

# A Speed-Up Strategy for Finite Volume WENO Schemes for Hyperbolic Conservation Laws

Fei Teng · Li Yuan · Tao Tang

Received: 3 February 2010 / Revised: 28 June 2010 / Accepted: 7 July 2010 / Published online: 22 July 2010  
© Springer Science+Business Media, LLC 2010

**Abstract** In this paper, a speed-up strategy for finite volume WENO schemes is developed for solving hyperbolic conservation laws. It adopts p-adaptive like reconstruction, which automatically adjusts from fifth order WENO reconstruction to first order constant reconstruction when nearly constant solutions are detected by the undivided differences. The corresponding order of accuracy for the solutions is shown to be the same as obtained by original WENO schemes. The strategy is implemented with both WENO and mapped WENO schemes. Numerical examples in different space dimensions show that the strategy can reduce the computational cost by 20–40%, especially for problems with large fraction of constant regions.

**Keywords** Conservation law · Finite volume WENO method · Speed-up strategy · Multi-dimension

## 1 Introduction

The discontinuities in solutions of hyperbolic conservation laws often produce spurious oscillations in high-order accurate numerical schemes. Many strategies have been developed to eliminate such oscillations. A popular one developed in 1980s was the limiter methods, most of which satisfy the total variation diminishing (TVD) property for one-dimensional scalar

---

F. Teng · L. Yuan (✉)

LSEC and Institute of Computational Mathematics and Scientific/Engineering Computing, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China  
e-mail: [lyuan@lsec.cc.ac.cn](mailto:lyuan@lsec.cc.ac.cn)

F. Teng

e-mail: [tengfei@lsec.cc.ac.cn](mailto:tengfei@lsec.cc.ac.cn)

T. Tang

Department of Mathematics, Hong Kong Baptist University, Hong Kong, China  
e-mail: [ttang@hkbu.edu.hk](mailto:ttang@hkbu.edu.hk)

conservation laws. Although TVD limiter methods can meet the need of many practical applications, they suffer from being only first order accurate near smooth extrema, thus are not suitable for accuracy-demanding simulations like shock-turbulence interaction problems.

The essentially non-oscillatory (ENO) schemes as first introduced in [7] and later developed in [21, 23, 24] were designed to overcome the flaw of accuracy degeneration near smooth extrema. The stencil-choosing technique is used to avoid cross-shock interpolation so as to reduce spurious oscillations and at the same obtain more than second-order accuracy near smooth extrema. Later, it was found that spending computational time in the dropped stencils was a disadvantage of ENO schemes. Weighted essentially non-oscillatory (WENO) schemes were then developed in [15] and [12]. In a WENO scheme, all candidate stencils are utilized by a convex combination, resulting in  $(2r - 1)$ -th order accuracy rather than  $r$ -th order accuracy as in the ENO scheme which uses only one of the several candidate stencils. Also, the smoothness indicators are used to allocate a weight to a candidate stencil, and this achieves stencil-choosing task automatically.

Because of its robustness and adaptivity, the WENO scheme drew much attention soon after its introduction. WENO schemes are nowadays widely used for solving conservation laws as well as other non-linear equations such as Hamilton-Jacobi equations [10, 11, 29], and great efforts were undertaken to make the schemes more applicable. In [9, 27, 31], some reconstruction procedures on unstructured meshes were developed, extending the schemes to applications in more complicated domains. In [20], a splitting technique was introduced to resolve the instability caused by negative weights. In [5, 19, 32], some strategies on combination of WENO and compact schemes were suggested to increase the wave resolution capability. Other developments include reducing dispersion errors and relaxing CFL condition [1], adopting Hermite polynomials as building blocks [16, 17], designing new smoothness indicator [30], etc. See [4] for a review of recent progresses on WENO schemes.

In 2005, a more accurate analysis of the fifth order WENO scheme was given in [8], showing probable accuracy degeneration in critical points of solutions. A correction to this flaw by a transform with regard to the non-linear weights led to the mapped WENO (hereafter WENOM) scheme. In this new version of WENO scheme and subsequent work in [3], the uniformly high-order accuracy was rigorously proved for smooth solutions. However, additional transforms are needed in computing the smoothness indicators.

In spite of great improvement made, the disadvantage of high computational cost for finite volume type WENO schemes remains intact, and the situation becomes even worse for late WENO versions like the WENOM scheme. In this paper, we aim at decreasing computational cost of finite volume WENO schemes. We developed a p-adaptive like strategy. For sufficiently smooth and nearly constant solutions, it is sufficient to use a constant reconstruction instead of a full WENO reconstruction. We showed by a simple theoretical analysis that the constant reconstruction can keep the same order of accuracy as the original WENO reconstruction. Since in many cases the region with nearly constant solutions will occupy a significant fraction of the whole domain, the constant reconstruction can save a lot of computational cost. Several numerical examples for scalar conservation laws and the Euler equations are given to demonstrate the effectiveness of the proposed strategy.

This paper is organized as follows. Section 2 briefly describes the general finite volume method to solve hyperbolic conservation laws. In Sect. 3, the finite volume WENO and WENOM reconstruction procedures are stated. In Sect. 4, the p-adaptive like strategy is presented in detail. Numerical examples are shown in Sect. 5 to verify the accuracy preserving property and the computational efficiency.

## 2 Finite Volume Method for Hyperbolic Conservation Laws

### 2.1 General Framework

A general system of hyperbolic conservation laws

$$\frac{\partial \mathbf{u}}{\partial t} + \sum_{i=1}^d \frac{\partial \mathbf{F}_i(\mathbf{u})}{\partial x_i} = 0 \tag{2.1}$$

is considered where  $d$  is the space dimension,  $\mathbf{u}$  is the solution vector, and  $\mathbf{F}_i$  is the flux vector in the  $i$ -th direction. For completeness, we will give a brief description of the general finite volume framework. For detail, [2] and [13] are recommended. Let the computational domain,  $\Omega \in \mathbb{R}^d$ , be tessellated into non-overlapping control volumes which cover the whole solution domain, that is

$$\bigcup_{j \in \mathcal{J}} \bar{T}_j = \bar{\Omega}, \tag{2.2}$$

$$T_j \cap T_i = \phi, \quad \forall i, j \in \mathcal{J}, \tag{2.3}$$

where  $\mathcal{J}$  is the set of index. The cell average of the solution vector is defined as

$$\bar{\mathbf{u}}_j = \frac{1}{|T_j|} \int_{T_j} \mathbf{u} dx. \tag{2.4}$$

According to the integral form of equation (2.1)

$$\int_{T_j} \mathbf{u}_t dx + \oint_{\partial T_j} \vec{\mathbf{f}} \cdot d\vec{\mathbf{n}} = 0, \tag{2.5}$$

and the Gauss quadrature rule, the following semi-discrete equation with regard to cell averages is obtained,

$$\frac{d\bar{\mathbf{u}}_j}{dt} + \frac{1}{|T_j|} \sum_{e \in \partial T_j} \sum_{s \in G_e} \alpha_s \mathbf{g}_{jk}(\mathbf{u}_j^s, \mathbf{u}_k^s) = 0, \tag{2.6}$$

where  $|T_j|$  is the volume (or area) of cell  $T_j$ ,  $e$  is an edge of  $\partial T_j$ ,  $\vec{\mathbf{n}}$  is the unit outward normal vector to  $e$ ,  $G_e$  is the collection of Gauss points on edge  $e$ ,  $\alpha_s$  is the weight of Gauss quadrature,  $\mathbf{u}_j^s$  is the limit value of the reconstructed  $\mathbf{u}_j(x)$  as  $x$  approaches a Gauss point  $s$ ,  $k$  is the index of the opposite cell which shares the same boundary with cell  $j$ , and the numerical flux function  $\mathbf{g}_{jk}$  is required to satisfy

$$\begin{aligned} \mathbf{g}_{jk}(\mathbf{u}, \mathbf{v}) &= -\mathbf{g}_{kj}(\mathbf{v}, \mathbf{u}), \\ \mathbf{g}_{jk}(\mathbf{u}, \mathbf{u}) &= \vec{\mathbf{f}}(\mathbf{u}) \cdot \vec{\mathbf{n}}, \end{aligned} \tag{2.7}$$

for conservation and consistency.

### 2.2 Flux Function $\mathbf{g}(\mathbf{u}, \mathbf{v})$

In this paper, two kinds of flux functions are used. One is the local Lax-Friedrichs flux, i.e.

$$\mathbf{g}(\mathbf{u}, \mathbf{v}) = \frac{1}{2} \left[ \vec{\mathbf{f}}(\mathbf{u}) \cdot \vec{\mathbf{n}} + \vec{\mathbf{f}}(\mathbf{v}) \cdot \vec{\mathbf{n}} - \max\{\alpha(\mathbf{u}), \alpha(\mathbf{v})\}(\mathbf{v} - \mathbf{u}) \right],$$

where  $\alpha(\mathbf{u})$  is the largest magnitude of eigenvalues of the Jacobian  $\frac{\partial \vec{f}}{\partial \mathbf{u}}(\mathbf{u}) \cdot \vec{n}$ . The other one is the HLLC flux especially designed for the Euler equations for which Refs. [25, 26] are good references for detailed description. These two approximate Riemann solvers are used to test the robustness of the speedup strategy.

### 2.3 Time Discretization

The time discretization used in this paper is the 3rd-order TVD Runge-Kutta method [23], i.e.

$$\mathbf{u}^{(1)} = \mathbf{u}^n + \Delta t L(\mathbf{u}^n, t^n), \tag{2.8}$$

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

$$\mathbf{u}^{n+1} = \frac{1}{3}\mathbf{u}^n + \frac{2}{3}\mathbf{u}^{(2)} + \frac{2}{3}\Delta t L\left(\mathbf{u}^{(2)}, t^n + \frac{1}{2}\Delta t\right), \tag{2.10}$$

where  $L$  denotes the spatial discretization operator, which may also depends on  $t$  for some time-dependent problems or boundary conditions.

## 3 WENO and WENOM Reconstruction

We study reconstruction only on Cartesian grids in this paper. For general shaped meshes, the reconstruction procedure is quite involved and is actually ongoing research topics, see [9], [27] and [31] for good reference.

The main goal of reconstruction is to obtain point-wise values  $\{\mathbf{u}_j^s\}$  at Gauss points from cell averages  $\{\bar{\mathbf{u}}_j\}$ . WENO and WENOM reconstructions are described in this section to achieve this task. We present the reconstruction procedure only for a scalar function  $u$  below. There are two approaches available for extending the scalar procedure to system of conservation laws: component-wise and characteristic-wise. In component-wise approach, one simply applies the procedure for the scalar function to each component of the vector function. In characteristic-wise approach, an additional characteristic transformation is made before using the component-wise approach, and corresponding inverse transformation is then used to change the variables from characteristic-wise ones back to component-wise ones, which means that the WENO reconstruction procedure is used on characteristic variables instead of original variables. The transformation is based on the left and right eigenvector matrices of the Jacobian  $\frac{\partial \vec{f}}{\partial \mathbf{u}}(\mathbf{u}) \cdot \vec{n}$  with  $\mathbf{u}$  being some average of  $\bar{\mathbf{u}}_j$  and  $\bar{\mathbf{u}}_{j+1}$  at  $x_{j+\frac{1}{2}}$ . In this paper, for the Euler equations, the characteristic-wise approach and the commonly used Roe average [18] are used.

### 3.1 Polynomial Reconstruction

The principle to achieve high order accuracy for ENO or WENO schemes is the polynomial reconstruction, which means a  $(k - 1)$ th order polynomial  $p(x)$  is constructed on a fixed stencil  $\{T_{j_1}, \dots, T_{j_k}\}$  to approximate a smooth function  $v(x)$ , requiring that for each cell  $T_{j_i}$  on this stencil, the following conservative property is satisfied:

$$\bar{v}_{j_i} = \frac{1}{\Delta x} \int_{T_{j_i}} p(x) dx. \tag{3.1}$$

The objective of reconstruction is to get  $k$ -th order approximation to the scalar function  $v(x)$ , i.e.

$$p(x) = v(x) + O(\Delta x^k) \quad \text{for each } T_{j_l}. \tag{3.2}$$

Unfortunately, whether this objective can be accomplished or not depends on the choice of the stencil, which is a very complicated task for multidimensional general meshes (see [9] for detail). However, on Cartesian grids, the task becomes simpler because the polynomial reconstruction can be done in a dimension-by-dimension way. So only the one-dimensional reconstruction is required. Notice that the value of  $p(x)$  is a linear combination of  $k$  cell averages  $\{\bar{v}_{j_l}\}$  on the stencil of  $\{T_{j_1}, \dots, T_{j_k}\}$ .

### 3.2 One-Dimensional WENO Reconstruction

#### 3.2.1 Weighted Reconstruction Based on Multi-stencils

In one-dimensional case, the final objective of the reconstruction is to obtain the left and right limiting states on the grid interface of two adjacent cells  $j$  and  $j + 1$ , which means that for each cell  $j$ , the reconstruction values at end points  $x_{j-\frac{1}{2}}$  and  $x_{j+\frac{1}{2}}$  are required. More generally, for later use in multi-dimensional case in Sect. 3.3, the reconstructed value at any point  $x \in [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$  is required. For a  $r$ -th order accurate ENO reconstruction in this interval,  $r$  different stencils can be chosen, which are

$$\{\{T_{j+l-r+1}, \dots, T_{j+l}\} : l = 0, \dots, r - 1\}. \tag{3.3}$$

The union of these stencils constitutes a wider stencil allowing a  $(2r - 1)$ th order polynomial reconstruction for  $u(x)$ . The  $(2r - 1)$ th order WENO reconstruction is a weighted combination of  $r$  ENO reconstructions on  $r$  different stencils, i.e.

$$\tilde{u}_W(x) = \sum_{l=0}^{r-1} \omega_l(x) \tilde{u}_{l,L}(x), \tag{3.4}$$

where  $\tilde{u}_W(x)$  is the WENO reconstruction value,  $\tilde{u}_{l,L}(x)$  is the ENO reconstruction value on stencil  $l$ , and  $\omega_l(x)$  is the nonlinear weight which depends on the smoothness of the solution. The purpose of WENO reconstruction is to obtain high-order accuracy and essentially non-oscillatory property.

#### 3.2.2 Optimal Weights

For achieving highest order of accuracy, the weights  $\omega_l$  can be chosen to result in a  $(2r - 1)$ th order polynomial reconstruction, which are called optimal weights. The existence of these weights can be proved by Taylor series expansion (see [6] for detail). However, depending on the location of  $x \in [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$ , the weights may be non-positive sometimes, resulting in numerical instability. In [20], a splitting technique is introduced to resolve this problem effectively. This technique is used in all numerical examples presented in Sect. 5, but we will not discuss this technique in this paper due to limited space.

Tables of coefficients of the polynomial reconstruction on uniform meshes and the optimal weights were given in [6] for  $x = x_{j-\frac{1}{2}}^+$  and  $x_{j+\frac{1}{2}}^-$  up to  $r = 9$  for convenience of programming.

### 3.2.3 Non-linear Weights and Smooth Indicators

The stability is realized by giving the smoother stencil more weight, resulting in an approximation to ENO schemes, which use only the smoothest stencil. Assuming  $\beta_l$  to be the quantification of the smoothness of stencil  $l$ , which is called smoothness indicator, the nonlinear weights are obtained by

$$\omega_l = \frac{\alpha_l}{\sum_{s=0}^{r-1} \alpha_s}, \quad l = 0, \dots, r - 1, \tag{3.5}$$

$$\alpha_l = \frac{c_l}{(\epsilon + \beta_l)^p}, \quad l = 0, \dots, r - 1 \tag{3.6}$$

where  $c_l$  is the optimal weight. The relationship between the exponent  $p$  and the order  $r$  was discussed in [6], with the conclusion that for  $r \leq 5$ ,  $p = 2$  is an appropriate choice, and  $p = r$  is recommended otherwise.

The smooth indicator  $\beta_l$  is computed as in [12]:

$$\beta_l = \sum_{m=1}^{r-1} \Delta x^{2m-1} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \left( \frac{\partial^m p_l(x)}{\partial x^m} \right)^2 dx, \tag{3.7}$$

where  $p_l(x)$  is the ENO polynomial reconstruction on stencil  $l$ .

### 3.2.4 Mapped WENO Schemes

In [8], sufficient conditions to obtain uniform  $(2r - 1)$ th order are formulated as

$$\omega_l - c_l = \mathcal{O}(\Delta x^r), \quad l = 0, \dots, r - 1, \tag{3.8}$$

and mapped weights for the fifth order ( $r = 3$ ) are given to satisfy such conditions. They are computed by

$$\omega_{l,M} = \frac{\alpha_{l,M}}{\sum_{s=0}^{r-1} \alpha_{s,M}}, \tag{3.9}$$

$$\alpha_{l,M} = g(\omega_l, c_l), \tag{3.10}$$

$$g(a, b) = \frac{a(b + b^2 - 3ab + a^2)}{b^2 + a(1 - 2b)}. \tag{3.11}$$

### 3.3 Multi-dimensional WENO Reconstruction

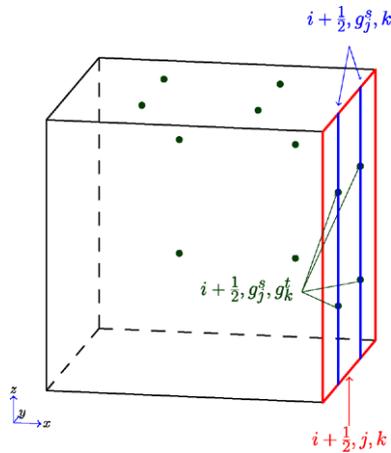
The WENO reconstruction on multi-dimensional Cartesian grids is much simpler than on general grids [22]. The procedure for the three-dimensional reconstruction can be summarized as follows.

Step 1 Follow the 1D reconstruction procedure described in Sect. 3.2 in  $i$  direction to obtain face averages  $\{\bar{u}_{i+\frac{1}{2},j,k}\}$  from cell averages  $\{\bar{u}_{i,j,k}\}$ .

Step 2 Follow the same 1D procedure in  $j$  direction to obtain line averages  $\{\bar{u}_{i+\frac{1}{2},g_j^s,k}\}$  from face averages  $\{\bar{u}_{i+\frac{1}{2},j,k}\}$ , where  $g_j^s$  is the  $y$  coordinate of Gauss quadrature point.

Step 3 Follow the same 1D procedure in  $k$  direction to obtain point values  $\{u_{i+\frac{1}{2},g_j^s,g_k^t}\}$  from line averages  $\{\bar{u}_{i+\frac{1}{2},g_j^s,k}\}$ , where  $g_k^t$  is the  $z$  coordinate of Gauss quadrature point.

**Fig. 1** Gauss points on a three-dimensional Cartesian grid



Step 4 Repeat previous steps 1–3 in directions  $j$  and  $k$  to obtain  $\{u_{g_i^q, j+\frac{1}{2}, g_k^t}\}$  and  $\{u_{g_i^q, g_j^s, k+\frac{1}{2}}\}$ , respectively, where  $g_i^q$  is the  $x$  coordinate of Gauss quadrature point in  $i$  direction.

Notice that  $\{u_{i+\frac{1}{2}, g_j^s, g_k^t}\}$ ,  $\{u_{g_i^q, j+\frac{1}{2}, g_k^t}\}$  and  $\{u_{g_i^q, g_j^s, k+\frac{1}{2}}\}$  include all Gauss points at all boundaries of every Cartesian grid, as illustrated in Fig. 1.

### 4 Speed Up Strategy on WENO Schemes

In many applications, solutions are almost uniform in large portions of the computational domain, especially at the beginning of computation. So the computational cost associated with using full WENO reconstruction in these regions is superfluous. Some simplifications are proposed as following.

Our speed up strategy can be stated as follows. For  $(2r - 1)$ th order WENO reconstruction of variable  $u$  at any point  $x$  in cell  $T_j := [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$ , compute the undivided difference

$$d_k := |\bar{u}_k - \bar{u}_j|, \quad \text{for } k = j - r + 1, \dots, j + r - 1. \tag{4.1}$$

If

$$d_k < K \Delta x^{2r-1}, \quad \text{for } k = j - r + 1, \dots, j + r - 1, \tag{4.2}$$

holds (where  $K = \mathcal{O}(1)$  is a problem-dependent constant), then simply set the WENO reconstruction  $\tilde{u}_w(x)$  in (3.4) to  $\bar{u}_j$ , otherwise, do a conventional WENO reconstruction as before.

The following proposition guarantees the original  $(2r - 1)$ -order accuracy of the WENO scheme for smooth functions.

**Proposition 1** *Assuming  $u$  is smooth enough and the condition (4.2) is satisfied, then the following relation*

$$\tilde{u}(x) = \bar{u}_j + \mathcal{O}(\Delta x^{2r-1}), \tag{4.3}$$

*is true, where  $\tilde{u}(x)$  denotes the  $(2r - 1)$ th order polynomial reconstruction for  $u(x)$  in the interval  $[x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$ .*

*Proof* First let

$$\begin{aligned}
 c_0 &= x_{j-\frac{1}{2}}, \\
 c_1 &= x_{j+\frac{1}{2}}, \\
 c_{i+2} &= x_{j-\frac{3}{2}-i}, \quad \text{for } 0 \leq i \leq r-2, \\
 c_{i+r+1} &= x_{j+\frac{3}{2}+i}, \quad \text{for } 0 \leq i \leq r-2,
 \end{aligned}$$

for ease of presentation. Consider the primitive function  $U$  of  $u$  as

$$U(x) = \int_{-\infty}^x u(y)dy. \tag{4.4}$$

The polynomial reconstruction for  $u$  on  $2r-1$  cells is equivalent to the Newton interpolation for primitive function  $U$  on  $2r$  end points due to  $u(x) = U'(x)$  [22]. Let  $\tilde{U}$  denote the Newton interpolation polynomial for  $U$ , then  $\tilde{U}$  can be written as the divided-difference form:

$$\begin{aligned}
 \tilde{U}(x) &= U(x_{j-r+\frac{1}{2}}) + \sum_{i=1}^{2r-1} V[c_0, \dots, c_i] \prod_{k=0}^{i-1} (x - c_k) \\
 &\quad + R(x) \prod_{k=0}^{2r-1} (x - c_k),
 \end{aligned} \tag{4.5}$$

where the third term is the  $\mathcal{O}(\Delta x^{2r})$  error term for the Newton interpolation and the recursive definition of the divided-difference is

$$V[a_1, a_2, \dots, a_n] = \frac{V[a_2, \dots, a_n] - V[a_1, \dots, a_{n-1}]}{a_n - a_1}. \tag{4.6}$$

Notice that condition (4.2) implies

$$V[c_0, \dots, c_i] = \mathcal{O}(\Delta x^{2r-i}), \quad \forall i \geq 2 \tag{4.7}$$

in accordance with (4.4) and (4.6). Differentiating (4.5) yields to

$$\begin{aligned}
 \tilde{u}(x) &= \sum_{i=1}^{2r-1} V[c_0, \dots, c_i] \left[ \prod_{k=0}^{i-1} (x - c_k) \right]_x \\
 &\quad + \left[ R(x) \prod_{k=0}^{2r-1} (x - x_{j-r+\frac{1}{2}+k}) \right]_x,
 \end{aligned} \tag{4.8}$$

and from (4.7), each  $i \geq 2$  term has a quantity of  $\mathcal{O}(\Delta x^{2r-i}) \times \mathcal{O}(\Delta x^{i-1}) = \mathcal{O}(\Delta x^{2r-1})$ , thus the equation

$$\begin{aligned}
 \tilde{u}(x) &= V[x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}] \left( x - x_{j-\frac{1}{2}} \right)_x + \mathcal{O}(\Delta x^{2r-1}) \\
 &= \bar{u}_j + \mathcal{O}(\Delta x^{2r-1}),
 \end{aligned} \tag{4.9}$$

is obtained. □

The free non-negative parameter  $K$  in (4.2) will depend on the spatial derivatives of the solution. However, present study shows that neither the numerical solution nor the required computational time is sensitive to this parameter. The numerical test in Sect. 5.1 will verify this point.

### 5 Numerical Examples

#### 5.1 Accuracy Test

Two equations are considered in accuracy tests. The first one is the scalar linear equation:

$$\frac{\partial u}{\partial t} + 2 \frac{\partial u}{\partial x} = 0, \tag{5.1}$$

$$u(x, 0) = e^{-x^2} \sin(x). \tag{5.2}$$

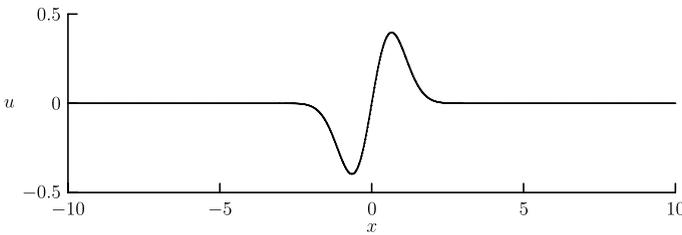
The initial condition is illustrated in Fig. 2. The accuracy and computational time at  $t = 1$  are listed in Tables 1 and 2, where SWENO and SWENOM denote the speed-up WENO and WENOM schemes, respectively, and  $K = 0.5$  is used.

It can be seen from Table 1 that the accuracy of the WENO and SWENO schemes is degenerated. Notice that the derivatives of function  $f(x) = e^{-x^2} \sin x$  are listed as

$$f'(x) = e^{-x^2} [-2x \sin x + \cos x], \tag{5.3}$$

$$f''(x) = e^{-x^2} [(4x^2 - 3) \sin x - 4x \cos x], \tag{5.4}$$

$$f'''(x) = e^{-x^2} [(-8x^3 + 18x) \sin x + 4x \cos x], \tag{5.5}$$



**Fig. 2** Initial conditions for the accuracy test of the scalar linear equation

**Table 1**  $L^\infty$  error and order of accuracy for the accuracy test of the scalar linear equation

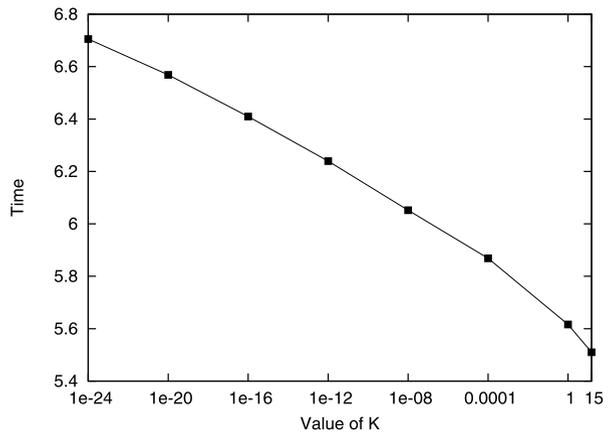
Grids		200	400	800
WENO	$L^\infty$ error	1.3754e-03	9.2553e-05	6.9870e-06
	order		3.89	3.73
SWENO	$L^\infty$ error	1.3754e-03	9.2553e-05	6.9870e-06
	order		3.89	3.73
WENOM	$L^\infty$ error	1.5206e-04	3.7115e-06	1.0395e-07
	order		5.36	5.16
SWENOM	$L^\infty$ error	1.5206e-04	3.7115e-06	1.0395e-07
	order		5.36	5.16

**Table 2** Computational time (seconds) and efficiency for the accuracy test of the scalar linear equation

Grids	WENO	SWENO	Time ratio	WENOM	SWENOM	Time ratio
200	0.167	0.136	81.4%	0.175	0.148	84.6%
400	0.850	0.674	79.3%	0.938	0.697	74.3%
800	5.457	3.953	72.4%	5.813	4.238	72.9%

**Table 3** Relationship between errors and the parameter  $K$ 

$K$	Error
1.0E-24	1.03947e-07
1	1.03947e-07
5	1.03947e-07
10	1.03947e-07
12	1.14143e-07
15	1.50095e-07

**Fig. 3** Relationship between time cost and the parameter  $K$ 

which means many critical points with  $f' = 0$  and  $f''' \neq 0$ . This is in agreement with the conclusion in [8], which states that the accuracy degenerates at such kind of points. The speed-up strategy does not change the data at all. Nevertheless, it effectively reduces the computational time by about 20% as seen from Table 2. Furthermore, it can reduce more time on finer meshes.

It will also be worth to see the impact on speed-up effectiveness and accuracy by choosing different values of  $K$ . The numerical tests on 800 grids verify that the errors do not change in six significant decimals from  $K = 0$  to  $K = 10$ , and increase only when  $K > 10$ , see Table 3. Figure 3 shows the relationship between time cost and the value of  $K$ . It can be seen that the time cost depends almost linearly on the log scale of  $K$ . This means that the CPU time essentially does not change much for a large range of  $K$ . Therefore, it is safe

**Table 4**  $L^\infty$  error and order of accuracy for the accuracy test of the Burger’s equation

Grids		200	400	600	800
WENOM	$L^\infty$ error	2.485e-05	3.013e-06	3.376e-07	6.174e-08
	order		3.04	5.40	5.91
SWENOM	$L^\infty$ error	2.485e-05	3.013e-06	3.376e-07	6.174e-08
	order		3.04	5.40	5.91

**Table 5** Computational time (seconds) and efficiency for the accuracy test of the Burger’s equation

Grids	WENOM	SWENOM	Time ratio
200	0.071	0.059	83.1%
400	0.235	0.178	75.7%
600	0.629	0.464	73.8%
800	1.291	0.975	75.5%

to say that both computational cost and numerical results are almost the same within a large range of  $K$ .

Secondly, the Burger’s equation is considered, i.e.

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x} \left( \frac{1}{2} u^2 \right) = 0, \tag{5.6}$$

where the initial condition is the mollifier function, i.e.

$$\eta(x) := \begin{cases} e^{\frac{1}{x^2-1}} & \text{if } |x| < 1, \\ 0 & \text{if } |x| \geq 1, \end{cases} \tag{5.7}$$

and computational domain is set to  $[-4, 4]$ . The computation is conducted until  $t = 0.2$  to show the accuracy and computational time, which are listed in Table 4 and 5 respectively. Only WENOM and SWENOM schemes with  $K = 1$  are used in this case. Figure 4 illustrates the numerical results at  $t = 1$  and  $t = 2$ .

### 5.2 One-Dimensional Riemann Problem

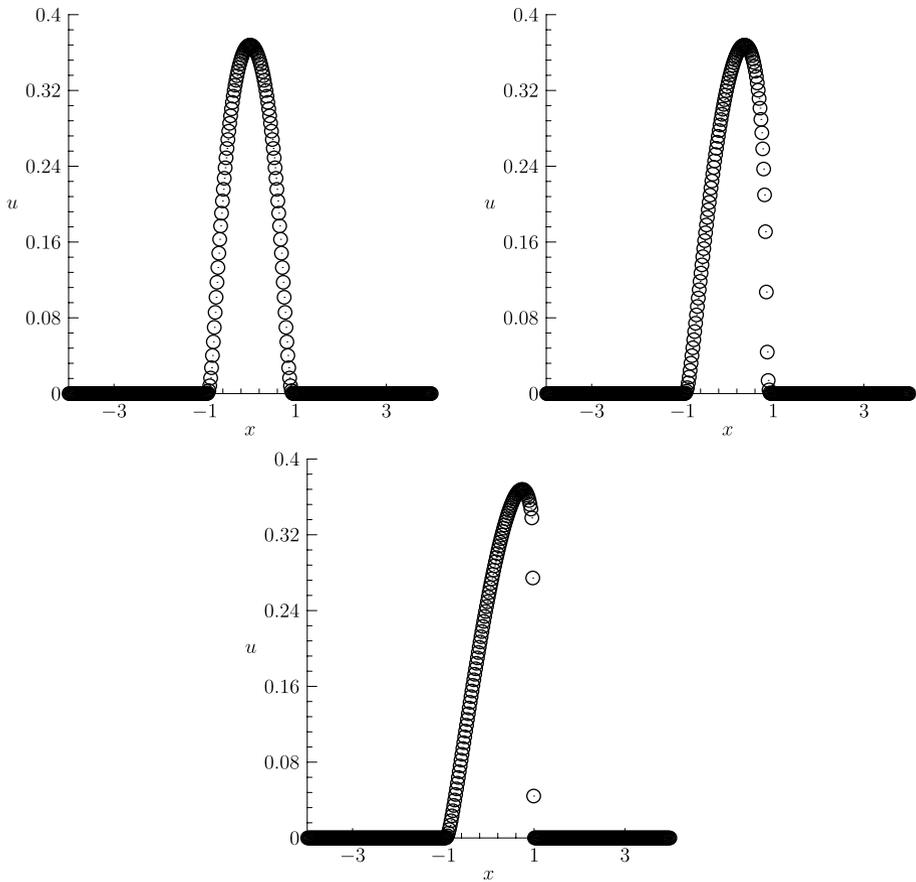
Hereafter, the Euler equations are considered, and only the numerical results computed with the SWENOM scheme are illustrated in the figures, for they are indistinguishable in these figures from those computed with the original SWENOM scheme. First, consider the 1D Euler equations

$$\frac{\partial}{\partial t} \begin{pmatrix} \rho \\ \rho u \\ \rho E \end{pmatrix} + \frac{\partial}{\partial x} \begin{pmatrix} \rho u \\ \rho u^2 + p \\ u(\rho E + p) \end{pmatrix} = 0, \tag{5.8}$$

where

$$\rho E = \frac{p}{\gamma - 1} + \frac{1}{2} \rho u^2, \tag{5.9}$$

with specific ratio  $\gamma = 1.4$ . The initial conditions are Sod’s shock tube problem, i.e.



**Fig. 4** Solutions for the accuracy test of the Burger’s equation on 400 grids. *Up left: t = 0; Up right: t = 1; Down: t = 2*

$$(\rho, u, p) = \begin{cases} (1, 0, 1) & \text{if } x \leq 0, \\ (0.125, 0, 0.1) & \text{if } x > 0. \end{cases} \tag{5.10}$$

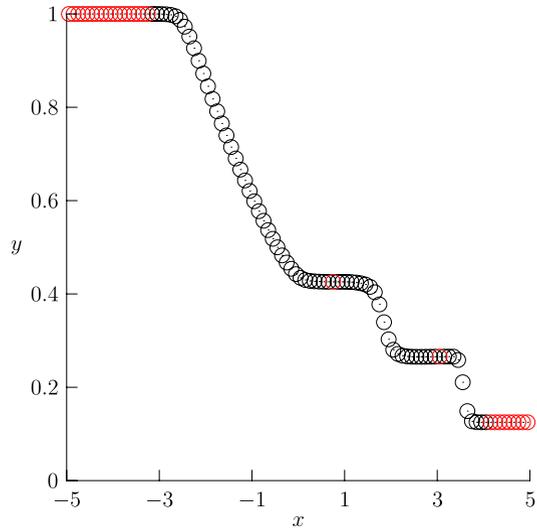
The computed density distribution by the SWENOM scheme is shown in Fig. 5, where the red circles indicate the constant region decided by the speed-up strategy with  $K = 1$ . and the computational costs are listed in Table 6. Again, the speedup strategy can reduce computational time by 20–30%, and can reduce more on finer grids.

### 5.3 Two-Dimensional Riemann Problem

Next, consider the two-dimensional Euler equations:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}_1(\mathbf{U})}{\partial x} + \frac{\partial \mathbf{F}_2(\mathbf{U})}{\partial y} = 0, \tag{5.11}$$

**Fig. 5** Density-space profile at  $t = 2$  for Sod’s shock tube problem with 100 grids, CFL=0.8 and local Lax-Friedrichs flux



**Table 6** Computational time and efficiency for the Sod’s shock tube problem

Grids	WENOM	SWENOM	Time ratio
200	0.464	0.347	74.8%
400	1.745	1.257	72.0%
800	6.759	4.865	72.0%

where

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{pmatrix}, \quad \mathbf{F}_1(\mathbf{U}) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(\rho E + p) \end{pmatrix}, \quad \mathbf{F}_2(\mathbf{U}) = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v(\rho E + p) \end{pmatrix}, \quad (5.12)$$

and

$$\rho E = \frac{p}{\gamma - 1} + \frac{1}{2} \rho (u^2 + v^2). \quad (5.13)$$

The initial conditions of the Riemann problem can be written as

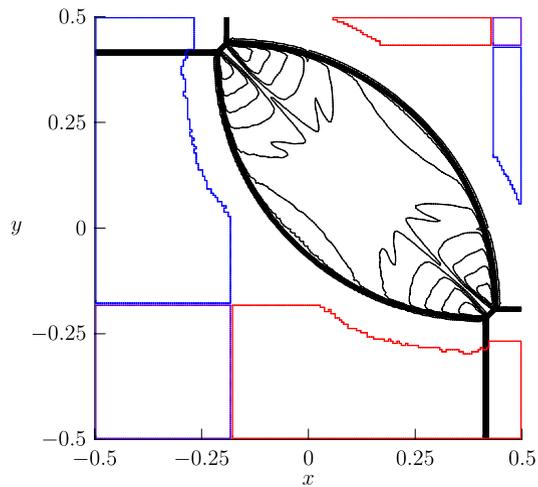
$$(\rho, u, v, p) = \begin{cases} (\rho_1, u_1, v_1, p_1), & \text{if } x > 0.5 \text{ and } y > 0.5, \\ (\rho_2, u_2, v_2, p_2), & \text{if } x \leq 0.5 \text{ and } y > 0.5, \\ (\rho_3, u_3, v_3, p_3), & \text{if } x \leq 0.5 \text{ and } y \leq 0.5, \\ (\rho_4, u_4, v_4, p_4), & \text{if } x > 0.5 \text{ and } y \leq 0.5. \end{cases} \quad (5.14)$$

Solutions of this problem depend only on configurations of the four phases. In [14], all 19 configurations are considered. However, in order to show just the speed-up efficiency, only three configurations listed in Table 7 are computed in this paper, and corresponding results are shown in Figs. 6, 7, and 8 with  $K = 10$ , where the areas with blue, red and purple boundaries are detected as constant solution areas in  $x-$ ,  $y-$  and both directions, respectively.

**Table 7** Three configurations for 2D Riemann problem

Conf.	Phase	$\rho$	$u$	$v$	$p$
1	1	1.1	0	0	1.1
	2	0.5065	0.8939	0	0.35
	3	1.1	0.8939	0.8939	1.1
	4	1.1	0.8939	0.8939	1.1
2	1	0.5197	0.1	0.1	0.4
	2	1	-0.6259	0.1	1
	3	0.8	0.1	0.1	1
	4	1	0.1	-0.6259	1
3	1	0.5313	0.1	0.1	0.4
	2	1.0222	-0.6179	0.1	1
	3	0.8	0.1	0.1	1
	4	1	0.1	0.8276	1

**Fig. 6** Density contours at  $t = 0.25$  for configuration 1 of 2D Riemann problem with  $200 \times 200$  grids, CFL = 0.6, and HLLC flux



The computational costs are tabulated in Table 8.

We can see from Figs. 6, 7, and 8 that configurations 1 and 3 have larger constant solution regions, while configuration 2 has smaller constant solution regions. Therefore, the time ratios of SWENOM to WENOM in Table 8 correctly reflect their dependence on the relative sizes of the nearly constant solution domain to the whole domain  $[-0.5, 0.5] \times [-0.5, 0.5]$ .

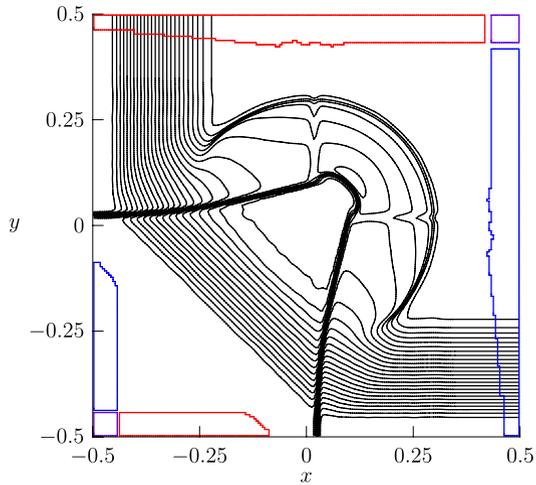
### 5.4 Double Mach Reflection

This is a standard two-dimensional test problem on compressible gas dynamics. The detailed description is given in [28]. The domain for computation is  $[0, 4] \times [0, 1]$ . Initially a Mach 10 shock makes a  $60^\circ$  angle with  $x$ -axis, positioned at  $x = \frac{1}{6}, y = 0$  and moving right. The exact states of the shock's two sides are

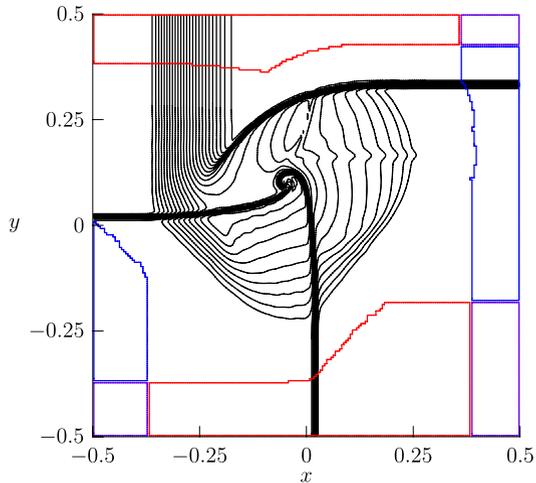
$$(\rho_l, u_l, v_l, p_l) = (8, 8.25 \cos \theta, -8.25 \sin \theta, 116.5), \tag{5.15}$$

$$(\rho_r, u_r, v_r, p_r) = (1.4, 0, 0, 1), \tag{5.16}$$

**Fig. 7** Density contours at  $t = 0.25$  for configuration 2 of 2D Riemann problem with  $200 \times 200$  grids, CFL = 0.6, and HLLC flux



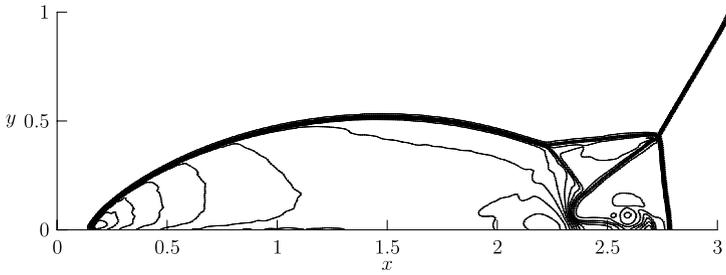
**Fig. 8** Density contours at  $t = 0.2$  for configuration 3 of 2D Riemann problem with  $200 \times 200$  grids, CFL = 0.6, and HLLC flux



**Table 8** Computing time of 2D Riemann problem with  $200 \times 200$  grids

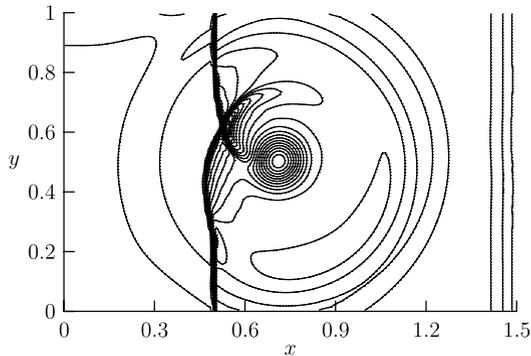
Conf.	WENOM	SWENOM	Time ratio
1	739.718	478.360	64.7%
2	513.681	396.443	77.2%
3	488.691	339.743	69.5%

with  $\theta = 30^\circ$ . At the left and top boundary, the exact states of the moving shock are set. At the bottom boundary, the exact post-shock state is set from  $x = 0$  to  $x = \frac{1}{6}$  and the reflective boundary condition is used for the rest part. The computed density contours are shown in Fig. 9. The overall computational time is reduced by 33% with  $K = 0.5$  for this problem.



**Fig. 9** Density contours at  $t = 0.2$  for double Mach reflection, with  $480 \times 120$  grids,  $Ma = 10$ ,  $CFL = 0.6$ , and local Lax-Friedrichs flux

**Fig. 10** Density contours at  $t = 0.4$  for shock vortex interaction, with  $200 \times 100$  grids,  $CFL = 0.6$ , and HLLC flux



### 5.5 Tow-Dimensional Shock Vortex Interaction

This example is often used to show the advantages of high-order methods. The computational domain is  $[0, 2] \times [0, 1]$ . At position  $x = 0.5$ , a vertical stationary shock is set with Mach number 1.1. A vortex is initialized at  $(x_c, y_c) = (0.25, 0.5)$ . The vortex is described as a perturbation to the initial condition, written as

$$\tilde{u} = \varepsilon \tau e^{\alpha(1-\tau^2)} \sin \theta, \tag{5.17}$$

$$\tilde{v} = -\varepsilon \tau e^{\alpha(1-\tau^2)} \cos \theta, \tag{5.18}$$

$$\tilde{T} = -\frac{(\gamma - 1)\varepsilon^2 e^{2\alpha(1-\tau^2)}}{4\alpha\gamma}, \tag{5.19}$$

$$\tilde{S} = 0, \tag{5.20}$$

where  $\tau = \frac{r}{r_c}$  and  $r = \sqrt{(x - x_c)^2 + (y - y_c)^2}$ . Here  $\tilde{T}$  is the perturbation to temperature ( $T = \frac{p}{\rho}$ ) and  $\tilde{S}$  is the perturbation to entropy ( $S = \ln \frac{p}{\rho^\gamma}$ ), see [22] for a detailed description. The numerical results are illustrated in Fig. 10. We record 20.0% time is saved with  $K = 10$  for this problem.

### 5.6 Explosion test in three space dimensions

Finally, a spherically symmetric problem is tested to show the speed-up efficiency in three dimensions. The problem is described in detail in Chap. 17 of [25]. The three-dimensional

Euler equations are written as

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}_1(\mathbf{U})}{\partial x} + \frac{\partial \mathbf{F}_2(\mathbf{U})}{\partial y} + \frac{\partial \mathbf{F}_3(\mathbf{U})}{\partial z} = 0, \tag{5.21}$$

where

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{pmatrix}, \quad \mathbf{F}_1(\mathbf{U}) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ u(\rho E + p) \end{pmatrix}, \tag{5.22}$$

$$\mathbf{F}_2(\mathbf{U}) = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ v(\rho E + p) \end{pmatrix}, \quad \mathbf{F}_3(\mathbf{U}) = \begin{pmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ w(\rho E + p) \end{pmatrix},$$

and

$$\rho E = \frac{p}{\gamma - 1} + \frac{1}{2} \rho (u^2 + v^2 + w^2). \tag{5.23}$$

The solution domain is in the range of  $[-1, 1] \times [-1, 1] \times [-1, 1]$ , and the initial conditions which consist of a spherical discontinuity are

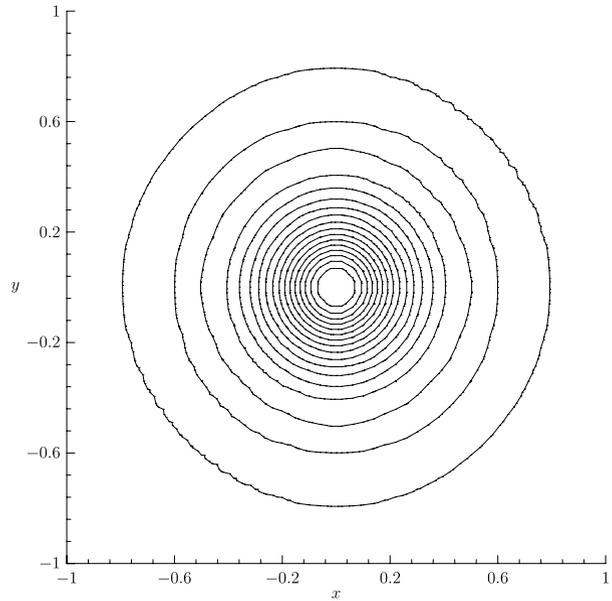
$$(\rho, u, v, w, p) = \begin{cases} (1.0, 0.0, 0.0, 0.0, 1.0) & \text{if } r \leq 0.4, \\ (0.125, 0.0, 0.0, 0.0, 0.1) & \text{if } r > 0.4, \end{cases} \tag{5.24}$$

where  $r = \sqrt{x^2 + y^2 + z^2}$ , and  $50 \times 50 \times 50$  uniform grids are used. Computed density and pressure distributions with the SWENOM scheme are illustrated in Figs. 11 and 12 respectively. 47% computational time is saved with  $K = 0.5$ . Thus we see the speed-up effect is more evident in high spatial dimensions. However, this effect relies on the relative size of nearly constant solution region to the whole domain, which is problem dependent.

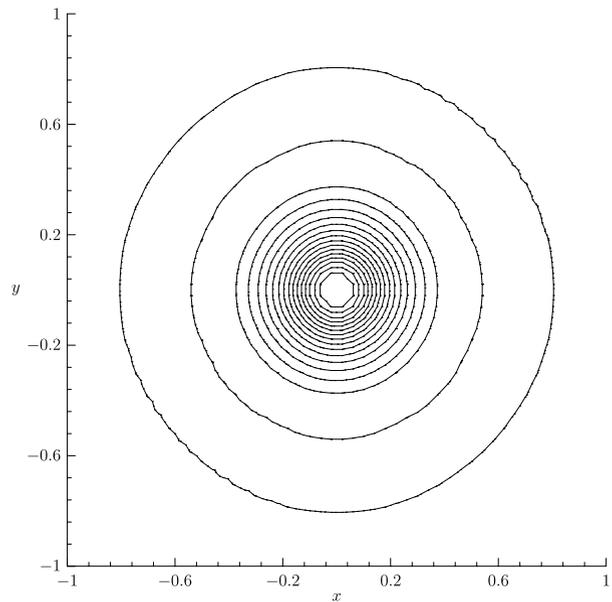
### 6 Conclusions

In this paper, a p-adaptive type speedup strategy for the finite volume WENO scheme is proposed. The reconstruction polynomial will adjust from the original WENO polynomial to the cell average constant when nearly constant solution regions are detected by using undivided differences of cell averages. The strategy is implemented with both original WENO and mapped WENO schemes. Both theoretical analysis and numerical tests demonstrate that the desired order of accuracy is maintained as long as the nearly constant solution condition is satisfied. The numerical results show the proposed strategy can reduce the computational time of the WENO schemes by about 20–40%, and this effect can be more pronounced in higher spatial dimensions.

**Fig. 11** Density contours at  $t = 0.25$  and  $z = 0$  for explosion test in 3D with  $50 \times 50 \times 50$  grids, CFL = 0.15, and local Lax-Friedrichs flux



**Fig. 12** Pressure contours at  $t = 0.25$  and  $z = 0$  for explosion test in 3D with  $50 \times 50 \times 50$  grids, CFL = 0.15, and local Lax-Friedrichs flux



**Acknowledgements** This work is supported by Natural Science Foundation of China (10531080, 10729101, 10972230) and state key program for developing basic sciences (2005CB321703, 2010CB731505).

## References

1. Balsara, D., Shu, C.-W.: Monotonicity preserving weighted essentially non-oscillatory schemes with increasingly high order of accuracy. *J. Comput. Phys.* **160**, 405–452 (2000)
2. Barth, T.: Numerical methods for conservation laws on structured and unstructured meshes. VKI March 2003 Lecture Series
3. Borges, R., Carmona, M., Costa, B., Don, W.S.: An improved weighted essentially non-oscillatory scheme for hyperbolic conservation laws. *J. Comput. Phys.* **227**, 3191–3211 (2008)
4. Cheng, J., Shu, C.-W.: High order schemes for CFD: a review, Chinese. *J. Comput. Phys.* **26**, 633–655 (2009)
5. Deng, X., Zhang, H.: Developing high-order weighted compact nonlinear schemes. *J. Comput. Phys.* **165**, 22–44 (2000)
6. Gerolymos, G.A., Sénéchal, D., Vallet, I.: Very-high-order WENO schemes. *J. Comput. Phys.* **228**, 8481–8524 (2009)
7. Harten, A., Engquist, B., Osher, S., Chakravarthy, S.: Uniformly high order essentially non-oscillatory schemes, III. *J. Comput. Phys.* **71**, 279–309 (1987)
8. Henrick, A.K., Aslam, T.D., Powers, J.M.: Mapped weighted essentially non-oscillatory schemes: achieving optimal order near critical points. *J. Comput. Phys.* **207**, 542–567 (2005)
9. Hu, C., Shu, C.-W.: Weighted essentially non-oscillatory schemes on triangular meshes. *J. Comput. Phys.* **150**, 97–127 (1999)
10. Huang, L., Shu, C.-W., Zhang, M.: Numerical boundary conditions for the fast sweeping high order WENO methods for solving the Eikonal equation. *J. Comput. Math.* **26**, 336–346 (2008)
11. Jiang, G.-S., Peng, D.: Weighted ENO schemes for Hamilton-Jacobi equations. *SIAM J. Sci. Comput.* **21**, 2126–2143 (2000)
12. Jiang, G., Shu, C.-W.: Efficient implementation of weighted ENO schemes. *J. Comput. Phys.* **126**, 202–228 (1996)
13. Kröner, D., Ohlberger, M.: A posteriori error estimates for upwind finite volume schemes for nonlinear conservation laws in multi dimensions. *Math. Comput.* **69**, 25–39 (2000)
14. Kurganov, A., Tadmor, E.: Solution of two-dimensional Riemann problems for gas dynamics without Riemann problem solvers. *Numer. Methods Partial Differ. Equ.* **18**, 584–608 (2002)
15. Liu, X.-D., Osher, S., Chan, T.: Weighted essentially nonoscillatory schemes. *J. Comput. Phys.* **115**, 200–212 (1994)
16. Qiu, J., Shu, C.-W.: Hermite WENO schemes and their application as limiters for Runge-Kutta discontinuous Galerkin method: one dimensional case. *J. Comput. Phys.* **193**, 115–135 (2003)
17. Qiu, J., Shu, C.-W.: Hermite WENO schemes and their application as limiters for Runge-Kutta discontinuous Galerkin method II: two dimensional case. *Comput. Fluids* **34**, 642–663 (2005)
18. Roe, P.L.: Approximate Riemann solvers, parameter vectors, and difference schemes. *J. Comput. Phys.* **43**, 357–372 (1981)
19. Shen, Y., Yang, G.: Hybrid finite compact-WENO schemes for shock calculation. *Int. J. Numer. Methods Fluids* **53**, 531–560 (2007)
20. Shi, J., Hu, C., Shu, C.-W.: A technique of treating negative weights in WENO schemes. *J. Comput. Phys.* **175**, 108–127 (2002)
21. Shu, C.-W.: Numerical experiments on the accuracy of ENO and modified ENO schemes. *J. Sci. Comput.* **5**, 127–149 (1990)
22. Shu, C.-W.: Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. ICASE Report, No. 97-65, pp. 127–149 (1997)
23. Shu, C.-W., Osher, S.: Efficient implementation of essentially non-oscillatory shock-capturing schemes. *J. Comput. Phys.* **77**, 439–471 (1988)
24. Shu, C.-W., Osher, S.: Efficient implementation of essentially non-oscillatory shock capturing schemes II. *J. Comput. Phys.* **83**, 32–78 (1989)
25. Toro, E.F.: *Riemann Solvers and Numerical Methods for Fluid Dynamics—A Practical Introduction*, 3rd edn. Springer, Dordrecht (2009)
26. Toro, E.F., Spruce, M., Speares, W.: Restoration of the contact surface in the HLL-Riemann solver. *Shock Waves* **4**, 25–34 (1994)
27. Wolf, W.R., Azevedo, J.L.F.: High-order ENO and WENO schemes for unstructured grids. *Int. J. Numer. Methods Fluids* **55**, 917–943 (2007)
28. Woodward, P., Colella, P.: The numerical simulation of two-dimensional fluid flow with strong shocks. *J. Comput. Phys.* **54**, 115–173 (1984)

29. Zhang, Y.-T., Shu, C.-W.: High order WENO schemes for Hamilton-Jacobi equations on triangular meshes. *SIAM J. Sci. Comput.* **24**, 1005–1030 (2003)
30. Zhang, S., Shu, C.-W.: A new smoothness indicator for the WENO schemes and its effect on the convergence to steady state solutions. *J. Sci. Comput.* **31**, 273–305 (2007)
31. Zhang, Y.-T., Shu, C.-W.: Third order WENO scheme on three dimensional tetrahedral meshes. *Commun. Comput. Phys.* **5**, 836–848 (2009)
32. Zhang, S., Jiang, S., Shu, C.-W.: Development of nonlinear weighted compact schemes with increasingly higher order accuracy. *J. Comput. Phys.* **227**, 7294–7321 (2008)