Computers and Fluids 251 (2023) 105746

Contents lists available at ScienceDirect

Computers and Fluids

journal homepage: www.elsevier.com/locate/compfluid

Efficient implementation of high-order WENO schemes with sharing function for solving Euler equations

Shengping Liu [a], Yiqing Shen [b,c], Shaodong Guo [a], Heng Yong [a,*], Guoxi Ni [a]

[a] Institute of Applied Physics and Computational Mathematics, Beijing 100094, China
[b] State Key Laboratory of High Temperature Gas Dynamics, Institute of Mechanics, Chinese Academy of Sciences, Beijing 100190, China
[c] School of Engineering Science, University of Chinese Academy of Sciences, Beijing 100049, China

ARTICLE INFO

Keywords:
Euler equations
WENO
Sharing function
Common-weights

ABSTRACT

Due to the high-order accuracy and essentially non-oscillatory (ENO) property, the weighted ENO (WENO) schemes have a wide range of successful applications. The component-wise reconstruction WENO (CP WENO) scheme for fluxes or variables has simple formulations but it may produce numerical oscillations near discontinuities when solving the Euler equations. Although the characteristic-wise reconstruction WENO (CH WENO) scheme can reduce such oscillations, it involves too many characteristic projection operations. In this paper, first, we introduced a sharing function to indicate the discontinuities in the Euler equations and then constructed new adaptive characteristic-wise WENO (Ada-WENO) scheme and common-weights WENO (Co-WENO) scheme with this function. Several one and two dimensional problems are used to test the performances of Ada-WENO and Co-WENO. Numerical results show that, Ada-WENO can reduce the computational cost of CH WENO while maintaining its oscillation-free property since it only switches from CP WENO to CH WENO near discontinuities, and Co-WENO can reduce the cost and oscillations of CP WENO, but it may still generate few oscillations.

1. Introduction

Due to the high-order accuracy and the essentially non-oscillatory (ENO) property, the WENO schemes have been widely applied in many areas [1]. The WENO schemes' concept was firstly proposed by Liu et al. [2], and then greatly simplified and improved by Jiang and Shu [3] (WENO-JS). Many high order WENO schemes were constructed in the general framework of WENO-JS [4,5]. Henrick et al. [6] found that WENO-JS will lose its accuracy near critical points and proposed a mapping function to fix that. Later, Borges et al. [7] proposed a new weighting function (WENO-Z) to improve the accuracy with a lower computational cost. The WENO-Z scheme provides a straight and efficient formulation for calculating the weights, hence many improved WENO-Z-type schemes were proposed [8–18].

Although these WENO schemes perform well for scalar problems, they may produce numerical oscillations near discontinuities when solving Euler equations with the component-wise reconstruction procedure (denoted as CP WENO) for the fluxes or variables. The characteristic-wise reconstruction procedure (denoted as CH WENO) is often used to eliminate such oscillations, while CH WENO involves a lot of characteristic projection operations and hence is too costly. To reduce the computational cost of CH WENO, Jiang and Shu [3] tried to replace the weights which should be computed from projected values with those weights calculated directly from the entropy and pressure, while the resulting scheme (WENO-LF-5-PS) cannot handle problems containing strong shock and reflective waves. Later, based on a characteristic-wise hybridization, Hu et al. [19] proposed a simple hybrid WENO scheme, which can save the computational cost of the non-linear weights of the WENO scheme in smooth regions. While the discontinuity-detector used in this hybrid WENO scheme [19] is based on the characteristic-variables and hence still involves a lot of characteristic projection operations. Since the CP method is more computational efficient than the CH method, Puppo [20] proposed a indicators to switch from CP method to CH method, the resulting adaptive method showed good performance and efficiency. While this method requires the computation of the smoothness indicators for each flux component, and hence still needs more computational time than the CP method [21]. Peng et al. [21] proposed a set of sharing functions to indicate the discontinuities of the fluxes, and then constructed the adaptive characteristic-wise WENO (Ada-WENO) scheme, which only switches from CP WENO to CH WENO near discontinuities. The analysis and results [21] showed that Ada-WENO could achieve the oscillation-free property same as CH WENO while reducing its computational cost

* Corresponding author.
E-mail address: yong_heng@iapcm.ac.cn (H. Yong).

https://doi.org/10.1016/j.compfluid.2022.105746
Received 11 November 2021; Received in revised form 10 September 2022; Accepted 25 November 2022
Available online 29 November 2022

a lot. Although, Ada-WENO showed good performance and efficiency, we noticed that the sharing functions [21] derived from Lax–Friedrichs (LF) flux vector splitting (FVS) method have different dimension units. As is known, all equations must be dimensionally consistent (dimensional homogeneity), otherwise, some spurious phenomena will be generated (See [22] for an example), therefore the applications of this Ada-WENO scheme is limited.

To avoid the characteristic projection operations and reduce oscillations, He et al. [23] proposed to combine a global FVS with a consistent discretization between different equations. In [23], a consistent discretization method that uses one set of common weights is proposed for nonlinear WENO schemes. Shen et al. [24] call this kind of WENO schemes common-weights WENO (Co-WENO) schemes. Analysis and numerical results [23] showed that Co-WENO could reduce the oscillations of CP WENO. For Co-WENO, the key is how to calculate the common-weights. He et al. [23] calculated the common-weights from the smooth factors of mass and energy equations. Recently, Peng et al. [21] proposed to calculate the common-weights directly through a set of carefully designed sharing functions. Since only one set of smooth factors based on the sharing functions is needed, the new Co-WENO scheme [21] is more efficient than CP WENO. Although, as mentioned above, the sharing functions [21] have different units, lately, Shen et al. [24] derived new dimensionless sharing functions based on the Steger–Warming [25] FVS method, and the resulting Co-WENO scheme performs well.

Although both Ada-WENO and Co-WENO mentioned above can reduce the oscillations and computational cost of high-order WENO schemes when solving Euler equations, they [21,23,24] are all derived from the FVS methods, and not suitable for the high-resolution finite-difference splitting (FDS) method (or approximate Riemann Solvers, Roe [26], HLL [27], et al.). In this paper, we want to focus on the efficient implementation of high-order WENO schemes for the FDS methods. First, we designed a new simple sharing function independent of the FVS and FDS methods and then constructed new highly efficient Ada-WENO and Co-WENO schemes based on the classical fifth-order WENO-Z scheme [7] with the new sharing function.

This paper is organized as follows: the implementation of CP WENO and CH WENO with the FVS and FDS methods are introduced in Section 2; Ada-WENO and Co-WENO for the FVS methods proposed by Peng et al. [21] and Shen et al. [24] are introduced in Section 3. Section 4 introduced our new sharing function and the new Ada-WENO and Co-WENO schemes for the FDS method. Various numerical examples are presented in Section 5 to demonstrate the good performance of these new schemes. Concluding remarks are given in Section 6.

2. Solving the Euler equations with the WENO scheme

In this section, we briefly introduce the implementation of CP WENO and CH WENO with FVS and FDS methods to solve the one-dimensional Euler equations. Although, for simplicity, the fifth-order WENO-Z [7] is used in this paper, the implementation of other high-order WENO schemes is similar.

2.1. The WENO-Z scheme

For convenience, we review the WENO-Z scheme [7] by using the scalar conservative law equation [3],

$$\frac{\partial u}{\partial t}+\frac{\partial f}{\partial x}=0. \tag{1}$$

By defining the points $x_i = i\Delta x$, $(i = 0, \dots, N)$, where Δx is the uniform grid spacing, the Eq. (1) can be approximated by a conservative finite difference formula,

$$\frac{du_i}{dt}=-\frac{\hat{f}_{i+1/2}-\hat{f}_{i-1/2}}{\Delta x}, \tag{2}$$

where $\hat{f}_{i\pm1/2}$ is the numerical flux.

Generally, the numerical flux $\hat{f}_{i+1/2}$ of a fifth-order WENO-Z [7] scheme is calculated on a five-points stencil $S_{i+1/2} = (f_{i-2}, f_{i-1}, f_i, f_{i+1}, f_{i+2})$,

$$\hat{f}_{i+1/2}=P^{WENO-Z}(S_{i+1/2})=\sum_{k=0}^{2}\omega_k q_k, \tag{3}$$

where q_k are the third-order fluxes given by,

$$\begin{cases} q_0 = \frac{1}{3}f_{i-2}-\frac{7}{6}f_{i-1}+\frac{11}{6}f_i, \\ q_1 = -\frac{1}{6}f_{i-1}+\frac{5}{6}f_i+\frac{1}{3}f_{i+1}, \\ q_2 = \frac{1}{3}f_i+\frac{5}{6}f_{i+1}-\frac{1}{6}f_{i+2}. \end{cases} \tag{4}$$

The non-linear weights are calculated as follows,

$$\omega_k=\frac{\alpha_k}{\alpha_0+\alpha_1+\alpha_2},\ \alpha_k=c_k\left(1+\frac{\tau_5}{IS_k+\epsilon}\right),\ \epsilon=10^{-40}, \tag{5}$$

where, $c_0 = 0.1, c_1 = 0.6, c_2 = 0.3$ are the optimal weights and τ_5 is the global smoothness indicator,

$$\tau_5=\left|IS_2-IS_0\right|, \tag{6}$$

and IS_k are the local smoothness indicators,

$$\begin{cases} IS_0=\frac{13}{12}(f_{i-2}-2f_{i-1}+f_i)^2+\frac{1}{4}(f_{i-2}-4f_{i-1}+3f_i)^2, \\ IS_1=\frac{13}{12}(f_{i-1}-2f_i+f_{i+1})^2+\frac{1}{4}(-f_{i-1}+f_{i+1})^2, \\ IS_2=\frac{13}{12}(f_i-2f_{i+1}+f_{i+2})^2+\frac{1}{4}(-3f_i+4f_{i+1}-f_{i+2})^2. \end{cases} \tag{7}$$

2.2. The Euler equations

The one-dimensional Euler equations of the inviscid ideal gas are given by,

$$\frac{\partial \boldsymbol{U}}{\partial t}+\frac{\partial \boldsymbol{F}}{\partial x}=0, \tag{8}$$

in which $\boldsymbol{U}$ and $\boldsymbol{F}$ are the conserved variable and the convective flux,

$$\boldsymbol{U}=\begin{bmatrix}\rho\\ \rho u\\ E\end{bmatrix},\ \text{and}\ \ \boldsymbol{F}=\begin{bmatrix}\rho u\\ \rho u^2+p\\ u(E+p)\end{bmatrix} \tag{9}$$

where, ρ, u, E and p are density, velocity, total energy and pressure. The equation of state is given by $E=\frac{p}{\gamma-1}+\frac{1}{2}\rho u^2$ with $\gamma = 1.4$.

For the convective flux $\boldsymbol{F}$, its Jacobian matrix $\boldsymbol{A}$ is defined as,

$$\frac{\partial \boldsymbol{F}(\boldsymbol{U})}{\partial x}=\boldsymbol{A}\cdot\frac{\partial \boldsymbol{U}}{\partial x}. \tag{10}$$

where, $\boldsymbol{A}=\boldsymbol{R}\cdot\boldsymbol{\Lambda}\cdot\boldsymbol{L}$. And $\boldsymbol{L}$ and $\boldsymbol{R}$ are the left and right eigenvectors,

$$\boldsymbol{L}=\begin{bmatrix} \frac{\gamma-1}{4}\frac{u^2}{c^2}+\frac{1}{2}\frac{u}{c}, & -\frac{\gamma-1}{2}\frac{u}{c^2}-\frac{1}{2}\frac{1}{c}, & \frac{\gamma-1}{2}\frac{1}{c^2} \\ 1-\frac{\gamma-1}{2}\frac{u^2}{c^2}, & \frac{\gamma-1}{2}\frac{u}{c^2}, & -\frac{\gamma-1}{2}\frac{1}{c^2} \\ \frac{\gamma-1}{4}\frac{u^2}{c^2}-\frac{1}{2}\frac{u}{c}, & -\frac{\gamma-1}{2}\frac{u}{c^2}+\frac{1}{2}\frac{1}{c}, & \frac{\gamma-1}{2}\frac{1}{c^2} \end{bmatrix},$$
$$\boldsymbol{R}=\begin{bmatrix} 1, & 1, & 1 \\ u-c, & u, & u+c \\ \frac{u^2}{2}+\frac{c^2}{\gamma-1}-uc, & \frac{u^2}{2}, & \frac{u^2}{2}+\frac{c^2}{\gamma-1}+uc \end{bmatrix}. \tag{11}$$

where, $c=\sqrt{\gamma p/\rho}$ is the sound speed.

2.3. Implementation with flux vector splitting method

With a FVS method (LF [3], Steger–Warming [25], and so on), we can split the nodal flux $\boldsymbol{F}_i$ into the positive and negative parts,

$$\boldsymbol{F}_i=\boldsymbol{F}_i^{+}+\boldsymbol{F}_i^{-}. \tag{12}$$

And then calculate the positive and negative numerical fluxes via the WENO-Z scheme to obtain the numerical flux,

$$\boldsymbol{F}_{i+1/2} = \boldsymbol{F}^{+}_{i+1/2} + \boldsymbol{F}^{-}_{i+1/2}. \tag{13}$$

For CP WENO-Z, we can implement Eq. (3) directly on stencil $S^{+}_{i+1/2} = (\boldsymbol{F}^{+}_{i-2}, \boldsymbol{F}^{+}_{i-1}, \boldsymbol{F}^{+}_{i}, \boldsymbol{F}^{+}_{i+1}, \boldsymbol{F}^{+}_{i+2})$ (i.e., implement the WENO reconstruction procedure for each component in $\boldsymbol{F}^{+}_{i}$ one by one) to obtain the positive numerical flux,

$$\boldsymbol{F}^{CP+}_{i+1/2} = P^{WENO-Z}(S^{+}_{i+1/2}), \tag{14}$$

For CH WENO-Z, we should first calculate the left and right eigenvectors ($\boldsymbol{L}$ and $\boldsymbol{R}$) on $x_{i+1/2}$ from the Roe-averaged primitive variables (See [21,28] for details), and then project the positive nodal fluxes on stencil $S^{+}_{i+1/2}$ to the characteristic field,

$$\boldsymbol{G}^{+}_{j} = \boldsymbol{L} \cdot \boldsymbol{F}^{+}_{j}, j = i-2, \dots, i+2. \tag{15}$$

Thus, we can implement Eq. (3) on the projected stencil $\hat{S}^{+}_{i+1/2} = (\boldsymbol{G}^{+}_{i-2}, \boldsymbol{G}^{+}_{i-1}, \boldsymbol{G}^{+}_{i}, \boldsymbol{G}^{+}_{i+1}, \boldsymbol{G}^{+}_{i+2})$ to compute the projected positive numerical flux,

$$\boldsymbol{G}^{+}_{i+1/2} = P^{WENO-Z}(\hat{S}^{+}_{i+1/2}), \tag{16}$$

Finally, we need to transform this back to the physical space to obtain the positive numerical flux,

$$\boldsymbol{F}^{CH+}_{i+1/2} = \boldsymbol{R} \cdot \boldsymbol{G}^{+}_{i+1/2}. \tag{17}$$

The negative numerical fluxes $\boldsymbol{F}^{CP-}_{i+1/2}$ and $\boldsymbol{F}^{CH-}_{i+1/2}$ on stencil $S^{-}_{i+1/2} = (\boldsymbol{F}^{-}_{i+3}, \boldsymbol{F}^{-}_{i+2}, \boldsymbol{F}^{-}_{i+1}, \boldsymbol{F}^{-}_{i}, \boldsymbol{F}^{-}_{i-1})$ can be calculated with the same equations similarly.

2.4. Implementation with finite-difference splitting method

Given the left-side value $\boldsymbol{U}^{L}_{i+1/2}$ and right-side value $\boldsymbol{U}^{R}_{i+1/2}$ from the conserved variable $\boldsymbol{U}_i$, we can use a FDS method to calculate the numerical flux, for example, the Roe solver [26],

$$\begin{aligned}\boldsymbol{F}_{i+1/2} = \boldsymbol{F}(\boldsymbol{U}^{L}_{i+1/2}, \boldsymbol{U}^{R}_{i+1/2}) = \frac{1}{2}\left(\boldsymbol{F}(\boldsymbol{U}^{L}_{i+1/2}) + \boldsymbol{F}(\boldsymbol{U}^{R}_{i+1/2})\right) \\ - \frac{|\boldsymbol{\Lambda}|_{i+1/2}}{2}\left(\boldsymbol{U}^{R}_{i+1/2} - \boldsymbol{U}^{L}_{i+1/2}\right).\end{aligned} \tag{18}$$

For CP WENO-Z, $\boldsymbol{U}^{L}_{i+1/2}$ is calculated by Eq. (3) on stencil $S^{L}_{i+1/2} = (\boldsymbol{U}_{i-2}, \boldsymbol{U}_{i-1}, \boldsymbol{U}_{i}, \boldsymbol{U}_{i+1}, \boldsymbol{U}_{i+2})$,

$$\boldsymbol{U}^{CPL}_{i+1/2} = P^{WENO-Z}(S^{L}_{i+1/2}), \tag{19}$$

and $\boldsymbol{U}^{R}_{i-1/2}$ is calculated by Eq. (3) on stencil $S^{R}_{i-1/2} = (\boldsymbol{U}_{i+2}, \boldsymbol{U}_{i+1}, \boldsymbol{U}_{i}, \boldsymbol{U}_{i-1}, \boldsymbol{U}_{i-2})$. We noticed that stencils $S^{L}_{i+1/2}$ and $S^{R}_{i-1/2}$ share the same nodal values, and the local smoothness indicators in Eq. (7) is symmetric with respect to node x_i, hence the indicators $(IS_0, IS_1, IS_2, \tau_5)$ calculated on $S^{L}_{i+1/2}$ for $\boldsymbol{U}^{L}_{i+1/2}$ will be equal to the ones $(IS_2, IS_1, IS_0, \tau_5)$ calculated on $S^{R}_{i-1/2}$ for $\boldsymbol{U}^{R}_{i-1/2}$, that means,

$$\begin{bmatrix} IS_0 \\ IS_1 \\ IS_2 \\ \tau_5 \end{bmatrix}_{\boldsymbol{U}^{R}_{i-1/2}} = \begin{bmatrix} IS_2 \\ IS_1 \\ IS_0 \\ \tau_5 \end{bmatrix}_{\boldsymbol{U}^{L}_{i+1/2}}, \tag{20}$$

and hence CP WENO-Z with the FDS method can save a lot of computational cost when computing the right-side value $\boldsymbol{U}^{CPR}_{i+1/2}$.

For CH WENO-Z, we transform the conserved variables $\boldsymbol{U}_i$ on stencil $S^{L}_{i+1/2}$ to the local characteristic field,

$$\boldsymbol{V}_{j} = \boldsymbol{L} \cdot \boldsymbol{U}_{j}, \ j = i-2, \dots, i+2, \tag{21}$$

and then implement Eq. (3) on the projected stencil $\hat{S}^{L}_{i+1/2} = (\boldsymbol{V}_{i-2}, \boldsymbol{V}_{i-1}, \boldsymbol{V}_{i}, \boldsymbol{V}_{i+1}, \boldsymbol{V}_{i+2})$,

$$\boldsymbol{V}^{L}_{i+1/2} = P^{WENO-Z}(\hat{S}^{L}_{i+1/2}). \tag{22}$$

Finally, we project this back to obtain the left-side value,

$$\boldsymbol{U}^{CHL}_{i+1/2} = \boldsymbol{R} \cdot \boldsymbol{V}^{L}_{i+1/2}. \tag{23}$$

The right-side value $\boldsymbol{U}^{CHR}_{i+1/2}$ on stencil $S^{R}_{i+1/2} = (\boldsymbol{U}_{i+3}, \boldsymbol{U}_{i+2}, \boldsymbol{U}_{i+1}, \boldsymbol{U}_{i}, \boldsymbol{U}_{i-1})$ is computed with the same equations in a similar way.

3. Ada-WENO and Co-WENO for the FVS method

In this section, we briefly introduce the sharing functions and the Ada-WENO and Co-WENO schemes derived for the FVS method in [21,24].

3.1. The sharing functions

To indicate the discontinuities of all the components of the positive and negative fluxes in Eq. (12), Peng et al. designed the sharing functions,

$$H^{\pm} = \rho + (\rho u^2 + p \pm \alpha \rho u), \tag{24}$$

where, α is a parameter computed by the LF FVS method, $\pm$ represents for the positive and negative functions, respectively (See [21] for more information). It can be found that these sharing functions do not meet the principle of dimensional homogeneity (ρ and ρu^2 have different units), and hence the Co-WENO scheme and Ada-WENO schemes based on these functions may encounter severe problems in applications [22]. Lately, Shen et al. [24] derived a set of new dimensionless sharing functions for the Steger–Warming FVS method.

$$H^{\pm} = \rho \cdot p \cdot f^{\pm}_{E}, \tag{25}$$

where, $f^{\pm}_{E}$ is the split energy fluxes obtained by the Steger–Warming FVS method (Refer [24] for more details).

3.2. The Ada-WENO scheme

To construct the Ada-WENO scheme, a discontinuity-detecting method is needed to detect the discontinuities of the positive and negative sharing functions, respectively. In [21], the switch function derived in [29] is used,

$$\theta^{\pm} = \frac{1}{1 + (\sum_{k=0}^{2} \alpha^{\pm}_{k}, -1)^{z}}, \tag{26}$$

where, $\alpha^{\pm}_{k}$ are the non-linear weighting functions calculated directly from the sharing functions on the five-points stencils $S^{H+}_{i+1/2} = (H^{+}_{i-2}, H^{+}_{i-1}, H^{+}_{i}, H^{+}_{i+1}, H^{+}_{i+2})$ and $S^{H-}_{i+1/2} = (H^{-}_{i+3}, H^{-}_{i+2}, H^{-}_{i+1}, H^{-}_{i}, H^{-}_{i-1})$ by Eq. (5) (Refer [21,29] for the details). With this detector, we can directly implement CP WENO, Co-WENO [21] or a linear scheme (for simplicity, we use the fifth-order upwind scheme here) in the smooth regions, and CH WENO-Z near discontinuities to reduce the oscillations,

$$\boldsymbol{F}^{\pm}_{i+1/2} = \begin{cases} \boldsymbol{F}^{UP\pm}_{i+1/2}, & \text{if } \theta^{\pm} > 0.5, \text{ smooth, upwind scheme,} \\ \boldsymbol{F}^{CH\pm}_{i+1/2}, & \text{otherwise, discontinuities, CH WENO-Z.} \end{cases} \tag{27}$$

Since Ada-WENO only switches to CH WENO near discontinuities, it can significantly decrease the computational cost. As the analysis and numerical results shown in [21], Ada-WENO performs as well as CH WENO near discontinuities, but with a lower computational cost.

3.3. The Co-WENO scheme

As shown in [21], the weights ($\hat{\omega}_k^{\pm}$) computed directly from the sharing functions on stencils $S_{i+1/2}^{H\pm}$ by Eq. (5) can be used as the common-weights to construct the Co-WENO scheme. Replacing all the weights of CP WENO-Z in Eq. (14), we can obtain the numerical flux of Co-WENO-Z,

$$\boldsymbol{F}_{i+1/2}^{Co\pm} = P^{WENO-Z}(S_{i+1/2}^{\pm}), \text{ with } \omega_k = \hat{\omega}_k^{\pm}. \tag{28}$$

Since Co-WENO can maintain consistent discretization between different equations, it produces fewer oscillations than CP WENO. Moreover, considering that only one set of common-weights is calculated in Co-WENO, it can greatly reduce the computational cost of CP WENO when solving the Euler equations.

4. The new method

As the numerical results shown in [21,23,24], constructing the Ada-WENO and Co-WENO schemes with the sharing functions is a good way to reduce the oscillations of high-order WENO schemes while maintain high efficiency. However, the sharing functions designed by Peng et al. [21] and Shen et al. [24] are both derived from the FVS methods, hence the applications of the Ada-WENO and Co-WENO schemes are limited. In this paper, we want to focus on implementing the WENO schemes with high-resolution FDS methods, hence, we need to design a new sharing function which is independent of the FVS methods, and then construct high-efficient Ada-WENO and Co-WENO schemes with this sharing function for the FDS methods.

4.1. The new sharing function

For a FVS method, two sharing functions (positive and negative) are needed to detect the discontinuities of the positive and negative fluxes $\boldsymbol{F}^{\pm}$, respectively. While for a FDS method, we only need one single sharing function to indicate the discontinuities of the conserved variables $\boldsymbol{U}$.

Obviously, this sharing function must indicate all the jump discontinuities in $\boldsymbol{U}$, which means it should contain the discontinuities in density, pressure and velocity (ρ, p, u). After extensive searching and trial, we get,

$$Q = \rho \cdot p \cdot E. \tag{29}$$

For convenience, near the interface $x_{i+1/2}$, we use the symbols L and R to denote the left-side and right-side values, respectively. It is easy to know that, if only one of the variables (ρ, p and u) is discontinuous near the interface, Q will be discontinuous. For example, given $\rho^L \ll \rho^R$(discontinuity), $p^L \sim p^R$(smooth) and $u^L \sim u^R$(smooth), we have $Q^L \ll Q^R$. Moreover, near shock regions, although ρ, p, u are all discontinuous, we still have $\rho^L \ll \rho^R, p^L \ll p^R, E^L \ll E^R$(L for pre-shock) according to shock theory, and hence $Q^L \ll Q^R$. Therefore all of those discontinuities in the density, pressure and velocity can be indicated by this single sharing function.

4.2. The Ada-WENO-Z scheme

In this paper, the discontinuity-detecting method proposed by Shen and Zha [30] is used to detect the discontinuities of the new sharing function and then construct the Ada-WENO scheme. As analyses showed in [30], this detector is derived from the indicators of the WENO-Z scheme, and has similar behavior with the WENO-Z scheme near discontinuities. Therefore, it is beneficial to reduce the influences of detectors in the comparisons between the Ada-WENO scheme and WENO-Z scheme. It should be pointed out that, those discontinuity-detecting methods mentioned in [19,29,31,32] can be implemented in a similar way to construct the Ada-WENO scheme.

First, we calculate the global/local smoothness indicators in Eqs. (6) and (7) via the sharing function on stencil $S_{i+1/2}^{QL} = (Q_{i-2}, Q_{i-1}, Q_i, Q_{i+1}, Q_{i+2})$, then we can distinguish the stencil $S_{i+1/2}^{QL}$ as a smooth stencil or a non-smooth stencil as shown in [30],

$$\tau_{i+1/2}^{L} = \begin{cases} 1, \text{ if } \tau_5 > \min(IS_0, IS_2), \text{non-smooth} \\ 0, \text{ otherwise, smooth.} \end{cases} \tag{30}$$

To save the computational cost, the local smoothness indicator IS_1 is not used in this detector, and we found that the influence of this indicator can be negligible for all the cases tested in this paper. With this detector, we can compute the left-side value,

$$\boldsymbol{U}_{i+1/2}^{L} = \begin{cases} \boldsymbol{U}_{i+1/2}^{UPL}, \text{ if } \tau_{i+1/2}^{L} = 0, \text{ smooth, upwind scheme,} \\ \boldsymbol{U}_{i+1/2}^{CHL}, \text{ otherwise, non-smooth, CH WENO-Z.} \end{cases} \tag{31}$$

Similar as CP WENO-Z for a FDS method (see Eq. (20)), the indicator $\tau_{i+1/2}^{R}$ on stencil $S_{i+1/2}^{QR} = (Q_{i+3}, Q_{i+2}, Q_{i+1}, Q_i, Q_{i-1})$ is equal to $\tau_{i+3/2}^{L}$ on stencil $S_{i+3/2}^{QL}$, hence we can save some computation when calculating the right-side value $\boldsymbol{U}_{i+1/2}^{R}$.

4.3. The Co-WENO-Z scheme

First, we calculate one set of common-weights ($\hat{\omega}_k^{L}$) on stencil $S_{i+1/2}^{QL}$ with Eq. (5) (i.e., replace all the f_i in Eq. (7) with the scalar value Q_i in Eq. (29) to calculate the indicators IS_k and then the weights ω_k in Eq. (5), here, ω_k are the common-weights $\hat{\omega}_k^{L}$), and then replace the weights in Eq. (19) with this common-weights to obtain the left-side value,

$$\boldsymbol{U}_{i+1/2}^{CoL} = P^{WENO-Z}(S_{i+1/2}^{L}), \text{ with } \omega_k = \hat{\omega}_k^{L}. \tag{32}$$

The right-side value $\boldsymbol{U}_{i+1/2}^{CoR}$ can be obtained on stencil $S_{i+1/2}^{QR}$, and same as CP WENO-Z for the FDS method, we can save a lot of computation when computing $\boldsymbol{U}_{i+1/2}^{CoR}$, too.

5. Numerical results

In this section, several Euler problems are presented to test the performance of these new schemes. For convenience, we list the schemes as follows:

1. CP WENO-Z: the component-wise WENO-Z scheme (19);
2. CH WENO-Z: the characteristic-wise WENO-Z scheme (23);
3. Ada-WENO-Z: the new adaptive characteristic-wise WENO-Z scheme (31);
4. Co-WENO-Z: the new common-weights WENO-Z scheme (32);

First, three one-dimensional shock-tube problems [6,21] and two two-dimensional Riemann problems [33,34] are used to study the shock capturing ability of these WENO schemes, and then the Double-Mach problem [3,21] to test their behavior near strong shock and reflective waves. Finally, the two dimensional Rayleigh–Taylor problem [9,35] is used to show the good efficiency of these new WENO schemes. The high resolution Roe [26] solver is used in all the problems except the Double-Mach problem, and the HLLR solver is used in this problem to prevent the Carbuncle problem [36]. In this paper, time advancement is performed with the third-order Runge–Kutta method [37].

5.1. One-dimensional shock-tube problems

The first shock-tube problem is the Sod problem, its initial condition is,

$$(\rho, u, p) = \begin{cases} (1, 0, 1), & 0 \le x < 0.5, \\ (0.125, 0, 0.1), & 0.5 \le x \le 1. \end{cases} \tag{33}$$

The second one is the Shu–Osher problem with the initial condition,

$$(\rho, u, p) = \begin{cases} (3.857143, 2.629369, 31/3), & -5 \le x < -4, \\ (1 + 0.2\sin(5x), 0, 1), & -4 \le x \le 5. \end{cases} \tag{34}$$

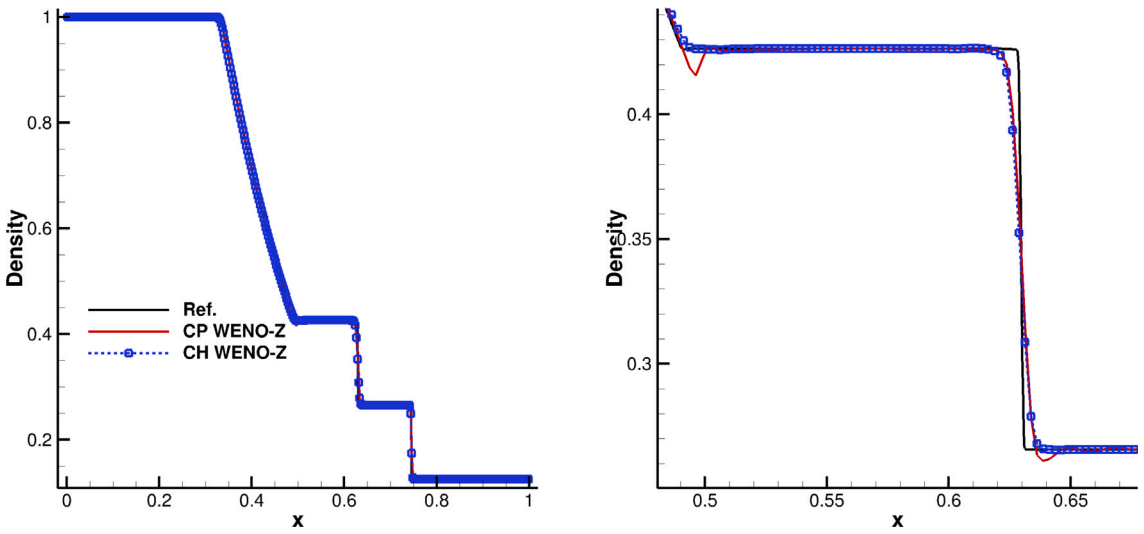

Fig. 1. Density distribution of Sod computed by CP WENO-Z and CH WENO-Z at $t = 0.14$.

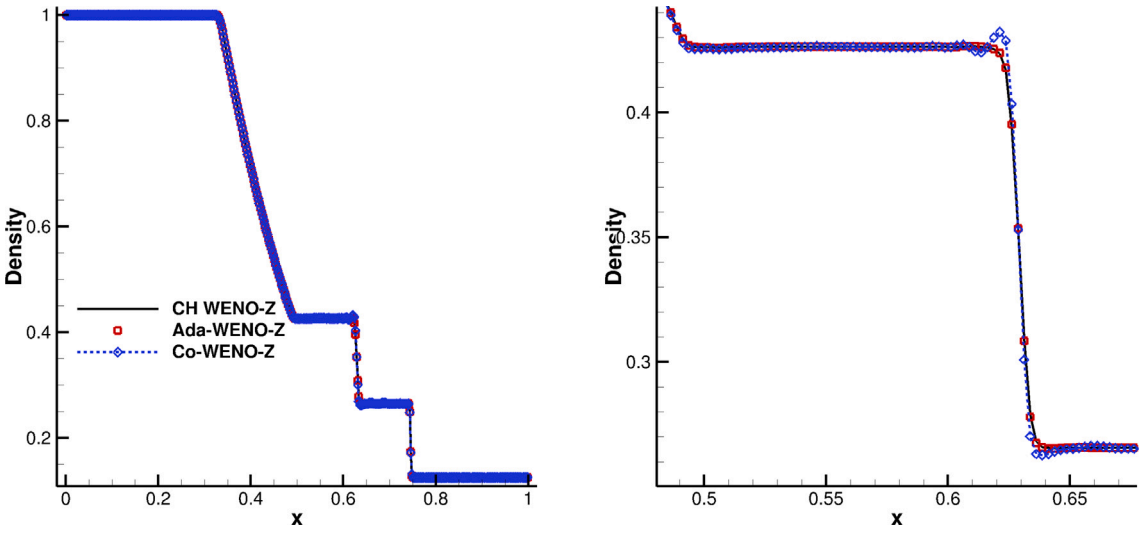

Fig. 2. Density distribution of Sod computed by CH WENO-Z, Co-WENO-Z and Ada-WENO-Z at $t = 0.14$.

The last one is the Lax problem with the initial condition,

$$(\rho, u, p) = \begin{cases} (0.445, 0.698, 3.528), & -5 \le x < 0, \\ (0.5, 0, 0.571), & 0 \le x \le 5. \end{cases} \tag{35}$$

The solutions with the grid of $N = 400$ are given in Figs. 1–6. It can be seen that, (a) CP WENO-Z generates overshoots (Fig. 3) and oscillations (Fig. 5); (b) Co-WENO can reduce the oscillations (Figs. 5 and 6) and overshoots (Figs. 3 and 4) of CP WENO-Z, but it may still generates few oscillations; (c) Ada-WENO-Z is oscillation-free and its results are comparable with those of CH WENO-Z.

5.2. Two-dimensional Riemann problems

Two two-dimensional Riemann problems [33,34] are presented to show the performance of these WENO schemes. The first Riemann problem (denoted as RM1) has the following initial condition,

$$(\rho, u, v, p) = \begin{cases} (1.1,\ 0,\ 0,\ 1.1) & 0.5 \le x \le 1, 0.5 \le y \le 1, \\ (0.5065,\ 0.8939,\ 0,\ 0.35) & 0 \le x < 0.5, 0.5 \le y \le 1, \\ (1.1,\ 0.8939,\ 0.8939,\ 1.1) & 0 \le x < 0.5, 0 \le y < 0.5, \\ (0.5065,\ 0,\ 0.8939,\ 0.35) & 0.5 \le x \le 1, 0 \le y < 0.5. \end{cases} \tag{36}$$

And the second one's (denoted as RM2) initial condition is,

$$(\rho, u, v, p) = \begin{cases} (1,\ 0.1,\ 0,\ 1) & 0.5 \le x \le 1, 0.5 \le y \le 1, \\ (0.5313,\ 0.8276,\ 0,\ 0.4) & 0 \le x < 0.5, 0.5 \le y \le 1, \\ (0.8,\ 0.1,\ 0,\ 0.4) & 0 \le x < 0.5, 0 \le y < 0.5, \\ (0.5313,\ 0.1,\ 0.7276,\ 0.4) & 0.5 \le x \le 1, 0 \le y < 0.5. \end{cases} \tag{37}$$

The RM1 case describes the interaction of two backward shocks and two forward shocks, and the RM2 case is the interaction of two backward shocks and two contact discontinuities (Refer Section 3.1 Configuration 4 and Section 3.2 Configuration E in [33] for more details). The solutions of RM1 and RM2 are shown in Figs. 7–10. As we can see, the results of Ada-WENO-Z are similar to those of CH WENO-Z, and Co-WENO-Z obtains smoother solutions than CP WENO-Z.

5.3. Two-dimensional Double-Mach case

As mentioned above, the simplified WENO-LF-5-PS scheme with pressure and entropy by Jiang and Shu [3] does not work for the Double-Mach problem which containing strong shock and reflective waves, hence this case is used to further test the behavior of these WENO schemes. Since the Roe solver will encounter the carbuncle

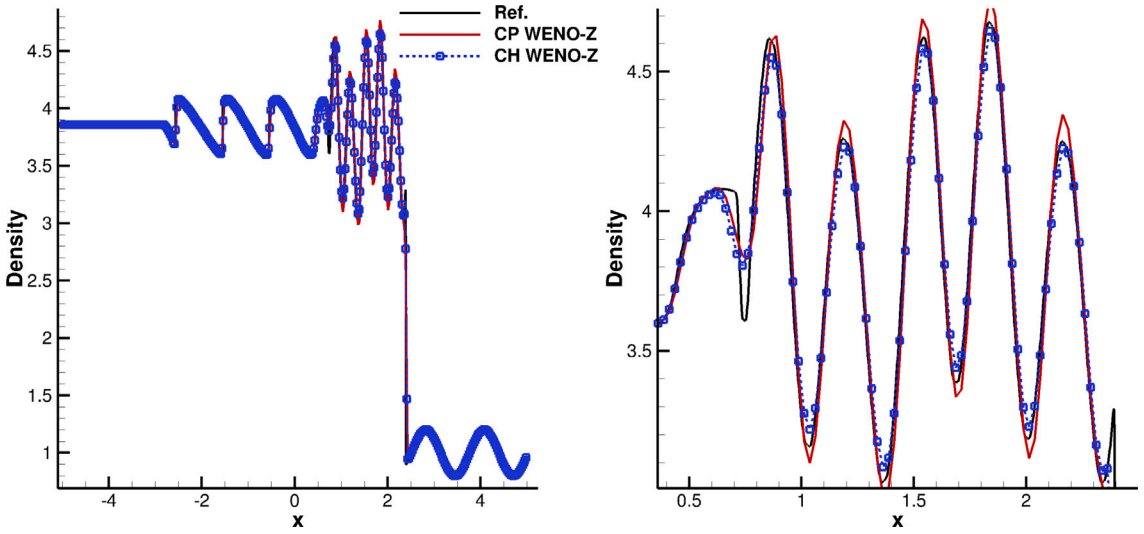

Fig. 3. Density distribution of Shu–Osher computed by CP WENO-Z and CH WENO-Z at $t = 1.8$.

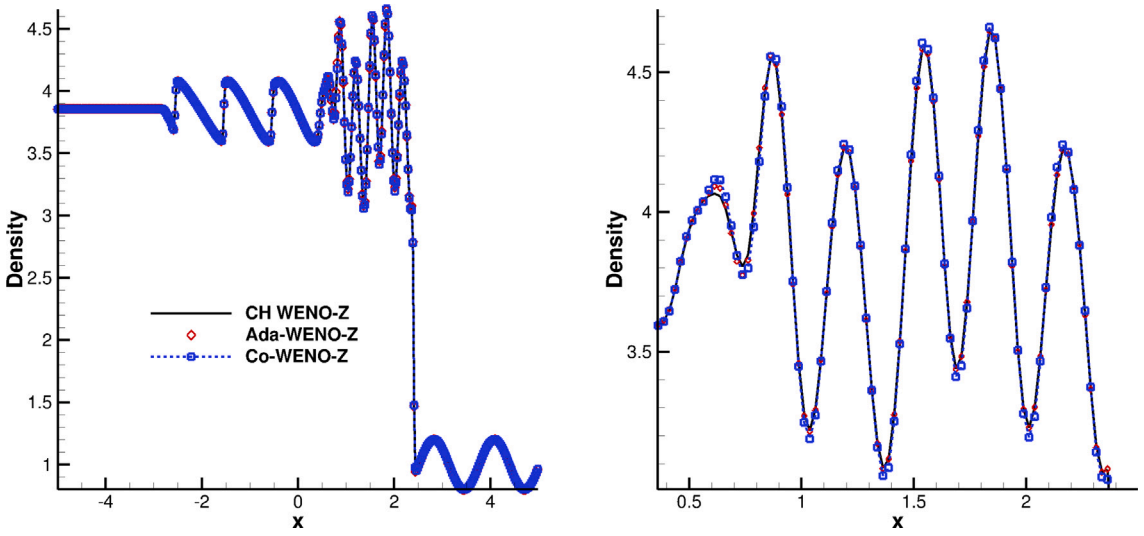

Fig. 4. Density distribution of Shu–Osher computed by CH WENO-Z, Co-WENO-Z and Ada-WENO-Z at $t = 1.8$.

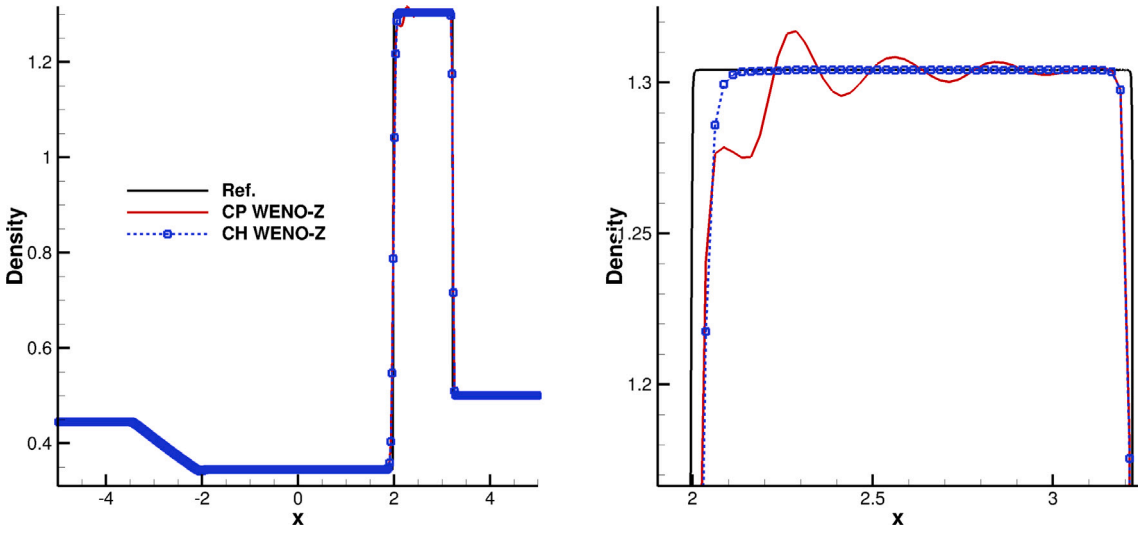

Fig. 5. Density distribution of Lax computed by CP WENO-Z and CH WENO-Z at $t = 1.3$.

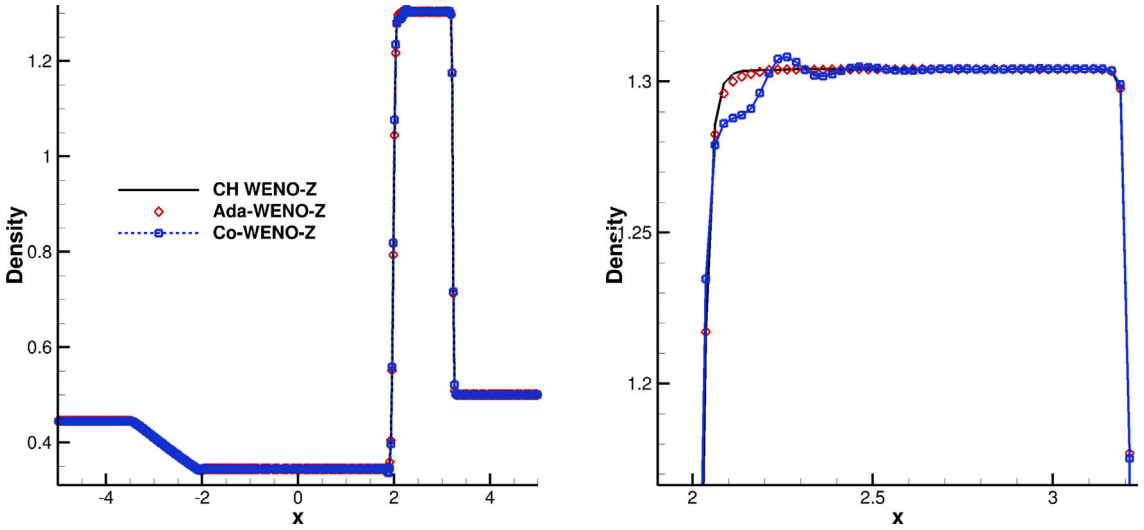

Fig. 6. Density distribution of Lax computed by CH WENO-Z, Co-WENO-Z and Ada-WENO-Z at $t = 1.3$.

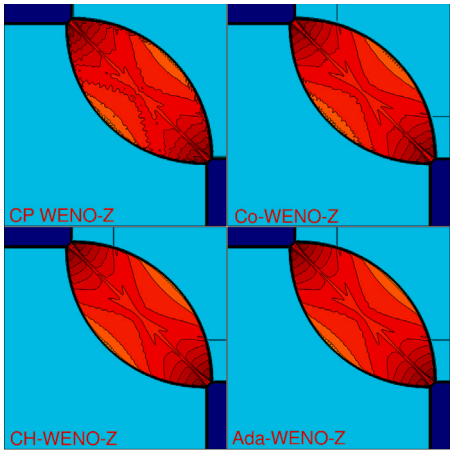

Fig. 7. Density contours of RM1 with 200 × 200 at $t = 0.25$.

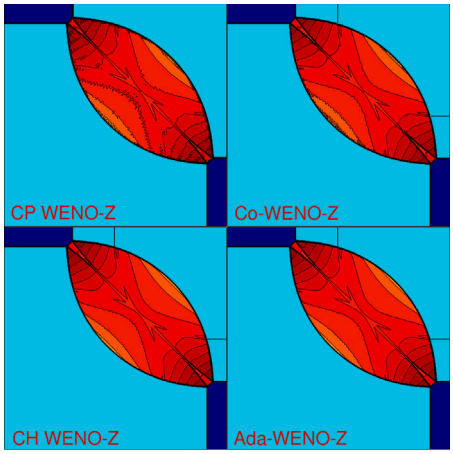

Fig. 8. Density contours of RM1 with 400 × 400 at $t = 0.25$.

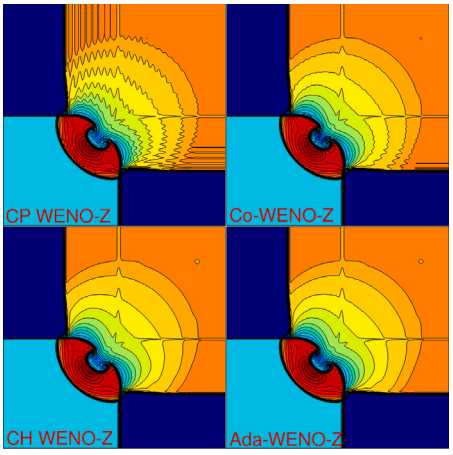

Fig. 9. Density contours of RM2 with 200 × 200 at $t = 0.3$.

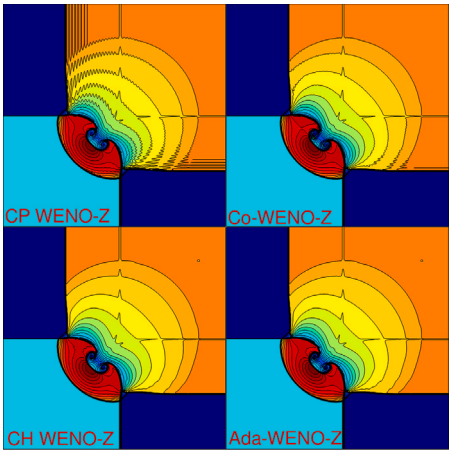

Fig. 10. Density contours of RM2 with 400 × 400 at $t = 0.3$.

phenomenon [36] in this problem, the carbuncle-free HLLR solver[36] is used here. The computational conditions can be found in [21]. The density contours are plotted in Figs. 11 and 12. As we can see, both Co-WENO-Z and Ada-WENO-Z resolve this problem well.

5.4. Two-dimensional Rayleigh–Taylor problem

The Rayleigh–Taylor problem (denoted as RT) is used to test the dissipation and efficiency of these WENO schemes. The computational

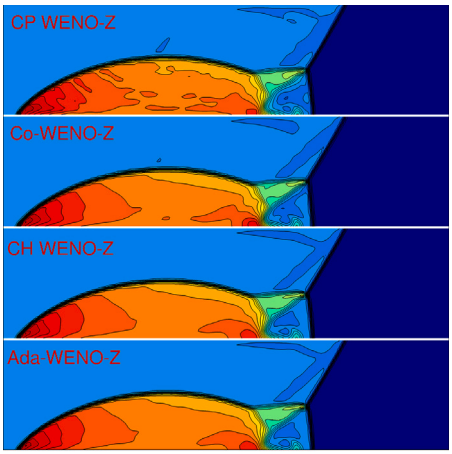

Fig. 11. Density contours of Double-Mach with 200 × 50 at $t = 0.2$.

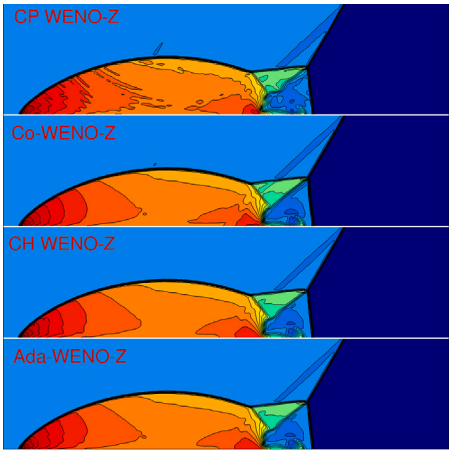

Fig. 12. Density contours of Double-Mach with 400 × 100 at $t = 0.2$.

Table 1
CPU_TIME of the WENO schemes for RT with 50 × 200.

	Total (rate)	Weno (rate)
CP WENO-Z	158.783 (1.00)	120.691 (1.00)
CH WENO-Z	367.244 (2.31)	328.691 (2.72)
Co-WENO-Z	125.213 (0.79)	87.255 (0.72)
Ada-WENO-Z	94.205 (0.59)	56.100 (0.46)

conditions for RT can be found in [9]. The density contours are plotted in Figs. 13 and 14. The CPU_TIME of these WENO schemes are presented in Table 1, where "total" means the total computational time for solving the RT problem, and "weno" indicate the computational time for the WENO reconstruction procedure (i.e., the total computational time for computing the left/right-side values $\boldsymbol{U}_{i+1/2}^{L/R}$, and for convenience, our Fortran code is available via https://gitee.com/liusp1988/co-weno.git).

As shown in Figs. 13 and 14, the result of Ada-WENO-Z is similar to the one of CH WENO-Z. That means, their dissipations are comparable. And among all of these schemes, CP WENO-Z produced the most complex structures, then is Co-WENO-Z. Although Co-WENO-Z and Ada-WENO-Z may be a little more dissipative than CP WENO-Z, they need much less computation than CP WENO-Z. As we can see in Table 1, both Co-WENO-Z and Ada-WENO-Z can save a lot of "weno" time of CP WENO-Z and CH WENO-Z. Since there are only a small percentage of discontinuous stencils in the whole computational domain (See Fig. 15), Ada-WENO-Z is even much more efficient than CP WENO-Z.

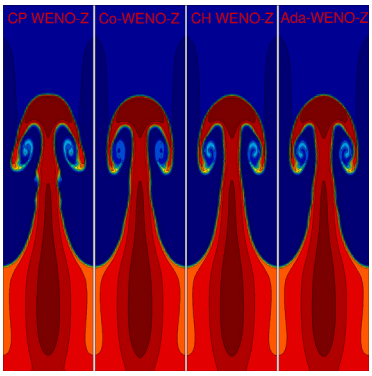

Fig. 13. Density contours of RT with 50 × 200 at $t = 1.95$.

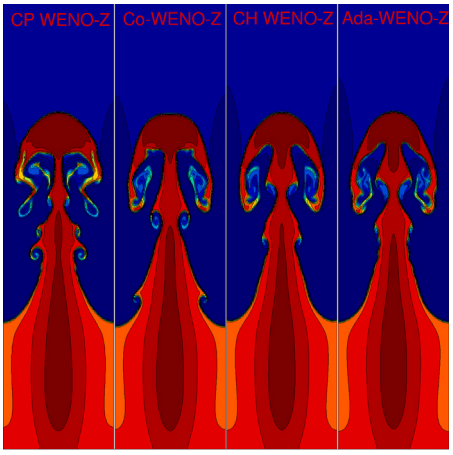

Fig. 14. Density contours of RT with 100 × 400 at $t = 1.95$.

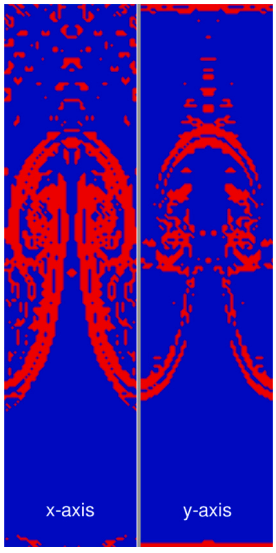

Fig. 15. Distribution of discontinuous stencils (Red) at the last step of Ada-WENO with 50 × 200 at $t = 1.95$.

6. Concluding remarks

In this paper, we propose a new sharing function independent of both the FVS and FDS methods, and then construct high-efficient adaptive characteristic-wise WENO (Ada-WENO) and common-weights WENO (Co-WENO) schemes with this function for the FDS methods:

1. The Ada-WENO scheme behaves similarly as the characteristic-wise WENO (CH WENO) scheme, it can keep the oscillation-free property near discontinuities with a much lower computational cost.

2. The Co-WENO scheme can reduce the oscillations and computational cost of the component-wise WENO (CP WENO) scheme, but it still generates few oscillations.

3. Based on the frameworks of the Ada-WENO and Co-WENO schemes, it is easy to combine other discontinuity-detecting methods with the new sharing function to develop high efficient and high resolution WENO schemes.

CRediT authorship contribution statement

Shengping Liu: Conceptualization, Writing – original draft, Writing – review & editing. **Yiqing Shen:** Supervision, Funding acquisition. **Shaodong Guo:** Software, Validation, Investigation. **Heng Yong:** Supervision, Funding acquisition, Project administration, Validation. **Guoxi Ni:** Project administration, Validation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

https://gitee.com/liusp1988/co-weno

Acknowledgments

This research work was supported by the National Key R&D Program of China under Grant Nos.2022YFA1004500, NSAF under Grant NO.U2230208, The Key Laboratory of Nuclear Data foundation under Grant JCKY2022201C155, National Natural Science Foundation of China under Grants 11872067 and 12172364.

References

[1] Shu C-W. High order WENO and DG methods for time-dependent convection-dominated PDEs: A brief survey of several recent developments. J Comput Phys 2016;316:598–613.
[2] Liu X-D, Osher S, Chan T. Weighted essentially non-oscillatory schemes. J Comput Phys 1994;115:200–12.
[3] Jiang G-S, Shu C-W. Efficient implementation of weighted ENO schemes. J Comput Phys 1996;126:202–28.
[4] Balsara DS, Shu C-W. Monotonicity preserving weighted essentially non-oscillatory schemes with increasingly high order of accuracy. J Comput Phys 2000;160:405–52.
[5] Gerolymos G, Senechal D, Vallet I. Very-high-order WENO schemes. J Comput Phys 2009;228:8481–524.
[6] Henrick AK, Aslam TD, Powers JM. Mapped weighted essentially non-oscillatory schemes: Achieving optimal order near critical points. J Comput Phys 2005;207:542–67.
[7] Borges R, Carmona M, Costa B, Don WS. An improved weighted essentially non-oscillatory scheme for hyperbolic conservation laws. J Comput Phys 2008;227:3191–211.
[8] Marcos C, Bruno C, Sun DW. High order weighted essentially non-oscillatory WENO-Z schemes for hyperbolic conservation laws. J Comput Phys 2011;230:1766–92.
[9] Liu S, Shen Y, Zeng F, Yu M. A new weighting method for improving the WENO-Z scheme. Internat J Numer Methods Fluids 2018;87:271–91.
[10] Fan P, Shen Y-Q, Tian B-L, Yang C. A new smoothness indicator for improving the weighted essentially non-oscillatory scheme. J Comput Phys 2014;269:329–54.
[11] Kim CH, Ha Y, Yoon J. Modified non-linear weights for fifth-order weighted essentially non-oscillatory schemes. J Sci Comput 2016;67:299–323.
[12] Ha Y, Kim CH, Lee YJ, Yoon J. An improved weighted essentially non-oscillatory scheme with a new smoothness indicator. J Comput Phys 2013;232:68–86.
[13] Hu X, Wang Q, Adams N. An adaptive central-upwind weighted essentially non-oscillatory scheme. J Comput Phys 2010;229:8952–65.
[14] Acker F, de R. Borges RB, Costa B. An improved WENO-Z scheme. J Comput Phys 2016;313:726–53.
[15] Shen Y, Liu L, Yang Y. Multistep weighted essentially non-oscillatory scheme. Internat J Numer Methods Fluids 2014;75:231–49.
[16] Ma Y, Yan Z, Zhu H. Improvement of multistep WENO scheme and its extension to higher orders of accuracy. Internat J Numer Methods Fluids 2016;82:818–38.
[17] Zeng F, Shen Y, Liu S, Liu L. A high performance fifth-order multistep WENO scheme. Internat J Numer Methods Fluids 2019;91:159–82.
[18] Peng J, Shen Y. Improvement of weighted compact scheme with multi-step strategy for supersonic compressible flow. Comput & Fluids 2015;115:243–55.
[19] Hu X, Wang B, Adams N. An efficient low-dissipation hybrid weighted essentially non-oscillatory scheme. J Comput Phys 2015;301:415–24.
[20] Puppo G. Adaptive application of characteristic projection for central schemes. In: Hyperbolic problems: Theory, numerics, applications. Springer; 2003, p. 819–29.
[21] Peng J, Zhai C, Ni G, Yong H, Shen Y. An adaptive characteristic-wise reconstruction WENO-Z scheme for gas dynamic euler equations. Comput & Fluids 2019;179:34–51.
[22] Liu S, Shen Y. Discontinuity-detecting method for a four-point stencil and its application to develop a third-order hybrid-WENO scheme. J Sci Comput 2019;81:1732–66.
[23] He Z, Zhang Y, Li X, Li L, Tian B. Preventing numerical oscillations in the flux-split based finite differece method for compressible flows with discontinuities. J Comput Phys 2015;300:269–87.
[24] Shen Y, Li S, Liu S, Peng J, Zheng G. A robust common-weights WENO scheme based on the flux vector splitting for Euler equations. 2021, (submitted for publication).
[25] Steger JL, Warming R. Flux vector splitting of the inviscid gasdynamic equations with application to finite-difference methods. J Comput Phys 1981;40:263–93.
[26] Roe PL. Approximate Riemann solvers, parameter vectors, and difference schemes. J Comput Phys 1981;43:357–72.
[27] Toro E. Riemann solvers and numerical methods for fluid dynamics. Springer-Verlag; 1997.
[28] Shu C-W. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. In: ICASE report 97-65. 1997.
[29] Peng J, Shen Y. A novel weighting weitch function for uniformly high-order hybrid shock-capturing schemes. Internat J Numer Methods Fluids 2017;83:681–703.
[30] Shen Y, Zha G. Generalized finite compact difference scheme for shock/complex flowfield interaction. J Comput Phys 2011;230:4419–36.
[31] Li G, Qiu J. Hybrid weighted essentially non-oscillatory schemes with different indicators. J Comput Phys 2010;229:8105–29.
[32] Ren Y, Liu M, Zhang H. A characteristic-wise hybrid compact-WENO scheme for solving hyperbolic conservation laws. J Comput Phys 2003;192:365–86.
[33] Schulz-Rinne C, Collins J, Glaz H. Numerical solution of the Riemann problem for two-dimensional gas dynamics. SIAM J Sci Comput 1993;14:1394–414.
[34] Kurganov A, Tadmor E. Solution of two-dimensional Riemann problems for gas dynamics without Riemann problem solvers. Numer Methods Partial Differ Equ 2002;18:584–608.
[35] Yong YN, Tufo H, dubey A, Rosner R. On the miscible Rayleigh-Taylor instability: two and three dimensions. J Fluid Mech 2001;447:337–408.
[36] Nishikawa H, Kitamura K. Very simple, carbuncle-free, boundary-layer-resolving, rotated-hybrid Riemann solvers. J Comput Phys 2008;227:2560–81.
[37] Shu C-W, Osher S. Efficient implementation of essentially non-oscillatory shock-capturing schemes. J Comput Phys 1988;77:439–71.