{\mathrm{d}}{\mathrm{d}t}\phi_i(t) = -H(u(x_i,t)), \\ \frac{\mathrm{d}}{\mathrm{d}t}\overline{u}_i(t) = -\frac{1}{\Delta x}(H(u(x_{i+1/2},t)) - H(u(x_{i-1/2},t))). \end{cases} \tag{2.2}$$

We approximate (2.2) by the following scheme:

$$\begin{cases} \frac{\mathrm{d}\phi_i(t)}{\mathrm{d}t} = -\tilde{H}_i, \\ \frac{\mathrm{d}\overline{u}_i(t)}{\mathrm{d}t} = -\frac{1}{\Delta x}(\hat{H}_{i+1/2} - \hat{H}_{i-1/2}), \end{cases} \tag{2.3}$$

where the numerical fluxes $\tilde{H}_i$ and $\hat{H}_{i+1/2}$ in (2.3) are subject to the usual conditions for numerical fluxes, such as monotonicity, Lipschitz continuity and consistency with the physical flux $H(u)$. For choices of numerical fluxes suitable for the HJ equations we refer to, e.g. [21]. In this paper we use the simple Lax–Friedrichs flux defined by:

$$\begin{aligned} \tilde{H}_i &= H\left(\tfrac{u_i^- + u_i^+}{2}\right) - \tfrac{\alpha}{2}\left(u_i^+ - u_i^-\right), \\ \hat{H}_{i+1/2} &= \tfrac{1}{2}\left(H(u_{i+1/2}^-) + H(u_{i+1/2}^+) - \alpha(u_{i+1/2}^+ - u_{i+1/2}^-)\right), \end{aligned} \tag{2.4}$$

where $u_i^{\pm}$ and $u_{i+1/2}^{\pm}$ are the numerical approximations to the point values of $u(x_i,t)$ and $u(x_{i+1/2},t)$, respectively, from left and right, and $\alpha = \max_u |H'(u)|$.

The method of lines ODE (2.3), written in the form

$$w_t = L(w)$$

is then discretized in time by a total variation diminishing (TVD) Runge–Kutta method [26], for example the third order version given by:

$$\begin{aligned} w^{(1)} &= w^n + \Delta t L(w^n), \\ w^{(2)} &= \tfrac{3}{4}w^n + \tfrac{1}{4}w^{(1)} + \tfrac{1}{4}\Delta t L(w^{(1)}), \\ w^{n+1} &= \tfrac{1}{3}w^n + \tfrac{2}{3}w^{(2)} + \tfrac{2}{3}\Delta t L(w^{(2)}). \end{aligned} \tag{2.5}$$

The key component of the HWENO schemes is the reconstruction, from point values $\{\phi_i\}$ and the cell averages $\{\overline{u}_i\}$ to the points values $\{u_i^{\pm}, u_{i+1/2}^{\pm}\}$. This reconstruction should be both high order accurate and essentially non-oscillatory. We outline the procedure of this reconstruction for the fifth order accurate case in the following.

Step 1. Reconstruction of $\{u_i^-\}$ by HWENO from $\{\phi_i, \overline{u}_i\}$.

1. Given the small stencils $S_0 = \{x_{i-2}, x_{i-1}, x_i, I_{i-1}\}$, $S_1 = \{x_{i-1}, x_i, x_{i+1}, I_{i+1}\}$, $S_2 = \{x_{i-2}, x_{i-1}, x_i, x_{i+1}\}$ and the bigger stencil $\mathscr{T} = \{S_0, S_1, S_2\}$, we construct Hermite cubic reconstruction polynomials $p_0(x)$, $p_1(x)$, $p_2(x)$ and a fifth-degree reconstruction polynomial $q(x)$ such that:

$$p_0(x_{i+j}) = \phi_{i+j}, \quad j = -2, -1, 0; \qquad \frac{1}{\Delta x}\int_{I_{i-1}} p_0'(x)\,\mathrm{d}x = \overline{u}_{i-1}.$$

$$p_1(x_{i+j}) = \phi_{i+j}, \quad j = -1, 0, 1; \qquad \frac{1}{\Delta x}\int_{I_{i+1}} p_1'(x)\,\mathrm{d}x = \overline{u}_{i+1}.$$

$$p_2(x_{i+j}) = \phi_{i+j}, \quad j = -2, -1, 0, 1.$$

$$q(x_{i+j}) = \phi_{i+j}, \quad j = -2, -1, 0, 1; \qquad \frac{1}{\Delta x}\int_{I_{i+j}} q'(x)\,\mathrm{d}x = \overline{u}_{i+j}, \quad j = -1, 1.$$

In fact, we only need the values of the derivatives of these polynomials at the grid points x_i, which have the following expressions:

$$p_0'(x_i) = \frac{1}{6}\left(5\frac{\Delta^-\phi_{i-1}}{\Delta x} + 17\frac{\Delta^-\phi_i}{\Delta x} - 16\overline{u}_{i-1}\right),$$

$$p_1'(x_i) = \frac{1}{18}\left(5\frac{\Delta^-\phi_i}{\Delta x} + 21\frac{\Delta^-\phi_{i+1}}{\Delta x} - 8\overline{u}_{i+1}\right),$$

$$p_2'(x_i) = \frac{1}{6}\left(-\frac{\Delta^-\phi_{i-1}}{\Delta x} + 5\frac{\Delta^-\phi_i}{\Delta x} + 2\frac{\Delta^-\phi_{i+1}}{\Delta x}\right),$$

$$q'(x_i) = \frac{1}{150}\left(23\frac{\Delta^-\phi_{i-1}}{\Delta x} + 185\frac{\Delta^-\phi_i}{\Delta x} + 70\frac{\Delta^-\phi_{i+1}}{\Delta x} - 110\overline{u}_{i-1} - 18\overline{u}_{i+1}\right),$$

where $\Delta^-\phi_i = \phi_i - \phi_{i-1}$.

2. We find the combination coefficients, also referred to as the linear weights, denoted by γ_0, γ_1 and γ_2, satisfying

$$q'(x_i) = \sum_{j=0}^{2} \gamma_j p_j'(x_i) \tag{2.6}$$

for all possible point values ϕ and cell averages of u in the bigger stencil $\mathscr{T}$. This leads to:

$$\gamma_0 = \frac{11}{40}, \quad \gamma_1 = \frac{27}{100}, \quad \gamma_2 = \frac{91}{200}.$$

3. We compute the smoothness indicator, denoted by β_j, for each stencil S_j, which measures how smooth the function $p_j(x)$ is near the target point x_i. The smaller this smoothness indicator β_j, the smoother the function $p_j(x)$ is near the target point. We use the same recipe for the smoothness indicator as in [12]

$$\beta_j = \sum_{l=2}^{3} \int_{I_i} \Delta x^{2l-1} \left(\frac{\partial^l}{\partial x^l} p_j(x) \right)^2 \mathrm{d}x. \tag{2.7}$$

Notice that the summation starts from $l = 2$ for the HJ equations, rather than from $l = 1$ for the conservation law case [13]. In the actual numerical implementation the smoothness indicators β_j are written out explicitly as quadratic forms of the point values of ϕ and the cell averages of u in the stencil:

$$\beta_0 = \frac{13}{12}\left[\frac{4}{\Delta x}(\Delta^- \phi_{i-1} + \Delta^- \phi_i) - 8\overline{u}_{i-1}\right]^2 + \left[\frac{1}{\Delta x}(3\Delta^- \phi_{i-1} + 5\Delta^- \phi_i) - 8\overline{u}_{i-1}\right]^2,$$

$$\beta_1 = \frac{13}{12}\left[\frac{1}{\Delta x}\left(\frac{4}{3}\Delta^- \phi_i - 4\Delta^- \phi_{i+1}\right) + \frac{8}{3}\overline{u}_{i+1}\right]^2 + \left[\frac{1}{\Delta x}(-\Delta^- \phi_i + \Delta^- \phi_{i+1})\right]^2,$$

$$\beta_2 = \frac{13}{12}\left[\frac{1}{\Delta x}(\Delta^- \phi_{i-1} - 2\Delta^- \phi_i + \Delta^- \phi_{i+1})\right]^2 + \left[\frac{1}{\Delta x}(-\Delta^- \phi_i + \Delta^- \phi_{i+1})\right]^2.$$

4. We compute the nonlinear weights based on the smoothness indicators:

$$\omega_j = \frac{\overline{\omega}_j}{\sum_k \overline{\omega}_k}, \quad \overline{\omega}_k = \frac{\gamma_k}{(\varepsilon + \beta_k)^2}, \tag{2.8}$$

where γ_k are the linear weights determined in Step 1.2 above, and ε is a small number to avoid the denominator to become 0. We use $\varepsilon = 10^{-6}$ in all the computation in this paper. The final HWENO reconstruction is then given by

$$u_i^- \approx \sum_{j=0}^{2} \omega_j p_j'(x_i). \tag{2.9}$$

The reconstruction to u_i^+ is mirror symmetric with respect to x_i of the above procedure.

Step 2. Reconstruction of the values $\{u_{i+1/2}^{\pm}\}$ by HWENO from $\{\phi_i, \overline{u}_i\}$.

1. Given the small stencils $S_0 = \{x_{i-1}, x_i, I_{i-1}, I_i\}$, $S_1 = \{x_i, x_{i+1}, I_i, I_{i+1}\}$, $S_2 = \{x_{i-1}, x_i, x_{i+1}, I_i\}$ and the bigger stencil $\mathscr{T} = \{S_0, S_1, S_2\}$, we construct Hermite cubic reconstruction polynomials $p_0(x)$, $p_1(x)$, $p_2(x)$ and a fifth-degree reconstruction polynomial $q(x)$ such that:

$$p_0(x_{i+j}) = \phi_{i+j}, \quad \frac{1}{\Delta x}\int_{I_{i+j}} p_0'(x)\,\mathrm{d}x = \overline{u}_{i+j}, \qquad j = -1, 0,$$

$$p_1(x_{i+j}) = \phi_{i+j}, \quad \frac{1}{\Delta x}\int_{I_{i+j}} p_1'(x)\,\mathrm{d}x = \overline{u}_{i+j}, \qquad j = 0, 1,$$

$$p_2(x_{i+j}) = \phi_{i+j}, \quad j = -1, 0, 1, \qquad \frac{1}{\Delta x}\int_{I_i} p_2'(x)\,\mathrm{d}x = \overline{u}_i,$$

$$q(x_{i+j}) = \phi_{i+j}, \quad \frac{1}{\Delta x}\int_{I_{i+j}} q'(x)\,\mathrm{d}x = \overline{u}_{i+j}, \qquad j = -1, 0, 1.$$

In fact, we only need the values of the derivative of these polynomials at the cell boundary $x_{i+1/2}$, which have the following expressions:

$$p_0'(x_{i+1/2}) = \frac{1}{6}\left(-16\frac{\Delta^- \phi_i}{\Delta x} + 5\overline{u}_{i-1} + 17\overline{u}_i\right),$$

$$p_1'(x_{i+1/2}) = \frac{1}{6}\left(8\frac{\Delta^- \phi_{i+1}}{\Delta x} - \overline{u}_i - \overline{u}_{i+1}\right),$$

$$p_2'(x_{i+1/2}) = \frac{1}{6}\left(-\frac{\Delta^- \phi_i}{\Delta x} + 5\frac{\Delta^- \phi_{i+1}}{\Delta x} + 2\overline{u}_i\right),$$

$$q'(x_{i+1/2}) = \frac{1}{30}\left(-4\frac{\Delta^- \phi_i}{\Delta x} + 36\frac{\Delta^- \phi_{i+1}}{\Delta x} + \overline{u}_{i-1} + \overline{u}_i - 4\overline{u}_{i+1}\right).$$

2. We compute the linear weights by requiring

$$q'(x_{i+1/2}) = \sum_{j=0}^{2} \gamma_j p_j'(x_{i+1/2})$$

for all possible point values ϕ and cell averages of u in the bigger stencil $\mathscr{T}$. This leads to:

$$\gamma_0 = \frac{1}{25}, \quad \gamma_1 = \frac{4}{5}, \quad \gamma_2 = \frac{4}{25}.$$

3. We compute smoothness indicators (2.7), obtaining:

$$\beta_0 = \frac{13}{12}\left[-\frac{8}{\Delta x}\Delta^- \phi_i + 4\overline{u}_{i-1} + 4\overline{u}_i\right]^2 + \left[-\frac{4}{\Delta x}\Delta^- \phi_i + \overline{u}_{i-1} + 3\overline{u}_i\right]^2,$$

$$\beta_1 = \frac{13}{12}\left[-\frac{8}{\Delta x}\Delta^- \phi_{i+1} + 4\overline{u}_i + 4\overline{u}_{i+1}\right]^2 + \left[\frac{4}{\Delta x}\Delta^- \phi_{i+1} - 3\overline{u}_i - \overline{u}_{i+1}\right]^2,$$

$$\beta_2 = \frac{13}{12}\left[\frac{4}{\Delta x}(\Delta^- \phi_i + \Delta^- \phi_{i+1}) - 8\overline{u}_i\right]^2 + \left[\frac{1}{\Delta x}(-\Delta^- \phi_i + \Delta^- \phi_{i+1})\right]^2.$$

4. We compute the nonlinear weights by (2.8).

The final HWENO reconstruction to $u^-_{i+1/2}$ is then given by

$$u^-_{i+1/2} \approx \sum_{j=0}^{2} \omega_j p'_j(x_{i+1/2}). \tag{2.10}$$

The reconstruction to $u^+_{i-1/2}$ is mirror symmetric with respect to x_i of the above procedure.

2.2. Two dimensional case

We now proceed to consider the two dimensional Hamilton–Jacobi equation (1.1). For simplicity of presentation, we again assume that the mesh is uniform with the cell sizes $x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}} = \Delta x,\ y_{j+\frac{1}{2}} - y_{j-\frac{1}{2}} = \Delta y$ and the cell centers $(x_i, y_j) = (\frac{1}{2}(x_{i+\frac{1}{2}} + x_{i-\frac{1}{2}}), \frac{1}{2}(y_{j+\frac{1}{2}} + y_{j-\frac{1}{2}}))$. This assumption is again non-essential: the algorithm can be easily defined for tensor product non-uniform meshes. We denote the cells by $I_{ij} = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] \times [y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}]$. Let $u = \frac{\partial \phi}{\partial x},\ v = \frac{\partial \phi}{\partial y}$. Taking the derivatives of (1.1), we obtain:

$$\begin{cases} u_t + H_x = 0, \\ u(x, y, 0) = \frac{\partial \phi_0(x,y)}{\partial x}, \end{cases} \tag{2.11}$$

$$\begin{cases} v_t + H_y = 0, \\ v(x, y, 0) = \frac{\partial \phi_0(x,y)}{\partial y}. \end{cases} \tag{2.12}$$

We denote $\phi_{ij}(t) = \phi(x_i, y_j, t)$, and the x cell average of u as $\overline{u}_{ij}(t) = \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} u(x, y_j, t)\, \mathrm{d}x$, the y cell average of v as $\overline{v}_{ij}(t) = \frac{1}{\Delta y} \int_{y_{j-1/2}}^{y_{j+1/2}} v(x_i, y, t)\, \mathrm{d}y$. Integrating (2.11) and (2.12) over the cell $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ and the cell $[y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}]$, respectively, we obtain an equivalent formulation:

$$\begin{cases} \frac{\mathrm{d}}{\mathrm{d}t} \phi_{ij}(t) = -H(u(x_i, y_j, t), v(x_i, y_j, t)), \\ \frac{\mathrm{d}}{\mathrm{d}t} \overline{u}_{ij}(t) = -\frac{1}{\Delta x}(H(u(x_{i+1/2}, y_j, t), v(x_{i+1/2}, y_j, t)) - H(u(x_{i-1/2}, y_j, t), v(x_{i-1/2}, y_j, t))), \\ \frac{\mathrm{d}}{\mathrm{d}t} \overline{v}_{ij}(t) = -\frac{1}{\Delta y}(H(u(x_i, y_{j+1/2}, t), v(x_i, y_{j+1/2}, t)) - H(u(x_i, y_{j-1/2}, t), v(x_i, y_{j-1/2}, t))). \end{cases} \tag{2.13}$$

We approximate (2.13) by the following scheme:

$$\begin{cases} \frac{\mathrm{d}}{\mathrm{d}t} \phi_{ij}(t) = -\tilde{H}_{ij}, \\ \frac{\mathrm{d}}{\mathrm{d}t} \overline{u}_{ij}(t) = -\frac{1}{\Delta x}(\hat{H}_{i+1/2,j} - \hat{H}_{i-1/2,j}), \\ \frac{\mathrm{d}}{\mathrm{d}t} \overline{v}_{ij}(t) = -\frac{1}{\Delta y}(\hat{H}_{i,j+1/2} - \hat{H}_{i,j-1/2}), \end{cases} \tag{2.14}$$

where the numerical fluxes $\tilde{H}_{ij}, \hat{H}_{i+1/2,j}$ and $\hat{H}_{i,j+1/2}$ in (2.14) are again subject to the usual conditions for numerical fluxes, such as monotonicity, Lipschitz continuity and consistency with the physical flux $H(u,v)$. For choices of two dimensional numerical fluxes suitable for the HJ equations we again refer to, e.g. [21]. In this paper, we use the simple Lax–Friedrichs flux defined by:

$$\begin{aligned} \tilde{H}_{ij} &= H\left(\frac{u^-_{ij} + u^+_{ij}}{2}, \frac{v^-_{ij} + v^+_{ij}}{2}\right) - \frac{\alpha_x}{2}\left(u^+_{ij} - u^-_{ij}\right) - \frac{\alpha_y}{2}\left(v^+_{ij} - v^-_{ij}\right), \\ \hat{H}_{i+1/2,j} &= \frac{1}{2}\left(H(u^-_{i+1/2,j}, v^-_{i+1/2,j}) + H(u^+_{i+1/2,j}, v^+_{i+1/2,j}) - \alpha_x(u^+_{i+1/2,j} - u^-_{i+1/2,j})\right), \\ \hat{H}_{i,j+1/2} &= \frac{1}{2}\left(H(u^-_{i,j+1/2}, v^-_{i,j+1/2}) + H(u^+_{i,j+1/2}, v^+_{i,j+1/2}) - \alpha_y(v^+_{i,j+1/2} - v^-_{i,j+1/2})\right), \end{aligned} \tag{2.15}$$

where $u^{\pm}_{ij}$, $u^{\pm}_{i+1/2,j}$ and $v^{\pm}_{i+1/2,j}$ are the numerical approximations to the point values of $u(x_i,y_j,t)$, $u(x_{i+1/2},y_j,t)$ and $v(x_{i+1/2},y_j,t)$, respectively, from left and right, and $v^{\pm}_{ij}$, $u^{\pm}_{i,j+1/2}$ and $v^{\pm}_{i,j+1/2}$ are the numerical approximations to the point values of $v(x_i,y_j,t)$, $u(x_i,y_{j+1/2},t)$ and $v(x_i,y_{j+1/2},t)$, respectively, from bottom and top. The constants α_x and α_y are defined by $\alpha_x = \max_{u,v}|\frac{\partial}{\partial u}H(u,v)|$ and $\alpha_y = \max_{u,v}|\frac{\partial}{\partial v}H(u,v)|$.

The values $u^{\pm}_{ij}$, $u^{\pm}_{i+1/2,j}$, $v^{\pm}_{ij}$ and $v^{\pm}_{i,j+1/2}$ can be reconstructed by the one dimensional reconstruction methods presented in the previous subsection with the grid index for the other dimension fixed.

We now summarize the reconstruction procedure of $u^{\pm}_{i,j+1/2}$ and $v^{\pm}_{i+1/2,j}$ from $\{\phi_{ij},\overline{u}_{ij},\overline{v}_{ij}\}$ based either on the fourth order HWENO reconstruction or on the fourth order linear reconstruction (i.e., reconstruction with constant coefficients) below. We will denote $I_i = [x_{i-1/2},x_{i+1/2}]$ and $J_j = [y_{j-1/2},y_{j+1/2}]$.

1. We construct Hermite cubic reconstruction polynomials $p_1(x,y),\ldots,p_8(x,y)$ such that:

$$p_n(x_{k_1},y_{k_2}) = \phi_{k_1,k_2},$$

$$\frac{1}{\Delta x}\int_{I_{k_3}} \frac{\partial p_n(x,y_{k_4})}{\partial x}\,\mathrm{d}x = \overline{u}_{k_3,k_4}, \quad \frac{1}{\Delta y}\int_{J_{k_6}} \frac{\partial p_n(x_{k_5},y)}{\partial y}\,\mathrm{d}y = \overline{v}_{k_5,k_6},$$

where:

- $n=1$, $(k_1,k_2) = (i-1,j-1),(i,j-1),(i-1,j),(i,j)$, $(k_3,k_4) = (i-1,j-1),(i-1,j),(i,j)$, $(k_5,k_6) = (i-1,j-1),(i,j-1),(i,j)$;
- $n=2$, $(k_1,k_2) = (i,j-1),(i+1,j-1),(i,j),(i+1,j)$, $(k_3,k_4) = (i+1,j-1),(i,j),(i+1,j)$, $(k_5,k_6) = (i,j-1),(i+1,j-1),(i,j)$;
- $n=3$, $(k_1,k_2) = (i-1,j),(i,j),(i-1,j+1),(i,j+1)$, $(k_3,k_4) = (i-1,j),(i,j),(i-1,j+1)$, $(k_5,k_6) = (i,j),(i-1,j+1),(i,j+1)$;
- $n=4$, $(k_1,k_2) = (i,j),(i+1,j),(i,j+1),(i+1,j+1)$, $(k_3,k_4) = (i,j),(i+1,j),(i+1,j+1)$, $(k_5,k_6) = (i,j),(i,j+1),(i+1,j+1)$;
- $n=5$, $(k_1,k_2) = (i-1,j-1),(i,j-1),(i+1,j-1),(i-1,j),(i,j),(i-1,j+1)$, $(k_3,k_4) = (i-1,j),(i,j)$, $(k_5,k_6) = (i,j-1),(i,j)$;
- $n=6$, $(k_1,k_2) = (i-1,j-1),(i,j-1),(i+1,j-1),(i,j),(i+1,j),(i+1,j+1)$, $(k_3,k_4) = (i,j),(i+1,j)$, $(k_5,k_6) = (i,j-1),(i,j)$;
- $n=7$, $(k_1,k_2) = (i-1,j-1),(i-1,j),(i,j),(i-1,j+1),(i,j+1),(i+1,j+1)$, $(k_3,k_4) = (i-1,j),(i,j)$, $(k_5,k_6) = (i,j),(i,j+1)$;
- $n=8$, $(k_1,k_2) = (i+1,j-1),(i,j),(i+1,j),(i-1,j+1),(i,j+1),(i+1,j+1)$, $(k_3,k_4) = (i,j),(i+1,j)$, $(k_5,k_6) = (i,j),(i,j+1)$.

2. We combine the cubic polynomials to obtain a fourth-order approximation of u at the point $G = (x_i,y_{j+1/2})$.

If we choose the linear weights denoted by $\gamma_1,\ldots,\gamma_8$ such that

$$u(G) = \sum_{n=1}^{8}\gamma_n\frac{\partial}{\partial x}p_n(G) \tag{2.16}$$

is valid for any polynomial ϕ of degree at most 4, then we can obtain a fourth-order approximation of u at the point G for all sufficiently smooth functions.

Notice that (2.16) holds for any polynomial ϕ of degree at most 3 if $\sum_{n=1}^{8}\gamma_n = 1$. This is because each individual $p_n(x,y)$ reconstructs cubic polynomials exactly. There are five other constraints on the linear weights $\gamma_1,\ldots,\gamma_8$ from requiring (2.16) to hold for $\phi = x^4$, x^3y, x^2y^2, xy^3 and y^4, respectively. This leaves at least 2 free parameters in determining the linear weights $\gamma_1,\ldots,\gamma_8$. These free parameters are uniquely determined by a least square procedure

$$\min\left(\sum_{n=1}^{8}(\gamma_n)^2\right)$$

subject to the constraints listed above. The linear weights $\gamma_1,\ldots,\gamma_8$ chosen this way are positive.

If we now simply use a linear fourth order reconstruction, namely we simply use the linear weights determined above, then we can write out the explicit formula for the reconstruction

$$\begin{aligned} u^-_{i,j+1/2} = \frac{1}{144}\Big\{ & \frac{1}{\Delta x}\left[19\left(\phi_{i-1,j-1}-\phi_{i+1,j-1}\right)+2\left(\phi_{i-1,j}-\phi_{i+1,j}\right)-45\left(\phi_{i-1,j+1}-\phi_{i+1,j+1}\right)\right] \\ & +4\left(\overline{u}_{i-1,j-1}+\overline{u}_{i+1,j-1}-\overline{u}_{i-1,j}+30\overline{u}_{ij}-\overline{u}_{i+1,j}-2\overline{u}_{i-1,j+1}-2\overline{u}_{i+1,j+1}\right) \\ & +6\left(\overline{v}_{i-1,j-1}-\overline{v}_{i+1,j-1}+\overline{v}_{i-1,j+1}-\overline{v}_{i+1,j+1}\right)\Big\}, \end{aligned} \tag{2.17}$$

which can be easily implemented, without the need to go through the small stencils in the actual coding. However, if we would like to use a WENO reconstruction here, we would need to continue into the next step.

3. We compute the smoothness indicator, denoted by β_n, for each stencil S_n, which measures how smooth the function $p_n(x,y)$ is in the target cell I_{ij}. The smaller this smoothness indicator β_n, the smoother the function $p_n(x,y)$ is in the target cell. We use a similar recipe for the smoothness indicator as in [10]

$$\beta_n = \sum_{|k|=2}^{3} |I_{ij}|^{2|k|-1} \int_{I_{ij}} \left(\frac{\partial^{|k|}}{\partial x^{k_1}\partial y^{k_2}} p_n(x,y)\right)^2 \mathrm{d}x\,\mathrm{d}y, \tag{2.18}$$

where $k=(k_1,k_2)$. Notice that the summation starts from $|k|=2$ for the HJ equations, rather than from $|k|=1$ for the conservation law case [10]. In practice, the smoothness indicator is represented as a quadratic form of the point values and cell averages involved in the stencil with the coefficients in this quadratic form precomputed and stored.

4. We compute the nonlinear weights based on the smoothness indicators:

$$\omega_n = \frac{\overline{\omega}_n}{\sum_k \overline{\omega}_k}, \quad \overline{\omega}_k = \frac{\gamma_k}{(\varepsilon+\beta_k)^2}, \tag{2.19}$$

where γ_k are the linear weights determined in (2.16), and ε is a small number to avoid the denominator to become 0. We use $\varepsilon = 10^{-6}$ in all the computation in this paper. The final HWENO reconstruction is then given by:

$$u^-(G) \approx \sum_{n=1}^{8} \omega_n \frac{\partial}{\partial x} p_n(G). \tag{2.20}$$

The reconstruction for $u^+_{i,j+1/2}$ is mirror symmetric of that for $u^-_{i,j+1/2}$ with respect to $y_{j+1/2}$, and reconstruction for $v^\pm_{i+1/2,j}$ is the same as that for $u^\pm_{i,j+1/2}$ with i and j interchanged.

Clearly, the linear reconstruction (2.17) is much simpler and cost effective than the WENO reconstruction (2.20). While WENO reconstruction is important for the main terms $u^\pm_{ij}$, $u^\pm_{i+1/2,j}$, $v^\pm_{ij}$ and $v^\pm_{i,j+1/2}$ in each dimension, there is reason to believe that the cross terms $u^\pm_{i,j+1/2}$ and $v^\pm_{i+1/2,j}$ play a lesser role towards spurious oscillations and a linear reconstruction for those terms might be enough. This is indeed verified by our extensive numerical experiments in the following section.

3. Numerical results

In this section we present the results of our numerical experiments for the fifth order HWENO schemes for one-dimensional and two-dimensional examples with the third order TVD Runge–Kutta method. For the two-dimensional examples, both the linear reconstruction (2.17) and the WENO reconstruction (2.20) have been tested for the cross terms $u^{\pm}_{i,j+1/2}$ and $v^{\pm}_{i+1/2,j}$. Similar results have been obtained, hence we will show only the results obtained with the linear reconstruction (2.17) for the cross terms $u^{\pm}_{i,j+1/2}$ and $v^{\pm}_{i+1/2,j}$. A uniform mesh is used for all the test cases. The CFL number is taken as 0.8 for all test cases except for some accuracy tests where a suitably reduced time step is used to guarantee that spatial error dominates. The original fifth order WENO scheme for Hamilton–Jacobi equations by Jiang and Peng [12] with the same Lax–Friedrichs flux is used for comparison.

3.1. Accuracy tests

We first test the accuracy of the schemes on linear and nonlinear problems.

Example 3.1. We solve the following linear equation:

$$\phi_t + \phi_x = 0 \tag{3.1}$$

with the initial condition $\phi(x,0) = \sin(\pi x)$, and a 2-periodic boundary condition. We compute the solution up to $t = 2$, i.e., after one period by the HWENO scheme and the WENO scheme. The numerical results are shown in Table 1. We can see that both schemes achieve their designed order of accuracy with comparable errors for the same mesh. In fact, the HWENO scheme has smaller errors than the WENO schemes for most meshes.

Example 3.2. We solve the following nonlinear scalar Burgers' equation:

$$\phi_t + \frac{(\phi_x + 1)^2}{2} = 0 \tag{3.2}$$

with the initial condition $\phi(x,0) = -\cos(\pi x)$, and a 2-periodic boundary condition. When $t = 0.5/\pi^2$ the solution is still smooth. The errors and numerical orders of accuracy by the HWENO scheme and the WENO scheme are shown in Table 2. We can see that both schemes achieve their designed order of accuracy, and the HWENO scheme has smaller errors than the WENO scheme for the same mesh.

Table 1
$\phi_t + \phi_x = 0$, $\phi(x,0) = \sin(\pi x)$, HWENO and WENO schemes with periodic boundary conditions

N	HWENO				WENO			
	L_1 error	Order	L_∞ error	Order	L_1 error	Order	L_∞ error	Order
10	2.54E − 02		3.53E − 02		2.75E − 02		4.70E − 02	
20	1.14E − 03	4.47	1.80E − 03	4.29	1.13E − 03	4.60	2.34E − 03	4.33
40	4.48E − 05	4.67	7.30E − 05	4.63	4.11E − 05	4.78	7.17E − 05	5.03
80	1.55E − 06	4.85	2.50E − 06	4.87	1.37E − 06	4.91	2.23E − 06	5.00
160	3.65E − 08	5.41	6.05E − 08	5.37	4.39E − 08	4.97	6.97E − 08	5.00
320	3.28E − 10	6.80	5.60E − 10	6.76	1.38E − 09	4.99	2.18E − 09	5.00

$t = 2$. L_1 and L_∞ errors and numerical orders of accuracy. Uniform meshes with N cells.

Table 2
Burgers' equation $\phi_t + (\phi_x + 1)^2/2 = 0$ with initial condition $\phi(x,0) = -\cos(\pi x)$ by HWENO and WENO schemes with periodic boundary conditions

N	HWENO				WENO			
	L_1 error	Order	L_∞ error	Order	L_1 error	Order	L_∞ error	Order
10	1.69E − 03		7.81E − 03		1.70E − 02		7.05E − 02	
20	1.06E − 04	4.01	9.01E − 04	3.12	6.23E − 04	4.77	4.15E − 03	4.09
40	4.63E − 06	4.51	5.25E − 05	4.10	2.84E − 05	4.45	2.69E − 04	3.94
80	1.66E − 07	4.80	2.25E − 06	4.54	1.10E − 06	4.70	1.26E − 05	4.42
160	4.11E − 09	5.34	7.31E − 08	4.94	3.94E − 08	4.80	4.41E − 07	4.84
320	8.27E − 11	5.63	1.41E − 09	5.70	1.36E − 09	4.86	1.42E − 08	4.96

$t = 0.5/\pi^2$. L_1 and L_∞ errors and numerical orders of accuracy. Uniform meshes with N cells

Example 3.3. We solve the following nonlinear scalar two dimensional Burgers' equation:

$$\phi_t + \frac{(\phi_x + \phi_y + 1)^2}{2} = 0 \tag{3.3}$$

with the initial condition $\phi(x,y,0) = -\cos(\pi(x + y)/2)$, and a 4-periodic boundary condition. When $t = 0.5/\pi^2$ the solution is still smooth. The errors and numerical orders of accuracy by the HWENO scheme and the WENO scheme are shown in Table 3. We can see that both schemes achieve their designed order of accuracy, and the HWENO scheme has smaller errors than the WENO scheme for the same mesh.

3.2. Test cases with discontinuous derivatives

Example 3.4. We solve the same linear equation (3.1) as in Example 3.1 but with the discontinuous initial condition $\phi(x,0) = \phi_0(x-0.5)$, with periodic condition, where:

$$\phi_0(x) = -\left(\frac{\sqrt{3}}{2} + \frac{9}{2} + \frac{2\pi}{3}\right)(x+1) + \begin{cases} 2\cos\left(\frac{3\pi x^2}{2}\right) - \sqrt{3}, & -1 \leqslant x < -\frac{1}{3}, \\ \frac{3}{2} + 3\cos(2\pi x), & -\frac{1}{3} \leqslant x < 0, \\ \frac{15}{2} - 3\cos(2\pi x), & 0 \leqslant x < \frac{1}{3}, \\ \frac{28+4\pi+\cos(3\pi x)}{3} + 6\pi x(x-1), & \frac{1}{3} \leqslant x < 1. \end{cases} \tag{3.4}$$

We plot the results at $t = 2.0$ and $t = 8.0$ in Fig. 1. We can observe that the results by the HWENO scheme have better resolution for the corner singularity than that of the WENO scheme.

Table 3
Two dimensional Burgers' equation $\phi_t + (\phi_x + \phi_y + 1)^2/2 = 0$ with the initial condition $\phi(x,y,0) = -\cos(\pi(x + y)/2)$ by HWENO and WENO schemes with periodic boundary conditions

$N_x \times N_y$	HWENO				WENO			
	L_1 error	Order	L_∞ error	Order	L_1 error	Order	L_∞ error	Order
20 × 20	1.33E − 04		7.55E − 04		3.14E − 03		1.64E − 02	
40 × 40	5.76E − 06	4.53	5.39E − 05	3.81	1.16E − 04	4.75	6.18E − 04	4.73
80 × 80	1.95E − 07	4.89	2.17E − 06	4.63	3.77E − 06	4.95	1.90E − 05	5.02
160 × 160	4.94E − 09	5.30	6.98E − 08	4.96	1.19E − 07	4.98	6.01E − 07	4.98
320 × 320	1.06E − 10	5.54	1.34E − 09	5.71	3.78E − 09	4.98	1.90E − 08	4.98

$t = 0.5/\pi^2$. L_1 and L_∞ errors and numerical orders of accuracy. Uniform meshes with $N_x \times N_y$ cells.

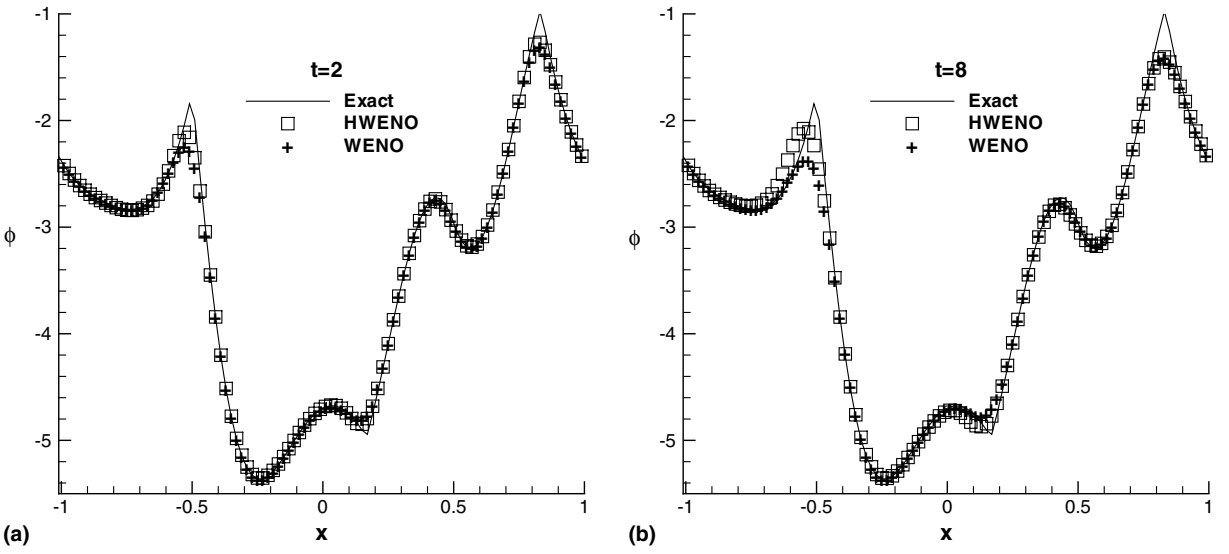

Fig. 1. One dimensional linear equation. $N = 100$ cells; (a) $t = 2$; (b) $t = 8$; solid lines: the exact solution; square symbols: the HWENO scheme; plus symbols: the WENO scheme.

Example 3.5. We solve the same nonlinear Burgers' equation (3.2) as in Example 3.2 with the same initial condition $\phi(x,0) = -\cos(\pi x)$, except that we now plot the results at $t = 3.5/\pi^2$ when discontinuous derivative has already appeared in the solution. In Fig. 2, the solutions of the HWENO scheme and the WENO scheme with $N = 40$ and $N = 80$ cells are shown. We can see that both schemes give good results for this problem.

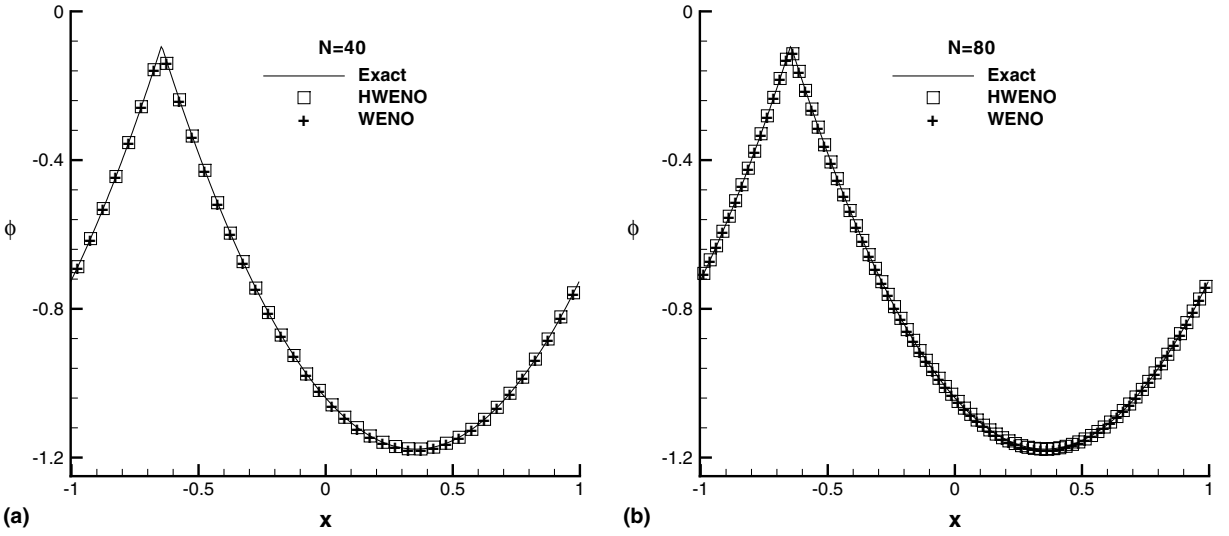

Fig. 2. Burgers' equation. $t = 3.5/\pi^2$; (a) $N = 40$ cells; (b) $N = 80$ cells; solid lines: the exact solution; square symbols: the HWENO scheme; plus symbols: the WENO scheme.

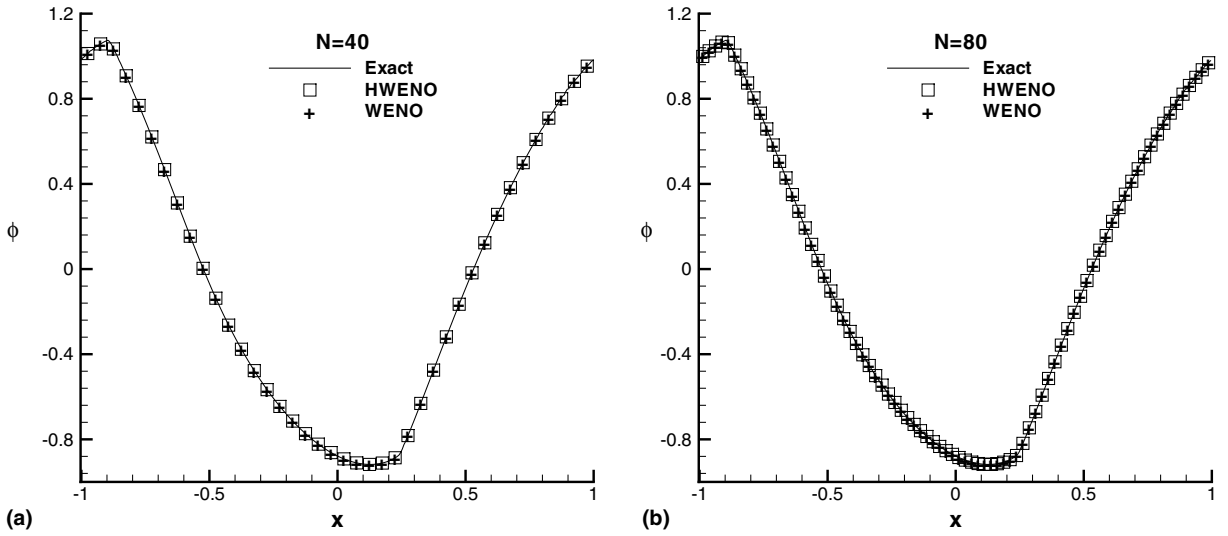

Fig. 3. Problem with the non-convex flux $H(u) = -\cos(u + 1)$. $t = 1.5/\pi^2$; (a) $N = 40$ cells; (b) $N = 80$ cells; solid lines: the exact solution; square symbols: the HWENO scheme; plus symbols: the WENO scheme.

Example 3.6. We solve the nonlinear equation with a non-convex flux

$$\phi_t - \cos(\phi_x + 1) = 0 \tag{3.5}$$

with the initial data $\phi(x,0) = -\cos(\pi x)$ and periodic boundary conditions. We plot the results at $t = 1.5/\pi^2$ when the discontinuous derivative has already appeared in the solution. In Fig. 3, the solutions of the

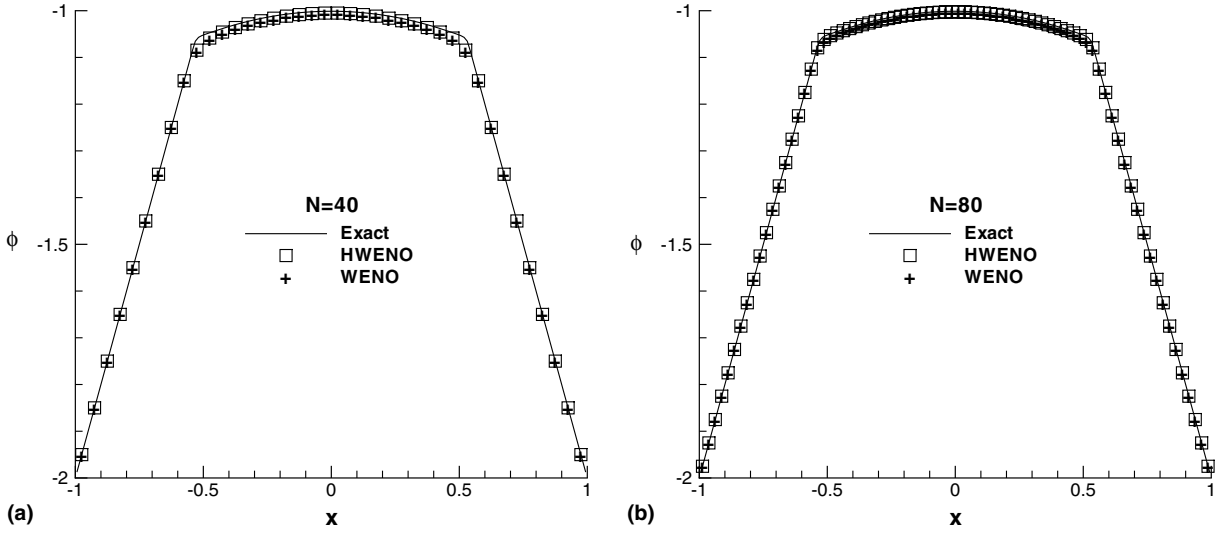

Fig. 4. Problem with the non-convex flux $H(u) = (1/4)(u^2 - 1)(u^2 - 4)$. $t = 1$; (a) $N = 40$ cells; (b) $N = 80$ cells; solid lines: the exact solution; square symbols: the HWENO scheme; plus symbols: the WENO scheme.

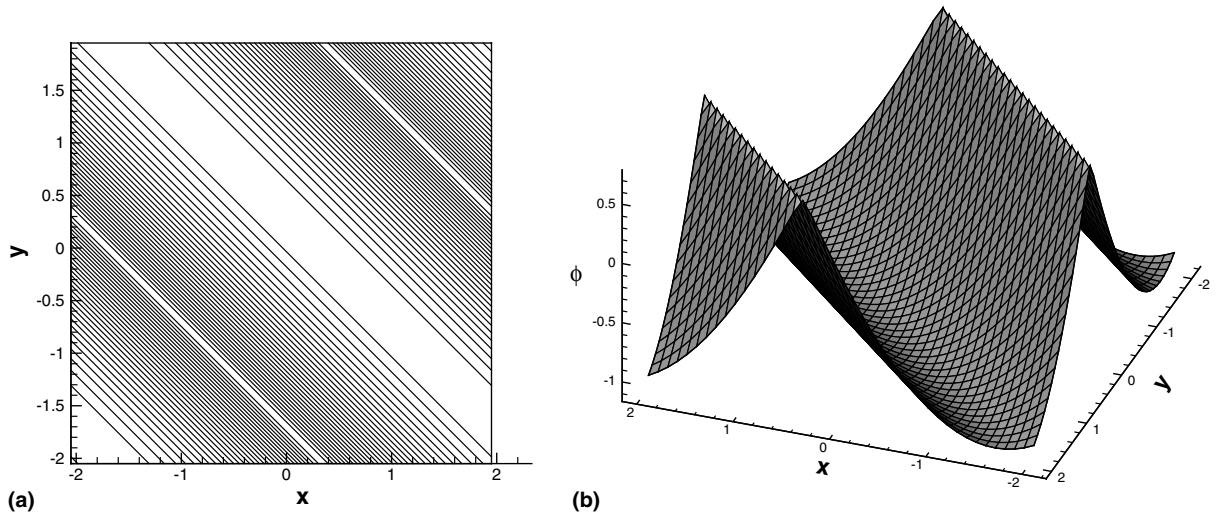

Fig. 5. Two dimensional Burgers' equation. $t = 1.5/\pi^2$ by the HWENO scheme with $N_x \times N_y = 40 \times 40$ cells. Contours of the solution (a) and surface of the solution (b).

HWENO scheme and the WENO scheme with $N = 40$ and $N = 80$ cells are shown. We can see that both schemes give good results for this problem.

Example 3.7. We solve the one dimensional Riemann problem with a non-convex flux:

$$\begin{cases} \phi_t - \frac{1}{4}(\phi_x^2 - 1)(\phi_x^2 - 4) = 0, & -1 < x < 1, \\ \phi(x,0) = -2|x|. \end{cases} \tag{3.6}$$

This is a demanding test case, for many schemes have poor resolutions or could even converge to a non-viscosity solution for this case. We plot the results at $t = 1$ by the HWENO scheme and the WENO scheme with $N = 40$ and $N = 80$ cells in Fig. 4. We can see that both schemes give good results for this problem.

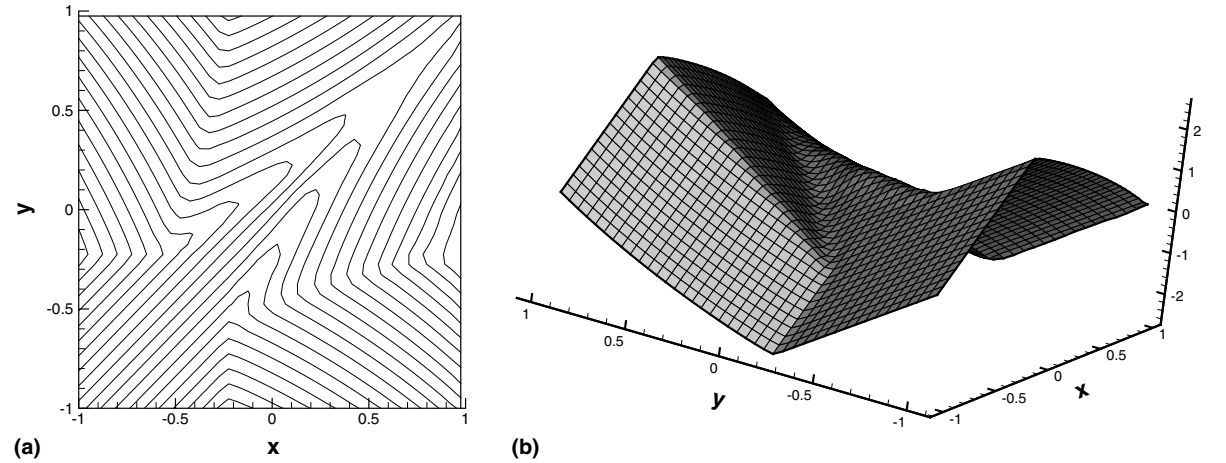

Fig. 6. Two dimensional Riemann problem with a non-convex flux $H(u,v) = \sin(u + v)$. $t = 1$ by the HWENO method with $N_x \times N_y = 40 \times 40$ cells. Contours of the solution (a) and surface of the solution (b).

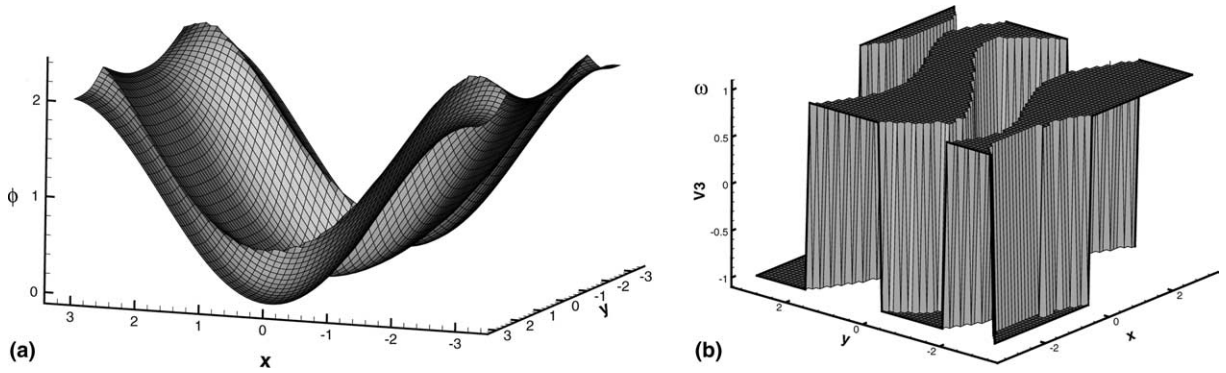

Fig. 7. The optimal control problem. $t = 1$ by the HWENO scheme with $N_x \times N_y = 60 \times 60$ cells. Surfaces of the solution (a) and of the optimal control $\omega = \text{sign}(\phi_y)$ (b).

Example 3.8. We solve the same two dimensional nonlinear Burgers' equation (3.3) as in Example 3.3 with the same initial condition $\phi(x,0) = -\cos(\pi(x + y)/2)$, except that we now plot the results at $t = 1.5/\pi^2$ when the discontinuous derivative has already appeared in the solution. The solution of the HWENO scheme with $N_x \times N_y = 40 \times 40$ cells are shown in Fig. 5. We observe good resolution for this example.

Example 3.9. The two dimensional Riemann problem with a non-convex flux:

$$\begin{cases} \phi_t + \sin(\phi_x + \phi_y) = 0, & -1 < x, y < 1, \\ \phi(x, y, 0) = \pi(|y| - |x|). \end{cases} \tag{3.7}$$

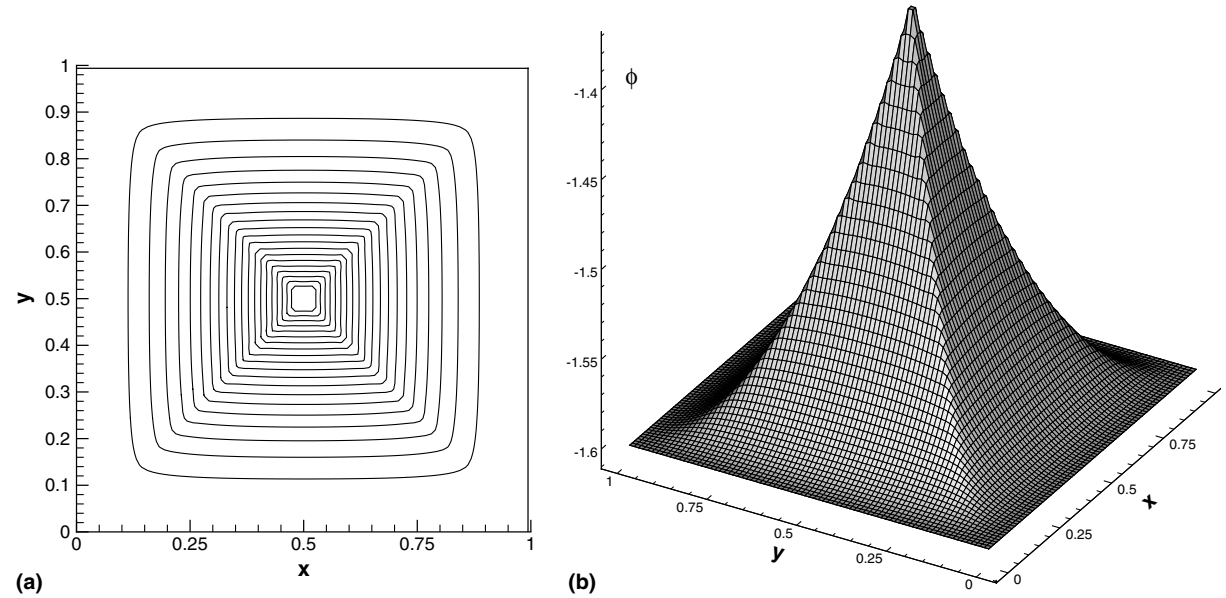

Fig. 8. Eikonal equation with a non-convex Hamiltonian. $t = 1$ by the HWENO schemes with $N_x \times N_y = 80 \times 80$ cells. Contours of the solution (a) and surface of the solution (b).

The solution of the HWENO scheme with $N_x \times N_y = 40 \times 40$ cells are shown in Fig. 6. We observe good resolution for this example.

Example 3.10. A problem from optimal control:

$$\begin{cases} \phi_t + \sin(y)\phi_x + (\sin x + \operatorname{sign}(\phi_y))\phi_y - \frac{1}{2}\sin^2 y - (1 - \cos x) = 0, \quad -\pi < x, y < \pi, \\ \phi(x, y, 0) = 0. \end{cases} \tag{3.8}$$

with periodic conditions, see [21]. The solution of the HWENO scheme with $N_x \times N_y = 60 \times 60$ cells and the optimal control $\omega = \operatorname{sign}(\phi_y)$ are shown in Fig. 7.

Example 3.11. A two dimensional eikonal equation with a non-convex Hamiltonian, which arises in geometric optics [11], is given by:

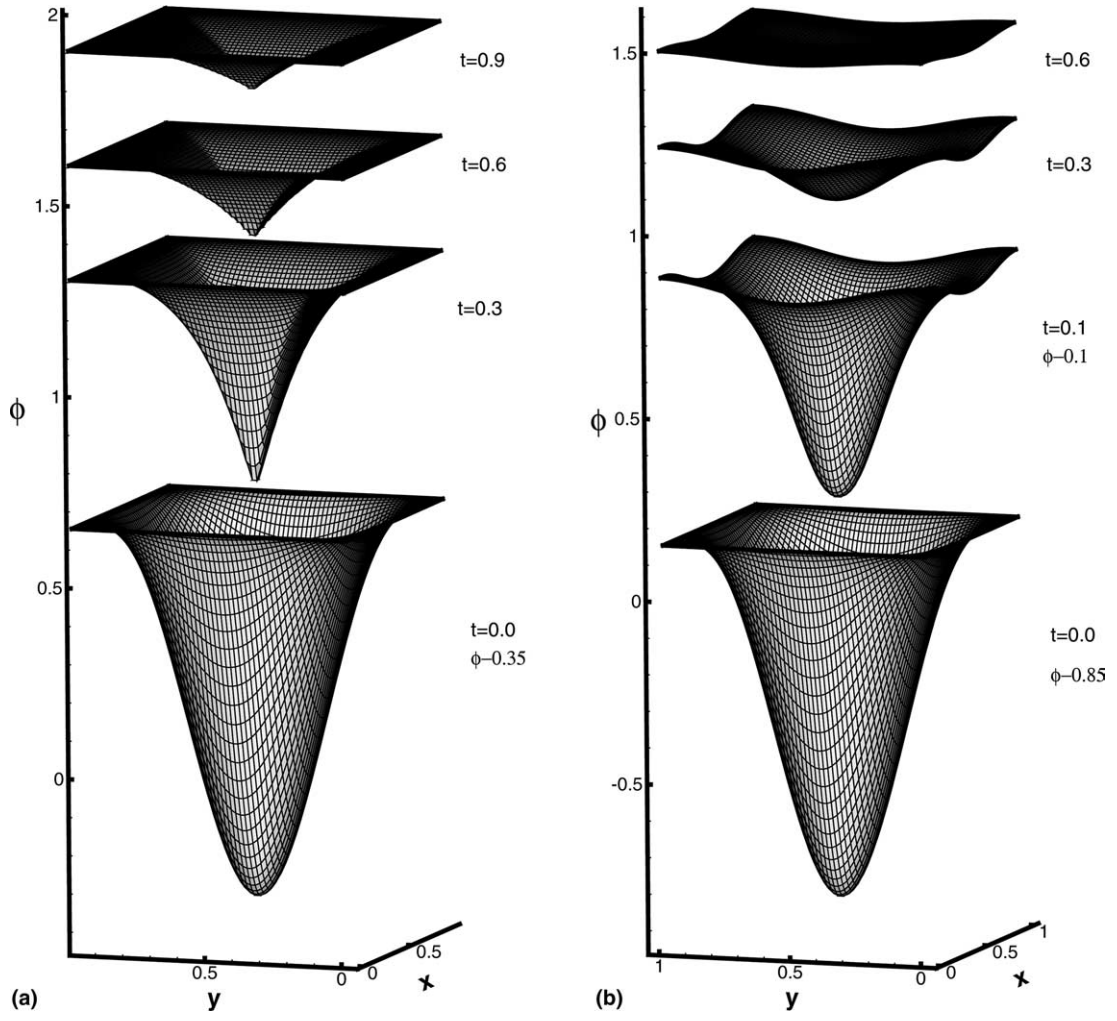

Fig. 9. Propagating surface. HWENO method with $N_x \times N_y = 60 \times 60$ cells. (a) $\varepsilon = 0$; (b) $\varepsilon = 0.1$.

$$\begin{cases} \phi_t + \sqrt{\phi_x^2 + \phi_y^2 + 1} = 0, \quad 0 \leqslant x, y < 1, \\ \phi(x, y, 0) = \frac{1}{4}(\cos(2\pi x) - 1)(\cos(2\pi y) - 1) - 1. \end{cases} \tag{3.9}$$

The solutions of the HWENO scheme with $N_x \times N_y = 80 \times 80$ cells is shown in Fig. 8. Good resolution is observed.

Example 3.12. The problem of a propagating surface [20]:

$$\begin{cases} \phi_t - (1 - \varepsilon K)\sqrt{\phi_x^2 + \phi_y^2 + 1} = 0, \quad 0 \leqslant x, y < 1, \\ \phi(x, y, 0) = 1 - \frac{1}{4}(\cos(2\pi x) - 1)(\cos(2\pi y) - 1), \end{cases} \tag{3.10}$$

where K is the mean curvature defined by

$$K = -\frac{\phi_{xx}(1 + \phi_y^2) - 2\phi_{xy}\phi_x\phi_y + \phi_{yy}(1 + \phi_x^2)}{(1 + \phi_x^2 + \phi_y^2)^{3/2}}.$$

and ε is a small constant. A periodic boundary condition is used. The approximation of the second derivative terms are constructed by the methods similar to that of the first derivative terms, but we only use linear weights in the reconstruction. The results of $\varepsilon = 0$ (pure convection) and $\varepsilon = 0.1$ by the HWENO method with $N_x \times N_y = 60 \times 60$ cells are presented in Fig. 9. The surfaces at $t = 0$ for $\varepsilon = 0$ and for $\varepsilon = 0.1$, and at $t = 0.1$ for $\varepsilon = 0.1$, are shifted downward in order to show the detail of the solution at later time.

4. Concluding remarks

In this paper, we have constructed a new class of fifth order WENO schemes, which we called the HWENO (Hermite WENO) schemes, for solving one and two dimensional Hamilton–Jacobi equations. The construction of HWENO schemes for Hamilton–Jacobi equations is based on Hermite interpolation and Runge–Kutta methods. The idea of the reconstruction for the HWENO schemes comes from the WENO schemes. In the HWENO schemes, both the solution and its first derivatives are evolved in time and used in the reconstruction, in contrast to the regular WENO schemes where only the solution value is evolved in time and used in the reconstruction. Comparing with the regular WENO schemes, one major advantage of HWENO schemes is their relatively compact stencil. Extensive numerical experiments are performed to illustrate the capability of the method.

Acknowledgements

The first author thank Professor B.C. Khoo and Professor G.W. Wei for their support and help.

References

[1] S. Augoula, R. Abgrall, High order numerical discretization for Hamilton–Jacobi equations on triangular meshes, J. Sci. Comput. 15 (2000) 198–229.
[2] F. Bouchut, C. Bourdarias, B. Perthame, A MUSCL method satisfying all the numerical entropy inequalities, Math. Comput. 65 (1996) 1439–1461.

[3] S. Bryson, D. Levy, High-order semi-discrete central-upwind schemes for multi-dimensional Hamilton–Jacobi equations, J. Comput. Phys. 189 (2003) 63–87.
[4] S. Bryson, D. Levy, High-order central WENO schemes for multidimensional Hamilton–Jacobi equations, SIAM J. Numer. Anal. 41 (2003) 1339–1369.
[5] T.J. Barth, J.A. Sethian, Numerical schemes for the Hamilton–Jacobi and level set equations on triangulated domains, J. Comput. Phys. 145 (1998) 1–40.
[6] T. Cecil, J. Qian, S. Osher, Numerical methods for high dimensional Hamilton–Jacobi equations using radial basis functions, J. Comput. Phys. 196 (2004) 327–347.
[7] R.L. Dougherty, A.S. Edelman, J.M. Hyman, Nonnegativity-, monotonicity- or convexity-preserving cubic and quintic Hermite interpolation, Math. Comput. 52 (1989) 471–494.
[8] O. Friedrichs, Weighted essentially non-oscillatory schemes for the interpolation of mean values on unstructured grids, J. Comput. Phys. 144 (1998) 194–212.
[9] A. Harten, B. Engquist, S. Osher, S. Chakravathy, Uniformly high order accurate essentially non-oscillatory schemes, III, J. Comput. Phys. 71 (1987) 231–303.
[10] C. Hu, C.-W. Shu, A discontinuous Galerkin finite element method for Hamilton–Jacobi equations, SIAM J. Sci. Comput. 21 (1999) 666–690.
[11] S. Jin, Z. Xin, Numerical passage from systems of conservation laws to Hamilton–Jacobi equations, and relaxation schemes, SIAM J. Numer. Anal. 35 (1998) 2163–2186.
[12] G. Jiang, D. Peng, Weighted ENO schemes for Hamilton–Jacobi equations, SIAM J. Sci. Comput. 21 (2000) 2126–2143.
[13] G. Jiang, C.-W. Shu, Efficient implementation of weighted ENO schemes, J. Comput. Phys. 126 (1996) 202–228.
[14] A. Kurganov, E. Tadmor, New high-resolution semi-discrete central schemes for Hamilton–Jacobi equations, J. Comput. Phys. 160 (2000) 720–742.
[15] F. Lafon, S. Osher, High order two dimensional nonoscillatory methods for solving Hamilton–Jacobi scalar equations, J. Comput. Phys. 123 (1996) 235–253.
[16] O. Lepsky, C. Hu, C.-W. Shu, Analysis of the discontinuous Galerkin method for Hamilton–Jacobi equations, Appl. Numer. Math. 33 (2000) 423–434.
[17] D. Levy, G. Puppo, G. Russo, Central WENO schemes for hyperbolic systems of conservation laws, Math. Model. Numer. Anal. 33 (1999) 547–571.
[18] X. Liu, S. Osher, T. Chan, Weighted essentially non-oscillatory schemes, J. Comput. Phys. 115 (1994) 200–212.
[19] T. Nakamura, R. Tanaka, T. Yabe, K. Takizawa, Exactly conservative semi-Lagrangian scheme for multi-dimensional hyperbolic equations with directional splitting technique, J. Comput. Phys. 174 (2001) 171–207.
[20] S. Osher, J. Sethian, Fronts propagating with curvature dependent speed: algorithms based on Hamilton–Jacobi formulations, J. Comput. Phys. 79 (1988) 12–49.
[21] S. Osher, C.-W. Shu, High-order essentially nonoscillatory schemes for Hamilton–Jacobi equations, SIAM J. Numer. Anal. 28 (1991) 907–922.
[22] J. Qiu, C.-W. Shu, On the construction, comparison, local characteristic decomposition for high order central WENO schemes, J. Comput. Phys. 183 (2002) 187–209.
[23] J. Qiu, C.-W. Shu, Hermite WENO schemes and their application as limiters for Runge–Kutta discontinuous Galerkin method: one dimensional case, J. Comput. Phys. 193 (2003) 115–135.
[24] J. Qiu, C.-W. Shu, Hermite WENO schemes and their application as limiters for Runge–Kutta discontinuous Galerkin method II: two dimensional case, Comp. Fluid (in press).
[25] C.-W. Shu, Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws, in: B. Cockburn, C. Johnson, C.-W. Shu, E. Tadmor (Eds.), Advanced Numerical Approximation of Nonlinear Hyperbolic Equations, A. Quarteroni (Ed.), Lecture Notes in Mathematics, vol. 1697, Springer, Berlin, 1998, pp. 325–432.
[26] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, J. Comput. Phys. 77 (1988) 439–471.
[27] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock capturing schemes II, J. Comput. Phys. 83 (1989) 32–78.
[28] H. Takewaki, A. Nishiguchi, T. Yabe, Cubic interpolated pseudoparticle method (CIP) for solving hyperbolic type equations, J. Comput. Phys. 61 (1985) 261–268.
[29] Y.-T. Zhang, C.-W. Shu, High-order WENO schemes for Hamilton–Jacobi equations on triangular meshes, SIAM J. Sci. Comput. 24 (2003) 1005–1030.