Journal of Applied Mathematics and Physics, 2017, 5, 59-82
http://www.scirp.org/journal/jamp
ISSN Online: 2327-4379
ISSN Print: 2327-4352

Conservative and Easily Implemented Finite Volume Semi-Lagrangian WENO Methods for 1D and 2D Hyperbolic Conservation Laws

Fuxing Hu

Department of Mathematics, Huizhou University, Huizhou, China
Email: fuxing_hoo@163.com

How to cite this paper: Hu, F.X. (2017) Conservative and Easily Implemented Finite Volume Semi-Lagrangian WENO Methods for 1D and 2D Hyperbolic Conservation Laws. *Journal of Applied Mathematics and Physics*, **5**, 59-82.
http://dx.doi.org/10.4236/jamp.2017.51008

Received: December 12, 2016
Accepted: January 16, 2017
Published: January 19, 2017

Abstract

The paper is devised to propose finite volume semi-Lagrange scheme for approximating linear and nonlinear hyperbolic conservation laws. Based on the idea of semi-Lagrangian scheme, we transform the integration of flux in time into the integration in space. Compared with the traditional semi-Lagrange scheme, the scheme devised here tries to directly evaluate the average fluxes along cell edges. It is this difference that makes the scheme in this paper simple to implement and easily extend to nonlinear cases. The procedure of evaluation of the average fluxes only depends on the high-order spatial interpolation. Hence the scheme can be implemented as long as the spatial interpolation is available, and no additional temporal discretization is needed. In this paper, the high-order spatial discretization is chosen to be the classical 5th-order weighted essentially non-oscillatory spatial interpolation. In the end, 1D and 2D numerical results show that this method is rather robust. In addition, to exhibit the numerical resolution and efficiency of the proposed scheme, the numerical solutions of the classical 5th-order WENO scheme combined with the 3rd-order Runge-Kutta temporal discretization (WENOJS) are chosen as the reference. We find that the scheme proposed in the paper generates comparable solutions with that of WENOJS, but with less CPU time.

Keywords

Semi-Lagrangian Method, Average Flux, WENO Scheme, High-Order Scheme, Hyperbolic Conservation Laws

1. Introduction

The characteristic of hyperbolic conservation laws is that they can generate discontinuous solutions even if the initial condition is smooth. It is this property

that leads to the challenge in developing the high-order numerical methods. The main difficulty of designing high-order methods is how to maintain the high-order accuracy around the smooth regions and meanwhile suppress the spurious oscillations around the regions with large gradients. In the past few decades, many high-order numerical methods were developed to evolve the hyperbolic problem, e.g. the essentially non-oscillatory schemes (ENO) [1] [2] [3] [4], the weighted essentially non-oscillatory schemes (WENO) [5] [6] and the discontinuous Galerkin finite element method [7] [8], and so on.

The discretization procedure of the classical WENO schemes [5] can be divided into two stages. First, through spatial discretization implemented by WENO reconstruction, leaving the equations continuous in time; this leads to a system of ordinary differential equations in time. Then, any numerical methods for ordinary differential equations, e.g. Runge-Kutta methods, can be used to evolve this system. In order to make the solution stable in total variation norm, Shu and Osher [3] and Shu [9] devised a set of TVD Runge-Kutta methods as ODE solvers. Taking three-stage TVD Runge-Kutta methods for example, three WENO reconstruction processes are needed to evolve one time step. And they also suffer from a CFL time step stability restriction.

To fix the problem above, the semi-Lagrangian methods [10] [11] [12] [13] avoid the time discretization by characteristic tracing technique and have more alleviative restriction for CFL number. Recently, in the papers [14] [15] [16] [17], the authors combined the idea of semi-Lagrangian with WENO reconstruction for solving advection problems. The semi-Lagrangian methods above depend on the characteristic tracing. However, for generally nonlinear cases, it is impossible to find the trace points exactly (even finding the trace points with high accuracy is very hard). It is this restriction that makes the semi-Lagrangian methods very hard to apply to nonlinear problems. The authors Huang and Arbogast in [18] devised a semi-Lagrangian method for nonlinear conservation laws. However, a costly flux correction is needed for keeping high accuracy and numerical stability.

In this paper, we try to directly evaluate the average flux

$$F_{i+1/2}^{n} = \frac{1}{\Delta t}\int_{t_n}^{t_{n+1}} f\left(u\left(x_{i+1/2}, t\right)\right)\mathrm{d}t, \tag{1}$$

where t_n is the time step and $x_{i+1/2}$ is the mesh boundary. The main idea of evaluation of the average fluxes is the transformation of the integration of flux function in time into the integration in space. A similar transformation had appeared in [16]. The computation for integration in space only depends on the spatial interpolation polynomial. Hence, the scheme here can be implemented easily as long as the procedure of spatial interpolation is available. Several high-order polynomial interpolations can be chosen, e.g. the PPM [19], ENO [1] [2] or WENO [5] [6], etc. In this paper, we choose the classical 5th-order WENO interpolation to reconstruct the polynomial in space and we call the scheme here as WENOEL. It is noted that the scheme is obviously conservative since we directly evaluate the average flux at each cell edge. As the semi-Lagrangian methods presented in literature, we also need the tracking back points along cha-

racteristic lines to evaluate the average flux. Since the characteristic points can be tracked back exactly (or with high-order accuracy) only for some simple linear equations, we do not try to find the characteristic points with high-order accuracy. For linear problems, the scheme can achieve optimal accuracy because the average flux can be evaluated with high order of accuracy. For nonlinear problems, in testing the order of accuracy for smooth solution, no optimal accuracy can be guaranteed. For testing the resolution in nonlinear problems of the scheme, we choose the classical WENOJS (the 5th-order WENO reconstruction with 3rd-order Runge-Kutta time discretization) as benchmark and compare them for several 1D and 2D examples. We find that the scheme proposed in the paper generates comparable solutions with that of WENOJS but needs less than half computing time. The free CFL restriction is another advantage of semi-Lagrangian methods. For advection problems, the CFL number can be chosen generally larger than 1. However, for nonlinear hyperbolic conservation laws, the $\mathrm{CFL}>1$ will lead to numerical instability generally if there exist shocks. In this paper, the $\mathrm{CFL}=0.6$ is chosen for the WENOEL scheme in all the numerical examples.

In the paper that follows, we will present the evaluation procedure of the average flux in Section 2 in detail. In Section 3, to show the high resolution and efficiency of our scheme, we compare the schemes WENOEL proposed here and WENOJS for several 1D and 2D examples. And from the comparisons of resolution and efficiency for these 1D and 2D examples, we can find that the proposed method in this paper can present comparable results with the classical WENOJS scheme, but with less CPU time.

2. The Finite Volume WENOEL Scheme

Consider the conservation laws

$$u_t(x,t)+f_x(u(x,t))=0,\quad a\le x\le b,\quad t\ge 0 \tag{2}$$

subject to initial condition

$$u(x,0)=u_0(x)$$

with proper boundary conditions. Integrating the Equation (2) on control volume $\left[x_{i-1/2},x_{i+1/2}\right]\times\left[t_n,t_{n+1}\right]$ gives

$$\begin{aligned}\int_{x_{i-1/2}}^{x_{i+1/2}}u(x,t_n)\,\mathrm{d}x=&\int_{x_{i-1/2}}^{x_{i+1/2}}u(x,t_n)\mathrm{d}x\\&+\int_{t_n}^{t_{n+1}}f\left(u\left(x_{i-1/2},t\right)\right)\mathrm{d}t-\int_{t_n}^{t_{n+1}}f\left(u\left(x_{i+1/2},t\right)\right)\mathrm{d}t.\end{aligned}$$

Rearranging this equation and dividing by Δx yields

$$\begin{aligned}\frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_{i+1/2}}u(x,t_{n+1})\,\mathrm{d}x=&\frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_{i+1/2}}u(x,t_n)\,\mathrm{d}x\\&-\frac{1}{\Delta x}\left(\int_{t_n}^{t_{n+1}}f\left(u\left(x_{i+1/2},t\right)\right)\mathrm{d}t-\int_{t_n}^{t_{n+1}}f\left(u\left(x_{i-1/2},t\right)\right)\mathrm{d}t\right)\end{aligned} \tag{3}$$

Denote the i th cell average by

$$\bar{u}_i^n = \frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_{i+1/2}} u(x,t_n)\,\mathrm{d}x,$$

and average flux at cell edge $x_{i+1/2}$ by

$$f_{i+1/2}^n = \frac{1}{\Delta t}\int_{t_n}^{t_{n+1}} f\left(u\left(x_{i+1/2},t\right)\right)\mathrm{d}t, \tag{4}$$

then (3) can be written as

$$\bar{u}_i^{n+1} = \bar{u}_i^n - \frac{\Delta t}{\Delta x}\left(f_{i+1/2}^n - f_{i-1/2}^n\right). \tag{5}$$

The value U_i^n will approximate the average value $\bar{u}_i^j$. Given the approximated cell average U_i^n, the formula (5) tells us that U_i^{n+1} at next time step can be updated if the time integrals on the right of (4) can be evaluated efficiently. Denoting the approximated flux function $f_{i+1/2}^n$ by $F_{i+1/2}^n$

$$F_{i+1/2}^n \approx \frac{1}{\Delta t}\int_{t_n}^{t_{n+1}} f\left(u\left(x_{i+1/2},t\right)\right)\mathrm{d}t,$$

then we obtain

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{\Delta x}\left(F_{i+1/2}^n - F_{i-1/2}^n\right). \tag{6}$$

2.1. The Linear Equation

Here, we mainly consider the 5th-order finite volume version because of its popularity and other cases can be obtained similarly. Consider the linear advection equation

$$u_t(x,t) + cu_x(x,t) = 0, \quad a \le x \le b, \quad t \ge 0. \tag{7}$$

In this case, the flux function $f\left(u(x,t)\right) = cu(x,t)$. For this equation, the initial condition $u_0(x)$ simply propagates right if $c > 0$ (or left if $c < 0$) with unchanged shape.

For evaluating the average flux $f_{i+1/2}^n$ in (4), we can firstly apply a 5th-order reconstruction based on piecewise constant average fluxes $\{f_k\}_{k=1}^N$ to obtain interpolation polynomial $P_i(x)$ on cell I_i

$$f\left(u(x,t_n)\right) = P_i(x) + \mathcal{O}\left(\Delta x^5\right). \tag{8}$$

The 5th-order polynomial $P_i(x)$ is reconstructed over the stencil

$$S_i = \{I_{i-2}, I_{i-1}, I_i, I_{i+1}, I_{i+2}\}.$$

Here, we assume the advection velocity $c > 0$, then $u(x,t_n)$ simply spreads to the right with velocity c, which gives

$$f\left(u\left(x_{i+1/2},t\right)\right) = f\left(u\left(x_{i+1/2} - c(t-t_n), t_n\right)\right) \tag{9}$$

for any time $t \in [t_n, t_{n+1}]$. Combining the Formulas (8) and (9), the flow rate $f\left(u(x,t)\right)$ at cell edge $x_{i+1/2}$ is

$$f\left(u\left(x_{i+1/2},t\right)\right) = P_i\left(x_{i+1/2} - c(t-t_n)\right) + \mathcal{O}\left(\Delta x^5\right). \tag{10}$$

In this case, the average flux $f_{i+1/2}^n$ can be expressed approximately as

$$\begin{aligned} f_{i+1/2}^{n} &= \frac{1}{\Delta t}\int_{t_n}^{t_{n+1}} f\left(u\left(x_{i+1/2},t\right)\right)\mathrm{d}t \\ &= \frac{1}{\Delta t}\int_{t_n}^{t_{n+1}} P_i\left(x_{i+1/2}-c\left(t-t_n\right)\right)\mathrm{d}t+\mathcal{O}\left(\Delta x^5\right) \\ &= \frac{1}{c\Delta t}\int_{x_{i+1/2}-c\Delta t}^{x_{i+1/2}} P_i\left(x\right)\mathrm{d}x+\mathcal{O}\left(\Delta x^5\right). \end{aligned} \tag{11}$$

Omitting the high-order term $\mathcal{O}\left(\Delta x^5\right)$, we obtain the numerical flux

$$F_{i+1/2}^{n} = \frac{1}{c\Delta t}\int_{x_{i+1/2}-c\Delta t}^{x_{i+1/2}} P_i\left(x\right)\mathrm{d}x. \tag{12}$$

The last equality in (11) is obtained by integration of substitution $x = x_{i+1/2}-c\left(t-t_n\right)$. From the Equation (11), the integration in time $\left[t_n,t_{n+1}\right]$ is transformed into integration in space $\left[x_{i+1/2}-c\Delta t,x_{i+1/2}\right]$. Due to the integrand $P_i\left(x\right)$ is reconstruction polynomial, the last integration in (12) can be computed exactly.

Remark 1. *This transformation is the main idea of this paper. And it is similar to the equation* (3.9) *in* [16] *but with a slightly different way. The main difference is that the operation in* [16] *transforms the flux integration in time at cell center into the integration of* $u\left(x,t_n\right)$ *in space*

$$\int_{t_n}^{t_{n+1}} f\left(u\left(x_i,t\right)\right)\mathrm{d}t = \int_{x_i^*}^{x_i} u\left(x,t_n\right)\mathrm{d}x,$$

where x_i^* is the backward characteristic point of cell center x_i. And here we transform the flux integration in time at cell edge into the integration of interpolation polynomial of flux function in space. This difference makes the extension of the scheme to nonlinear cases much easier.

In the following of this subsection, we will present the 5th-order WENO reconstruction process to approximate the integral in (12)

$$\int_{x_{i+1/2}-c\Delta t}^{x_{i+1/2}} P_i\left(x\right)\mathrm{d}x.$$

The 5th-order WENO reconstruction procedure is represented as the convex combination of three 3rd-order reconstructions. First, we intend to reconstruct the 3rd-order conservative polynomials on cell $I_i = \left[x_{i-1/2},x_{i+1/2}\right]$ based on the piecewise constant average fluxes.

$$\left\{\overline{f}_k^{\,n}\right\},\quad k = i-2,\cdots,i+2.$$

It is noted that there are 3 three-point stencils contained the cell I_i can be used to reconstruct the polynomials. We denote these stencils respectively by

$$S_i^0 = \left\{I_i,I_{i+1},I_{i+2}\right\},$$

$$S_i^1 = \left\{I_{i-1},I_i,I_{i+1}\right\},$$

$$S_i^2 = \left\{I_{i-2},I_{i-1},I_i\right\}.$$

The corresponding polynomial of each stencil is

$$P_i^j\left(x\right) = b_0^j + b_1^j x + b_2^j x^2,\quad j = 0,1,2, \tag{13}$$

where the subscript i denotes the polynomial on cell I_i and superscript j

denotes the reconstruction based on stencil S_i^j. The coefficients in (13) are given by (for simplicity, we omit superscript in f_i^n)

$$b_0^j = \frac{11\overline{f}_{i-j} - 7\overline{f}_{i+1-j} + 2\overline{f}_{i+2-j}}{6} + \frac{\left(2\overline{f}_{i-j} - 3\overline{f}_{i+1-j} + \overline{f}_{i+2-j}\right)x_{i-1/2-j}}{\Delta x} + \frac{\left(\overline{f}_{i-j} - 2\overline{f}_{i+1-j} + \overline{f}_{i+2-j}\right)x_{i-1/2-j}^2}{2\Delta x^2},$$

$$b_1^j = -\frac{2\overline{f}_{i-j} - 3\overline{f}_{i+1-j} + \overline{f}_{i+2-j}}{\Delta x} - \frac{\left(\overline{f}_{i-j} - 2\overline{f}_{i+1-j} + \overline{f}_{i+2-j}\right)x_{i-1/2-j}}{\Delta x^2},$$

$$b_2^j = \frac{\overline{f}_{i-j} - 2\overline{f}_{i+1-j} + \overline{f}_{i+2-j}}{2\Delta x^2},$$

where $j = 0,1,2$ and $\Delta x = x_{i+1/2} - x_{i-1/2}$ is the length of cell I_i. So far, we have obtained the conservative interpolation polynomials $P_i^j(x)$ on each stencil S_i^j. In the end, the integral in (12) can be expressed as

$$\int_{x_{i+1/2}-c\Delta t}^{x_{i+1/2}} P_i(x)\,\mathrm{d}x = \sum_{j=0}^{2} d^j \int_{x_{i+1/2}-c\Delta t}^{x_{i+1/2}} P_i^j(x)\,\mathrm{d}x,$$

where the linear optimal weights d^j are

$$d^0 = \frac{3}{10} - \frac{1}{4}\lambda + \frac{1}{20}\lambda^2,$$

$$d^1 = \frac{6}{10} + \frac{1}{10}\lambda - \frac{1}{10}\lambda^2,$$

$$d^2 = \frac{1}{10} + \frac{3}{20}\lambda + \frac{1}{20}\lambda^2,$$

where $\lambda = \frac{c\Delta t}{\Delta x}$. For alleviating the effect of the non-smooth stencils, the nonlinear weights can be constructed as follows

$$w_i^j = \frac{a_i^j}{a_i^0 + a_i^1 + a_i^2} \tag{14}$$

and

$$a_i^j = \frac{d^j}{\left(\varepsilon + \beta_i^j\right)^2}. \tag{15}$$

The β_i^j is the indicator of smoothness based on the stencil S_i^j,

$$\beta_i^0 = \frac{13}{12}\left(\overline{f}_i - 2\overline{f}_{i+1} + \overline{f}_{i+2}\right)^2 + \frac{1}{4}\left(3\overline{f}_i - 4\overline{f}_{i+1} + \overline{f}_{i+2}\right)^2,$$

$$\beta_i^1 = \frac{13}{12}\left(\overline{f}_{i-1} - 2\overline{f}_i + \overline{f}_{i+1}\right)^2 + \frac{1}{4}\left(\overline{f}_{i-1} - \overline{f}_{i+1}\right)^2,$$

$$\beta_i^2 = \frac{13}{12}\left(\overline{f}_{i-2} - 2\overline{f}_{i-1} + \overline{f}_i\right)^2 + \frac{1}{4}\left(\overline{f}_{i-2} - 4\overline{f}_{i-1} + 3\overline{f}_i\right)^2.$$

Finally, the numerical flux should be expressed as

$$F_{i+1/2}^n = \frac{1}{c\Delta t}\sum_{j=0}^{2} w_i^j \int_{x_{i+1/2}-c\Delta t}^{x_{i+1/2}} P_i^j(x)\,\mathrm{d}x. \tag{16}$$

Substituting the Formulas (13) into (16) gives

$$
\begin{aligned}
F_{i+1/2}^{n} &= \frac{1}{c\Delta t}\sum_{j=0}^{2} w_i^j \int_{x_{i+1/2}-c\Delta t}^{x_{i+1/2}} P_i^j(x)\,\mathrm{d}x \\
&= \frac{1}{6}\Big[w_i^0\left(\lambda^2\left(\overline{f}_i - 2\overline{f}_{i+1} + \overline{f}_{i+2}\right) + \lambda\left(3\overline{f}_i - 3\overline{f}_{i+1}\right) + \left(2\overline{f}_i + 5\overline{f}_{i+1} - \overline{f}_{i+2}\right)\right) \\
&\quad + w_i^1\left(\lambda^2\left(\overline{f}_{i-1} - 2\overline{f}_i + \overline{f}_{i+1}\right) + \lambda\left(3\overline{f}_i - 3\overline{f}_{i+1}\right) + \left(-\overline{f}_{i-1} + 5\overline{f}_i + 2\overline{f}_{i+1}\right)\right) \\
&\quad + w_i^2\left(\lambda^2\left(\overline{f}_{i-2} - 2\overline{f}_{i-1} + \overline{f}_i\right) + \lambda\left(-3\overline{f}_{i-2} + 9\overline{f}_{i-1} - 6\overline{f}_i\right) + \left(2\overline{f}_{i-2} - 7\overline{f}_{i-1} + 11\overline{f}_i\right)\right)\Big].
\end{aligned} \tag{17}
$$

In contrast, when the advection velocity $c<0$, $u(x,t_n)$ spreads to the left with velocity c, which gives

$$
f\left(u\left(x_{i+1/2},t\right)\right) = f\left(u\left(x_{i+1/2} - c\left(t-t_n\right),t_n\right)\right) \tag{18}
$$

and the flow rate $f\left(u(x,t)\right)$ at cell edge $x_{i+1/2}$ is

$$
f\left(u\left(x_{i+1/2},t\right)\right) = P_{i+1}\left(x_{i+1/2} - c\left(t-t_n\right)\right) + \mathcal{O}\left(\Delta x^5\right). \tag{19}
$$

Similarly, the average flux $F_{i+1/2}^n$ can be expressed as

$$
F_{i+1/2}^{n} = \frac{1}{-c\Delta t}\sum_{j=0}^{2} w_{i+1}^j \int_{x_{i+1/2}}^{x_{i+1/2}-c\Delta t} P_{i+1}^j(x)\,\mathrm{d}x. \tag{20}
$$

The nonlinear weight w_{i+1}^j can be constructed similarly as (14) and (15). The linear optimal weights are

$$
d^0 = \frac{1}{10} - \frac{3}{20}\lambda + \frac{1}{20}\lambda^2,
$$

$$
d^1 = \frac{6}{10} - \frac{1}{10}\lambda - \frac{1}{10}\lambda^2,
$$

$$
d^2 = \frac{3}{10} + \frac{1}{4}\lambda + \frac{1}{20}\lambda^2.
$$

Substituting the formula (13) into (20) gives

$$
\begin{aligned}
F_{i+1/2}^{n} &= \frac{1}{6}\Big[w_{i+1}^0\left(\lambda^2\left(\overline{f}_{i+1} - 2\overline{f}_{i+2} + \overline{f}_{i+3}\right) + \lambda\left(6\overline{f}_{i+1} - 9\overline{f}_{i+2} + 3\overline{f}_{i+3}\right) + \left(11\overline{f}_{i+1} - 7\overline{f}_{i+2} + 2\overline{f}_{i+3}\right)\right) \\
&\quad + w_{i+1}^1\left(\lambda^2\left(\overline{f}_i - 2\overline{f}_{i+1} + \overline{f}_{i+2}\right) + \lambda\left(3\overline{f}_i - 3\overline{f}_{i+1}\right) + \left(2\overline{f}_i + 5\overline{f}_{i+1} - \overline{f}_{i+2}\right)\right) \\
&\quad + w_{i+1}^2\left(\lambda^2\left(\overline{f}_{i-1} - 2\overline{f}_i + \overline{f}_{i+1}\right) + \lambda\left(3\overline{f}_i - 3\overline{f}_{i+1}\right) + \left(-\overline{f}_{i-1} + 5\overline{f}_i + 2\overline{f}_{i+1}\right)\right)\Big].
\end{aligned} \tag{21}
$$

The scheme proposed in the paper is conservative for linear and nonlinear cases. The property of conservation is obvious since the scheme (6) is represented in conservation form.

Remark 2. *In* [16], *the authors Qiu and Shu presented a semi-Lagrangian finite difference WENO method for advection in incompressible flow. Actually, there are two reconstruction procedures needed to obtain the numerical flux at cell edge* $x_{i+1/2}$. *It is noted that, for linear equation with constant coefficient, the formulas of average flux* (17) *and* (21) *are totally same as ones in* [16]. *However, these two methods are devised by totally different procedure. They mainly have the following differences: the method present here is finite volume; our method can be easily extended to nonlinear cases by freezing locally and detailed description is given in Subsection* 2.3; *there are always exist linear optimal weights to construct the WENO schemes for our scheme which is not satisfied the schemes in* [16].

2.2. Algorithm

Here, we conclude the algorithm for computing numerical flux $F_{i+1/2}^n$ at $x = x_{i+1/2}$ during a time step $\Delta t = t_{n+1} - t_n$.

1) Distinguish the propagation direction of solution at $x = x_{i+1/2}$ used Rankine-Hugoniot jump condition

$$c_{i+1/2} = \frac{f(u_{i+1}) - f(u_i)}{u_{i+1} - u_i}.$$

2) The propagation velocity can be chosen to be $c = \frac{\partial f(u)}{\partial u}\Big|_{u=u_i}$ if $c_{i+1/2} > 0$, or $c = \frac{\partial f(u)}{\partial u}\Big|_{u=u_{i+1}}$ if $c_{i+1/2} < 0$.

3) Insert the propagation velocity into (17) or (21) to get numerical flux $F_{i+1/2}^n$.

2.3. The Nonlinear Equations

In this subsection, we will extent the method above to nonlinear case. Firstly, it is noted that for nonlinear case the method here cannot approach the optimal accuracy since the solution no longer simply translates uniformly. Instead, it deforms as it evolves and the characteristic curves are no longer parallel straight lines, which lead to the evaluation of average fluxes is hard to obtain. And generally the tracking back points cannot be found exactly (even cannot find the points with high accuracy). Hence for nonlinear case, rather than trying to find the tracking back points, we freeze the equation to linear formation locally and apply the procedure in Subsection 2.1 to it. For solving the nonlinear case, the propagation direction is firstly distinguished by Rankine-Hugoniot jump conditions and propagation velocity is chosen to be $c = \frac{\partial f(u)}{\partial u}\Big|_{u=u_i}$ $\left(\text{if propagation direction} > 0\right)$ or $c = \frac{\partial f(u)}{\partial u}\Big|_{u=u_{i+1}}$ $\left(\text{if propagation direction} < 0\right)$.

Remark 3. *The semi-Lagrangian finite volume method proposed here is largely dependent on the reconstruction from the cell flux average*

$$\overline{f}_i = \frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_{i+1/2}} f\left(u(x,t_n)\right)\mathrm{d}x.$$

For finite volume method, only the cell average

$$\overline{u}_i^n = \frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_{i+1/2}} u(x,t_n)\,\mathrm{d}x$$

is available. For linear problem, $f\left(u(x,t)\right) = au(x,t)$, *the cell flux average can be apparently chosen to be*

$$\overline{f}_i = a\overline{u}_i^n$$

However, for nonlinear problem, the cell flux average must be computed by some numerical quadrature. In this paper, we choose Gauss quadrature to evaluate the cell flux average

$$\overline{f}_i = \frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_{i+1/2}} f\left(u\left(x,t_n\right)\right)\mathrm{d}x = \frac{1}{\Delta x}\sum_{k=1}^{k=3} A_k^g f\left(u\left(x_k^g,t_n\right)\right),$$

where A_k^g *and* x_k^g *are the Gauss weights and Gauss points, respectively. For simple, the value of* $u\left(x,t_n\right)$ *at Gauss point* x_k^g *can be obtained by the* 5*th-order Lagrangian interpolation.*

In addition, for solving Euler equations, the Roe speed is used to identify the upwind direction which may lead to some numerical instabilities, especially for 2D cases. In Subsection 3.3, in computing the numerical tests: double Mach reflection and Mach 3 wind tunnel with a step, there indeed exist numerical instabilities. In addition, it is noted that this numerical instability not only appears in the WENOEL scheme, but also appears in the WENOJS scheme. It is said that the numerical instability is mainly due to the choice of Roe speed. The detailed introductions for the numerical instabilities of upwind schemes are referred to [20] [21] [22]. In order to eliminate such kind of instabilities, the H-correction procedure in [21] is taken to solve these flaws. For these two 2D problems, the numerical flux $F_{i+1/2,j}^n$ at the cell edge $\left(x_{i+1/2},y_j\right)$ is computed as follows. [1]

1) If $\min\left(\left|\lambda_{i,j}\right|,\left|\lambda_{i+1,j}\right|\right) \geq \eta_{i+1/2,j}$, then the numerical flux $F_{i+1/2,j}^n$ can be chosen to be (17) or (21) and the average fluxes f_k in these two equations are correspondingly substituted by $f_{k,j}\left(k=i-2,\cdots,i+3\right)$. The $\lambda_{i,j}$ is the eigenvalue and the value $\eta_{i+1/2,j}$ is determined by

$$\eta_{i+1/2,j} = \max\left(\eta_{i+1/2,j}^x,\eta_{i,j-1/2}^y,\eta_{i,j+1/2}^y,\eta_{i+1,j-1/2}^y,\eta_{i+1,j+1/2}^y\right),$$

where

$$\eta_{i+1/2,j}^x = \left|u_{i,j}-u_{i+1,j}\right| + \left|a_{i,j}-a_{i+1,j}\right|,$$

$$\eta_{i,j+1/2}^y = \left|v_{i,j}-v_{i,j+1}\right| + \left|a_{i,j}-a_{i,j+1}\right|,$$

and $a_{i,j}$ is the speed of sound.

2) Otherwise, a more dissipative Lax-Friedrich flux splitting method is used to split the flux $f_{k,j}$ into positive and negative fluxes

$$f_{k,j}^+ = \frac{1}{2}\left(f_{k,j}+\alpha u_{k,j}^n\right),$$

$$f_{k,j}^- = \frac{1}{2}\left(f_{k,j}-\alpha u_{k,j}^n\right).$$

For the positive and negative fluxes, the propagation velocities are chosen to be

$$c^+ = \frac{1}{2}\left(\lambda_{k,j}+\alpha\right),$$

$$c^- = \frac{1}{2}\left(\lambda_{k,j}-\alpha\right).$$

Inserting the $f_{k,j}^+$, c^+ and $f_{k,j}^-$, c^- into (17) and (21), respectively, we can get the positive and negative numerical fluxes $F_{i+1/2,j}^+$ and $F_{i+1/2,j}^-$. Finally, the numerical flux at cell edge $\left(x_{i+1/2},y_j\right)$ is

$$F_{i+1/2,j}^n = F_{i+1/2,j}^+ + F_{i+1/2,j}^-.$$

3. Numerical Results

In this section, we will apply this method to 1D and 2D hyperbolic problems. At present, the Strang dimensional splitting method [23] [24] is applied to 2D problems, which is used to split the 2D equation into two 1D equation. In each coordinate direction, the equation is evolved by the proposed 1D WENOEL method. The detailed comparisons of resolution and efficiency are shown between our method and WENOJS method (5th-order spatial and 3rd-order TVD RK temporal discretization). In the end, it is noted that, in the WENOJS method, the upwinding (not flux vector splitting) is used to construct the numerical flux. So the WENOJS method in this paper can also be called by WENOJS-Roe. The detailed description of WENOJS-Roe is referred to the procedure 2.4 on page 33 in [25].

For the choice of CFL number, It is found that when $0 < \mathrm{CFL} < 1$ the WENOEL scheme is numerically stable in all the examples we have tested. Especially, we had chose CFL = 0.9 to simulate all the examples appeared in this paper and no unstable phenomenon had been found. For comparing the resolution and computing time between the WENOEL and WENOJS schemes, we choose CFL = 0.6 for these two schemes for all the tests contained the discontinuities. This choice of CFL number for the WENOJS scheme is also adopted in many classical literatures, e.g. [25]. For the problem (22) (23), which is used to test the order of accuracy of schemes, we choose $\Delta t = \Delta x^{5/3}$ for WENOJS in order that spatial error is dominant but still let CFL = 0.6 for WENOEL.

3.1. 1D Scalar Examples

In this subsection, we will consider linear advection, Burger's and Buckley-Leverett equations. For the linear advection equation, two initial conditions are used to test the schemes. We use a smooth initial condition to test the order of accuracy of the scheme and a condition contained discontinuities and high-frequency waves to test the performance of the scheme in simulating discontinuous and large-gradient solution. For the Burger equation, we also consider two initial conditions which commonly used to validate the proposed scheme. For the nonconvex Buckley-Leverett equation, we study a simple model for two-phase fluid flow in a porous medium. Except for the comparison of resolution between the WENOJS and WENOEL schemes, the computing time is also shown in this subsection.

3.1.1. Linear Advection Equation

Consider the linear advection equation

$$u_t(x,t) + u_x(x,t) = 0, \quad -1 \le x \le 1, \quad t \ge 0, \tag{22}$$

subject to two initial conditions and periodic boundary conditions. One initial condition is

$$u(x,0) = \sin \pi x, \tag{23}$$

and the other is

$$u(x,0)=\begin{cases}\left(G(x,\beta,z-\delta)+G(x,\beta,z+\delta)+4G(x,\beta,z)\right)/6 & -0.8\le x\le -0.6,\\ 1 & -0.4\le x\le -0.2,\\ 1-|10x-1| & 0.0\le x\le 0.2,\\ \left(F(x,\alpha,a-\delta)+F(x,\alpha,a+\delta)+4F(x,\alpha,a)\right)/6 & 0.4\le x\le 0.6,\\ 0 & \text{otherwise,}\end{cases} \tag{24}$$

where $G(x,\beta,z)=\exp\left(-\beta(x-z)^2\right)$, $F(x,\alpha,a)=\sqrt{\max\left(1-(\alpha(x-a))^2,0\right)}$, and the parameters are given by

$$a=0.5,\ z=-0.7,\ \delta=0.005,\ \alpha=10,\ \delta=\frac{\ln 2}{36\delta^2}.$$

The problem (22) (23) here is used to test the accuracy of our method in linear case. The Table 1 and Table 2 present the comparison of errors and accuracy between the WENOEL and WENOJS methods. The output time is chosen to be $t=2$ and the time steps are chosen to be $\Delta t=0.6\Delta x$ and $\Delta t=(\Delta x)^{5/3}$ for the WENOEL and WENOJS, respectively. As the classical WENOJS method, the WENOEL also arrives 5th-order of accuracy. For the l_1 and l_∞ errors, the WENOEL method shows much better results than the WENOJS method. This

Table 1. The test of order of accuracy for the l_1 and l_∞ errors for initial value problem (22) (23) with WENOEL method. The time step is chosen to be $\Delta t=0.6\Delta x$ and final time is $t=2$.

WENOEL				
N	l_1	r	l_∞	r
20	1.36471e−03	-	1.26904e−03	
40	4.19678e−05	5.0232	4.18532e−05	4.9223
80	1.30400e−06	5.0083	1.32972e−06	4.9761
160	4.06743e−08	5.0027	4.08887e−08	5.0233
320	1.26992e−09	5.0013	1.24379e−09	5.0389
640	3.96726e−11	5.0005	3.81086e−11	5.0285

Table 2. The test of order of accuracy for the l_1 and l_∞ errors for initial value problem (22) (23) with WENOJS method. The time step is chosen to be $\Delta t=(\Delta x)^{5/3}$ and final time is $t=2$.

WENOJS				
N	l_1	r	l_∞	r
20	2.96529e−03	-	2.57899e−03	
40	9.27609e−05	4.9985	9.05453e−05	4.8320
80	2.89265e−06	5.0031	2.90709e−06	4.9610
160	9.03392e−08	5.0009	8.85753e−08	5.0365
320	2.82330e−09	4.9999	2.72457e−09	5.0228
640	9.04248e−11	4.9645	8.41008e−11	5.0178

result is expected since the tracking point can be found exactly and the numerical flux is obtained much accurately.

The solution of (22) (24) contains a discontinuous square pulse and several continuous but high-gradient profiles. This problem is conventionally employed to test high-resolution schemes. The output time is chosen to be $t = 2$ and the time step is chosen to be $\Delta t = 0.6\Delta x$ for the WENOEL and WENOJS schemes. Figure 1 gives the solutions of the WENOEL and WENOJS methods with cells $N = 200$. From Figure 1, we can find that the WENOEL and WENOJS methods can both capture the discontinuous and steep solutions. And for the first and third high-frequency waves, the WENOEL performs slightly better than WENOJS method. Figure 2 is used to test the long-time behavior of these two methods

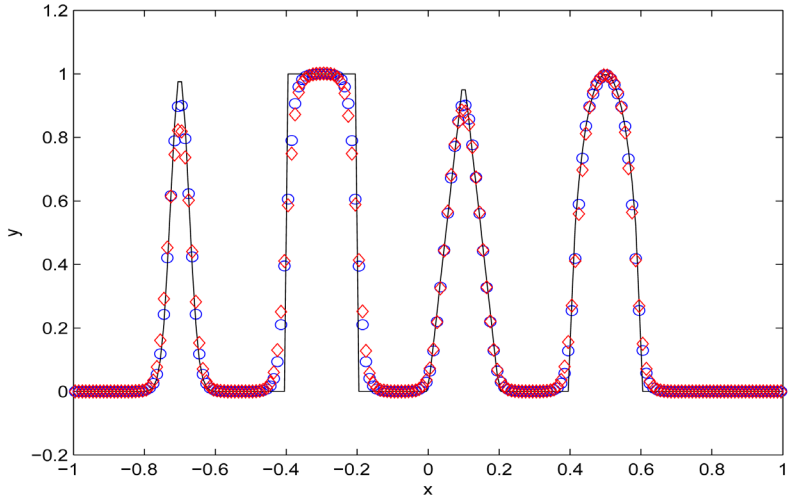

Figure 1. The numerical solutions of (22) (24) is computed by WENOEL and WENOJS methods with $N = 200$ until time $t = 2$. The blue "○" and red "◊" denote the solutions of WENOEL and WENOJS methods, respectively.

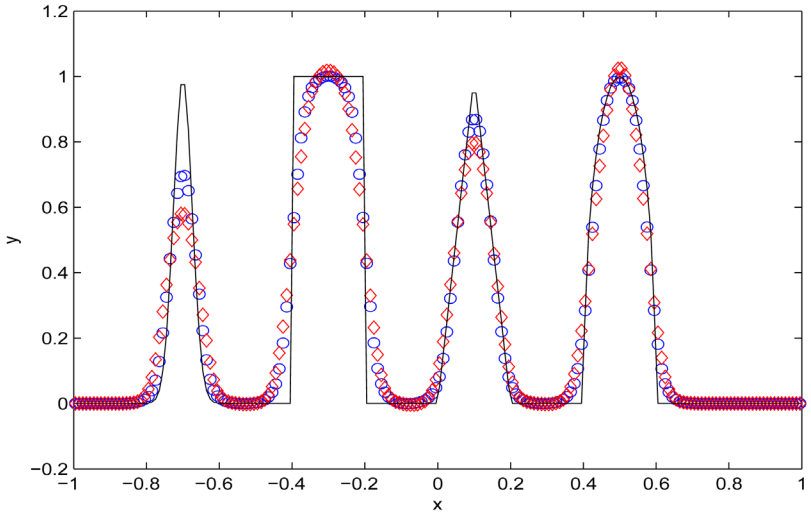

Figure 2. The numerical solutions of (22) (24) is computed by WENOEL and WENOJS methods with $N = 200$ until time $t = 20$. The blue "○" and red "◊" denote the solutions of WENOEL and WENOJS methods, respectively.

after 10 cycles, *i.e.* $t = 20$. Obviously, following long-time evolution, the WENOEL gives higher-resolution solution around four waves than that of WENOJS method. In addition, we present the comparison of CPU time (in seconds) and l_1 error for this problem between the WENOEL and WENOJS methods.

3.1.2. Burger Equation

In this subsection, we consider the Burger's equation

$$u_t + \left(\frac{u^2}{2}\right)_x = 0, \quad -1 \le x \le 1, \ t \ge 0 \tag{25}$$

subject to two initial conditions. one initial condition is

$$u(x,0) = \begin{cases} 1 & -1 \le x \le 0, \\ 0 & 0 \le x \le 1, \end{cases} \tag{26}$$

and the other is

$$u(x,0) = 1.5 - \sin(\pi x). \tag{27}$$

For the nonlinear scalar equation, we first use Rankine-Hugoniot jump condition to identify the upwinding direction, then compute the average flux by (17) or (21). In addition, an entropy fix is used to modify the average flux when rarefaction wave appears. For the problems (25) (26) and (25) (27), we both set $\Delta t = 0.6\frac{\Delta x}{\lambda}$ for the WENOEL and WENOJS schemes, where $\lambda = \max_{1\le i\le N}\left\{\frac{\partial f}{\partial u}\right\}_i$. For the problem (25) (26), the numerical solutions are computed with cells $N = 100$ until time $t = 1$. From Figure 3, the solution of this problem contains a rarefaction and a shock, and these two methods arrive almost same resolution. In Figure 4, the problem (25) (27) has the same set but the output time $t = 0.37$, and has the same conclusion with last problem. In a word, for nonlinear Burger's

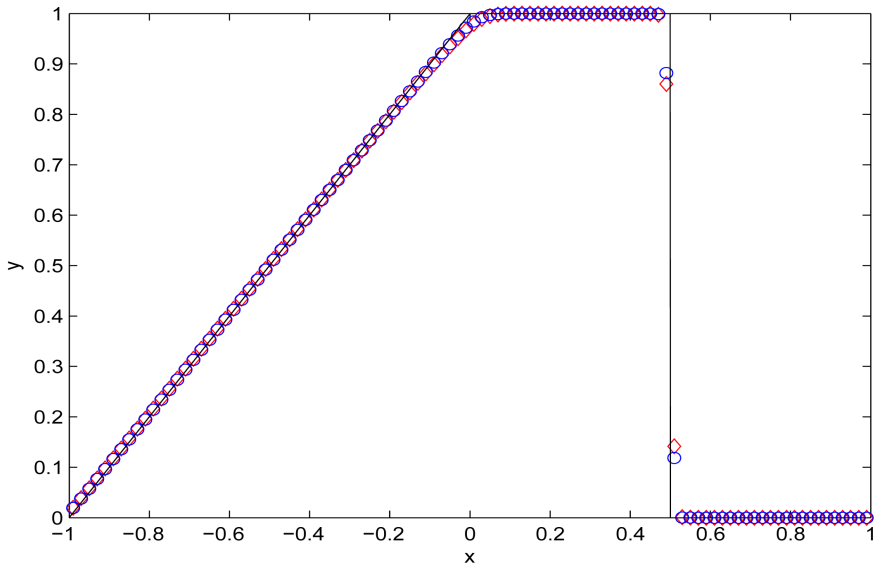

Figure 3. The numerical solutions of (25) (26) are computed by the WENOEL and WENOJS methods with $N = 100$ until time $t = 1$. The blue "○" and red "◊" denote the solutions of WENOEL and WENOJS methods, respectively.

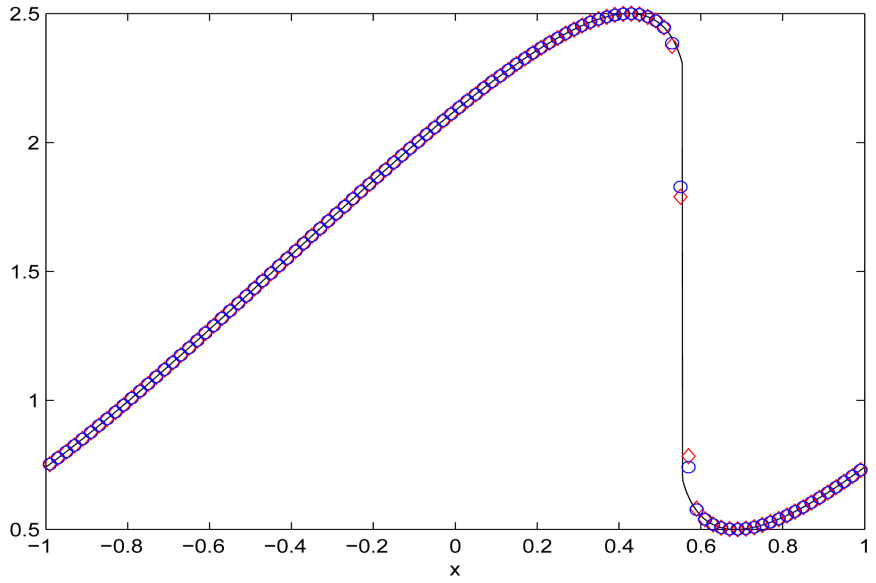

Figure 4. The numerical solutions of (25) (27) are computed by the WENOEL and WENOJS methods with $N=100$ until time $t=0.37$. The blue "$\circ$" and red "$\diamond$" denote the solutions of WENOEL and WENOJS methods, respectively.

equation, the WENOEL scheme can also obtain comparable numerical results and only need half CPU time. The presentation of CPU time is omitted for saving space.

3.1.3. Buckley-Leverett Equation

As an example where nonconvex flux functions arise, we study a simple model for two-phase fluid flow in a porous medium. Consider the scalar conservation law

$$u_t + \left(\frac{u^2}{u^2 + \frac{1}{4}(1-u)^2} \right)_x = 0, \quad -1 \le x \le 1, \quad t \ge 0, \tag{28}$$

with the initial condition

$$u(x,0) = \begin{cases} 1 & -0.5 \le x \le 0, \\ 0 & \text{other.} \end{cases} \tag{29}$$

Figure 5 shows the solutions computed by the WENOEL and WENOJS methods with cells $N=100$ until time $t=0.4$. Also, we choose the time steps $\Delta t = 0.6\frac{\Delta x}{\lambda}$ for WENOEL and WENOJS schemes, where $\lambda = \max_{1\le i\le N}\left\{\frac{\partial f}{\partial u}\right\}_i$.

From **Figure 5** the WENOEL method generates the solution which is comparable with that of the WENOJS method. For the nonconvex flux problem, the WENOEL scheme can also have a robust performance as WENOJS scheme.

3.2. The 1D Euler Equation

In this subsection, we consider 1D Euler equations since one of the main application areas of high-resolution scheme is compressible gas dynamics,

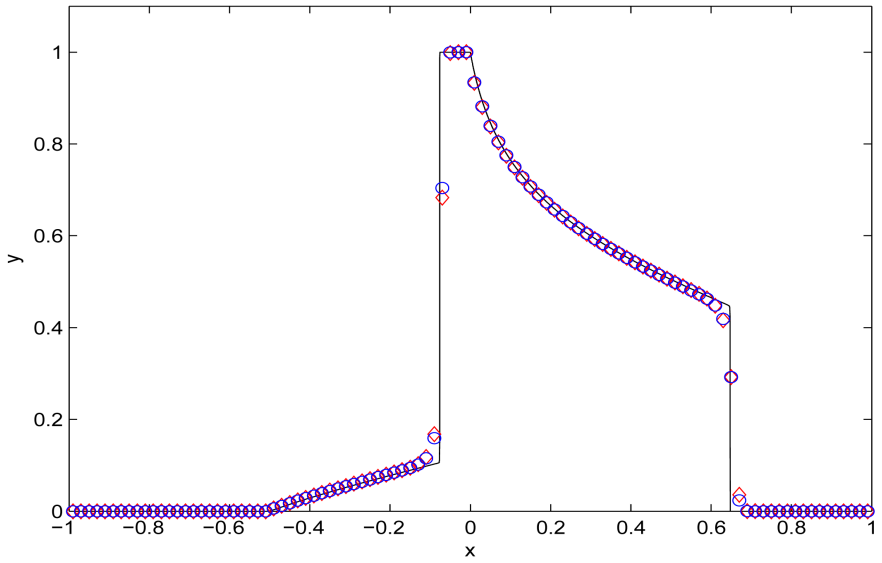

Figure 5. The numerical solutions of (28) (29) are computed by the WENOEL and WENOJS methods with $N=100$ until time $t=0.4$. The blue "$\circ$" and red "$\diamond$" denote the solutions of WENOEL and WENOJS methods, respectively.

$$\begin{pmatrix} \rho \\ \rho u \\ E \end{pmatrix}_t + \begin{pmatrix} \rho u \\ \rho u^2 + p \\ (E+p)u \end{pmatrix}_x = 0, \tag{30}$$

where ρ, u, p, E are density, velocity, pressure and total energy, respectively. The system of equations is closed by the equation of state for an ideal polytropic gas:

$$E = \frac{p}{\gamma - 1} + \frac{1}{2}\rho u^2,$$

where the ratio of specific heats $\gamma = 1.4$.

The following three initial conditions combined with Euler Equations (30) are considered, which are often used to examine the methods for solving Euler equations:

$$(\rho, u, p) = \begin{cases} (0.445, 0.698, 3.528), & \text{if } -5 \le x < 0, \\ (0.5, 0, 0.571), & \text{if } 0 \le x \le 5, \end{cases} \tag{31}$$

$$(\rho, v, p) = \begin{cases} \left(\frac{27}{7}, \frac{4\sqrt{35}}{9}, \frac{31}{3}\right), & \text{if } -5 \le x < -4, \\ (1 + 0.2\sin(5x), 0, 1), & \text{if } -4 \le x \le 5. \end{cases} \tag{32}$$

$$(\rho, u, p) = \begin{cases} (1, 0, 1000), & \text{if } -5 \le x < -4, \\ (1, 0, 0.01), & \text{if } -4 \le x < 4, \\ (1, 0, 100), & \text{if } 4 \le x \le 5, \end{cases} \tag{33}$$

For solving Euler equations, Roe linearization (*i.e.* Roe average) is used to locally freeze a nonlinear system to a linear system. Then this system is decoupled into three advection equations and each equation is solved by procedure

in Section 0. It is noted that the propagation direction of solution of each advection equation is distinguished by corresponding eigenvalue of Roe matrix.

For the problem (30) (31), called Lax problem [26], we solve it with $\Delta t = 0.6\Delta x/\lambda$ for the WENOEL and WENOJS methods. **Figure 6** is computed with cells $N = 200$ and output time $t = 1.3$. Apparently, the WENOEL method performs almost the same as WENOJS method and presents a robust solution for this shock-tube problem. The exact solution is obtained by the solver presented in [24].

The problem (30) (32), called shock entropy wave interaction [4], is usually used to test the high-order methods for capturing the high-frequency waves. The CFL number is also set to be $\Delta t = 0.6\Delta x/\lambda$ for the WENOEL and WENOJS methods. We evolve the equations up to $t = 1.8$ with cells $N = 400$. For this problem, a Mach 3 shock wave moved right interacts with sine waves in a density which lead to a field obtained smooth and discontinuous structures. The reference solution in **Figure 7** is computed by WENOJS method with cells $N = 2000$. It is plainly seen from **Figure 7** that, as WENOJS, the WENOEL method performs with the high resolution. Furthermore, from **Figure 8**, we can find that in the part of high-frequency waves the WENOEL method performs less dissipative than the WENOJS method.

The problem (30) (33) was originally proposed as a benchmark for testing several numerical methods by Woodward and Colella [19]. The reflective boundary conditions are applied to both boundaries. The CFL number is also set to be $\Delta t = 0.6\Delta x/\lambda$ for the WENOEL and WENOJS methods. We evolve the equations up to $t = 0.38$ with cells $N = 400$. The reference solution in **Figure 9** is computed by WENOJS method with cells $N = 2000$. As the last two problems, the WENOEL method gives almost the same resolution as WENOJS method.

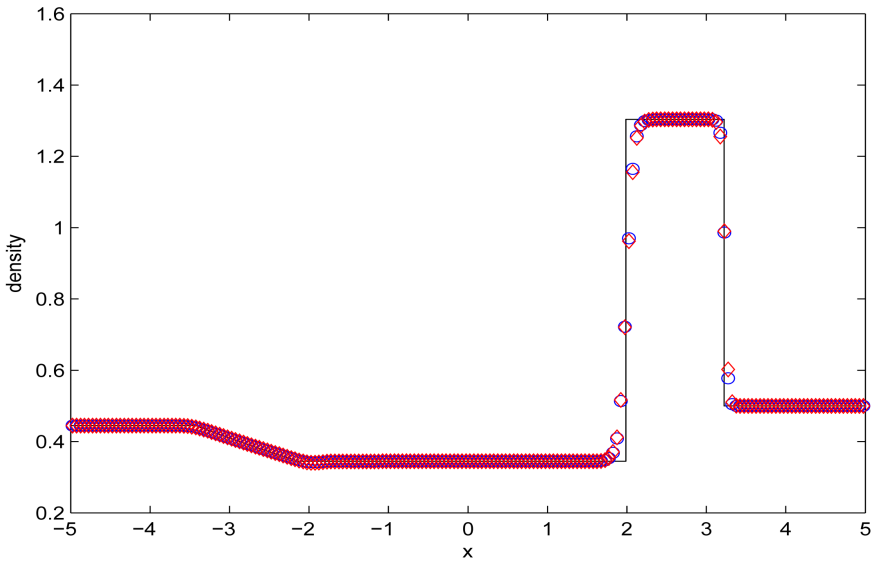

Figure 6. The shock-tube problem (30) (31) is computed by the WENOEL and WENOJS methods with cells $N = 200$ until time $t = 1.3$. The blue "$\circ$" and red "$\diamond$" denote the solutions of WENOEL and WENOJS methods, respectively.

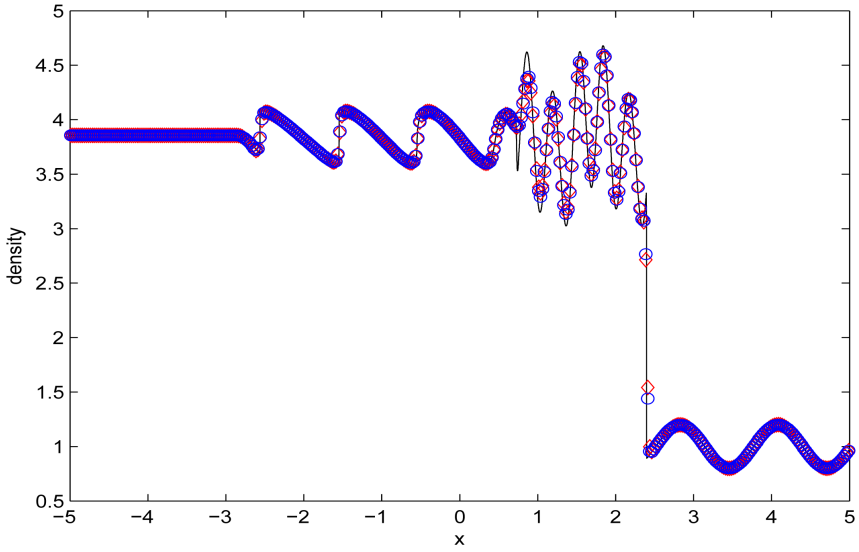

Figure 7. The shock entropy wave problem (30) (32) is computed by the WENOEL and WENOJS methods with cells $N=400$ until time $t=1.8$. The blue "$\circ$" and red "$\diamond$" denote the solutions of WENOEL and WENOJS methods, respectively.

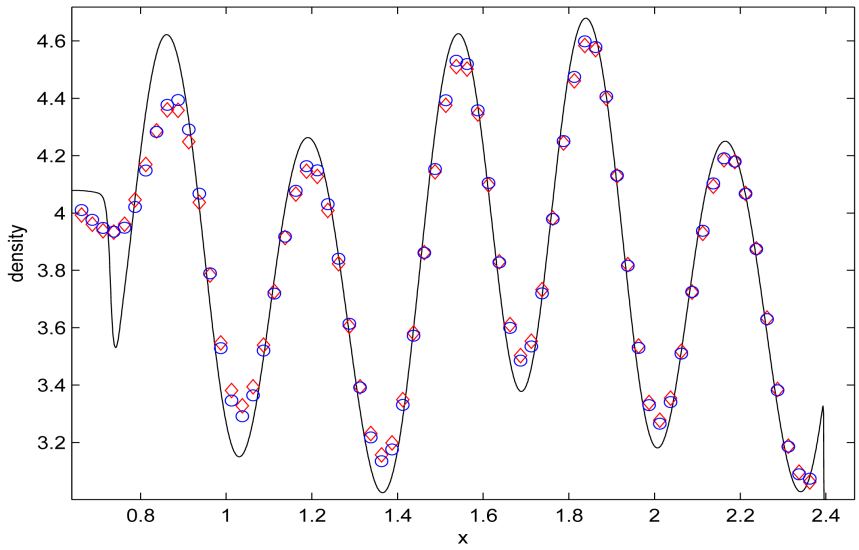

Figure 8. The zoomed figure of **Figure 7** around the high-frequency waves.

3.3. The 2D Euler System

Finally, we consider a numerical experiment for 2D Euler equations for gas dynamics,

$$\begin{bmatrix}\rho\\ \rho u\\ \rho v\\ E\end{bmatrix}_t+\begin{bmatrix}\rho u\\ \rho u^2+p\\ \rho uv\\ (E+p)u\end{bmatrix}_x+\begin{bmatrix}\rho v\\ \rho uv\\ \rho v^2+p\\ (E+p)v\end{bmatrix}_y=0, \tag{34}$$

where the equation of state is

$$p=(\gamma-1)\left(E-\frac{1}{2}\rho\left(u^2+v^2\right)\right),\quad \gamma=1.4.$$

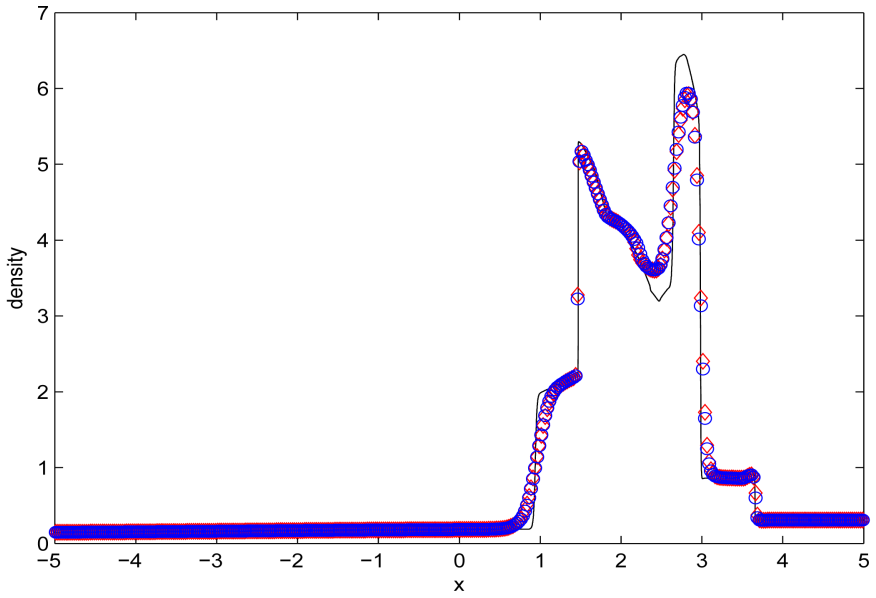

Figure 9. The blast wave problem (30) (33) is computed by the WENOEL and WENOJS methods with cells $N=400$ until time $t=0.38$. The blue "$\circ$" and red "$\diamond$" denote the solutions of WENOEL and WENOJS methods, respectively.

3.3.1. Double Mach Reflection

Here we firstly apply the WENOEL and WENOJS schemes to the 2D double-Mach shock reflection problem where a strong vertical shock moves horizontally into a wedge which inclined with some angle with the ratio of specific heats $\gamma=1.4$. Initially, this problem was proposed by Woodward and Colella [19] and had been taken extensively as a test example for high-order schemes. The computational domain is chosen to be $[0,4]\times[0,1]$ and the reflective wall lies on the bottom of the computational domain for $\frac{1}{6}\le x\le 4$. In the beginning, a Mach 10 shock, moving right, is located at $x=\frac{1}{6}$, $y=0$ and makes an angle 60° with the x-axis. For the boundary conditions, the exact postshock condition is imposed for bottom boundary from $x=0$ to $x=\frac{1}{6}$, and the reflective boundary condition is imposed for the rest; the flows are imposed on the top boundary such that there is no interaction with the Mach 10 shock; inflow and outflow boundary conditions are set for the left and right boundaries respectively. The unshocked fluid has a density of 1.4, a pressure of 1 and this problem is run until $t=0.2$ with cells 960×240. In addition, there exists numerical instability in computing this problem by unwind methods. A H-correction procedure presented in Subsection 2.3 is used to eliminate this instability. The H-correction procedure injects much dissipation in unstable region by the way of shifting the upwind scheme into flux splitting scheme. The contour lines of the density are shown **Figure 10**. These two methods both obtain high-resolution solutions. In region of complicated structure, the WENOJS scheme performs better than our scheme.

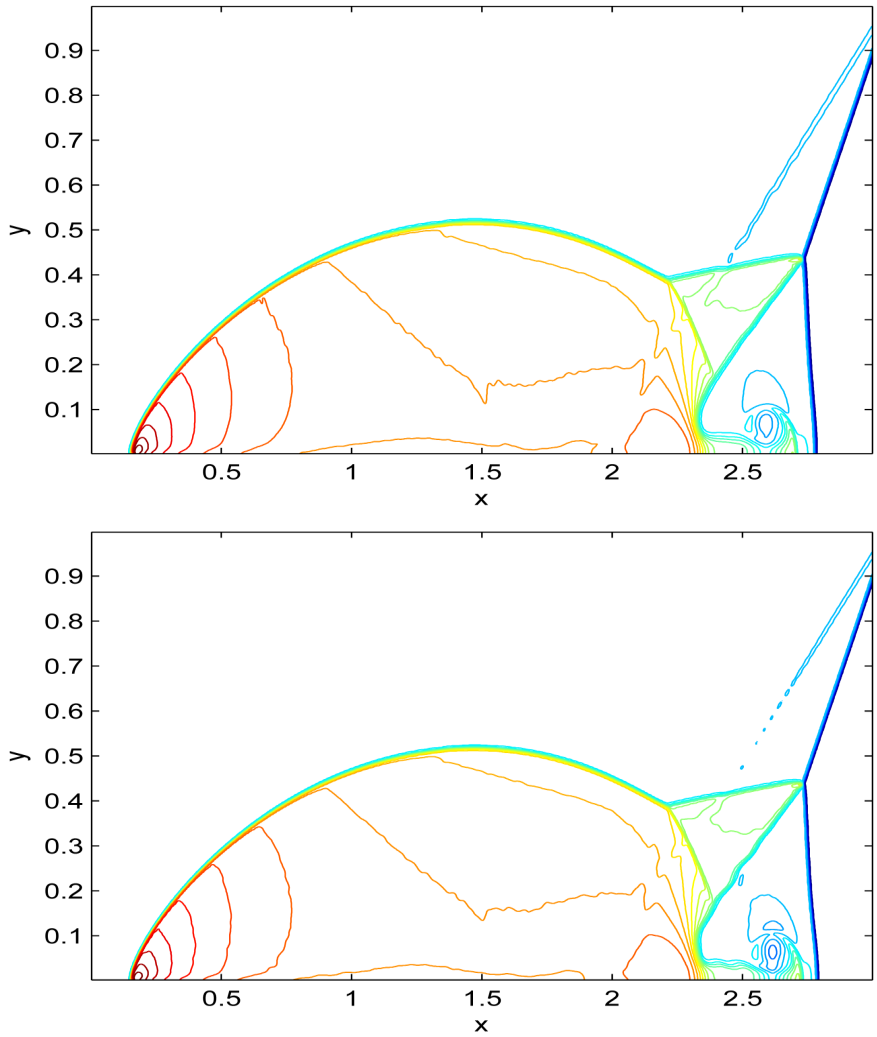

Figure 10. Double Mach problem. The density ρ is shown with meshes 960×240. 30 equally spaced contour lines are plotted from 1.731 to 22.9705. The top: WENOEL scheme. The bottom: WENOJS scheme.

3.3.2. Mach 3 Tunnel with a Step

This wind problem is also originally presented in [19]. This problem is set up as follows. The wind tunnel is 1 length unit wide and 3 length units long. The step is 0.2 length units high and is located 0.6 length units from the left-hand end of the tunnel. The problem is initialized by a uniform right-going Mach 3 flow. Reflective boundary conditions are applied along the wall of the tunnel. The inflow and outflow boundary conditions are applied at the entrance and exit, respectively. The corner of is a singular point and we use the same technique to correct it [19], which is based on the assumption of a nearly steady flow in the region near the corner. In addition, to eliminate the numerical instability originated from the upwind scheme, the same H-correction procedure as last problem is used to eliminate this flaw. We evolve the initial data until time 4 with a grid of 720×240 cells by the WENOEL and WENOJS methods. The CFL number is chosen to be 0.6 for these methods. The contour lines of the density are displayed in **Figure 11**. We observe the good resolution and strong

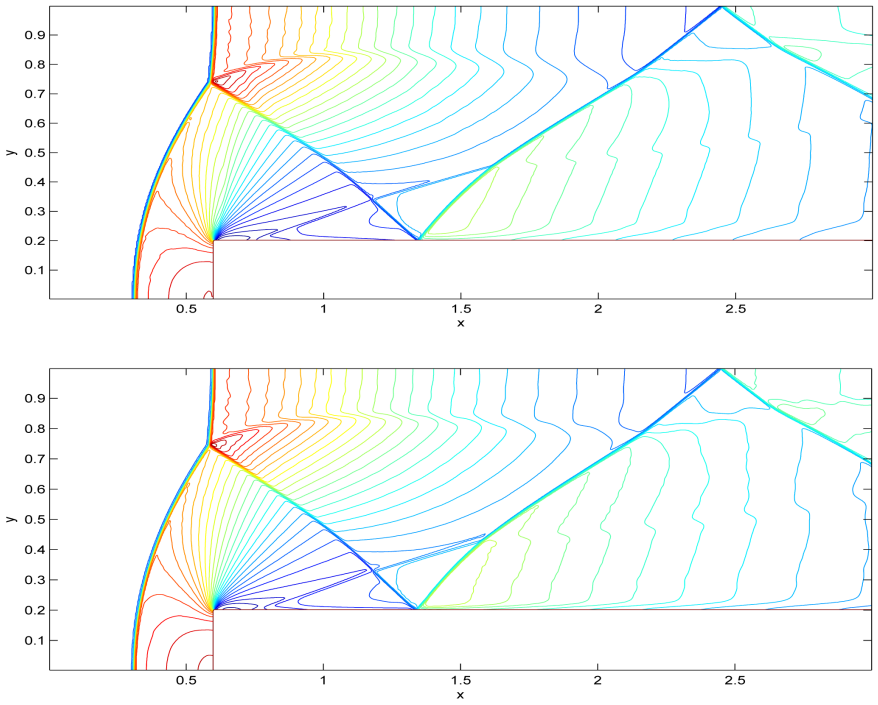

Figure 11. Mach 3 tunnel with a step problem. The density ρ is shown with meshes 20×240. 30 equally spaced contour lines are plotted from 0.12 to 6.41. The top: WENOEL scheme. The bottom: WENOJS scheme.

reflective waves in this test for these methods. Moreover, a better resolution can be observed for the WENOJS method.

3.3.3. 2D Riemann Problem

This is a simple 2D Riemann problem on square $[0,1]\times[0,1]$ is used to test our method which is originally proposed in [27]. The square is divided into four quadrants by straight lines $7x=0.8$ and $y=0.8$. Initially, four different constant states are defined on each of quadrants

$$(\rho,u,v,p)=\begin{cases}(0.5323,1.206,0,0.3), & \text{if } 0\le x<0.8, 0.8\le y\le 1,\\ (1.5,0,0,1.5), & \text{if } 0.8\le x\le 1, 0.8\le y\le 1,\\ (0.138,1.206,1.206,0.029), & \text{if } 0\le x<0.8, 0\le y<0.8,\\ (0.5323,0,1.206,0.3), & \text{if } 0.8<x\le 1, 0\le y<0.8.\end{cases} \quad (35)$$

We solve the problem until $t=0.8$ with cells 200×200. The time steps is set to be $\Delta t=0.6\Delta x/\lambda$ for the WENOEL and WENOJS methods. **Figure 12** gives the density profiles computed by the WENOEL and WENOJS methods, respectively, and these two high-resolution scheme both perform well.

3.3.4. Rayleigh-Taylor Instability Problem

Finally, we consider Rayleigh-Taylor instability problem [28]. This problem describes the flow motions on the interface between fluids with different density. The heavy fluid moves into the region of the light fluid with a fingering nature, which lead to the bubbles of light fluid rising into the heavy fluid and the spikes of heavy fluid falling into the light fluid. The computational domain is

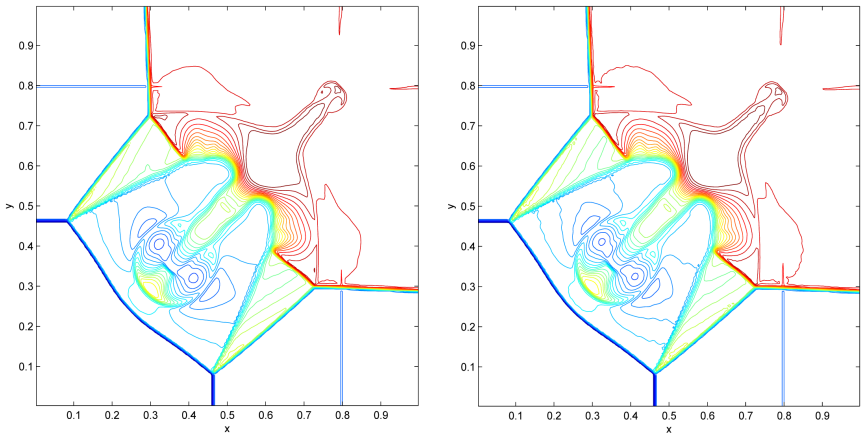

Figure 12. The solutions of 2D Riemann problem computed by the WENOEL and WENOJS methods. The left: WENOEL scheme. The right: WENOJS scheme.

$\left[0,\frac{1}{4}\right]\times[0,1]$, and the initial conditions is

$$(\rho,u,v,p)=\begin{cases}\left(2,0,-0.025\sqrt{\frac{\gamma p}{\rho}}\cos(8\pi x),2y+1\right), & \text{if } 0\le y<\frac{1}{2},\\ \left(1,0,-0.025\sqrt{\frac{\gamma p}{\rho}}\cos(8\pi x),y+1.5\right), & \text{if } \frac{1}{2}\le y\le 1,\end{cases} \tag{36}$$

where the ratio of specific heats $\gamma=\frac{5}{3}$. For the boundary conditions, the reflective boundary conditions are imposed on the left and right boundaries; the flows are set as $(\rho,u,v,p)=(1,0,0,2.5)$ and $(\rho,u,v,p)=(2,0,0,1)$ for the top and bottom boundaries, respectively. In addition, the source terms ρ and ρv are added to the right of the third and fourth equations of Euler equations (34), respectively. For this problem, from Figure 13, we can find that the WENOEL and WENOJS methods both generate high-resolution numerical solutions.

From the performance of these four 2D problems, as in the 1D problems, we can find that the similar resolution with the WENOJS scheme can be observed.

4. Conclusion

In this paper, we propose a simple and easily implemented method to solve hyperbolic conversation laws. The main idea of this method is the transformation of integration of flux function in time into integration in space. It is this procedure that leads to the average flux at cell edge which can be directly evaluated. In addition, the evaluation of the average flux is easily implemented and can be combined with any non-oscillatory spatial reconstruction. Through the performance on lots of classical examples, we can find this method is rather robust. We compare our scheme with the classical WENOJS scheme and almost the same performance on resolution is observed. And from the comparisons of resolution and efficiency for these 1D and 2D examples, we can find that the

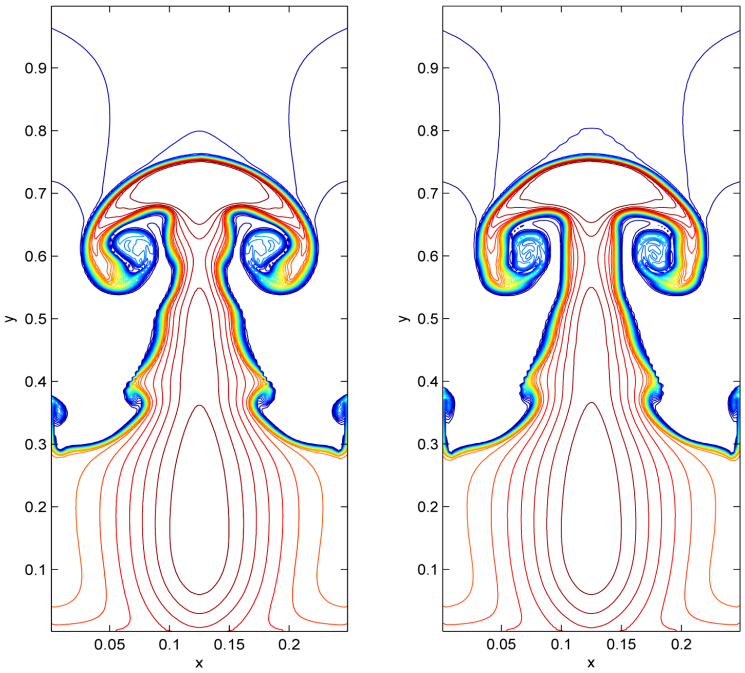

Figure 13. The solutions of Rayleigh-Taylor instability problem computed by the WENOEL and WENOJS methods with cells 100×400 . The left: WENOEL scheme. The right: WENOJS scheme.

proposed method in this paper can present comparable results with the classical WENOJS scheme, but with less CPU time.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (No. 11501238), Natural Science Foundation of Guangdong Province (No. 2012A030313119) and Supported by the Major Project Foundation of Guangdong Province Education Department (No. 2014KZDXM070).

References

[1] Harten, A., Engquist, B., Osher, S. and Chakravarthy, S. (1987) Uniformly High Order Essentially Non-Oscillatory Schemes III. *Journal of Computational Physics*, **71**, 231-303. https://doi.org/10.1016/0021-9991(87)90031-3

[2] Harten, A. and Osher, S. (1987) Uniformly High Order Essentially Non-Oscillatory Schemes I. *SIAM Journal on Numerical Analysis*, **24**, 279-309. https://doi.org/10.1137/0724022

[3] Shu, C.W. and Osher, S. (1988) Efficient Implementation of Essentially Non-Oscillatory shock-Capturing Schemes. *Journal of Computational Physics*, **77**, 439-471. https://doi.org/10.1016/0021-9991(88)90177-5

[4] Shu, C.W. and Osher, S. (1989) Efficient Implementation of Essentially Non-Oscillatory Shock-Capturing Schemes II. *Journal of Computational Physics*, **83**, 32-78. https://doi.org/10.1016/0021-9991(89)90222-2

[5] Jiang, G.S. and Shu, C.W. (1996) Efficient Implementation of Weighted ENO Schemes. *Journal of Computational Physics*, **126**, 202-228. https://doi.org/10.1006/jcph.1996.0130

[6] Liu, X.D., Osher, S. and Chan, T. (1994) Weighted Essentially Non-Oscillatory Schemes. *Journal of Computational Physics*, **115**, 200-212. https://doi.org/10.1006/jcph.1994.1187

[7] Cockburn, B. and Shu, C.W. (1998) The Runge-Kutta Discontinuous Galerkin Method for Conservation Laws V: Multidimensional Systems. *Journal of Computational Physics*, **141**, 199-224. https://doi.org/10.1006/jcph.1998.5892

[8] Cockburn, B. and Shu, C.W. (1989) TVB Runge-Kutta Local Projection Discontinuous Galerkin Finite Element Method for Conservation Laws II: General Framework. *Mathematics of Computation*, **52**, 411-435.

[9] Shu, C.W. (1998) Essentially Non-Oscillatory and Weighted Essentially Non-Oscillatory Schemes for Hyperbolic Conservation Laws. In: Cockburn, B., Shu, C.-W., Johnson, C. and Tadmor, E., Eds., *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*, Lecture Notes in Mathematics, Springer, Berlin, 325-432. https://doi.org/10.1007/BFb0096355

[10] Arbogast, T. and Huang, C. (2006) A Fully Mass and Volume Conserving Implementation of a Characteristic Method for Transport Problems. *SIAM Journal on Scientific Computing*, **28**, 2001-2022. https://doi.org/10.1137/040621077

[11] Arbogast, T. and Wheeler, M.F. (1995) A Characteristics-Mixed Finite Element Method for Advection-Dominated Transport Problems. *SIAM Journal on Numerical Analysis*, **32**, 404-424. https://doi.org/10.1137/0732017

[12] Douglas, J. and Russell, T.F. (1982) Numerical Methods for Convection-Dominated Diffusion Problems Based on Combining the Method of Characteristics with Finite Element or Finite Difference Procedures. *SIAM Journal on Numerical Analysis*, **19**, 871-885. https://doi.org/10.1137/0719063

[13] Wang, H. and Al-Lawatia, M. (2006) A Locally Conservative Eulerian-Lagrangian Control-Volume Method for Transient Advection-Diffusion Equations. *Numerical Methods for Partial Differential Equations*, **22**, 577-599. https://doi.org/10.1002/num.20106

[14] Huang, C., Arbogast, T. and Qiu, J. (2012) An Eulerian-Lagrangian WENO Finite Volume Scheme for Advection Problems. *Journal of Computational Physics*, **231**, 4028-4052. https://doi.org/10.1016/j.jcp.2012.01.030

[15] Qiu, J.M. and Christlieb, A. (2010) A Conservative High Order Semi-Lagrangian WENO Method for the Vlasov Equation. *Journal of Computational Physics*, **229**, 1130-1149. https://doi.org/10.1016/j.jcp.2009.10.016

[16] Qiu, J.M. and Shu, C.W. (2011) Conservative High Order Semi-Lagrangian Finite Difference WENO Methods for Advection in Incompressible Flow. *Journal of Computational Physics*, **230**, 863-889. https://doi.org/10.1016/j.jcp.2010.04.037

[17] Qiu, J.M. and Shu, C.W. (2011) Conservative Semi-Lagrangian Finite Difference WENO Formulations with Applications to the Vlasov Equation. *Communications in Computational Physics*, **10**, 979-1000. https://doi.org/10.4208/cicp.180210.251110a

[18] Huang, C. and Arbogast, T. (2016) An Eulerian-Lagrangian WENO Scheme for Nonlinear Conservation Laws. *Numerical Methods for Partial Differential Equations*, In Press. https://doi.org/10.1002/num.22091

[19] Woodward, P. and Colella, P. (1984) The Numerical Simulation of Two-Dimensional Fluid Flow with Strong Shocks. *Journal of Computational Physics*, **54**, 115-173. https://doi.org/10.1016/0021-9991(84)90142-6

[20] Kim, S., Kim, C., Rho, O. and Hong, S. (2003) Cures for the Shock Instability: Development of a Shock-Stable Roe Scheme. *Journal of Computational Physics*, **185**,

342-374. https://doi.org/10.1016/S0021-9991(02)00037-2

[21] Pandolfi, M. and D'Ambrosio, D. (2001) Numerical Instabilities in Upwind Methods: Analysis and Cures for the "Carbuncle" Phenomenon. *Journal of Computational Physics*, **166**, 271-301. https://doi.org/10.1006/jcph.2000.6652

[22] Sanders, R., Morano, E. and Druguet, M. (1998) Multidimensional Dissipation for Upwind Schemes: Stability and Applications to Gas Dynamics. *Journal of Computational Physics*, **145**, 511-537. https://doi.org/10.1006/jcph.1998.6047

[23] Strang, G. (1968) On the Construction and Comparison of Difference Schemes. *SIAM Journal on Numerical Analysis*, **5**, 506-517. https://doi.org/10.1137/0705041

[24] Toro, E.F. (2009) Riemann Solver and Numerical Methods for Fluid Dynamics. 3rd Edition, Springer, New York. https://doi.org/10.1007/b79761

[25] Shu, C.W. (1988) Total-Variation-Diminishing Time Discretizations. *SIAM Journal on Scientific and Statistical Computation*, **9**, 1073-1084. https://doi.org/10.1137/0909073

[26] Lax, P.D. (1954) Weak Solutions of Nonlinear Hyperbolic Equations and Their Numerical Computation. *Communications on Pure and Applied Mathematics*, **7**, 159-193. https://doi.org/10.1002/cpa.3160070112

[27] Schulz-Rinne, C.W., Collins, J.P. and Glaz, H.M. (1993) Numerical Solution of the Riemann Problems for Two-Dimentional Gas Dynamics. *SIAM Journal on Scientific Computing*, **14**, 1394-1414. https://doi.org/10.1137/0914082

[28] Gardner, C.L., Glimm, J., McBryan, O., Menikoff, R., Sharp, D.H. and Zhang, Q. (1988) The Dynamics of Bubble Growth for Rayleigh-Taylor Unstable Interfaces. *Physics of Fluids*, **31**, 447-465. https://doi.org/10.1063/1.866826