Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational and Applied Mathematics 200 (2007) 591–605
JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS
www.elsevier.com/locate/cam

WENO schemes with Lax–Wendroff type time discretizations for Hamilton–Jacobi equations☆

Jianxian Qiu

Department of Mathematics, Nanjing University, Nanjing, Jiangsu 210093, PR China

Received 18 June 2005; received in revised form 16 January 2006

Abstract

In this paper, a class of weighted essentially non-oscillatory (WENO) schemes with a Lax–Wendroff time discretization procedure, termed WENO-LW schemes, for solving Hamilton–Jacobi equations is presented. This is an alternative method for time discretization to the popular total variation diminishing (TVD) Runge–Kutta time discretizations. We explore the possibility in avoiding the nonlinear weights for part of the procedure, hence reducing the cost but still maintaining non-oscillatory properties for problems with strong discontinuous derivative. As a result, comparing with the original WENO with Runge–Kutta time discretizations schemes (WENO-RK) of Jiang and Peng [G. Jiang, D. Peng, Weighted ENO schemes for Hamilton–Jacobi equations, SIAM J. Sci. Comput. 21 (2000) 2126–2143] for Hamilton–Jacobi equations, the major advantages of WENO-LW schemes are more cost effective for certain problems and their 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; Lax–Wendroff type time discretization; High-order accuracy

1. Introduction

In this paper, we study an alternative method for time discretization, namely the Lax–Wendroff type time discretization [15], to the popular TVD Runge–Kutta time discretization in [25], for weighted essentially non-oscillatory (WENO) schemes[17,11,10], termed WENO-LW schemes, in solving the Hamilton–Jacobi (HJ) equations:

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

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 [18] and Osher and Shu [19] in

☆ Research partially supported by NNSFC Grant 10371118 and Nanjing University Talent Development Foundation.
E-mail address: jxqiu@nju.edu.cn.

0377-0427/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2006.01.022

constructing high-order essentially non-oscillatory (ENO) 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 [8,25,26]. ENO schemes for solving the HJ equations on unstructured meshes were constructed in [14]. More recently, ENO schemes based on radial basis functions were constructed in [5]. Central high resolution schemes were developed in [3,4,13]. Finite element methods suitable for arbitrary triangulations were developed in [1,2,9,16]. Finally, most relevant to our work, we mention the WENO schemes for solving the HJ equations [10] by Jiang and Peng, based on the WENO schemes for solving conservation laws [17,11], and Hermite WENO schemes in [22]. Zhang and Shu [29] further developed high-order WENO schemes on unstructured meshes for solving two-dimensional HJ equations.

WENO is a spatial discretization procedure, namely, it is a procedure to approximate the spatial derivative terms in (1.1). The time derivative term there must also be discretized. There are mainly two different approaches to approximate the time derivative. The first approach is to use an ODE solver, such as a Runge–Kutta or a multi-step method, to solve the method of lines ODE obtained after spatial discretization. The second approach is a Lax–Wendroff type time discretization procedure, which is also called the Taylor type referring to a Taylor expansion in time or the Cauchy–Kowalewski type referring to the similar Cauchy–Kowalewski procedure in PDE. This approach is based on the idea of the classical Lax–Wendroff scheme [15], and it relies on converting all the time derivatives in a temporal Taylor expansion into spatial derivatives by repeatedly using the PDE and its differentiated versions. The spatial derivatives are then discretized by, e.g., the WENO approximations.

The first approach, namely the method of lines plus an ODE solver, has the advantage of simplicity, both in concept and in coding. It also enjoys good stability properties when the TVD type Runge–Kutta or multi-step methods are used [25,24]. Thus, the majority of the WENO codes are using this type of time discretizations.

The second approach, the Lax–Wendroff type time discretization, usually produces the same high-order accuracy with a smaller effective stencil than that of the first approach, and it uses more extensively the original PDE. The original finite volume ENO schemes in [8] used this approach for the time discretization. More recently, a Lax–Wendroff type time discretization procedure for high-order finite difference WENO schemes was developed in [21]. This approach was also used in [27,28,23], termed ADER (arbitrary high-order schemes utilizing higher-order derivatives), to construct a class of high-order schemes for conservation laws in finite volume version. The Lax–Wendroff type time discretization was also used in discontinuous Galerkin method [6,7,20].

In this paper, based on the WENO-LW methodology for conservation laws in [21], we develop WENO-LW schemes to solve the HJ equations. Comparing with the original WENO-RK schemes of Jiang and Peng [10], one major advantage of WENO-LW schemes is their compactness in the reconstruction. For example, in the one-dimensional case, 19 or 25 points are needed in the stencil for WENO5-RK3 and WENO5-RK4, respectively, while only 15 or 17 points are needed for WENO5-LW3 or WENO5-LW4, respectively. In this paper, we use WENOn-LWk and WENOn-RKk, to denote the nth-order WENO scheme with the kth-order Lax–Wendroff time discretization and the kth-order Runge–Kutta time discretization, respectively.

The organization of this paper is as follows. In Section 2, we describe in detail the construction and implementation of the WENO-LW schemes, for one- and two-dimensional Hamilton–Jacobi equations (1.1). In Section 3, we provide extensive numerical examples to demonstrate the behavior of the schemes and to perform a comparison with the original WENO-RK schemes for HJ equations in [10]. Concluding remarks are given in Section 4.

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

In this section, we will present the details of the construction of WENO-LW 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 $\Delta x = x_{i+1} - x_i$.

We denote $\phi^{(r)}$ by the rth-order time derivative of ϕ, namely $\partial^r \phi / \partial t^r$. We also use ϕ', ϕ'' and ϕ''' to denote the first three time derivatives of ϕ. By a temporal Taylor expansion we obtain

$$\phi(x, t + \Delta t) = \phi(x, t) + \Delta t \phi' + \frac{\Delta t^2}{2}\phi'' + \frac{\Delta t^3}{6}\phi''' + \frac{\Delta t^4}{24}\phi^{(4)} + \cdots . \tag{2.1}$$

If we would like to obtain kth-order accuracy in time, we would need to approximate the first k time derivatives: $\phi', \ldots, \phi^{(k)}$. We will proceed up to fourth-order in time in this paper, although the procedure can be naturally extended to any higher orders.

From (1.1), we obtain

$$\phi' = -H(u), \tag{2.2}$$

$$\phi'' = -H'(u)\phi'_x, \tag{2.3}$$

$$\phi''' = -H''(u)(\phi'_x)^2 - H'(u)\phi''_x, \tag{2.4}$$

$$\phi^{(4)} = -H'''(u)(\phi'_x)^3 - 3H''(u)(\phi'_x)\phi''_x - H'(u)\phi'''_x, \tag{2.5}$$

where $u = \phi_x$.

After extensive numerical tests, we have found the following Lax–Wendroff procedure which produces the best balance between cost reduction and ensuring ENO properties to reconstruct ϕ_x, ϕ'_x, ϕ''_x, ϕ'''_x, and ϕ', ϕ'', ϕ''', $\phi^{(4)}$:

Step 1: The reconstruction of the first time derivative $\phi' = -H(u)$.

We approximate $\phi' = -H(u)$ by the following schemes:

$$\phi'_i = -\hat{H}_i, \tag{2.6}$$

where the numerical flux $\hat{H}_i$ in (2.6) are subject to the usual conditions for numerical fluxes, such as Lipschitz continuity and consistency with the physical fluxes $H(u)$, in this paper, we use the following Lax–Friedrichs flux defined by

$$\hat{H}_i = H\left(\frac{u_i^- + u_i^+}{2}\right) - \frac{\alpha}{2}(u_i^+ - u_i^-), \tag{2.7}$$

where $u_i^{\pm}$ are numerical approximations to the point values of $u(x_i, t)$, respectively, from left and right, and $\alpha = \sup_u |H'(u)|$.

Let us introduce $\phi_i = \phi(x_i)$, $\Delta^-\phi_i = \phi_i - \phi_{i-1}$. Reconstruction of $\{u_i^-\}$ is obtained by following $(2r+1)$th-order WENO procedure [11,10].

1. Given the small stencils $S_j = \{x_{i+j-r-1}, x_{i+j-r}, \ldots, x_{i+j}\}$, $j = 0, \ldots, r$, and the bigger stencil $\mathscr{T} = \{S_0, \ldots, S_{r-1}\}$, we construct rth degree reconstruction polynomials $p_0(x), \ldots, p_{r-1}(x)$ and a $2r$th degree reconstruction polynomial $q(x)$ such that

$$\int_{x_{i+l-1}}^{x_{i+l}} p_j(x)\,\mathrm{d}x = \Delta^-\phi_{i+l}, \quad j = 0, \ldots, r, \quad l = j-r, \ldots, j,$$

$$\int_{x_{i+l-1}}^{x_{i+l}} q(x)\,\mathrm{d}x = \Delta^-\phi_{i+l}, \quad l = -r, \ldots, r.$$

In fact, we only need the values of these polynomials at the x_i. For example, for $r = 2$, we have the following expressions [10]:

$$p_0(x_i) = \frac{1}{6\Delta x}(2\Delta^-\phi_{i-2} - 7\Delta^-\phi_{i-1} + 11\Delta^-\phi_i),$$

$$p_1(x_i) = \frac{1}{6\Delta x}(-\Delta^-\phi_{i-1} + 5\Delta^-\phi_i + 2\Delta^-\phi_{i+1}),$$

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

$$q(x_i) = \frac{1}{60\Delta x}(2\Delta^-\phi_{i-2} - 13\Delta^-\phi_{i-1} + 47\Delta^-\phi_i + 27\Delta^-\phi_{i+1} - 3\Delta^-\phi_{i+2}).$$

2. We find the combination coefficients, also called linear weights, denoted by $\gamma_0, \ldots, \gamma_r$, satisfying

$$q(x_i) = \sum_{j=0}^{r} \gamma_j p_j(x_i), \tag{2.8}$$

for all point values ϕ in the bigger stencil $\mathscr{T}$. For example, for $r = 2$, we have

$$\gamma_0 = \frac{1}{10}, \quad \gamma_1 = \frac{6}{10}, \quad \gamma_2 = \frac{3}{10}.$$

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 in the target point x_i. The smaller this smoothness indicator β_j, the smoother the function $p_j(x)$ is in the target point. We use the same recipe for the smoothness indicator as in [10]:

$$\beta_j = \sum_{l=1}^{r} \int_{x_{i-1}}^{x_i} \Delta x^{2l-1} \left(\frac{\partial^l}{\partial x^l} p_j(x) \right)^2 \mathrm{d}x. \tag{2.9}$$

In the actual numerical implementation the smoothness indicators β_j are written out explicitly as quadratic forms of the point values of ϕ in the stencil, for example for $r = 2$, we obtain [11,10]

$$\beta_0 = \frac{13}{12\Delta x}(\Delta^- \phi_{i-2} - 2\Delta^- \phi_{i-1} + \Delta^- \phi_i)^2 + \frac{1}{4\Delta x}(3\Delta^- \phi_{i-2} - 4\Delta^- \phi_{i-1} + \Delta^- \phi_i)^2,$$

$$\beta_1 = \frac{13}{12\Delta x}(\Delta^- \phi_{i-1} - 2\Delta^- \phi_i + \Delta^- \phi_{i+1})^2 + \frac{1}{4\Delta x}(\Delta^- \phi_{i-1} - \Delta^- \phi_{i+1})^2,$$

$$\beta_2 = \frac{13}{12\Delta x}(\Delta^- \phi_i - 2\Delta^- \phi_{i+1} + \Delta^- \phi_{i+2})^2 + \frac{1}{4\Delta x}(\Delta^- \phi_i - 4\Delta^- \phi_{i+1} + 3\Delta^- \phi_{i+2})^2.$$

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

$$\omega_j = \frac{\overline{\omega}_j}{\sum_k \overline{\omega}_k}, \quad \overline{\omega}_k = \frac{\gamma_k}{(\varepsilon + \beta_k)^2}, \tag{2.10}$$

where γ_k are the linear weights determined in sub-step 2 above, and ε is a small number to avoid the denominator to become 0. We are using $\varepsilon = 10^{-6}$ in all the computation in this paper. The final WENO reconstruction is then given by

$$u_i^- \approx \sum_{j=0}^{r} \omega_j p_j(x_i). \tag{2.11}$$

The reconstruction to u_i^+ is mirror symmetric with respect to x_i of the above procedure.
Then we get an approximation of ϕ_x at point x_i: $(\phi_x)_i \approx (u_i^+ + u_i^-)/2$, and ϕ_i' by (2.6).

Step 2. In order to reconstruct the second time derivative $\phi'' = -H'(\phi_x)\phi_x'$, we only need to reconstruct ϕ_x'. The reconstruction of ϕ_x' is obtained as following. Note that we will only need an approximation of order $2r$, one order lower than before, because of the extra Δt factor. Let $g_i = \phi_i'$, where ϕ_i' is the point value of ϕ' at the point (x_i, t^n) computed in Step 1 described above. *We can use a simple* $2r$*th-order central difference formula to approximate* ϕ_x' *at the point* (x_i, t^n). For example, when $r = 1$ we use the following second-order central difference approximation

$$(\phi_x')_i \approx -\frac{1}{2\Delta x}(g_{i+1} - g_{i-1}) \tag{2.12}$$

and when $r = 2$ we use the following fourth-order central difference approximation

$$(\phi_x')_i \approx -\frac{1}{12\Delta x}(g_{i-2} - 8g_{i-1} + 8g_{i+1} - g_{i+2}) \tag{2.13}$$

then we get an approximation of ϕ'' at point x_i:

$$\phi_i'' \approx -H'((\phi_x)_i)(\phi_x')_i \tag{2.14}$$

It seems that a more costly WENO approximation is *not* needed here to control spurious oscillations, presumably because this term is multiplied by an extra Δt anyway.

Step 3: The third time derivative $\phi''' = -H''(\phi_x)(\phi_x')^2 - H'(\phi_x)\phi_x''$. Similar to Step 2, we only need to reconstruct ϕ_x''. Let $g_i = \phi_i''$, here ϕ_i'' is the point value of ϕ'' at the point (x_i, t^n) computed in Step 2 above. Then we repeat Step 2 to get the approximation of ϕ_x'' using a central difference approximation of order $2r$. In fact, we only need an approximation of order $(2r-1)$, because of the extra Δt^2 factor, but we would like to use simple central differences which are all of even order. Then we get an approximation of ϕ''' at point x_i:

$$\phi_i''' \approx -H''((\phi_x)_i)(\phi_x')_i^2 - H'((\phi_x)_i)(\phi_x'')_i. \tag{2.15}$$

Again, it seems that a more costly WENO approximation is *not* needed here to control spurious oscillations.

Step 4. The fourth time derivative $\phi^{(4)} = -H'''(\phi_x)(\phi_x')^3 - 3H''(\phi_x)\phi_x''\phi_x' - H'(\phi_x)\phi_x'''$ is obtained in a similar fashion. Let $g_i = \phi_i'''$, where ϕ_i''' is the point value of ϕ''' at the point (x_i, t^n) computed in Step 3 above. In order to get a $(2r+1)$th-order scheme, we only need to use a $(2r-2)$th-order central difference approximation to ϕ_x''' at the point (x_i, t^n), because of the extra Δt^3 factor. For example, when $r=2$ we can use the following second-order approximation:

$$(\phi_x''')_i \approx -\frac{1}{2\Delta x}(g_{i+1} - g_{i-1}). \tag{2.16}$$

Then we get an approximation of $\phi^{(4)}$ at point x_i:

$$\phi_i^{(4)} = -H'''((\phi_x)_i)(\phi_x')_i^3 - 3H''((\phi_x)_i)(\phi_x'')_i(\phi_x')_i - H'((\phi_x)_i)(\phi_x''')_i. \tag{2.17}$$

If we require higher order accuracy in time this procedure can be continued in a similar fashion. The final approximation at the next time step is then given by

$$\phi(x_i, t^{n+1}) \approx \phi_i + \Delta t\phi_i' + \frac{\Delta t^2}{2}\phi_i'' + \frac{\Delta t^3}{6}\phi_i''' + \frac{\Delta t^4}{24}\phi_i^{(4)} + \cdots + \frac{\Delta t^k}{k!}\phi_i^{(k)}. \tag{2.18}$$

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 grid points $\{(x_i, y_j)\}$ are uniformly distributed with the cell size $\Delta x = x_{i+1} - x_i$, $\Delta y = y_{j+1} - y_j$.

From (1.1) in two-dimensional case, we have

$$\phi' = -H, \tag{2.19}$$

$$\phi'' = -H_1\phi_x' - H_2\phi_y', \tag{2.20}$$

$$\phi''' = -H_{11}(\phi_x')^2 - 2H_{12}\phi_x'\phi_y' - H_{22}(\phi_y')^2 - H_1\phi_x'' - H_2\phi_y'', \tag{2.21}$$

$$\begin{aligned}\phi^{(4)} = &-H_{111}(\phi_x')^3 - 3H_{112}(\phi_x')^2\phi_y' - 3H_{122}\phi_x'(\phi_y')^2 - H_{222}(\phi_y')^3\\ &- 3H_{11}\phi_x'\phi_x'' - 3H_{12}(\phi_x''\phi_y' + \phi_x'\phi_y'') - 3H_{22}\phi_y'\phi_y'' - H_1\phi_x''' - H_2\phi_y''',\end{aligned} \tag{2.22}$$

where H_i is the partial derivative of H with respect to ith argument, H_{ij} is the second partial derivative of H with respect to ith and jth arguments and H_{ijk} is the third partial derivative of H with respect to ith , jth and kth arguments.

Table 1
CPU time (in seconds) for the WENO5-LW4 and WENO5-RK4 schemes for the two-dimensional Burgers' equation at final time $t = 0.5/\pi^2$ and $3.5/\pi^2$

t	WENO5-LW4	WENO5-RK4
$0.5/\pi^2$	5.23	9.28
$3.5/\pi^2$	37.28	62.16

Total CPU time for $N = 10, 20, 40, 80, 160$ and 320 cells is recorded.

Similar to the one-dimensional case, what we want to do is to reconstruct ϕ_x, ϕ_y, ϕ'_x, ϕ'_y, ϕ''_x, ϕ''_y, ϕ'''_x , ϕ'''_y and ϕ', ϕ'', ϕ''', $\phi^{(4)}$ from point values $\{\phi_{ij} = \phi(x_i, y_j, t^n)\}$, respectively.

Let $u = \phi_x$, $v = \phi_y$, the first time derivative $\phi' = -H(u, v)$ is approximated by the following scheme:

$$\phi'_{ij} = -\hat{H}(u^-_{ij}, u^+_{ij}; v^-_{ij}, v^+_{ij}),$$

where $\hat{H}$ is a numerical flux. In this paper, we use the following Lax–Friedrichs flux:

$$\hat{H}_{ij} = H\left(\frac{u^-_{ij} + u^+_{ij}}{2}, \frac{v^-_{ij} + v^+_{ij}}{2}\right) - \frac{\alpha_x}{2}(u^+_{ij} - u^-_{ij}) - \frac{\alpha_y}{2}(v^+_{ij} - v^-_{ij}), \tag{2.23}$$

where $u^{\pm}_{ij}$ are numerical approximations to the point values of $\phi_x(x_i, y_j, t)$, respectively, from left and right, $v^{\pm}_{ij}$ are numerical approximations to the point values of $\phi_y(x_i, y_j, t)$, respectively, from bottom and top, $\alpha_x = \sup_{u,v} |H_1(u, v)|$ and $\alpha_y = \sup_{u,v} |H_2(u, v)|$.

The $u^{\pm}_{ij}$ and $v^{\pm}_{ij}$ are obtained by WENO reconstruction procedure described in Step 1 for one-dimensional case in a dimension-by-dimension fashion, and similar to one-dimensional case, we get an approximation of ϕ_x and ϕ_y at point (x_i, y_j): $(\phi_x)_{ij} \approx (u^+_{ij} + u^-_{ij})/2$ and $(\phi_y)_{ij} \approx (v^+_{ij} + v^-_{ij})/2$, respectively.

On the other hand, as in the one-dimensional situation, the derivatives ϕ'_x, ϕ'_y, ϕ''_x, ϕ''_y, ϕ'''_x and ϕ'''_y etc., can be approximated by simple central differences of suitable orders of accuracy, again in a dimension-by-dimension fashion, and the approximation of ϕ'' , ϕ''', $\phi^{(4)}$ are obtained from (2.20)–(2.22), respectively.

3. Numerical results

In this section, we present the results of our numerical experiments for the third- and the fifth-order WENO schemes for one-dimensional and two-dimensional examples with the third-order and the fourth-order Lax–Wendroff time discretization. A uniform mesh is used for all the test cases. The CFL number is taken as 0.6 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 WENO scheme with Runge–Kutta time discretization method for Hamilton–Jacobi equations in [10] with the same Lax–Friedrichs flux is used for comparison.

We first remark on the important issue of CPU timing and relevant efficiency of WENO-LW schemes compared with WENO-RK schemes. In general, the WENO-LW schemes have smaller CPU cost for the same mesh and same order of accuracy in our implementation. For example, in Table 1, we provide a CPU time comparison between WENO5-LW4 and WENO5-RK4 schemes for the two-dimensional Burgers' equation in Examples 3.3 and 3.8. We can see that the CPU cost for the WENO-LW schemes is about 60% for the WENO-RK schemes. The computations are performed on a Dell Precision Workstation 370, P4-2.8 with 2 GB ram.

3.1. Accuracy tests

We first test the accuracy of the schemes on linear and nonlinear problems. We have tested both WENO3 and WENO5, only the results of WENO5 are shown to save space in this subsection.

Table 2
$\phi_t + \phi_x = 0$. $\phi(x, 0) = \sin(\pi x)$. WENO5-LW4 and WENO5-RK4 schemes with periodic boundary conditions

N	WENO5-LW4				WENO5-RK4			
	L_1 error	Order	L_∞ error	Order	L_1 error	Order	L_∞ error	Order
10	2.51E−02		4.48E−02		2.69E−02		4.59E−02	
20	1.05E−03	4.59	2.19E−03	4.36	1.11E−03	4.60	2.30E−03	4.32
40	3.86E−05	4.76	6.77E−05	5.01	4.03E−05	4.78	7.04E−05	5.03
80	1.30E−06	4.89	2.12E−06	5.00	1.35E−06	4.91	2.19E−06	5.01
160	4.18E−08	4.96	6.66E−08	4.99	4.31E−08	4.96	6.85E−08	5.00
320	1.32E−09	4.98	2.09E−09	4.99	1.36E−09	4.99	2.14E−09	5.00

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

Table 3
Burgers' equation $\phi_t + (\phi_x + 1)^2/2 = 0$ with initial condition $\phi(x, 0) = -\cos(\pi x)$ by WENO5-LW4 and WENO5-RK4 schemes with periodic boundary conditions

N	WENO5-LW4				WENO5-RK4			
	L_1 error	Order	L_∞ error	Order	L_1 error	Order	L_∞ error	Order
10	3.57E−03		1.26E−02		4.41E−03		1.84E−02	
20	2.20E−04	4.02	1.81E−03	2.80	2.62E−04	4.07	2.31E−03	2.99
40	1.17E−05	4.24	1.20E−04	3.91	1.32E−05	4.32	1.41E−04	4.04
80	4.93E−07	4.57	5.89E−06	4.35	5.23E−07	4.65	6.40E−06	4.46
160	1.85E−08	4.73	2.14E−07	4.78	1.91E−08	4.78	2.23E−07	4.85
320	6.51E−10	4.83	7.00E−09	4.93	6.62E−10	4.85	7.14E−09	4.96

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

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 WENO5-LW4 scheme and the WENO5-RK4 scheme. The numerical results are shown in Table 2. We can see that both schemes achieve their designed order of accuracy with comparable errors for the same mesh. In fact, the WENO5-LW4 scheme has smaller errors than the WENO5-RK4 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 derivative of solution is still smooth. The errors and numerical orders of accuracy by the WENO5-LW4 scheme and the WENO5-RK4 scheme are shown in Table 3. We can also see that both schemes achieve their designed order of accuracy, and the WENO5-LW4 scheme has smaller errors than the WENO5-RK4 scheme for the same mesh.

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 WENO5-LW4 scheme and the

Table 4
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 WENO5-LW4 and WENO5-RK4 schemes with periodic boundary conditions

$N_x \times N_y$	WENO5-LW4				WENO5-RK4			
	L_1 error	Order	L_∞ error	Order	L_1 error	Order	L_∞ error	Order
20×20	1.39E−04		7.49E−04		2.58E−04		1.75E−03	
40×40	6.09E−06	4.51	5.57E−05	3.75	1.29E−05	4.32	1.56E−04	3.48
80×80	2.03E−07	4.91	2.28E−06	4.61	5.12E−07	4.66	6.59E−06	4.57
160×160	5.99E−09	5.08	7.45E−08	4.94	1.86E−08	4.78	2.24E−07	4.88
320×320	1.61E−10	5.22	1.55E−09	5.59	6.45E−10	4.85	7.14E−09	4.97

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

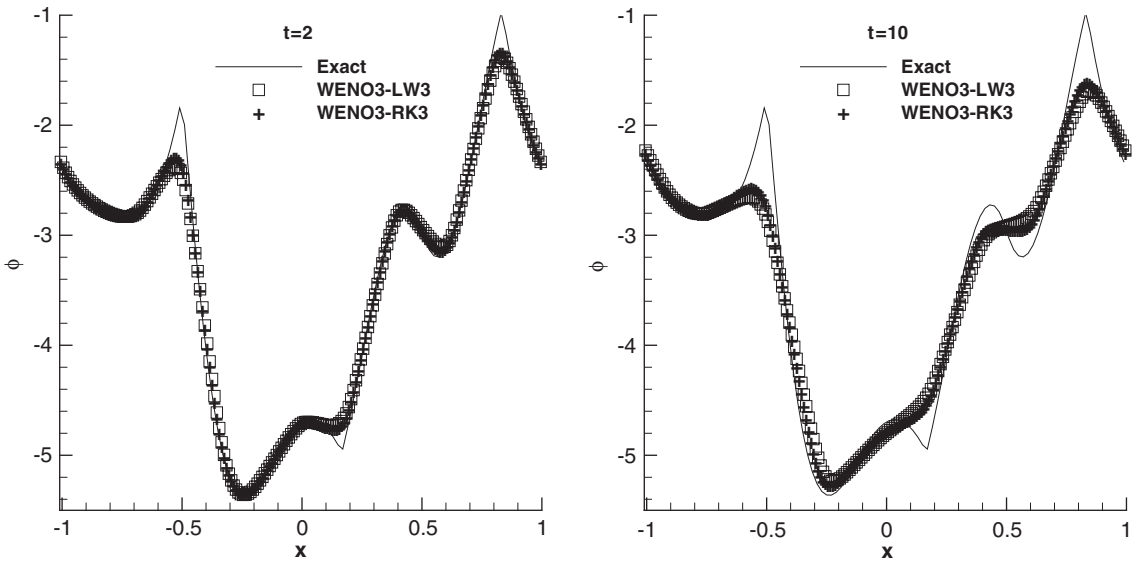

Fig. 1. One-dimensional linear equation. $N = 200$ cells. WENO3. Left: $t = 2$; right: $t = 10$. Solid lines: the exact solution; square symbols: the WENO-LW scheme; plus symbols: the WENO-RK scheme.

WENO5-RK4 scheme are shown in Table 4. We can also see that both schemes achieve their designed order of accuracy, and the WENO5-LW4 scheme has smaller errors than the WENO5-RK4 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

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

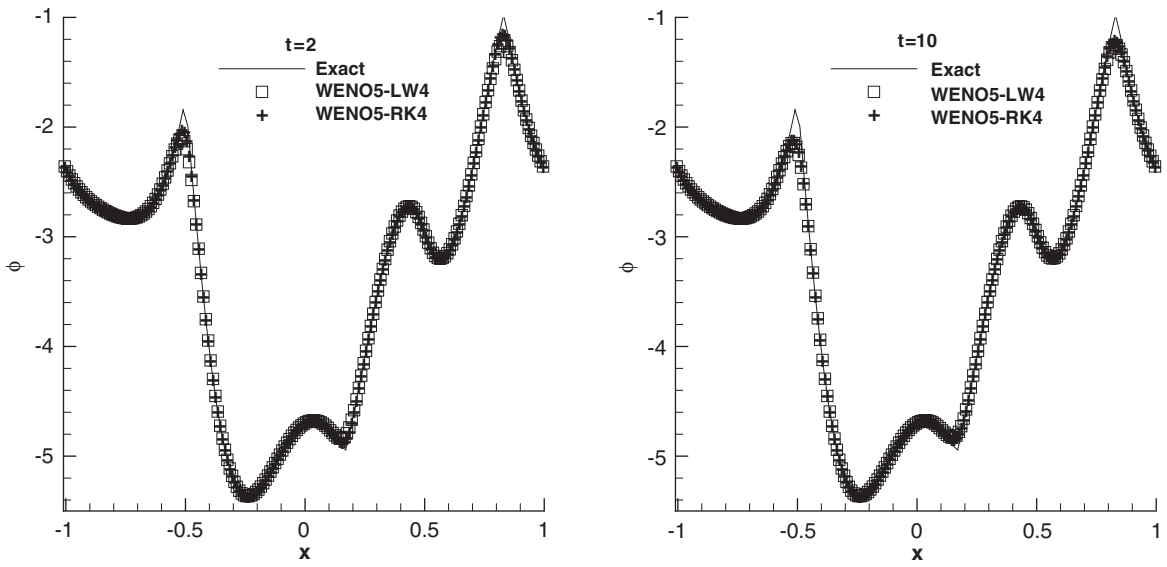

Fig. 2. One-dimensional linear equation. $N = 200$ cells. WENO5. Left: $t = 2$; right: $t = 10$. Solid lines: the exact solution; square symbols: the WENO-LW scheme; plus symbols: the WENO-RK scheme.

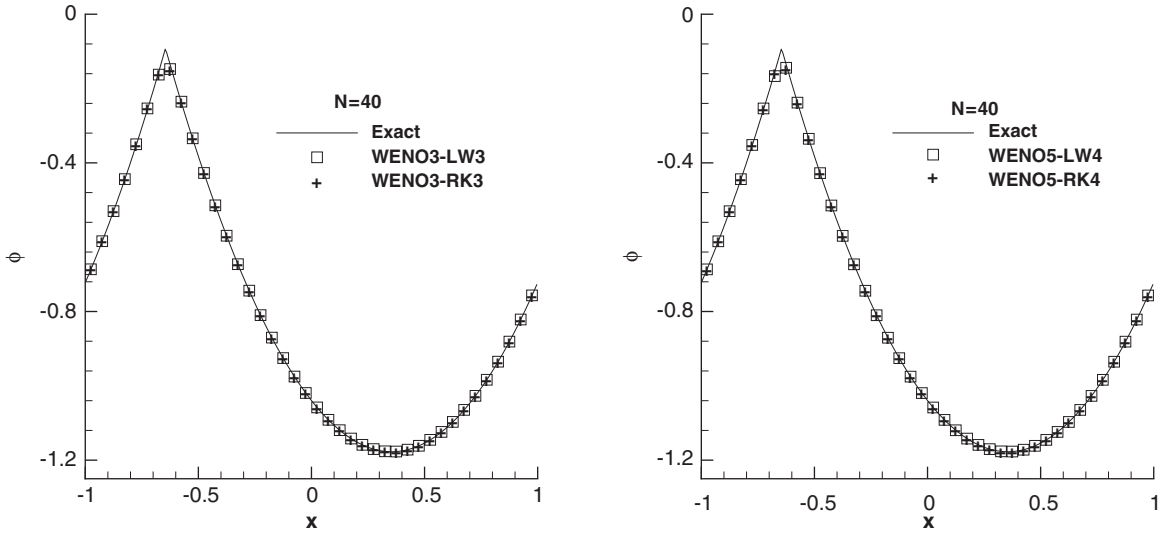

Fig. 3. Burgers' equation. $t = 3.5/\pi^2$. $N = 40$. Left: WENO3; right: WENO5. Solid lines: the exact solution; square symbols: the WENO-LW scheme; plus symbols: the WENO-RK scheme.

We plot the results at $t = 2.0$ and 10.0 in Fig. 1 for WENO3 and Fig. 2 for WENO5, respectively. We can observe that the results by both the WENO-LW schemes and WENO-RK schemes have a good resolution.

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. 3, the solutions of the WENO-LW scheme and the WENO-RK scheme with $N = 40$ cells are shown. We can see that both schemes give good results for this problem.

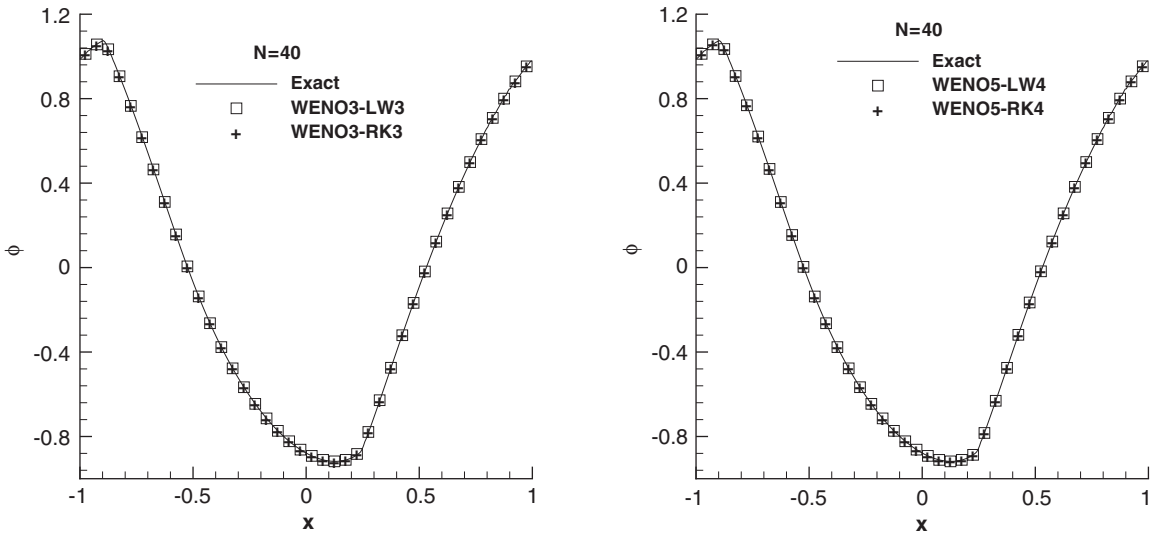

Fig. 4. Problem with the non-convex flux $H(u) = -\cos(u+1)$. $t = 1.5/\pi^2$. $N = 40$. Left: WENO3; right: WENO5. Solid lines: the exact solution; square symbols: the WENO-LW scheme; plus symbols: the WENO-RK scheme.

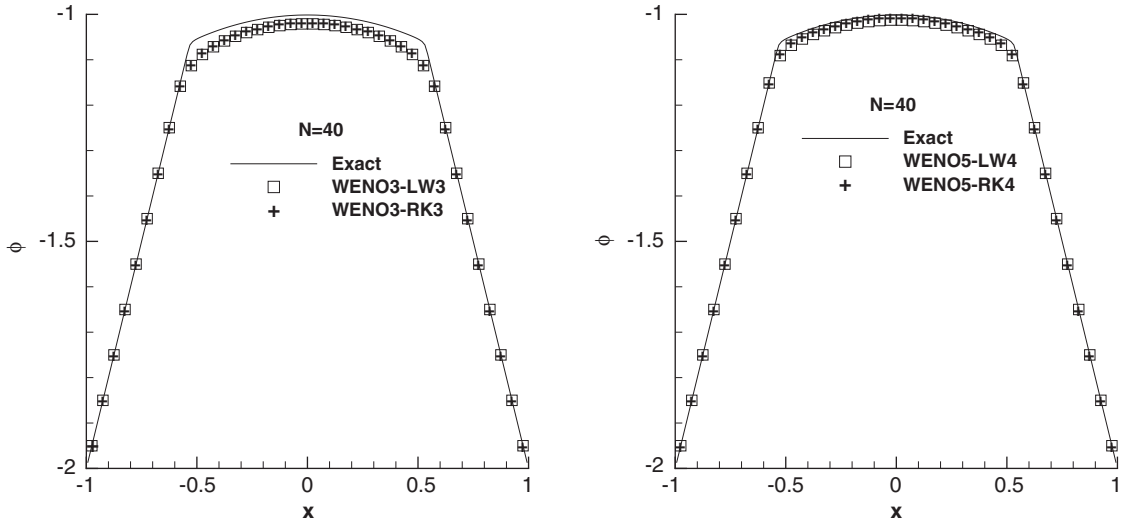

Fig. 5. Problem with the non-convex flux $H(u) = \frac{1}{4}(u^2-1)(u^2-4)$. $t = 1$. $N = 40$. Left: WENO3; right: WENO5. Solid lines: the exact solution; square symbols: the WENO-LW scheme; plus symbols: the WENO-RK scheme.

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. 4, the solutions of the WENO-LW scheme and the WENO-RK scheme with $N = 40$ cells are shown. We also can see that both schemes give good results for this problem.

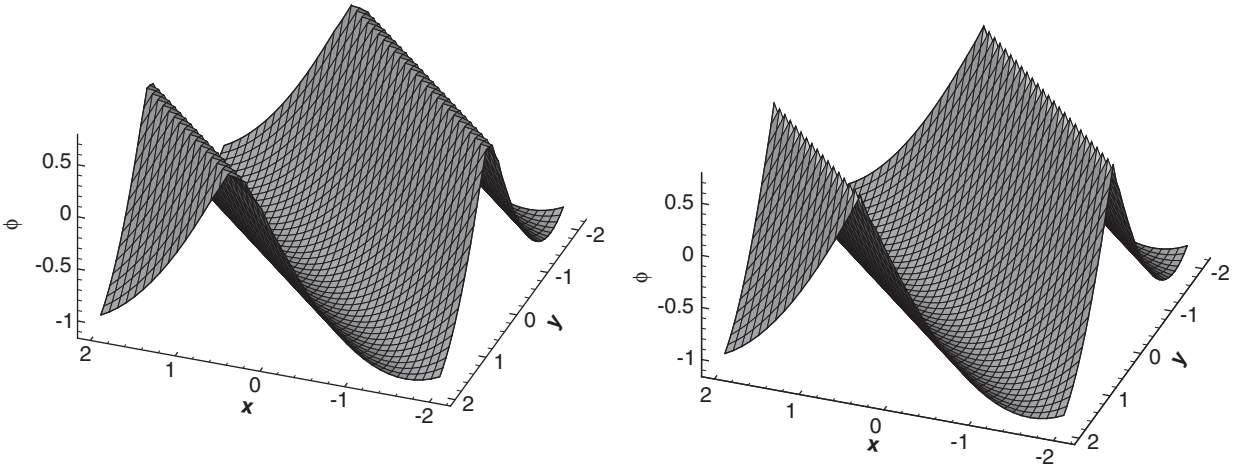

Fig. 6. Two-dimensional Burgers' equation. $t = 1.5/\pi^2$ by the WENO-LW scheme with $N_x \times N_y = 40 \times 40$ cells. Left: WENO3-LW3; right: WENO5-LW4.

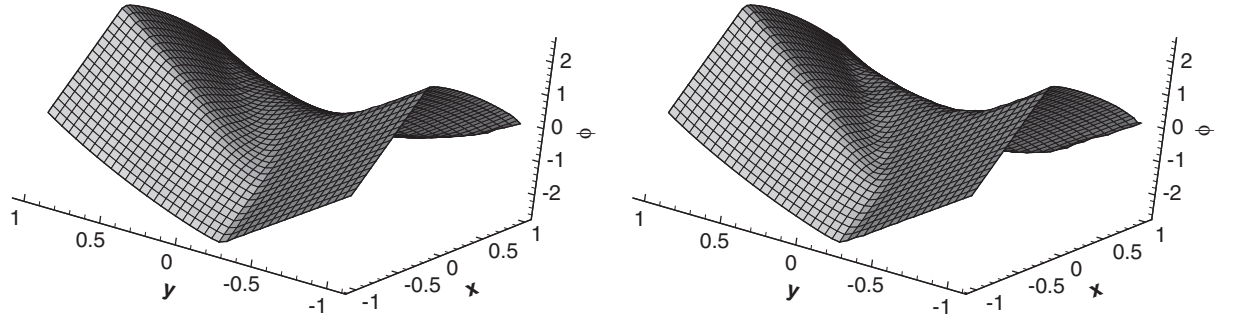

Fig. 7. Two-dimensional Riemann problem with a non-convex flux $H(u, v) = \sin(u + v)$. $t = 1$ by the WENO-LW method with $N_x \times N_y = 40 \times 40$ cells. Left: WENO3-LW3; right: WENO5-LW4.

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

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

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 WENO-LW scheme and the WENO-RK scheme with $N = 40$ cells in Fig. 5. We can also see that both schemes give good results for this problem again.

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 WENO-LW scheme with $N_x \times N_y = 40 \times 40$ cells are shown in Fig. 6. We observe good resolution for this example.

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

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

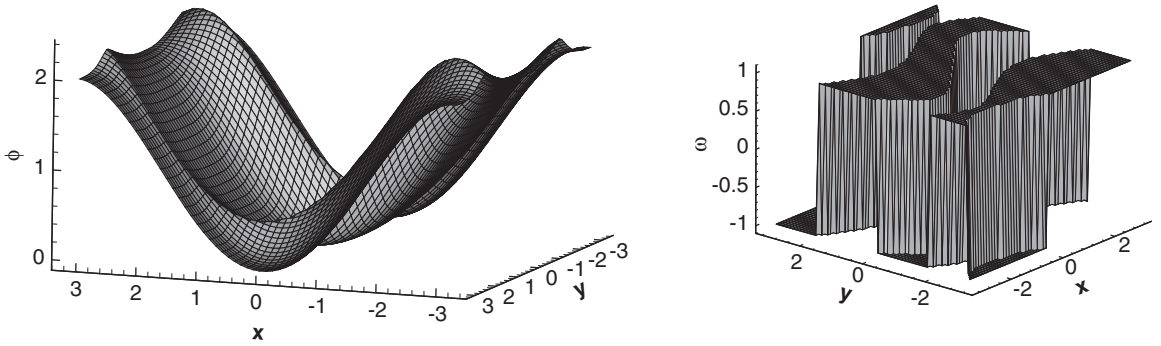

Fig. 8. The optimal control problem. $t = 1$ by the WENO3-LW3 scheme with $N_x \times N_y = 60 \times 60$ cells. Surfaces of the solution (left) and of the optimal control $\omega = \text{sign}(\phi_y)$ (right).

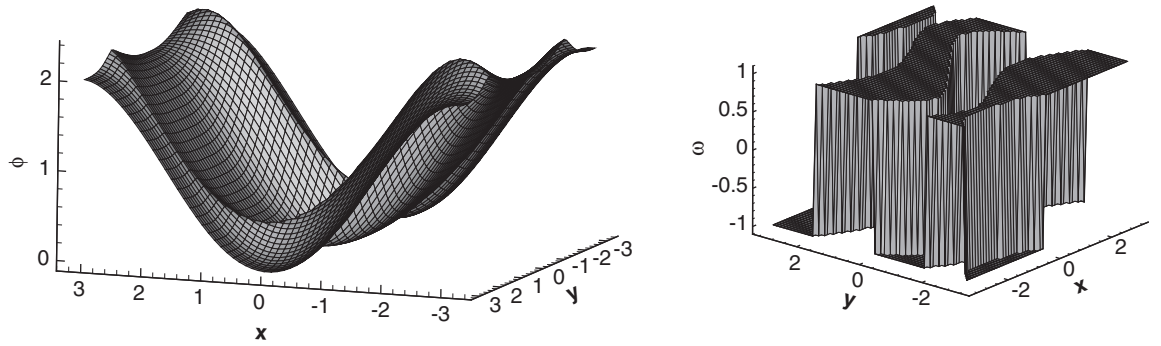

Fig. 9. The optimal control problem. $t = 1$ by the WENO5-LW4 scheme with $N_x \times N_y = 60 \times 60$ cells. Surfaces of the solution (left) and of the optimal control $\omega = \text{sign}(\phi_y)$ (right).

The solution of the WENO-LW scheme with $N_x \times N_y = 40 \times 40$ cells are shown in Fig. 7. We observe good resolution for this example.

Example 3.10. A problem from optimal control:

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

with periodic conditions, see [19]. The solution of the WENO-LW scheme with $N_x \times N_y = 60 \times 60$ cells and the optimal control $\omega = \text{sign}(\phi_y)$ are shown in Figs. 8 and 9.

Example 3.11. A two-dimensional eikonal equation with a non-convex Hamiltonian, which arises in geometric optics [12], is given by

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

The solutions of the WENO-LW scheme with $N_x \times N_y = 80 \times 80$ cells is shown in Fig. 10. Good resolution is observed.

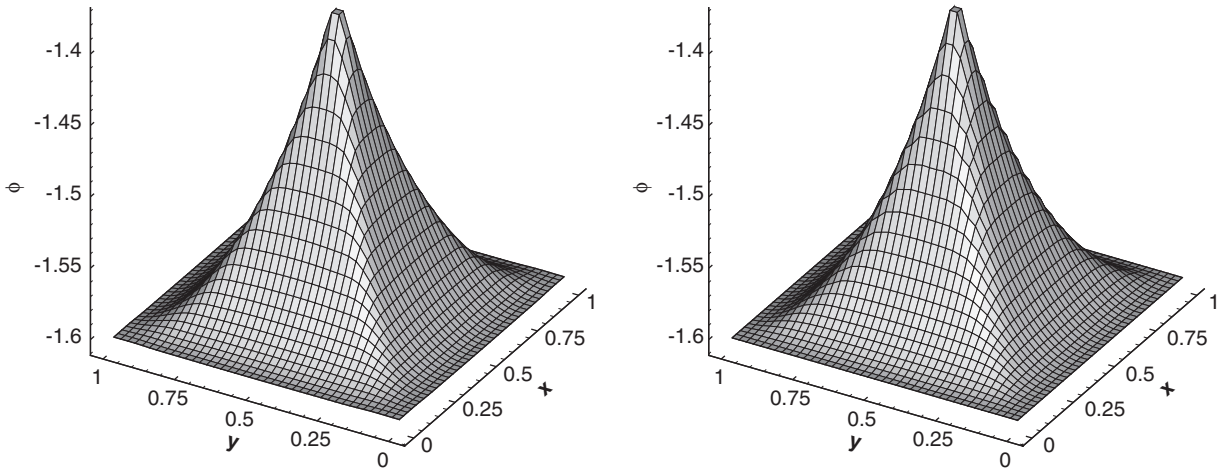

Fig. 10. Eikonal equation with a non-convex Hamiltonian. $t = 1$ by the WENO-LW schemes with $N_x \times N_y = 80 \times 80$ cells. Left: WENO3-LW3; right: WENO5-LW4.

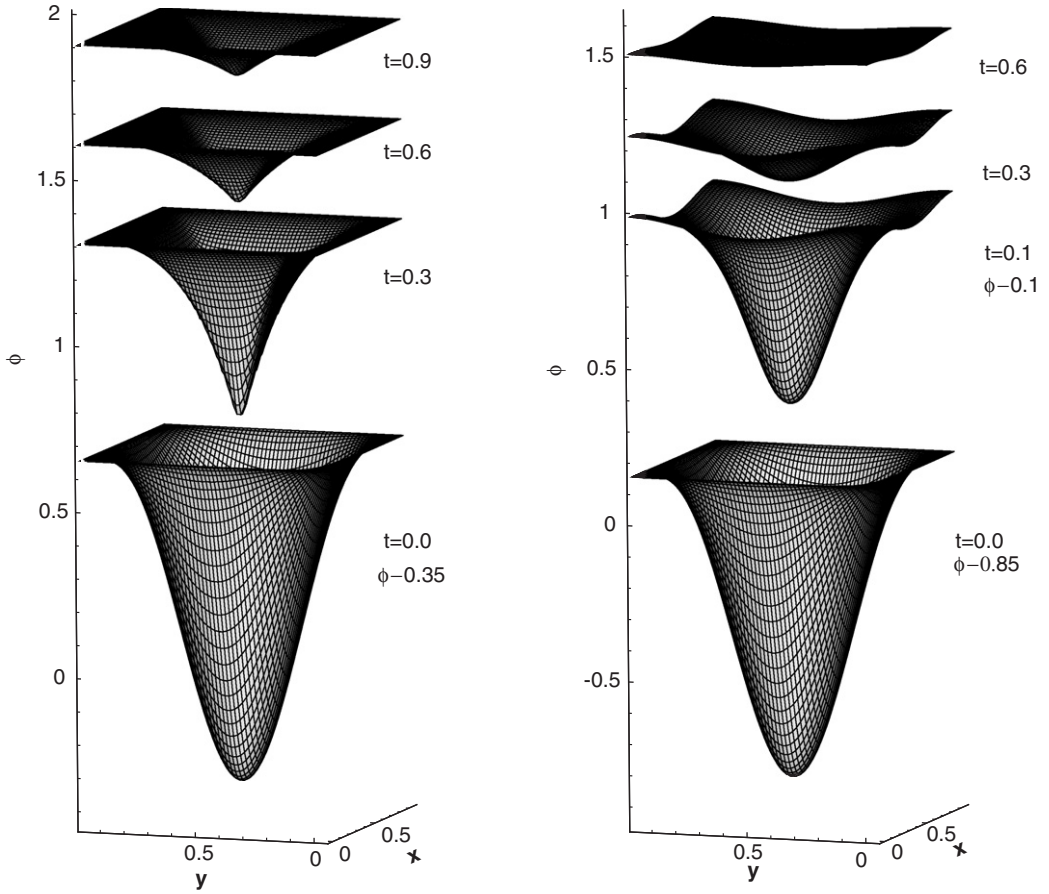

Fig. 11. Propagating surface. WENO3-LW3 method with $N_x \times N_y = 60 \times 60$ cells. Left: $\varepsilon = 0$; right: $\varepsilon = 0.1$.

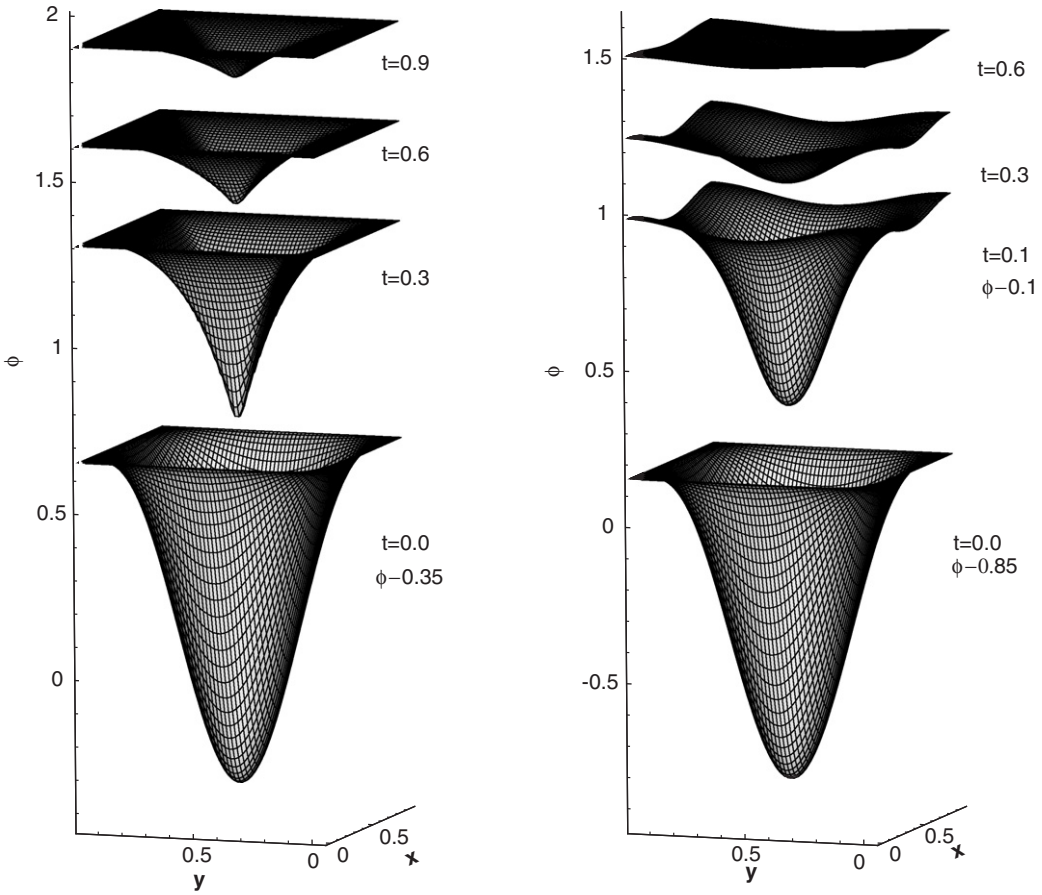

Fig. 12. Propagating surface. WENO5-LW4 method with $N_x \times N_y = 60 \times 60$ cells. Left: $\varepsilon = 0$; right: $\varepsilon = 0.1$.

Example 3.12. The problem of a propagating surface [18]:

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

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 WENO-LW method with $N_x \times N_y = 60 \times 60$ cells are presented in Figs. 11 and 12. 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, a class of weighted essentially non-oscillatory (WENO) schemes with a Lax–Wendroff time discretization procedure, termed WENO-LW schemes, for solving one- and two-dimensional Hamilton–Jacobi equations has

been constructed. This is an alternative method for time discretization to the popular TVD Runge–Kutta time discretizations. We explore the possibility in avoiding the nonlinear weights for part of the procedure, hence reducing the cost but still maintaining non-oscillatory properties for problems with strong discontinuous derivative. As a result, comparing with the original WENO with Runge–Kutta time discretizations schemes (WENO-RK) of Jiang and Peng [10] for Hamilton–Jacobi equations, the major advantages of WENO-LW schemes are more cost effective for certain problems and their compactness in the reconstruction. The accuracy by WENO-LW is slightly better than or is comparable with that given by WENO-RK at same meshes for all cases we have computed.

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] 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.
[3] S. Bryson, D. Levy, High-order semi-discrete central-upwind schemes for multidimensional 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. Cecil, J. Qian, S. Osher, Numerical methods for high dimensional Hamilton–Jacobi equations using radial basis functions, J. Comput. Phys. 196 (2004) 327–347.
[6] M. Dumbser, ADER discontinuous Galerkin schemes for aeroacoustics, Proceedings of the Euromech Colloquium, vol. 449, Chamonix, France, 2003.
[7] M. Dumbser, C.-D. Munz, Arbitrary high order discontinuous Galerkin schemes, in: S. Cordier, T. Goudon, E. Sonnendrcker (Eds.), Numerical Methods for Hyperbolic and Kinetic Problems, EMS Publishing House, 2005, pp. 295–333.
[8] A. Harten, B. Engquist, S. Osher, S. Chakravathy, Uniformly high order accurate essentially non-oscillatory schemes, III, J. Comput. Phys. 71 (1987) 231–303.
[9] C. Hu, C.-W. Shu, A discontinuous Galerkin finite element method for Hamilton–Jacobi equations, SIAM J. Sci. Comput. 21 (1999) 666–690.
[10] G. Jiang, D. Peng, Weighted ENO schemes for Hamilton–Jacobi equations, SIAM J. Sci. Comput. 21 (2000) 2126–2143.
[11] G. Jiang, C.-W. Shu, Efficient implementation of weighted ENO schemes, J. Comput. Phys. 126 (1996) 202–228.
[12] 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.
[13] A. Kurganov, E. Tadmor, New high-resolution semi-discrete central schemes for Hamilton–Jacobi equations, J. Comput. Phys. 160 (2000) 720–742.
[14] F. Lafon, S. Osher, High order two dimensional nonoscillatory methods for solving Hamilton–Jacobi scalar equations, J. Comput. Phys. 123 (1996) 235–253.
[15] P.D. Lax, B. Wendroff, Systems of conservation laws, Comm. Pure Appl. Math. 13 (1960) 217–237.
[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] X. Liu, S. Osher, T. Chan, Weighted essentially non-oscillatory schemes, J. Comput. Phys. 115 (1994) 200–212.
[18] S. Osher, J. Sethian, Fronts propagating with curvature dependent speed: algorithms based on Hamilton–Jacobi formulations, J. Comput. Phys. 79 (1988) 12–49.
[19] S. Osher, C.-W. Shu, High-order essentially nonoscillatory schemes for Hamilton–Jacobi equations, SIAM J. Numer. Anal. 28 (1991) 907–922.
[20] J. Qiu, M. Dumbser, C.-W. Shu, The discontinuous Galerkin method with Lax–Wendroff type time discretizations, Comput. Methods Appl. Mech. Eng. 194 (2005) 4528–4543.
[21] J. Qiu, C.-W. Shu, Finite difference WENO schemes with Lax–Wendroff-type time discretizations, SIAM J. Sci. Comput. 24 (2003) 2185–2198.
[22] J. Qiu, C.-W. Shu, Hermite WENO schemes for Hamilton–Jacobi equations, J. Comput. Phys. 204 (2005) 82–99.
[23] T. Schwartzkopff, C.D. Munz, E.F. Toro, ADER: a high-order approach for linear hyperbolic systems in 2D, J. Sci. Comput. 17 (2002) 231–240.
[24] C.-W. Shu, Total-variation-diminishing time discretizations, SIAM J. Sci. Statist. Comput. 9 (1988) 1073–1084.
[25] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, J. Comput. Phys. 77 (1988) 439–471.
[26] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock capturing schemes II, J. Comput. Phys. 83 (1989) 32–78.
[27] V.A. Titarev, E.F. Toro, ADER: arbitrary high order Godunov approach, J. Sci. Comput. 17 (2002) 609–618.
[28] V.A. Titarev, E.F. Toro, ADER schemes for three-dimensional nonlinear hyperbolic systems, J. Comput. Phys. 204 (2005) 715–736.
[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.