A simple weighted essentially non-oscillatory limiter for the correction procedure via reconstruction (CPR) framework on unstructured meshes

Jie Du[1], Chi-Wang Shu[2] and Mengping Zhang[3]

Abstract

In this paper, we adapt a simple weighted essentially non-oscillatory (WENO) limiter, originally designed for discontinuous Galerkin (DG) schemes on two-dimensional unstructured triangular meshes [39], to the correction procedure via reconstruction (CPR) framework for solving nonlinear hyperbolic conservation laws on two-dimensional unstructured triangular meshes with straight or curved edges. This is an extension of our earlier work [4] in which the WENO limiter was designed for the CPR framework on regular meshes. The objective of this simple WENO limiter is to simultaneously maintain uniform high order accuracy of the CPR framework in smooth regions and control spurious numerical oscillations near discontinuities. The WENO limiter we adopt in this paper uses information only from the target cell and its immediate neighbors. Hence, it is particularly simple to implement and will not harm the conservativeness and compactness of the CPR framework. Since the CPR framework with this WENO limiter does not in general satisfy the positivity preserving property, we also extend the positivity-preserving limiters [36, 26] to the CPR framework. Numerical results for both scalar equations and Euler systems of compressible gas dynamics are provided to illustrate the good behavior of this procedure.

Key Words: CPR framework, WENO limiter, positivity-preserving, unstructured meshes.

[1]School of Mathematical Sciences, University of Science and Technology of China, Hefei, Anhui 230026, P.R. China. E-mail: dujie@mail.ustc.edu.cn

[2]Division of Applied Mathematics, Brown University, Providence, RI 02912, USA. E-mail: shu@dam.brown.edu. Research supported by AFOSR grant F49550-12-1-0399 and NSF grant DMS-1418750.

[3]School of Mathematical Sciences, University of Science and Technology of China, Hefei, Anhui 230026, P.R. China. E-mail: mpzhang@ustc.edu.cn. Research supported by NSFC grants 11071234, 91130016 and 91024025.

1 Introduction

In this paper, we consider the following two-dimensional hyperbolic conservation law

$$\begin{cases} u_t + \nabla \cdot \vec{F}(u) = 0, \\ u(x, y, 0) = u_0(x, y), \end{cases} \tag{1}$$

where u is the state vector, $\vec{F} = (f, g)$ is the flux vector, and $\nabla = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y})$, to be solved on unstructured triangular meshes. There has been a surge of recent research activities on high-order methods capable of solving conservation laws on unstructured meshes, such as the discontinuous Galerkin (DG) method [22, 2, 3], the spectral and spectral-element type methods [8, 14], the spectral volume (SV) [27, 30, 31, 34] and spectral difference (SD) methods [18, 20, 32]. In this paper, we consider a correction procedure via reconstruction (CPR) framework developed more recently.

The CPR framework was originally developed in [9] to solve hyperbolic conservation laws on structured meshes, under the name of flux reconstruction (FR). In [28, 29], this method was extended to two-dimensional triangular and mixed grids, and the idea of flux reconstruction was generalized into a lifting collocation penalty (LCP) approach. Later, the FR and LCP methods were renamed the correction procedure via reconstruction or CPR in short. When performing the CPR method, one need to evaluate the divergence of the flux vector. Compared with the original more straight-forward Lagrange polynomial (LP) approach, the chain-rule (CR) approach suggested in [28] produces a one order higher approximation to the flux divergence, when the convective flux is nonlinear. However, the conservativeness of the CPR method will be harmed when using the CR approach for the interior flux divergence evaluation. In [6], Gao and Wang provided a fix by adding a correction term into the CPR formulation and obtained a conservative CPR formulation with the chain-rule divergence evaluation. In [33], the CPR framework was combined with a $P_N P_M$ method. The CPR framework can also be used to solve diffusion problems [10] and Navier-Stokes equations [5, 7]. In [16], it was also extended to compute broad-band waves.

The CPR formulation on unstructured meshes has some nice properties. With different choices of lifting coefficients, the CPR approach recovers several well-known methods in the framework, including DG, SV and SD methods and leads to more efficient versions of these methods, at least for linear equations. Compared with the original DG formulation based on volume and surface integral quadratures, the CPR framework solves the conservation laws in a nodal differential form. The extension of the CPR method to meshes with curved boundaries is straightforward because no surface or volume integrals are required. The reconstruction of solution polynomials to calculate the residual can be completely avoided through a judicious selection of solution and flux points. Moreover, the CPR method is compact because only immediate face neighbors play a role in updating the solutions in the current cell. Therefore, the complexity of implementation for multi-dimensional meshes including unstructured meshes can be reduced.

The main difficulty in solving conservation laws (1) is that solutions may contain discontinuities even if the initial conditions are smooth. However, the CPR method on unstructured meshes is only a high-order linear scheme, hence it may generate spurious oscillations for problems containing strong discontinuities. These spurious oscillations may lead to nonlinear instability and eventual blow-ups of the codes. Therefore, we need to apply nonlinear limiters to control these oscillations.

To achieve the full potential of high order accuracy and efficiency of the CPR method, we would like to find a robust high order limiting procedure to simultaneously maintain uniform high order accuracy in smooth regions and control spurious numerical oscillations near discontinuities. Limiters based on the WENO methodology [17, 13, 11] would serve such a purpose. Zhu et al. [38] designed limiters using the usual WENO reconstruction for the DG methods on two-dimensional unstructured meshes. They use the cell averages in an adaptive stencil to reconstruct the values of the solutions at certain points in the target cell. However, the reconstruction stencil contains not only the immediate neighboring cells of the target cell but also the neighbors' neighbors. To reduce the

width of the reconstruction stencil, a Hermite type WENO procedure for DG methods on unstructured meshes is adopted in [19], which uses not only the cell averages but also the first derivative or first order moment information in the stencil. However the information of neighbors' neighbors is still needed for higher order methods. Also, it is complicated to perform the usual WENO procedure or the Hermite type WENO procedure on unstructured meshes, with the possibility of negative linear weights, as we would need to use extra special treatments to handle them [23]. To maintain the compactness of the CPR framework, we would like to find another limiting procedure that uses the information only from the target cell and its immediate neighboring cells, with only positive linear weights in the WENO procedure if possible.

Very recently, a new and simple WENO limiter [39] was designed for the Runge-Kutta discontinuous Galerkin (RKDG) methods on two-dimensional triangulations. This is an extension of an earlier work [37] designed for RKDG methods on regular meshes. This WENO limiter attempts to reconstruct the entire polynomial on the target cell, instead of reconstructing point values or moments in the classical WENO reconstructions. In fact, the entire reconstruction polynomial is just a convex combination of polynomials on the target cell and its immediate neighboring cells (with suitable adjustments for conservation). Hence, it will not harm the compactness of the CPR framework. Also, the linear weights are always positive.

Recently, we have adapted the new WENO limiter in [37] to the CPR framework on structured meshes [4], to make it more robust for shocked flows while maintaining high order accuracy. In this paper, we will extend this WENO limiter to the CPR framework on two-dimensional unstructured triangular meshes with straight or curved edges along the lines of [39] for RKDG methods.

An important property of the entropy solution of a scalar conservation law is that it satisfies a strict maximum principle, i.e., if

$$M = \max_{(x,y)} u_0(x,y), \qquad m = \min_{(x,y)} u_0(x,y), \tag{2}$$

then $u(x, y, t) \in [m, M]$ for any (x, y) and t. In particular, the solution will not be negative if $u_0(x, y) \geqslant 0$. For hyperbolic conservation law systems, the entropy solutions generally do not satisfy the maximum principle. However, for the Euler equations for compressible flows, the density ρ and the pressure p should both be positive physically. The failure of preserving positivity of density or pressure may cause blow-ups of the numerical algorithm. In [36], a maximum-principle-satisfying and positivity-preserving high order DG scheme on triangular meshes was developed. [26] further discussed an extension to design arbitrarily high order positivity-preserving DG schemes for reactive Euler equations and presented a simpler and more robust implementation of the positivity-preserving limiter in [36].

In this paper, the CPR framework with the WENO limiter does not in general satisfy a strict positivity-preserving property. Hence, we also extend the limiters designed for the DG method in [36] and [26] to the CPR framework to get a positivity-preserving CPR method. This is done in our previous work [4] for the structured meshes.

This paper is organized as follows. We first review the CPR formulation on unstructured meshes in Section 2. In Section 3, we describe the details of how to introduce the WENO limiting procedure to the CPR framework. In Section 4, we describe the detailed procedure to construct a positivity-preserving CPR framework. In Section 5, numerical experiments are provided to verify the accuracy and stability of the CPR scheme with these limiters. Finally, concluding remarks are provided in Section 6.

2 Formulation of the CPR framework

In this section, we give an overview of the formulation of the CPR framework on two-dimensional unstructured triangular meshes for solving hyperbolic conservation law (1), which can be either a scalar equation or a system of equations. In Section 2.1, we give the formulation on unstructured triangular meshes with straight edges. The extension to curved boundary cells will be given in Section 2.2.

2.1 Triangular meshes with straight edges

Let the domain of calculation be divided into N non-overlapping elements with the i-th element denoted by V_i. By multiplying Eq. (1) with a test function w and integrating over V_i, the weighted residual of Eq. (1) can be written as

$$\int_{V_i} \big(u_t + \nabla \cdot \vec{F}(u)\big) w dxdy = \int_{V_i} u_t w dxdy + \int_{\partial V_i} w\vec{F}(u) \cdot \vec{\nu} ds - \int_{V_i} \nabla w \cdot \vec{F}(u) dxdy = 0, \tag{3}$$

where $\vec{\nu}$ is the outward unit normal vector on the element interface. We assume that the numerical solution belongs to P^k (the space of polynomials of degree k or less) within each element without continuity requirement across element interfaces. Let $u_i(x, y) \in P^k(V_i)$ be an approximate solution to u in element V_i, called the solution polynomial. It is required to satisfy the following equation

$$\int_{V_i} (u_i)_t w dxdy + \int_{\partial V_i} w\hat{F}(u_i, u_{i+}, \vec{\nu}) ds - \int_{V_i} \nabla w \cdot \vec{F}(u_i) dxdy = 0. \tag{4}$$

Since the solution is discontinuous across element interfaces, we use a common normal flux in the equation above

$$\hat{F}(u_i, u_{i+}, \vec{\nu}) \approx \vec{F}(u_i) \cdot \vec{\nu}, \tag{5}$$

where u_{i+} is the solution outside the current element V_i. Applying integration by parts again to the last term of Eq. (4), we obtain

$$\int_{V_i} (u_i)_t w dxdy + \int_{V_i} w\nabla \cdot \vec{F}(u_i) dxdy + \int_{\partial V_i} w\big(\hat{F}(u_i, u_{i+}, \vec{\nu}) - \vec{F}(u_i) \cdot \vec{\nu}\big) ds = 0. \tag{6}$$

Here $\hat{F}$ is the common numerical flux defined in (5), while $\vec{F}(u_i) \cdot \vec{\nu}$ is the boundary flux calculated only from the current cell V_i. Denote $[F^\nu] = \hat{F}(u_i, u_{i+}, \vec{\nu}) - \vec{F}(u_i) \cdot \vec{\nu}$ to be the normal flux difference. It represents the influence of the data in the immediate neighboring cells. Here we require that the space of the test functions have the same dimension as the solution space. The test space is chosen in a manner to guarantee that a unique solution exists for (6).

Next, we try to eliminate the test function w and transform the integral formulation into a differential one. By the introduction of a "correction field" $\delta_i \in P^k$,

$$\int_{V_i} w\delta_i dxdy = \int_{\partial V_i} w[F^\nu]ds, \tag{7}$$

the boundary integral is cast as a volume integral. The above equation is sometimes referred to as the "lifting operator". Substituting Eq. (7) into Eq. (6), we get

$$\int_{V_i} [(u_i)_t + \nabla \cdot \vec{F}(u_i) + \delta_i] w dxdy = 0. \tag{8}$$

In general, $\nabla \cdot \vec{F}(u_i)$ is not an element of $P^k(V_i)$. Here we approximate it by its projection onto $P^k(V_i)$, denoted by $\Pi(\nabla \cdot \vec{F}(u_i))$. In the case that the flux vector $\vec{F}$ is linear in u, we have $\Pi(\nabla \cdot \vec{F}(u_i)) = \nabla \cdot \vec{F}(u_i)$. We will deal with the nonlinear case later. Now the term $(u_i)_t + \Pi(\nabla \cdot \vec{F}(u_i)) + \delta_i$ belongs to $P^k(V_i)$. Because the test space is selected to ensure a unique solution, (8) is reduced to

$$(u_i)_t + \Pi(\nabla \cdot \vec{F}(u_i)) + \delta_i = 0, \tag{9}$$

which is satisfied everywhere in V_i. Hence, we have reduced the weighted residual formulation to an equivalent differential form, which involves no integrals. Note that for δ_i defined by (7), if $w \in P^k$, Eq. (9) is equivalent to the DG formulation; if w varies in another space, we will get a different method. We refer to [28] for more details.

Let the degrees-of-freedom be the solution values at a set of points $\vec{r}_{i,j}$ ($j = 1, \cdots, K = \frac{(k+1)(k+2)}{2}$), called solution points (SPs). Denote the numerical solution values on cell V_i at these solution points as $u_{i,j}(t)$, $j = 1, \cdots, K$. Note that each numerical solution value $u_{i,j}(t)$ depends on t, and that $u_{i,j}(t^n)$ is the numerical solution at time level n. For simplicity, $u_{i,j}(t^n)$ is abbreviated to $u_{i,j}$ when there is no confusion. Eq. (9) must be true at the SPs, i.e.,

$$(u_{i,j})_t + \Pi(\nabla \cdot \vec{F}(u_{i,j})) + \delta_{i,j} = 0, \qquad j = 1, \cdots, K. \tag{10}$$

Let us now deal with the term $\delta_{i,j}$ which can be computed by (7). Along each edge m, we need to define $k+1$ points named flux points. For efficiency, the solution points

are chosen to coincide with the flux points along cell edges. The normal flux difference values $[F^\nu]_{m,l}, l = 1, \cdots, k+1$ at the flux points along edge m define a one dimensional polynomial of degree k, denoted by $[F^\nu]_m$. Eq. (7) can then be written as

$$\int_{V_i} w_k \sum_{j=1}^{K} L_j \delta_{i,j} dxdy = \sum_{m \in \partial V_i} \int_m w_k [F^\nu]_m ds. \tag{11}$$

Here $L_j, j = 1, \cdots, K$ are the two dimensional Lagrange polynomials, and w_k is the test function. For example, we can let $w_k = L_k$, and this equation yields a linear system as k varies form 1 to K. The resulting formulation is equivalent to the DG method. The unknowns $\delta_{i,j}$ can be easily solved in terms of the normal flux difference values at the flux points, i.e.,

$$\delta_{i,j} = \frac{1}{|V_i|} \sum_{m \in \partial V_i} \sum_{l=1}^{k+1} \alpha_{j,m,l} [F^\nu]_{m,l} S_m, \tag{12}$$

where $\alpha_{j,m,l}$ are constants independent of the solution, j is the index for the solution points, i is the cell index, m is the edge index, l is the index of the flux points, S_m is the length of edge m and $|V_i|$ is the area of cell V_i. Substituting (12) into (10), we obtain the following CPR formulation

$$(u_{i,j})_t + \Pi(\nabla \cdot \vec{F}(u_{i,j})) + \frac{1}{|V_i|} \sum_{m \in \partial V_i} \sum_{l=1}^{k+1} \alpha_{j,m,l} [F^\nu]_{m,l} S_m = 0, \qquad j = 1, \cdots, K. \tag{13}$$

For more details of the selection of the solution points and flux points and the computation of coefficients $\alpha_{j,m,l}$ for various schemes, we refer to [28].

Now let us deal with the term $\Pi(\nabla \cdot \vec{F}(u_i))$. To compute this projection, two approaches can be used: Lagrange polynomial (LP) and chain-rule (CR). For the LP approach, the flux function is projected onto the P^k space with Lagrange polynomial reconstruction:

$$\Pi_{LP}\big(\nabla \cdot \vec{F}(u_i)\big) = \nabla \cdot \sum_{j \in SPs} L_j \vec{F}(u_{i,j}). \tag{14}$$

However, for nonlinear fluxes, the LP approach can cause the accuracy to deteriorate by about half an order due to the aliasing error [12]. The aliasing error could also

induce instabilities [12]. To remedy this accuracy loss and possible instabilities, the CR approach can be used instead. The idea is to obtain the exact divergence of the fluxes at solution points, based on the degree k solution polynomial. Then the divergences at the solution points are projected onto the P^k space:

$$\Pi_{CR}\big(\nabla \cdot \vec{F}(u_i)\big) = \sum_{j \in SPs} L_j\big(\frac{\partial \vec{F}}{\partial u} \cdot \nabla u_i\big)_j. \tag{15}$$

When using the CR approach, the CPR method is not fully conservative. In [6], this conservation issue is fixed by adding a source term to account for the conservation error. Define the source term as

$$S_{c,i} = -\frac{\int_{V_i} \Pi_{CR}\big(\nabla \cdot \vec{F}(u_i)\big) dxdy - \int_{\partial V_i} \vec{F}(u_i) \cdot \vec{\nu} ds}{|V_i|}, \tag{16}$$

then the fixed CPR formulation is

$$(u_{i,j})_t + \Pi_{CR}\big(\nabla \cdot \vec{F}(u_{i,j})\big) + \delta_{i,j} + S_{c,i} = 0, \qquad j = 1, \cdots, K, \tag{17}$$

which can be solved by various time discretizations such as a Runge-Kutta method. We will use this fixed CPR formulation in the rest of this paper. Integrating Eq. (17) over cell V_i with test function $w = 1$, we can obtain

$$\int_{V_i} (u_i)_t dxdy + \int_{\partial V_i} \hat{F}(u_i, u_{i+}, \vec{\nu}) ds = 0, \tag{18}$$

which means Eq. (17) is fully conservative. For more details of the conservation issue, we refer to [6].

2.2 Triangular meshes with curved boundary

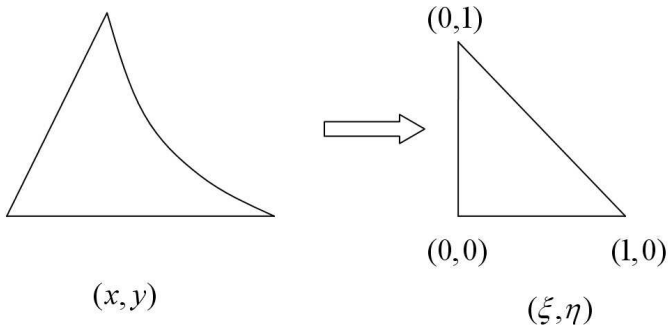


Figure 1: **Transformation of a curved boundary triangle to a standard element**

We first transform the curved boundary element from the physical domain (x, y) into the computational domain which is a standard element with the variables (ξ, η) as shown in Figure 1. The transformation can be written as

$$\begin{pmatrix} x \\ y \end{pmatrix} = \sum_{l=1}^{L} M_l(\xi, \eta) \begin{pmatrix} x_l \\ y_l \end{pmatrix}, \tag{19}$$

where L is the number of points used to define the curved boundary element, (x_l, y_l) are the Cartesian coordinates of these points, and $M_l(\xi, \eta)$ are the shape functions. Denote the Jacobian matrix of this transformation as

$$J = \frac{\partial(x, y)}{\partial(\xi, \eta)} = \begin{bmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{bmatrix}. \tag{20}$$

Assuming the transformation is non-singular so that the inverse matrix J^{-1} exists, we get that

$$J^{-1} = \frac{\partial(\xi, \eta)}{\partial(x, y)} = \begin{bmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{bmatrix}, \tag{21}$$

where

$$\xi_x = y_\eta/|J|, \qquad \xi_y = -x_\eta/|J|, \qquad \eta_y = -y_\xi/|J|, \qquad \eta_y = x_\xi/|J|. \tag{22}$$

Here, $|J|$ is the Jacobian determinant of J and is assumed to be positive. By replacing the variables (x, y) with (ξ, η), we transform Eq. (1) on the physical domain into the following equation on the standard element which has straight edges,

$$\tilde{u}_t + \tilde{\nabla} \cdot \vec{\tilde{F}} = 0, \tag{23}$$

where $\tilde{\nabla} = (\frac{\partial}{\partial \xi}, \frac{\partial}{\partial \eta})$, $\vec{\tilde{F}} = (\tilde{f}, \tilde{g})$ and

$$\begin{aligned} \tilde{u} &= |J|u, \\ \tilde{f} &= |J|(\xi_x f + \xi_y g), \\ \tilde{g} &= |J|(\eta_x f + \eta_y g). \end{aligned} \tag{24}$$

We now introduce the CPR formulation for the curved boundary element, for example, the i-th element V_i. Let us denote the standard element as $\tilde{V}$. Note that the Jacobian matrix of the transformation from V_i to $\tilde{V}$ depends on the shape of V_i. Here,

we assume that the new solution polynomial $\tilde{u}_i(\xi,\eta)$ is a member in $P^k(\tilde{V})$. Hence, we can obtain $\tilde{u}_i(\xi,\eta)$ by interpolating $\tilde{u}_{i,j} = |J|u_{i,j}, j = 1,\cdots,K$ in $\tilde{V}$. We then apply the CPR formulation (17) to the transformed equation on the standard triangle. Now, the formulation becomes

$$(\tilde{u}_{i,j})_t + \Pi_{CR}\big(\tilde{\nabla}\cdot\vec{\tilde{F}}\big) + \tilde{\delta}_{i,j} + \tilde{S}_{c,i} = 0, \qquad j = 1,\cdots,K, \tag{25}$$

where the superscript $\sim$ means the variables or operations computed by the transformed equation on the standard triangular element. For example,

$$\tilde{\delta}_{i,j} = 2\sum_{m\in\partial\tilde{V}}\sum_{l=1}^{k+1}\alpha_{j,m,l}[\tilde{F}^{\tilde{\nu}}]_{m,l}\tilde{S}_m, \tag{26}$$

where $\tilde{\nu}$ is the outward unit normal vector of $\tilde{V}$, $[\tilde{F}^{\tilde{\nu}}]_{m,l}$ are the normal jumps of the transformed fluxes across the edges of $\tilde{V}$, and $\tilde{S}_m, m\in\partial\tilde{V}$ are the lengths of the edges of $\tilde{V}$.

3 The WENO limiter

In this section, we attempt to adapt the WENO limiter in [39] to the CPR framework to control the oscillations for shocked flows as well as maintain the original high order accuracy in smooth regions. As mentioned in the introduction, the CPR method is compact because only immediate face neighbors play a role in updating the solutions in the current cell. For convenient, we label the target cell and its immediate neighboring cells as shown in Figure 2. When performing the WENO limiting procedure on V_i, we use the same stencil $S = \{V_i, V_{i1}, V_{i2}, V_{i3}\}$. Hence, the WENO limiter can maintain the compactness of CPR. For the target cell V_i, we first use the WENO limiter to get a new reconstructed function $u_i^{new}(x,y)$ if the original solution $u_i(x,y)$ contains possible shocks, then perform the normal CPR procedure on the new numerical solutions $u_{i,j}^{new} = u_i^{new}(\vec{r}_{i,j})$, $j = 1,\cdots,K$ at the solution points instead of $u_{i,j}, j = 1,\cdots,K$.

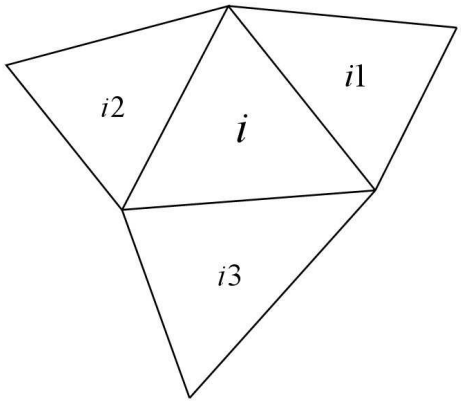


Figure 2: **The stencil** $S = \{V_i, V_{i1}, V_{i2}, V_{i3}\}$

In Sections 3.1 and 3.2, we assume that all the edges in the stencil S are straight. Hence, the solutions on those cells in S are approximated by the solution polynomials on the physical domain (x, y) by interpolating the numerical solutions at the solution points. We will discuss the extensions to the curved boundary cells in Section 3.3.

3.1 The WENO limiting procedure for the scalar case

Consider conservation laws as shown in Eq. (1), where u, f and g are scalars. For convenience, we denote the solution polynomials on cell $V_i, V_{i1}, V_{i2}, V_{i3}$ as

$$p_0(x, y) = u_i(x, y) \in P^k(V_i), \qquad p_l(x, y) = u_{il}(x, y) \in P^k(V_{il}),\ l = 1, 2, 3. \tag{27}$$

As in [39], we first use the KXRCF shock detection technique [15] to check whether the cell V_i is a troubled cell, namely, whether $p_0(x, y)$ contains possible shocks and needs the limiting procedure. We divide the boundary ∂V_i into two parts: ∂V_i^- and ∂V_i^+, where the flow is into $(\vec{F}'(u) \cdot \vec{\nu} < 0)$ and out of $(\vec{F}'(u) \cdot \vec{\nu} > 0)$ V_i respectively. The target cell V_i is identified as a troubled cell when

$$\frac{|\int_{\partial V_i^-} \big(p_0(x, y) - p_l(x, y)\big) ds|}{h^{\frac{k+1}{2}} |\partial V_i^-| \cdot \|p_0\|} > C_k, \tag{28}$$

where C_k is a constant, h is the radius of the circumscribed circle in V_i, $p_l(x, y)$ denote the solution polynomials on the neighboring cells sharing the edge(s) in ∂V_i^-, and $\|\cdot\|$ is

the standard L^2 norm in the cell V_i. We remark that the KXRCF troubled cell indicator is just one of the many possibilities and may not be the best one (see [21] for a detailed discussion). We use it here for its simplicity, as our main focus of this paper is not on troubled cell indicators.

Assuming that the cell V_i is a troubled cell, we now introduce the WENO limiting procedure in [39] to reconstruct the solution polynomial on it. We need to use the solution polynomials on the target cell and its three neighboring cells, namely, $p_l(x,y)$, $l = 0,1,2,3$. In order to maintain the original cell average of $p_0(x,y)$ in cell V_i, which is essential to keep the conservativeness of the original CPR framework, we make the following modifications:

$$\tilde{p}_l(x,y) = p_l(x,y) - \frac{1}{|V_i|}\int_{V_i} p_l(x,y)dxdy + \frac{1}{|V_i|}\int_{V_i} p_0(x,y)dxdy, \qquad l = 1,2,3. \tag{29}$$

For notational consistency we also denote $\tilde{p}_0(x,y) = p_0(x,y)$. The final nonlinear WENO reconstruction polynomial $u_i^{new}(x,y)$ on cell V_i is defined by a convex combination of these four modified polynomials:

$$u_i^{new}(x,y) = \omega_0\tilde{p}_0(x,y) + \omega_1\tilde{p}_1(x,y) + \omega_2\tilde{p}_2(x,y) + \omega_3\tilde{p}_3(x,y). \tag{30}$$

From Eq. (30), we know that $u_i^{new}(x,y)$ has the same cell average as $p_0(x,y)$ as long as $\omega_0 + \omega_1 + \omega_2 + \omega_3 = 1$. Hence, we define the normalized nonlinear weights as

$$\omega_l = \frac{\tilde{\omega}_l}{\sum_{m=0}^{3}\tilde{\omega}_m}, \qquad l = 0,1,2,3, \tag{31}$$

where the non-normalized nonlinear weights $\tilde{\omega}_l$ are defined as

$$\tilde{\omega}_l = \frac{\gamma_l}{(\varepsilon + \beta_l)^2}, \qquad l = 0,1,2,3. \tag{32}$$

Here $\varepsilon > 0$ is introduced to avoid the denominator to become 0. We take $\varepsilon = 10^{-6}$ in all our numerical tests. The linear weights $\gamma_l, l = 0,1,2,3$ are a set of positive numbers adding up to one. Note that since $\tilde{p}_l(x,y)$ for $l = 0,1,2,3$, are all $(k+1)$-th order approximations to the exact solution in smooth regions, there are no extra requirements

on the linear weights in order to maintain the original high order accuracy. As discussed in [39], since for smooth solutions the central cell is usually the best one, we put a larger linear weight on the central cell than on the neighboring cells. As in [39], we take

$$\gamma_0 = 0.997, \qquad \gamma_1 = \gamma_2 = \gamma_3 = 0.001, \tag{33}$$

which can maintain the original high order in smooth regions and can keep essentially non-oscillatory shock transitions in our numerical examples. β_l is the smoothness indicator, which measures how smooth the function $\tilde{p}_l(x, y)$ is on the target cell V_i. As in [13, 1], we define β_l as

$$\beta_l = \sum_{|s|=1}^{k} |V_i|^{|s|-1} \int_{V_i} \big(\frac{\partial^{|s|}}{\partial x^{s_1} \partial y^{s_2}} \tilde{p}_l(x, y)\big)^2 dxdy, \tag{34}$$

where $s = (s_1, s_2)$.

For each cell V_i, $i = 1, \cdots, N$, if it is a troubled cell, we replace the entire solution polynomial $u_i(x, y)$ with the new reconstructed polynomial $u_i^{new}(x, y)$, which is a convex combination of polynomials on this cell and its immediate neighboring cells. If the cell V_i is not a troubled cell, we just let $u_i^{new}(x, y) = u_i(x, y)$. Now we can get the new solutions at the solution points $u_{i,j}^{new} = u_i^{new}(\vec{r}_{i,j})$, $j = 1, \cdots, K$. After the WENO limiter procedure, we just use the original CPR procedure to march to the next time level by using the new numerical solutions $u_{i,j}^{new}$, $j = 1, \cdots, K$.

3.2 The WENO limiting procedure for the Euler system

Let us consider the two-dimensional Euler system which is given by

$$\mathbf{u}_t + \mathbf{f}(\mathbf{u})_x + \mathbf{g}(\mathbf{u})_y = 0,$$

$$\mathbf{u} = \begin{pmatrix} \rho \\ m \\ n \\ E \end{pmatrix}, \mathbf{f}(\mathbf{u}) = \begin{pmatrix} m \\ \rho u^2 + p \\ \rho uv \\ u(E + p) \end{pmatrix}, \mathbf{g}(\mathbf{u}) = \begin{pmatrix} n \\ \rho uv \\ \rho v^2 + p \\ v(E + p) \end{pmatrix}. \tag{35}$$

Here, ρ is the density, $(u, v)^T$ is the velocity vector, $m = \rho u$ and $n = \rho v$ are the momenta. E is the total energy, and p is the pressure, with

$$p(\mathbf{u}) = (\gamma - 1)\big(E - \frac{1}{2}\rho(u^2 + v^2)\big). \tag{36}$$

We present the WENO limiting procedure for the Euler system in this subsection. For convenience, we also denote the solution polynomials on cell $V_i, V_{i1}, V_{i2}, V_{i3}$ as

$$\mathbf{p}_0(x,y) = \mathbf{u}_i(x,y), \qquad \mathbf{p}_l(x,y) = \mathbf{u}_{il}(x,y), \quad l = 1,2,3, \tag{37}$$

respectively. Each of them is a 4-component vector.

As in the scalar case, we first identify the troubled cells using the KXRCF technique. We divide the boundary ∂V_i into two parts: ∂V_i^- and ∂V_i^+, where the flow is into $((u,v)^T \cdot \vec{\nu} < 0)$ and out of $((u,v)^T \cdot \vec{\nu} > 0)$ V_i respectively. In the system case, we take both the density ρ and the total energy E as the indicator variables. The target cell V_i is identified as a troubled cell when

$$\frac{|\int_{\partial V_i^-} \big(\rho_0(x,y) - \rho_l(x,y)\big) ds|}{h^{\frac{k+1}{2}} |\partial V_i^-| \cdot \|\rho_0\|} > C_k, \tag{38}$$

or

$$\frac{|\int_{\partial V_i^-} \big(E_0(x,y) - E_l(x,y)\big) ds|}{h^{\frac{k+1}{2}} |\partial V_i^-| \cdot \|E_0\|} > C_k, \tag{39}$$

where $\rho_l(x,y)$ and $E_l(x,y)$ denote the density polynomials and the total energy polynomials on the neighboring cells sharing the edge(s) in ∂V_i^-.

Assuming that the cell V_i is a troubled cell, we now perform the WENO limiting procedure on it. To maintain the original cell average of $\mathbf{p}_0(x,y)$ in cell V_i, we compute as before the four modified solution polynomials on the target cell and its three immediate neighboring cells:

$$\tilde{\mathbf{p}}_l(x,y) = \mathbf{p}_l(x,y) - \frac{1}{|V_i|}\int_{V_i} \mathbf{p}_l(x,y)dxdy + \frac{1}{|V_i|}\int_{V_i} \mathbf{p}_0(x,y)dxdy, \qquad l = 0,1,2,3, \tag{40}$$

each of them being a 4-component vector and each component of the vector is a k-th degree polynomial. In order to achieve better non-oscillatory qualities, the WENO limiter is used with a local characteristic field decomposition. Denote the Jacobian matrix as $\big(\mathbf{f}'(\mathbf{u}), \mathbf{g}'(\mathbf{u})\big)^T \cdot \vec{\nu}_l$, where $\vec{\nu}_l = (n_{lx}, n_{ly})^T, l = 1,2,3$ are the outward unit normal vectors

to different edges of the target cell V_i. Then the matrix with the left eigenvectors of such Jacobian matrix as rows is

$$L_l = \begin{pmatrix} \frac{B_2+(un_{lx}+vn_{ly})/c}{2} & -\frac{B_1u+n_{lx}/c}{2} & -\frac{B_1v+n_{ly}/c}{2} & \frac{B_1}{2} \\ n_{ly}u - n_{lx}v & -n_{ly} & n_{lx} & 0 \\ 1-B_2 & B_1u & B_1v & -B_1 \\ \frac{B_2-(un_{lx}+vn_{ly})/c}{2} & -\frac{B_1u-n_{lx}/c}{2} & -\frac{B_1v-n_{ly}/c}{2} & \frac{B_1}{2} \end{pmatrix}, \tag{41}$$

and the matrix with the right eigenvectors as columns is

$$R_l = \begin{pmatrix} 1 & 0 & 1 & 1 \\ u - cn_{lx} & -n_{ly} & u & u + cn_{lx} \\ v - cn_{ly} & n_{lx} & v & v + cn_{ly} \\ H - c(un_{lx} + vn_{ly}) & -n_{ly}u + n_{lx}v & \frac{u^2+v^2}{2} & H + c(un_{lx} + vn_{ly}) \end{pmatrix}, \tag{42}$$

where $c = \sqrt{\gamma p/\rho}$, $B_1 = (\gamma - 1)/c^2$, $B_2 = B_1(u^2 + v^2)/2$ and $H = (E + p)/\rho$.

Now we perform the characteristic-wise WENO limiting procedure in each $\vec{\nu}_l$ direction with the associated Jacobian $\left(\mathbf{f}'(\mathbf{u}), \mathbf{g}'(\mathbf{u})\right)^T \cdot \vec{\nu}_l$ to reconstruct a new polynomial vector $(\mathbf{p}_0)_l^{new}$. We first project the modified polynomial vectors $\tilde{\mathbf{p}}_m, m = 0, 1, 2, 3$ into the characteristic fields:

$$\bar{\mathbf{p}}_m = L_l \cdot \tilde{\mathbf{p}}_m, \qquad m = 0, 1, 2, 3. \tag{43}$$

Then we perform the scalar WENO limiting procedure that has been specified in the last subsection on each component of these vectors, and obtain a 4-component vector $\bar{\mathbf{p}}_0^{new}$. The new polynomial vector $(\mathbf{p}_0)_l^{new}$ in the $\vec{\nu}_l$ direction is then computed by projecting $\bar{\mathbf{p}}_0^{new}$ back into the physical space:

$$(\mathbf{p}_0)_l^{new} = R_l \cdot \bar{\mathbf{p}}_0^{new}. \tag{44}$$

The final new 4-component solution polynomial vector on the troubled cell V_i is defined as

$$\mathbf{u}_i^{new} = \frac{\sum_{l=1}^{3} (\mathbf{p}_0)_l^{new} |V_{il}|}{\sum_{l=1}^{3} |V_{il}|}. \tag{45}$$

For each cell V_i, $i = 1, \cdots, N$, if it is a troubled cell, we perform the characteristic-wise WENO limiting procedure discussed above in each $\vec{\nu}_l$ direction separately and get

three reconstructed new polynomial vectors $(\mathbf{p}_0)_l^{new}, l = 1, 2, 3$. We replace the entire solution polynomial $\mathbf{u}_i(x, y)$ with the final new 4-component solution polynomial vector $\mathbf{u}_i^{new}(x, y)$ as in (45), which is a combination of these three reconstructed polynomials. If the cell V_i is not a troubled cell, we just let $\mathbf{u}_i^{new}(x, y) = \mathbf{u}_i(x, y)$. Then we compute the new numerical solutions $\mathbf{u}_{i,j}^{new} = \mathbf{u}_i^{new}(\vec{r}_{i,j})$, $j = 1, \cdots, K$. We now use the normal CPR procedure to march to the next time level by using the new numerical solutions $\mathbf{u}_{i,j}^{new}$, $j = 1, \cdots, K$.

3.3 Extension to curved cells

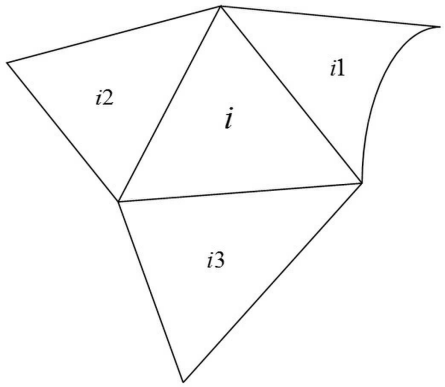


Figure 3: **The case that only the neighboring cells have curved edges**

We now turn to the cases that the reconstruction stencil S contains triangle(s) with curved edges. We first consider the case that only one or more of the neighboring cells, say V_{i1}, has curved edges, as shown in Fig. 3. Note that in the CPR framework, we transform V_{i1} into the standard element $\tilde{V}$ and assume that the transformed solution polynomial $\tilde{u}_{i1}(\xi, \eta) = |J|u_{i1}$ (it is a 4-component vector in the system case) is a member in $P^k(\tilde{V})$. Hence, $u_{i1} = \tilde{u}_{i1}/|J|$ is no longer a polynomial in the physical element. However, in the WENO limiting procedure, we can simply obtain a polynomial $p_1(x, y)$ by interpolating the numerical solution values $u_{i1,j} = \tilde{u}_{i1,j}/|J|, j = 1, \cdots, K$ in the physical element V_{i1} and extend it to the cell V_i. The remaining procedure remains exactly the same as in the case with all straight edges.

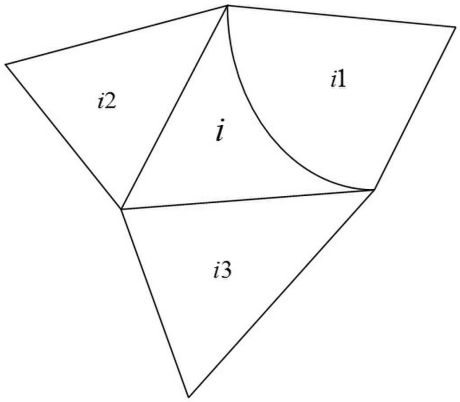


Figure 4: **The case that the target cell has curved edges**

Now we consider the case that the target cell V_i itself has curved edges as shown in Fig. 4. In the CPR framework, we transform $(x, y) \in V_i$ into $(\xi, \eta) \in \tilde{V}$ using Eq. (19). The Jacobian J of this transformation is a function of (ξ, η). Note that the transformed solution $\tilde{u}_i(\xi, \eta) = |J|u_i$ is a polynomial in $P^k(\tilde{V})$. Hence, $u_i = \tilde{u}_i/|J|$ is no longer a polynomial in the physical element (x, y). However, we still let $p_0(x, y) = u_i(x, y)$. The WENO limiter for this case has actually the same form as in the straight edges case, except that $p_0(x, y)$ is no longer a polynomial (or a vector of four polynomials in the system case) in the physical domain V_i. Hence, we need to be careful when doing the computations with p_0. Also, since V_i has curved edges, we need to transform it into $\tilde{V}$ when computing integrals on it. For example, we use the following equation to compute the cell averages on V_i:

$$\frac{1}{|V_i|}\int_{V_i} p_l(x, y)dxdy = \frac{1}{|V_i|}\int_{\tilde{V}} p_l(\xi, \eta)|J|(\xi, \eta)d\xi d\eta. \tag{46}$$

For $l = 0$, we know that $p_0(\xi, \eta)|J|(\xi, \eta) = \tilde{u}_i(\xi, \eta)$ is a polynomial in $P^k(\tilde{V})$. Hence the above integral is easy to compute. For $l = 1, 2, 3$, we know the expression of $p_l(x, y)$ which is a polynomial in (x, y). Using Eq. (19) which describes the relationship between (x, y) and (ξ, η), we can obtain the expression of $p_l(\xi, \eta)$ as well. Hence, we can compute the above integral.

4 Positivity-preserving limiter

In this section, we extend the positivity-preserving limiters originally designed for the DG schemes in [26, 36] to the CPR framework on triangular meshes. At each time level of the CPR framework, we first reconstruct the solution functions using the WENO limiter described in Section 3 to control oscillations, and thus can obtain a new set of numerical solutions at the solution points. Now we further modify these solution values in each cell to enforce the positivity of the solutions. The other procedures will remain the same as in the original CPR framework. As shown in [36] and [26], these limiters will not harm the original high order accuracy of CPR.

In Sections 4.1 and 4.2, we also assume that all the edges of the cell are straight. We will discuss the extensions to the curved boundary cells in Section 4.3.

4.1 Positivity-preserving limiter for scalar conservation laws

As shown in Equation (18), the CPR framework is conservative. Note that we only reconstruct the solution polynomials during the WENO limiting procedure and use the same CPR procedure thereafter. Hence, the solution after the WENO limiting procedure is also conservative. Assuming that $u_{i,j}^{new}$, $j = 1, \cdots, K$ are the new solution values in V_i modified by the WENO limiter and $u_i^{new}(x, y)$ is the new solution polynomial interpolating them, we have

$$\int_{V_i} (u_i^{new})_t dxdy + \int_{\partial V_i} \hat{F}(u_i^{new}, u_{i+}^{new}, \vec{\nu}) ds = 0, \tag{47}$$

Considering the Euler forward time discretization of this equation, we can get

$$\bar{u}_i^{n+1} = \bar{u}_i^n - \frac{\Delta t}{|V_i|} \int_{\partial V_i} \hat{F}(u_i^{new}, u_{i+}^{new}, \vec{\nu}) ds, \tag{48}$$

where $\bar{u}_i^n$ denotes the cell average of u_i^{new} on V_i at time level n:

$$\bar{u}_i^n = \frac{1}{|V_i|} \int_{V_i} u_i^{new} dxdy. \tag{49}$$

Note that Equation (48) is the same as the scheme satisfied by the cell averages of a DG method. Hence, we can extend the limiters in [36] and [26] designed for the DG method easily to the CPR framework. Here we can choose the global Lax-Friedrichs flux

$$\hat{F}(u,v,\vec{\nu}) = \frac{1}{2}\big(\vec{F}(u)\cdot\vec{\nu} + \vec{F}(v)\cdot\vec{\nu} - a(v-u)\big), \tag{50}$$

where $a = \max_{u,\vec{\nu}} |\vec{F}'(u)\cdot\vec{\nu}|$.

Assuming $\bar{u}_i^n \geqslant 0$ and the edge integral in Equation (48) is approximated by the $(k+1)$-point Gauss quadrature, a sufficient condition of the positivity property $\bar{u}_i^{n+1} \geqslant 0$ is that each $u_i^{new}(x,y)$ satisfies $u_i^{new}(x,y) \geqslant 0, \forall (x,y) \in Q_i$, under the CFL condition

$$a\frac{\Delta t}{|V_i|}\sum_{l=1}^{3} S_l \leqslant \frac{2}{3}\hat{\omega}_1. \tag{51}$$

Here $\hat{\omega}_1 = \frac{1}{6}$ for $k = 2,3$ and $\hat{\omega}_1 = \frac{1}{12}$ for $k = 4,5$. In the barycentric coordinates, the set Q_i can be written as

$$\begin{aligned} Q_i = \{&\big(\frac{1}{2}+v^\beta, (\frac{1}{2}+\hat{u}^\alpha)(\frac{1}{2}-v^\beta), (\frac{1}{2}-\hat{u}^\alpha)(\frac{1}{2}-v^\beta)\big), \\ &\big((\frac{1}{2}-\hat{u}^\alpha)(\frac{1}{2}-v^\beta), \frac{1}{2}+v^\beta, (\frac{1}{2}+\hat{u}^\alpha)(\frac{1}{2}-v^\beta)\big), \\ &\big((\frac{1}{2}+\hat{u}^\alpha)(\frac{1}{2}-v^\beta), (\frac{1}{2}-\hat{u}^\alpha)(\frac{1}{2}-v^\beta), \frac{1}{2}+v^\beta\big) : \\ &\alpha = 1,\cdots,M; \beta = 1,\cdots,k+1\}, \end{aligned} \tag{52}$$

where M is the smallest integer such that $2M-3 \geqslant k$, $\{\hat{u}^\alpha : \alpha = 1,\cdots,M\}$ denote the Gauss-Lobatto quadrature points on $[-\frac{1}{2},\frac{1}{2}]$ and $\{\hat{v}^\beta : \beta = 1,\cdots,k+1\}$ denote the Gauss quadrature points on $[-\frac{1}{2},\frac{1}{2}]$. For more details of this conclusion, we refer to see [36].

We now modify the solution polynomial $u_i^{new}(x,y)$ to get $\tilde{\tilde{u}}_i(x,y)$ such that $\tilde{\tilde{u}}_i(x,y) \geqslant 0, \forall (x,y) \in Q_i$. For all i, assume $\bar{u}_i^n \geqslant 0$, we define the following modified polynomial

$$\tilde{\tilde{u}}_i(x,y) = \theta(u_i^{new}(x,y) - \bar{u}_i^n) + \bar{u}_i^n, \qquad \theta = \min\left\{\left|\frac{\bar{u}_i^n}{\bar{u}_i^n - m_i}\right|, 1\right\}, \tag{53}$$

with $m_i = \min_{(x,y)\in Q_i} u_i^{new}(x,y)$. In the case of $m_i \geqslant 0$, we have $\theta = 1$ and $\tilde{\tilde{u}}_i(x,y) = u_i^{new}(x,y) \geqslant m_i \geqslant 0, \forall (x,y) \in Q_i$. In the case of $m_i < 0$, we have $\theta = \frac{\bar{u}_i^n}{\bar{u}_i^n - m_i}$ and

$\tilde{\tilde{u}}_i(x) \geqslant \theta(m_i - \bar{u}_i^n) + \bar{u}_i^n = 0, \forall (x,y) \in Q_i$. Then we replace the numerical solutions $u_{i,j}^{new}, j = 1, \cdots, K$ by the modified numerical solutions $\tilde{\tilde{u}}_i(\vec{r}_{i,j})$, $j = 1, \cdots, K$ for all i. The other procedures remain unchanged as in the original CPR framework with the WENO limiter.

We can also use SSP high order time discretization and it will keep the positivity-preserving property because of the convexity. In this case, we need to perform the procedure above in each stage for a Runge-Kutta method or in each step for a multistep method.

4.2 Positivity-preserving limiter for the Euler equations

Consider the two-dimensional Euler system which is given by Equation (35). We define the set of admissible states as

$$G = \left\{ \mathbf{u} = \begin{pmatrix} \rho \\ m \\ n \\ E \end{pmatrix} \middle| \, \rho > 0 \, and \, p(\mathbf{u}) > 0 \right\}. \tag{54}$$

Denote the solution polynomial reconstructed by the WENO limiter in the cell V_i at time level n as $\mathbf{u}_i^{new}(x,y) = \big(\rho_i(x,y), m_i(x,y), n_i(x,y), E_i(x,y)\big)^T$ with the cell average $\bar{\mathbf{u}}_i^n = \big(\bar{\rho}_i^n, \bar{m}_i^n, \bar{n}_i^n, \bar{E}_i^n\big)^T$. Since the CPR method is conservative, the scheme satisfied by the cell averages in the CPR method with Euler forward time discretization is

$$\bar{\mathbf{u}}_i^{n+1} = \bar{\mathbf{u}}_i^n - \frac{\Delta t}{|V_i|} \int_{\partial V_i} \hat{\mathbf{F}}(\mathbf{u}_i^{new}, \mathbf{u}_{i+}^{new}, \vec{\nu}) ds = 0. \tag{55}$$

Here, we also consider the Lax-Friedrichs flux

$$\hat{\mathbf{F}}(\mathbf{u}, \mathbf{v}, \vec{\nu}) = \frac{1}{2}\big(\mathbf{F}(\mathbf{u}) \cdot \vec{\nu} + \mathbf{F}(\mathbf{v}) \cdot \vec{\nu} - a(\mathbf{v} - \mathbf{u})\big), \tag{56}$$

where $\mathbf{F} = (\mathbf{f}, \mathbf{g})^T$, $a = \max\{\|(|u| + c)\|_\infty, \|(|v| + c)\|_\infty\}$.

Assuming the edge integral is approximated by the $(k+1)$-point Gauss quadrature, a sufficient condition of the positivity property $\bar{\mathbf{u}}_i^{n+1} \in G$ is that each $\mathbf{u}_i^{new}(x,y)$ satisfies $\mathbf{u}_i^{new}(x,y) \in G, \forall (x,y) \in Q_i$, where Q_i is defined in (52), under the CFL condition (51).

We use the following algorithm to get a new modified solution polynomial $\tilde{\tilde{\mathbf{u}}}_i(x,y)$ such that $\tilde{\tilde{\mathbf{u}}}_i(x,y) \in G, \forall (x,y) \in Q_i$.

1. In each cell, we enforce the positivity of density first. Set up a small number $\varepsilon > 0$ such that $\bar{\rho}_i^n \geqslant \varepsilon$ for all i. In practice, we can choose $\varepsilon = 10^{-13}$. Replace $\rho_i(x,y)$ by

$$\hat{\rho}_i(x,y) = \theta_1\big(\rho_i(x,y) - \bar{\rho}_i^n\big) + \bar{\rho}_i^n, \qquad \theta_1 = \min\{\frac{\bar{\rho}_i^n - \varepsilon}{\bar{\rho}_i^n - \rho_{min}}, 1\}, \tag{57}$$

   with $\rho_{min} = \min_{(x,y)\in Q_i} \rho_i(x,y)$. Now we have $\hat{\rho}_i(x,y) \geqslant \varepsilon > 0, \forall (x,y) \in Q_i$. This fact is proved in [36].

2. The second step is to enforce the positivity of the pressure. Define $\hat{\mathbf{u}}_i(x,y) = \big(\hat{\rho}_i(x,y), m_i(x,y), n_i(x,y), E_i(x,y)\big)^T$. For each $(x,y) \in Q_i$, if $p(\hat{\mathbf{u}}_i(x,y)) \geqslant 0$, set $\theta_{x,y} = 1$; otherwise, set

$$\theta_{x,y} = \frac{p(\bar{\mathbf{u}}_i^n)}{p(\bar{\mathbf{u}}_i^n) - p(\hat{\mathbf{u}}_i(x,y))}. \tag{58}$$

   Then we get the limited polynomial

$$\tilde{\tilde{\mathbf{u}}}_i(x,y) = \theta_2(\hat{\mathbf{u}}_i(x,y) - \bar{\mathbf{u}}_i^n) + \bar{\mathbf{u}}_i^n, \qquad \theta_2 = \min_{(x,y)\in Q_i} \theta_{x,y}. \tag{59}$$

   The proof of $p(\tilde{\tilde{\mathbf{u}}}_i(x,y)) \geqslant 0, \forall (x,y) \in Q_i$ can be found in [26]. It can be shown that this limiter does not destroy accuracy for smooth solutions [26].

We replace $\mathbf{u}_{i,j}^{new}$, $j = 1, \cdots, K$ in V_i by the modified solutions $\tilde{\tilde{\mathbf{u}}}_i(\vec{r}_{i,j})$, $j = 1, \cdots, K$ for all i. The other procedures remain unchanged as in the original CPR framework with the WENO limiter.

We can also use SSP high order time discretization and it will keep the positivity-preserving property because of the convexity. In this case, we need to perform the procedure above in each stage for a Runge-Kutta method or in each step for a multistep method.

4.3 Extension to curved boundary cells

In this subsection, we consider the cell with curved edges, for example, V_i. Assume that the new solutions in V_i modified by the WENO limiter are $u_{i,j}^{new}, j = 1, \cdots, K$ (scalars or vectors). As described in Section 2.2, we transform V_i into the standard element $\tilde{V}$ with the Jacobian denoted by J and then perform the normal CPR procedure on the transformed solutions $\tilde{u}_{i,j}^{new} = |J|u_{i,j}^{new}, j = 1, \cdots, K$ in $\tilde{V}$ using Eq. (25). Integrating Eq. (25) over cell $\tilde{V}$ with test function $w = 1$, we can obtain

$$\int_{\tilde{V}} (\tilde{u}_i^{new})_t d\xi d\eta + \int_{\partial\tilde{V}} \hat{\tilde{F}}^{new} ds = 0, \tag{60}$$

where $\tilde{u}_i^{new}(\xi, \eta)$ is a polynomial in $P^k(\tilde{V})$ interpolating $\tilde{u}_{i,j}^{new}$, $j = 1, \cdots, K$ and $\hat{\tilde{F}}^{new}$ is the transformed common normal flux in $\tilde{V}$. This equation has the same form as the equation satisfied by $u_i^{new}(x, y)$ in V_i.

Replacing $u_i^{new}(x, y)$ with $\tilde{u}_i^{new}(\xi, \eta)$, we transform the problem on V_i into a problem on $\tilde{V}$ which has straight edges. Now we only need to use the same positivity-preserving limiter described in Section 4.1 for the scalar case or the limiter described in Section 4.2 for the system case to modify $\tilde{u}_i^{new}(\xi, \eta)$ in $\tilde{V}$. For the scalar case, we can get $\bar{\tilde{u}}_i^{n+1} \geqslant 0$ after the limiting procedure, where $\bar{\tilde{u}}_i^{n+1}$ denotes the cell average of $\tilde{u}_i^{new}$ on $\tilde{V}$ at time level $n + 1$:

$$\bar{\tilde{u}}_i^{n+1} = \frac{1}{|\tilde{V}|} \int_{\tilde{V}} \tilde{u}_i^{new} d\xi d\eta. \tag{61}$$

Note the fact that

$$\bar{u}_i^{n+1} = \frac{1}{|V_i|} \int_{V_i} u_i^{new} dxdy = \frac{1}{|V_i|} \int_{\tilde{V}} u_i^{new} |J| d\xi d\eta = \frac{1}{|V_i|} \int_{\tilde{V}} \tilde{u}_i^{new} d\xi d\eta = \frac{1}{2|V_i|} \bar{\tilde{u}}_i^{n+1}. \tag{62}$$

Hence, we also have $\bar{u}_i^{n+1} \geqslant 0$.

For the system case, although the transformed equation is no longer an Euler equation, the numerical flux can still be computed using the flux of the original Euler equation. Suppose $\vec{\tilde{\nu}} = (\tilde{\nu}^{\xi}, \tilde{\nu}^{\eta})$ is the unit outward normal vector of $\tilde{V}$. Note that

$$\begin{aligned} \tilde{\mathbf{f}} &= \xi_x |J| \mathbf{f}(\mathbf{u}) + \xi_y |J| \mathbf{g}(\mathbf{u}) \\ &= \xi_x \mathbf{f}(\tilde{\mathbf{u}}) + \xi_y \mathbf{g}(\tilde{\mathbf{u}}) \end{aligned} \tag{63}$$

and

$$\tilde{\mathbf{g}} = \eta_x \mathbf{f}(\tilde{\mathbf{u}}) + \eta_y \mathbf{g}(\tilde{\mathbf{u}}), \tag{64}$$

we have

$$\begin{aligned} \tilde{\mathbf{F}} \cdot \vec{\tilde{\nu}} &= (\tilde{\mathbf{f}}, \tilde{\mathbf{g}}) \cdot \vec{\tilde{\nu}} = \big(\xi_x \mathbf{f}(\tilde{\mathbf{u}}) + \xi_y \mathbf{g}(\tilde{\mathbf{u}}), \eta_x \mathbf{f}(\tilde{\mathbf{u}}) + \eta_y \mathbf{g}(\tilde{\mathbf{u}})\big) \cdot \vec{\tilde{\nu}} \\ &= \big(\mathbf{f}(\tilde{\mathbf{u}}), \mathbf{g}(\tilde{\mathbf{u}})\big) \cdot \vec{r}, \end{aligned} \tag{65}$$

where $\vec{r}(\xi, \eta) = (\tilde{\nu}^{\xi} \xi_x + \tilde{\nu}^{\eta} \eta_x, \tilde{\nu}^{\xi} \xi_y + \tilde{\nu}^{\eta} \eta_y)$. Hence, the numerical flux of the transformed equation can be computed by the flux of the Euler equation multiplying $|\vec{r}|$:

$$\hat{\tilde{\mathbf{F}}}^{new} = \hat{\mathbf{F}}(\tilde{\mathbf{u}}_i^{new}, \tilde{\mathbf{u}}_{i+}^{new}, \frac{\vec{r}}{|\vec{r}|}) \, |\vec{r}|. \tag{66}$$

Now, the scheme satisfied by the cell averages of the transformed solution with Euler forward time discretization is

$$\bar{\tilde{\mathbf{u}}}_i^{n+1} = \bar{\tilde{\mathbf{u}}}_i^{n} - \frac{\Delta t}{|\tilde{V}|} \int_{\partial \tilde{V}} \hat{\mathbf{F}}(\tilde{\mathbf{u}}_i^{new}, \tilde{\mathbf{u}}_{i+}^{new}, \frac{\vec{r}}{|\vec{r}|}) \, |\vec{r}| ds. \tag{67}$$

By denoting $\{(\xi_{l,\beta}, \eta_{l,\beta}) : \beta = 1, \cdots, k+1\}$ as the Gauss quadrature points on the $l-th$ edge of $\tilde{V}$, we can get $\bar{\tilde{\mathbf{u}}}_i^{n+1} \in G$ under the CFL condition

$$a \frac{\Delta t}{|\tilde{V}|} \max_{\beta} \sum_{l=1}^{3} \tilde{S}_l |\vec{r}(\xi_{l,\beta}, \eta_{l,\beta})| \leqslant \frac{2}{3} \hat{\omega}_1. \tag{68}$$

We omit the details of the proof since it is almost the same as in [36]. Since $\bar{\mathbf{u}}_i^{n+1} = \frac{1}{2|V_i|} \bar{\tilde{\mathbf{u}}}_i^{n+1}$, we also have $\bar{\mathbf{u}}_i^{n+1} \in G$.

5 Numerical results

In this section, we provide numerical experiments to demonstrate the performance of the WENO limiter for the CPR framework on unstructured meshes.

We use the third order TVD Runge-Kutta method [25] for the time discretization. Second, third and fourth order CPR schemes with the WENO limiter are tested in the accuracy test examples where the CFL numbers are set to be 0.3. Since the time discretization is only third order accurate, we take $\Delta t \sim \Delta x^{4/3}$ to obtain the fourth order accurate results for the accuracy test examples. For the examples containing

discontinuities, the positivity-preserving limiter is used for the third order scheme. To satisfy the CFL condition (51) in case the positivity-preserving limiter is used, we use the CFL number 0.1 in all the examples containing discontinuities.

For all the numerical experiments, we use a non-uniform triangulation. The refinement is performed by a structured refinement (we simply break each triangle into four similar smaller triangles for each level of the refinement).

As we know, there are several different kinds of methods in the CPR framework. For δ_i defined by (7), if the test function w varies on different spaces, we obtain formulations corresponding to different methods [29]. In all our numerical examples, we choose $w \in P^k$, which result in a method similar to the DG formulation.

For the purpose of artificially generating a larger percentage of troubled cells in order to test accuracy when the WENO reconstruction procedure is enacted in more cells, we adjust the constant C_k in different examples when using KXRCF technique to identify troubled cells. We list in each table the percentage of troubled cells among all the cells.

Example 1. Consider the two-dimensional nonlinear scalar Burgers equation

$$u_t + (\frac{u^2}{2})_x + (\frac{u^2}{2})_y = 0, \qquad 0 \le x, y \le 2\pi, \tag{69}$$

with the initial condition $u(x, y, 0) = 0.5 + \sin(x + y)$ and periodic boundary conditions in both directions. For this test case the coarsest mesh we have used is shown in Figure 5. We show the errors and numerical orders of accuracy for the CPR method with the WENO limiter comparing with the original CPR method without limiter at $t = 0.25$ (smooth solution) in Table 1. We list in the last column of the table the percentage of troubled cells among all the cells. We can see that the WENO limiter keeps the designed order of accuracy, even when a large percentage of good cells are artificially identified as troubled cells.

At $t = 0.5$, a shock begins to appear in the solution. We plot the solution surfaces at $t = 0.75$ in Figure 6. We can see that the schemes give non-oscillatory shock transitions for this problem.

Table 1: 2D Burgers equation at $t = 0.25$.

cell	CPR without limiter				CPR with WENO limiter ($Ck = 0.01$)				
length	L^1 norm	order	L^∞ norm	order	L^1 norm	order	L^∞ norm	order	percentage
	2nd order ($C_k = 0.01$)								
$2\pi/10$	2.51E-02	–	1.68E-01	–	4.01E-02	–	3.88E-01	–	83.75%
$2\pi/20$	6.01E-03	2.06	5.49E-02	1.61	8.44E-03	2.25	8.67E-02	2.16	79.48%
$2\pi/40$	1.54E-03	1.97	1.88E-02	1.55	1.80E-03	2.23	1.86E-02	2.22	59.22%
$2\pi/80$	3.93E-04	1.97	8.74E-03	1.10	4.09E-04	2.14	8.64E-03	1.11	41.20%
$2\pi/160$	1.00E-04	1.97	4.21E-03	1.05	1.00E-04	2.03	4.17E-03	1.05	17.60%
	3rd order ($C_k = 0.01$)								
$2\pi/10$	2.58E-03	–	7.06E-02	–	2.57E-03	–	7.08E-02	–	17.92%
$2\pi/20$	3.97E-04	2.70	1.43E-02	2.30	3.96E-04	2.70	1.44E-02	2.30	5.83%
$2\pi/40$	5.96E-05	2.74	2.68E-03	2.42	5.92E-05	2.74	2.71E-03	2.41	1.38%
$2\pi/80$	8.89E-06	2.75	4.87E-04	2.46	8.89E-06	2.74	4.87E-04	2.48	0.06%
$2\pi/160$	1.30E-06	2.78	7.20E-05	2.76	1.30E-06	2.78	7.20E-05	2.76	0.00%
	4th order ($C_k = 0.001$)								
$2\pi/10$	4.52E-04	–	1.92E-02	–	4.55E-04	–	1.92E-02	–	21.67%
$2\pi/20$	3.40E-05	3.73	2.69E-03	2.83	3.37E-05	3.76	2.70E-03	2.83	5.83%
$2\pi/40$	2.22E-06	3.94	2.72E-04	3.31	2.19E-06	3.94	2.73E-04	3.31	1.46%
$2\pi/80$	1.46E-07	3.92	1.62E-05	4.07	1.45E-07	3.92	1.62E-05	4.07	0.08%
$2\pi/160$	1.02E-08	3.85	1.08E-06	3.91	1.01E-08	3.84	1.08E-06	3.91	0.00%

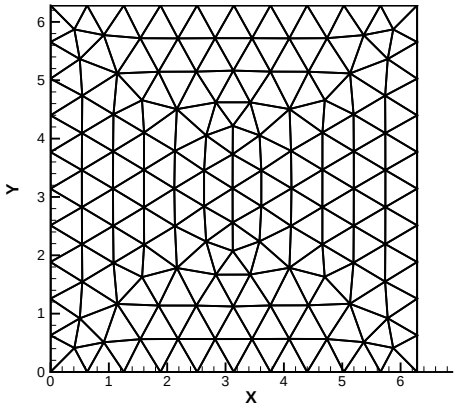
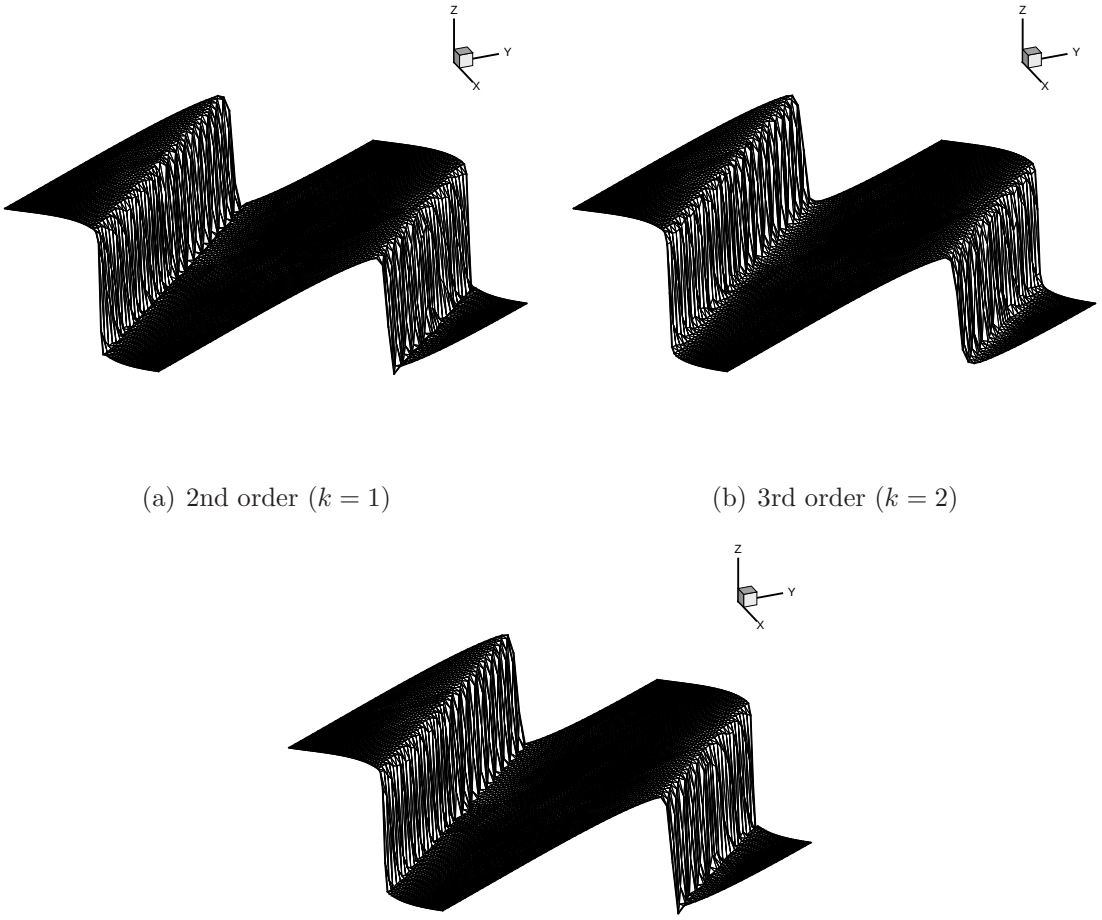


Figure 5: **Burgers equation. The coarsest mesh. The mesh points on the boundary are uniformly distributed with cell length** $2\pi/10$**.**



(a) 2nd order ($k = 1$)

(b) 3rd order ($k = 2$)

(c) 4th order ($k = 3$)

Figure 6: **2D Burgers solution at** $t = 0.75$**.**

Example 2. Let us consider the two-dimensional Euler system (35). The initial condition is set to be $\rho(x, y, 0) = 1 + 0.2\sin(x + y)$, $u(x, y, 0) = 0.7$, $v(x, y, 0) = 0.3$ and $p(x, y, 0) = 1$, $0 \leq x, y \leq 2\pi$. The boundary conditions are periodic. $\gamma = 1.4$ is used in the computation. The exact solution is $\rho(x, y, t) = 1 + 0.2\sin(x + y - t)$, $u(x, y, t) = 0.7$, $v(x, y, t) = 0.3$ and $p(x, y, t) = 1$. For this test case, we use the same mesh as in Example 1. The coarsest mesh is show in Figure 5. Table 2 shows the L_1 and L_∞ errors and numerical orders of accuracy of the density at $t = 2\pi$. Similar to the previous example, we can see that the WENO limiter again keeps the designed order of accuracy.

Table 2: 2D Euler equation with initial condition $\rho(x,y,0) = 1+0.2\sin(x+y)$, $u(x,y,0) = 0.7$, $v(x,y,0) = 0.3$ and $p(x,y,0) = 1$ at $t = 2\pi$.

cell	CPR without limiter				CPR with WENO limiter				
length	L^1 norm	order	L^∞ norm	order	L^1 norm	order	L^∞ norm	order	percentage
	2nd order ($C_k = 0.01$)								
$2\pi/10$	3.60E-03	–	1.16E-02	–	7.84E-03	–	3.19E-02	–	17.92%
$2\pi/20$	6.41E-04	2.49	3.43E-03	1.76	6.79E-04	3.53	3.92E-03	3.02	1.15%
$2\pi/40$	1.42E-04	2.17	9.42E-04	1.86	1.42E-04	2.25	9.42E-04	2.06	0.00%
$2\pi/80$	3.41E-05	2.06	2.61E-04	1.85	3.41E-05	2.06	2.61E-04	1.85	0.00%
$2\pi/160$	8.41E-06	2.02	6.89E-05	1.92	8.41E-06	2.02	6.89E-05	1.92	0.00%
	3rd order ($C_k = 0.0001$)								
$2\pi/10$	4.59E-04	–	3.94E-03	–	6.04E-04	–	4.89E-03	–	93.33%
$2\pi/20$	6.17E-05	2.89	6.51E-04	2.60	1.10E-04	2.46	6.71E-04	2.87	84.58%
$2\pi/40$	7.53E-06	3.03	9.16E-05	2.83	1.66E-05	2.72	1.14E-04	2.56	60.52%
$2\pi/80$	9.09E-07	3.05	1.23E-05	2.89	1.12E-06	3.89	2.07E-05	2.45	3.89%
$2\pi/160$	1.11E-07	3.04	1.60E-06	2.94	1.11E-07	3.34	1.60E-06	3.69	0.00%
	4th order ($C_k = 0.00001$)								
$2\pi/10$	1.52E-05	–	1.97E-04	–	1.44E-04	–	8.26E-04	–	88.75%
$2\pi/20$	8.46E-07	4.17	1.45E-05	3.76	1.08E-05	3.73	1.45E-04	2.51	43.33%
$2\pi/40$	5.15E-08	4.04	8.92E-07	4.02	1.48E-07	6.19	3.29E-06	5.46	2.16%
$2\pi/80$	3.18E-09	4.02	5.68E-08	3.97	3.17E-09	5.54	5.69E-08	5.85	0.00%
$2\pi/160$	1.97E-10	4.01	3.58E-09	3.99	1.97E-10	4.01	3.59E-09	3.99	0.00%

Example 3. Consider the two-dimensional vortex evolution problem, which is an idealized problem for the 2D Euler equations [24]. The set up of this problem is: The mean flow is $\rho = 1$, $p = 1$ and $(u, v) = (1, 1)$ (diagonal flow). We add, to this mean flow, an isentropic vortex (perturbation in (u, v) and the temperature $T = \frac{p}{\rho}$, no perturbation in the entropy $S = \frac{p}{\rho^\gamma}$):

$$(\delta u, \delta v) = \frac{\epsilon}{2\pi} e^{0.5(1-t^2)}(-\bar{y}, \bar{x}), \qquad \delta T = -\frac{(\gamma-1)\epsilon^2}{8\gamma\pi^2} e^{1-r^2}, \qquad \delta S = 0, \tag{70}$$

where $(\bar{x}, \bar{y}) = (x-7, y-7)$, $r^2 = \bar{x}^2 + \bar{y}^2$, and the vortex strength $\epsilon = 5$. The computational domain is taken as $[0, 14] \times [0, 14]$, extended periodically in both directions. It is clear that the exact solution of the Euler equation with the above initial and boundary conditions is just the passive convection of the vortex with the mean velocity. For this test case the coarsest mesh we have used is shown in Figure 7. The errors and orders of accuracy for the density at $t = 0.2$ are shown in Table 3. We can see that the WENO

limiter maintains both the designed order of accuracy and the magnitude of the errors of the original CPR method.

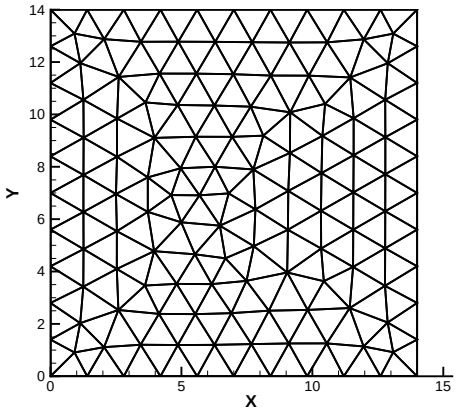


Figure 7: **2D Euler system of the smooth vortex evolution problem. The coarsest mesh. The mesh points on the boundary are uniformly distributed with cell length** 14/10**.**

Table 3: 2D Euler system of the smooth vortex evolution problem at $t = 0.2$.

cell	CPR without limiter				CPR with WENO limiter				
length	L^1 norm	order	L^∞ norm	order	L^1 norm	order	L^∞ norm	order	percentage
	2nd order ($C_k = 0.01$)								
14/10	2.81E-03	–	6.93E-02	–	2.69E-03	–	1.42E-01	–	8.97%
14/20	6.10E-04	2.20	2.03E-02	1.77	7.11E-04	1.92	4.36E-02	1.71	7.16%
14/40	1.23E-04	2.31	4.91E-03	2.05	1.30E-04	2.45	7.30E-03	2.58	2.80%
14/80	2.91E-05	2.09	1.56E-03	1.65	2.91E-05	2.16	1.55E-03	2.23	0.75%
14/160	7.21E-06	2.01	5.55E-04	1.49	7.20E-06	2.01	5.57E-04	1.48	0.20%
	3rd order ($C_k = 0.001$)								
14/10	6.40E-04	–	2.51E-02	–	6.76E-04	–	2.53E-02	–	16.24%
14/20	1.13E-04	2.50	7.82E-03	1.68	1.11E-04	2.61	7.86E-03	1.69	10.26%
14/40	1.76E-05	2.68	1.98E-03	1.98	1.73E-05	2.68	1.98E-03	1.99	4.78%
14/80	2.89E-06	2.61	2.96E-04	2.75	2.85E-06	2.60	3.00E-04	2.72	1.36%
14/160	4.67E-07	2.63	5.16E-05	2.52	4.67E-07	2.61	5.16E-05	2.54	0.00%
	4th order ($C_k = 0.0001$)								
14/10	1.64E-04	–	1.34E-02	–	9.15E-04	–	9.94E-02	–	25.64%
14/20	1.15E-05	3.84	1.15E-03	3.55	3.21E-05	4.83	4.98E-03	4.32	12.18%
14/40	7.94E-07	3.86	8.39E-05	3.77	1.43E-06	4.49	1.43E-04	5.12	3.77%
14/80	5.14E-08	3.95	5.90E-06	3.83	6.15E-08	4.54	9.23E-06	3.95	0.40%
14/160	3.09E-09	4.06	2.93E-07	4.33	3.09E-09	4.31	2.93E-07	4.98	0.00%

Example 4. We test the double Mach reflection problem. The computational domain is sketched in Figure 8. This problem is initialized by sending a horizontally moving Mach 10 shock into a solid wall inclined by a 30° angle. In order to impose the no-penetration boundary condition by the reflection technique, people usually solve an equivalent problem that puts the wall horizontal and puts the shock 60° angle inclined to the wall. We solve the original problem in this example. Initially the right-moving shock is positioned at $(0, 0)$. The initial pre-shock condition is

$$(\rho, p, u, v) = (1.4, 1, 0, 0), \tag{71}$$

and the post-shock condition is

$$(\rho, p, u, v) = (8, 116.5, 8.25, 0). \tag{72}$$

At $y = 0$, the exact post-shock condition is imposed. At the solid wall, a reflective boundary condition is used. At the top boundary, the flow values are set to describe the exact motion of the Mach 10 shock. Supersonic inflow and outflow boundary conditions are used at the left and right boundary, respectively. A sample mesh coarser than what is used is shown in Figure 9. $C_k = 0.01$ is used in this example. We compute the solutions of the second order and third order schemes up to $t = 0.2$. Figure 10 shows the density contours. The "zoomed-in" pictures around the double Mach stem to show more details are given in Figure 11. In all the plots, we use 29 contours equally distributed from $\rho = 1.3$ to 23. We can see that the resolution around the double Mach region improves with an increasing k.

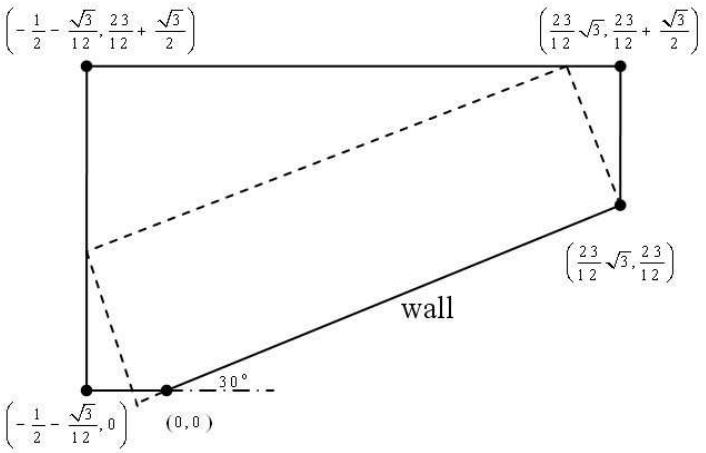


Figure 8: **The computational domain (solid line) of the double Mach reflection problem. The dashed line indicates the computational domain of the equivalent problem that puts the wall horizontal.**

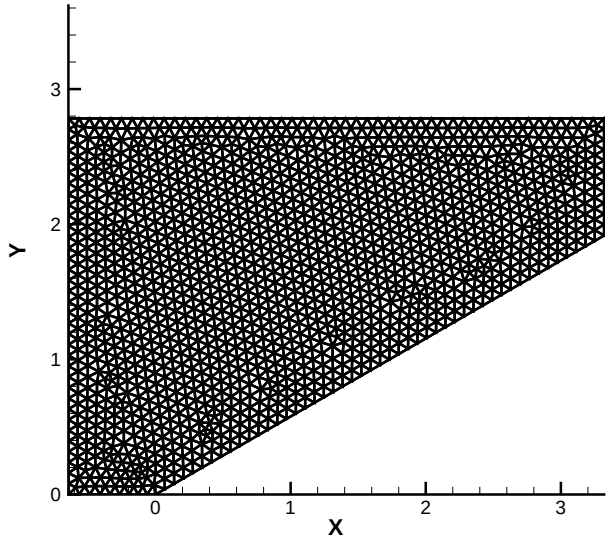


Figure 9: **Double Mach refection problem. Sample mesh. The mesh points on the boundary are uniformly distributed with cell length 0.08.**

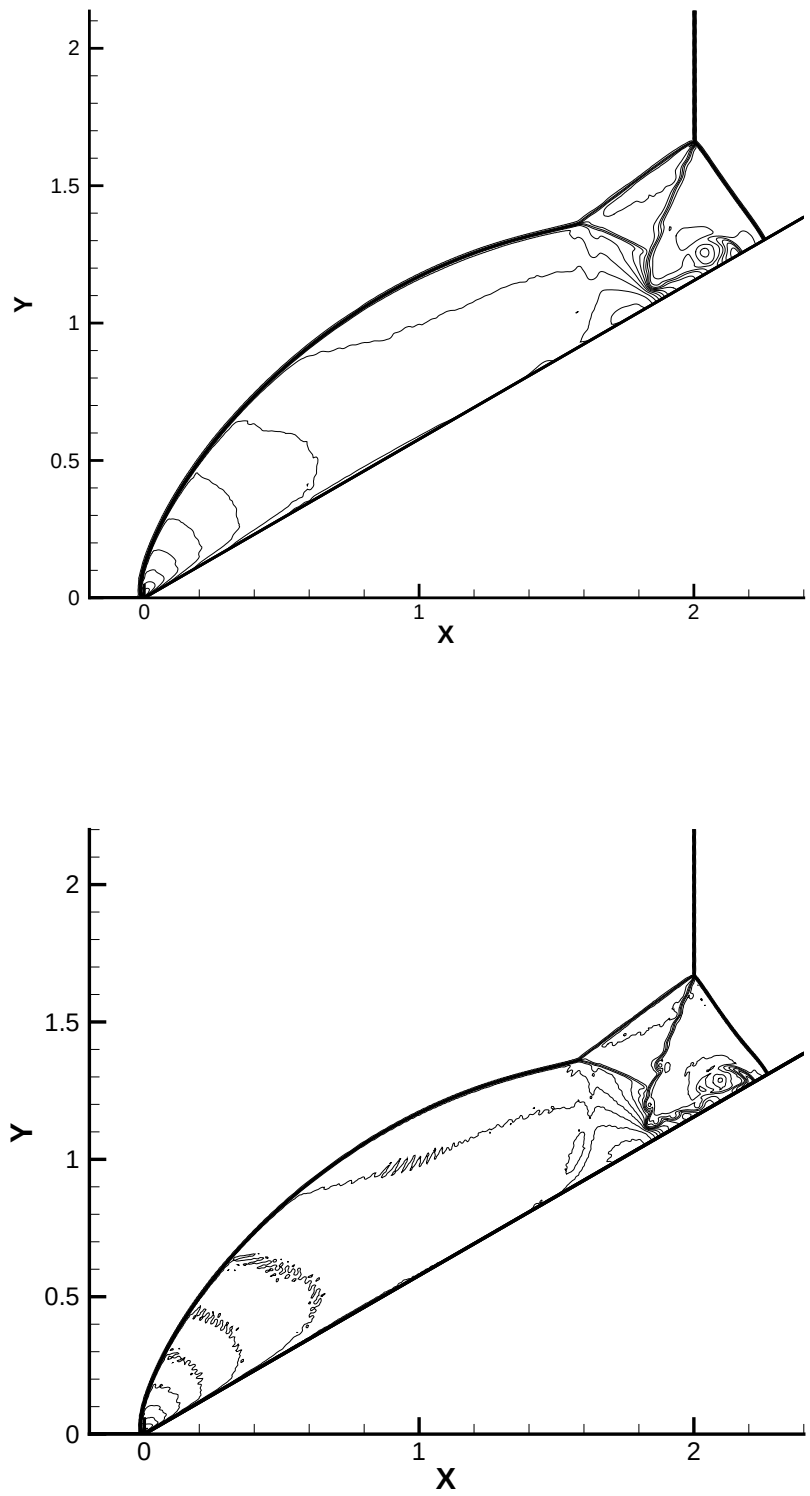


Figure 10: **Double Mach reflection problem. Top: second order ($k = 1$); bottom: third order ($k = 2$). 29 equally spaced density contours from 1.3 to 23. The mesh points on the boundary are uniformly distributed with cell length** $1/200$**.**

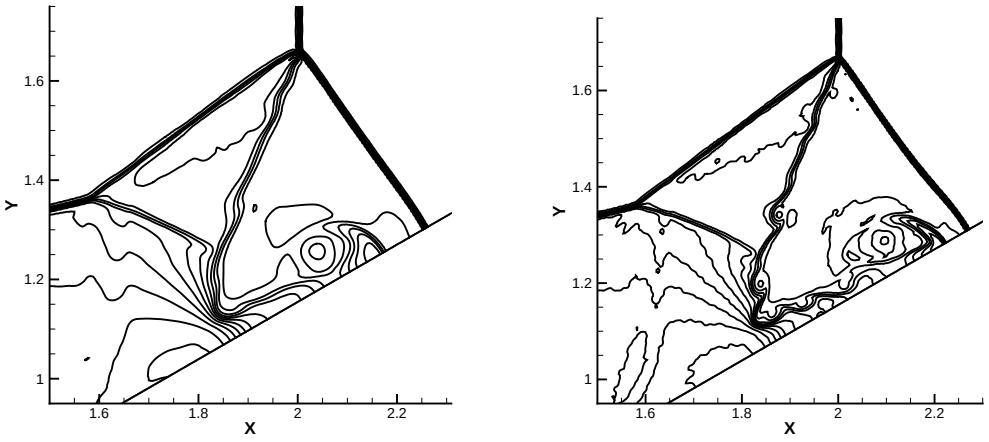


Figure 11: **Double Mach reflection problem. Zoom-in pictures around the Mach stem. Left: second order ($k = 1$); right: third order ($k = 2$). 29 equally spaced density contours from 1.3 to 23.**

Example 5. A Mach 3 wind tunnel with a step. The setup of the problem is 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 right-moving Mach 3 flow. Reflective boundary conditions are applied along the wall of the tunnel and inflow/outflow boundary conditions are applied at the entrance/exit. At the corner of the step, there is a singularity. For the second order scheme, we do not modify the scheme or refine the mesh near the corner. For the third order scheme, we adopt the Woodward-Collela fix technique [35] at the corner of the step. We present a sample triangulation coarser than what is used in Figure 12. $C_k = 0.01$ is used in this example. The results are shown at $t = 4$ in Figure 13. In all the plots, we use 30 contours equally distributed from 0.32 to 6.15. We can see that the third order scheme gives better resolution than the second order scheme.

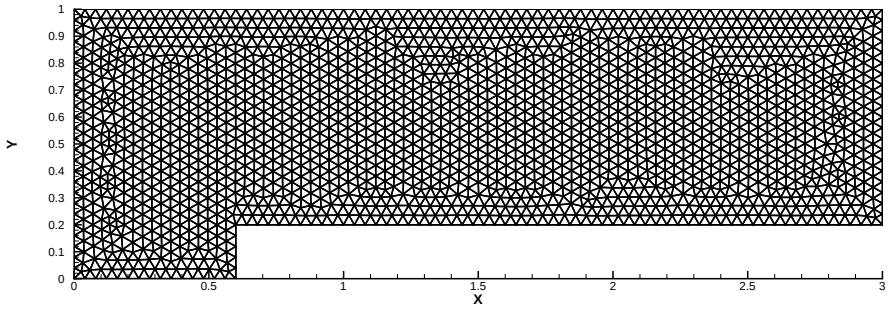


Figure 12: **Forward step problem. Sample mesh. The mesh points on the boundary are uniformly distributed with cell length $\frac{1}{25}$.**

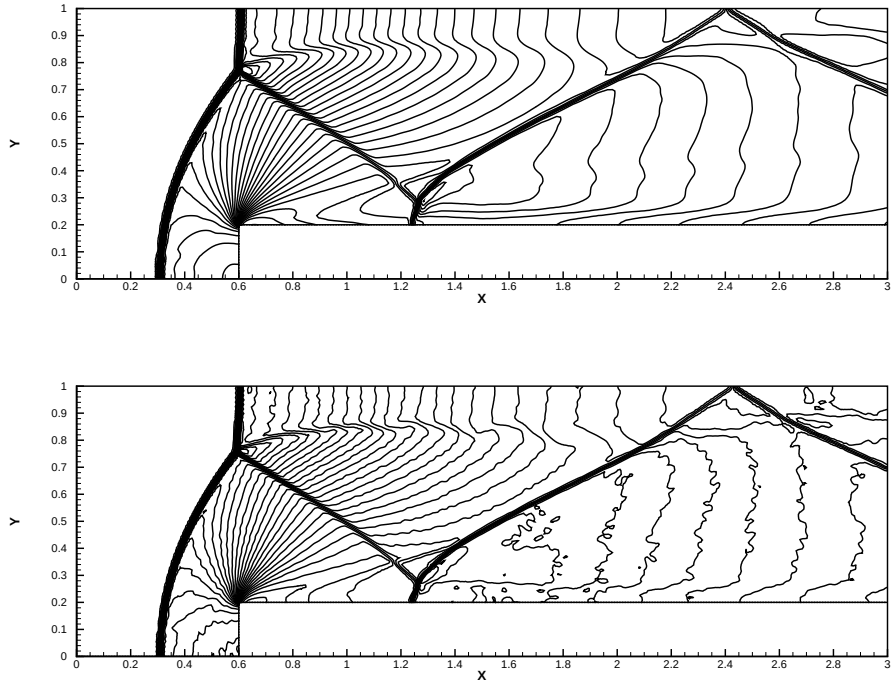


Figure 13: **Forward step problem. Top: second order ($k = 1$); bottom: third order ($k = 2$). 30 equally spaced density contours from 0.32 to 6.15. The mesh points on the boundary are uniformly distributed with cell length** $1/100$**.**

Example 6. Let consider the problem of a shock passing a backward facing corner (diffraction). The setup is the following: the computational domain is the union of $[0, 1] \times [6, 11]$ and $[1, 13] \times [0, 11]$; the initial condition is a pure right-moving Mach 5.09

shock, initially located at $x = 0.5$ and $6 \leqslant y \leqslant 11$, moving into undisturbed air ahead of the shock with a density of 1.4 and pressure of 1. The boundary conditions are inflow at $x = 0$, $6 \leqslant y \leqslant 11$, outflow at $x = 13$, $0 \leqslant y \leqslant 11$ and $1 \leqslant x \leqslant 13$, $y = 0$, and reflective at the walls $0 \leqslant x \leqslant 1$, $y = 6$ and $x = 1$, $0 \leqslant y \leqslant 6$. At the top boundary, we use the exact solution of a free-moving Mach 5.09 shock. We present a sample triangulation coarser than what is used in Figure 14. $C_k = 0.0001$ is used in this example. The density and pressure at $t = 2.3$ are presented in Figure 15.

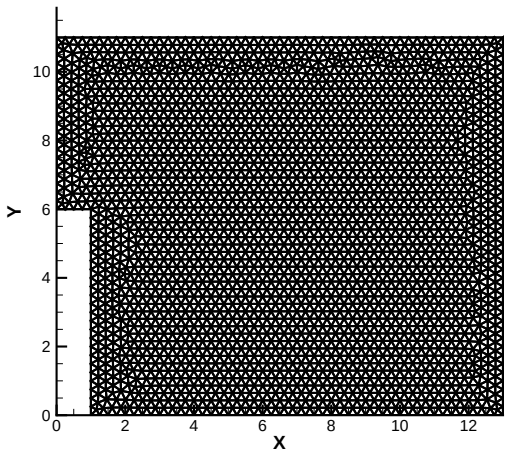


Figure 14: **Shock diffraction problem. Sample mesh. The mesh points on the boundary are uniformly distributed with cell length $\frac{1}{4}$.**

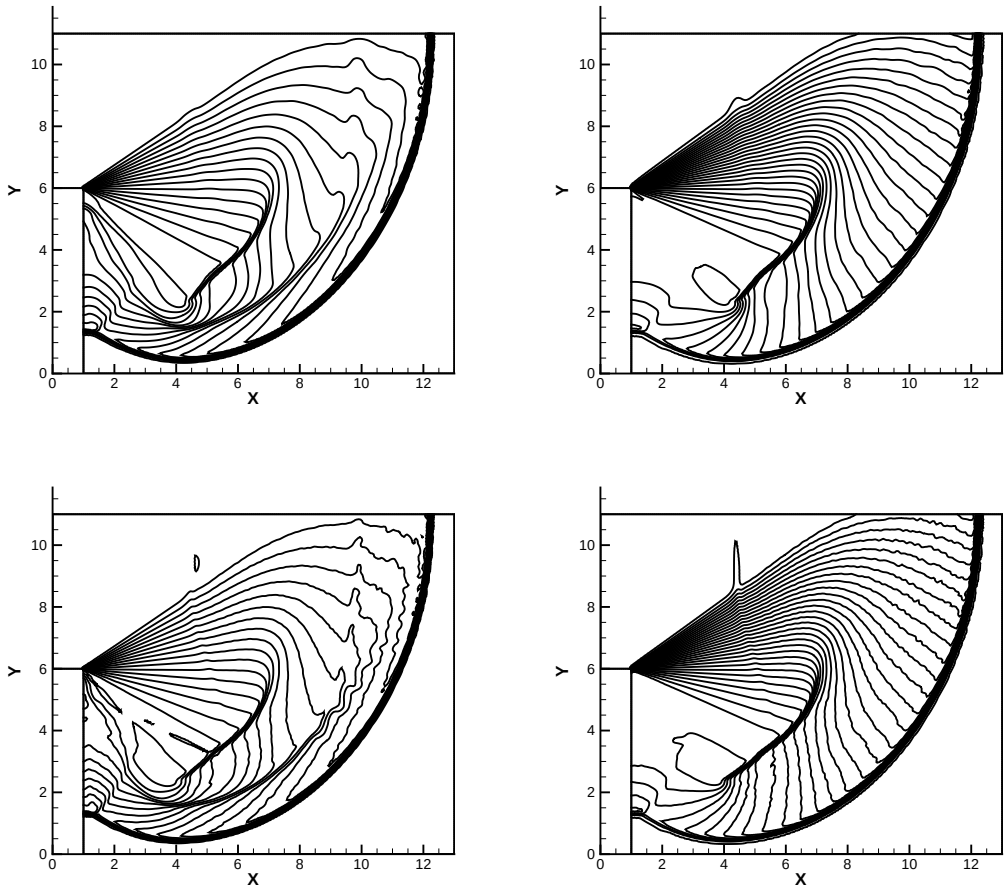


Figure 15: **Shock diffraction problem. Top: second order ($k = 1$); bottom: third order ($k = 2$). Left: density, 20 equally spaced contour lines from 0.066227 to 7.0668; right: pressure, 40 equally spaced contour lines from 0.091 to 37. The mesh points on the boundary are uniformly distributed with cell length** $1/16$**.**

Example 7. A shock wave diffracts at a convex corner. Here we study a Mach 10 shock diffracting at a 120° convex corner. The initial condition is a pure right-moving Mach 10 shock, initially located at $x = 3.4$ and $6 \leqslant y \leqslant 11$, moving into undisturbed air ahead of the shock with a density of 1.4 and pressure of 1. We present a sample triangulation coarser than what is used in Figure 16. $C_k = 0.0001$ is used in this example. The contour plots of density and pressure at $T = 0.9$ are given in Figure 17. The high order CPR method without the positivity limiter may blow up since the lowest density is very close

to zero.

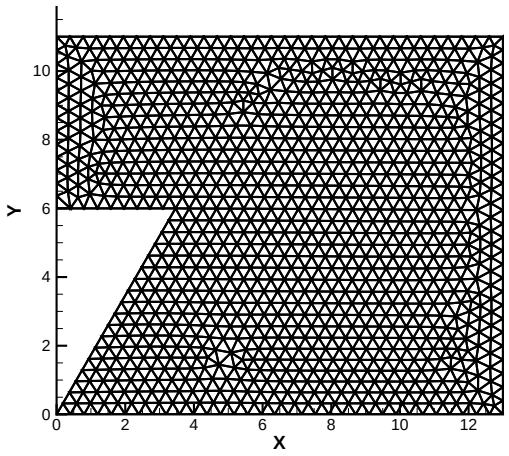


Figure 16: **Mach 10 Shock diffracting at a** 120° **corner. Sample mesh. The mesh points on the boundary are uniformly distributed with cell length** 0.4**.**

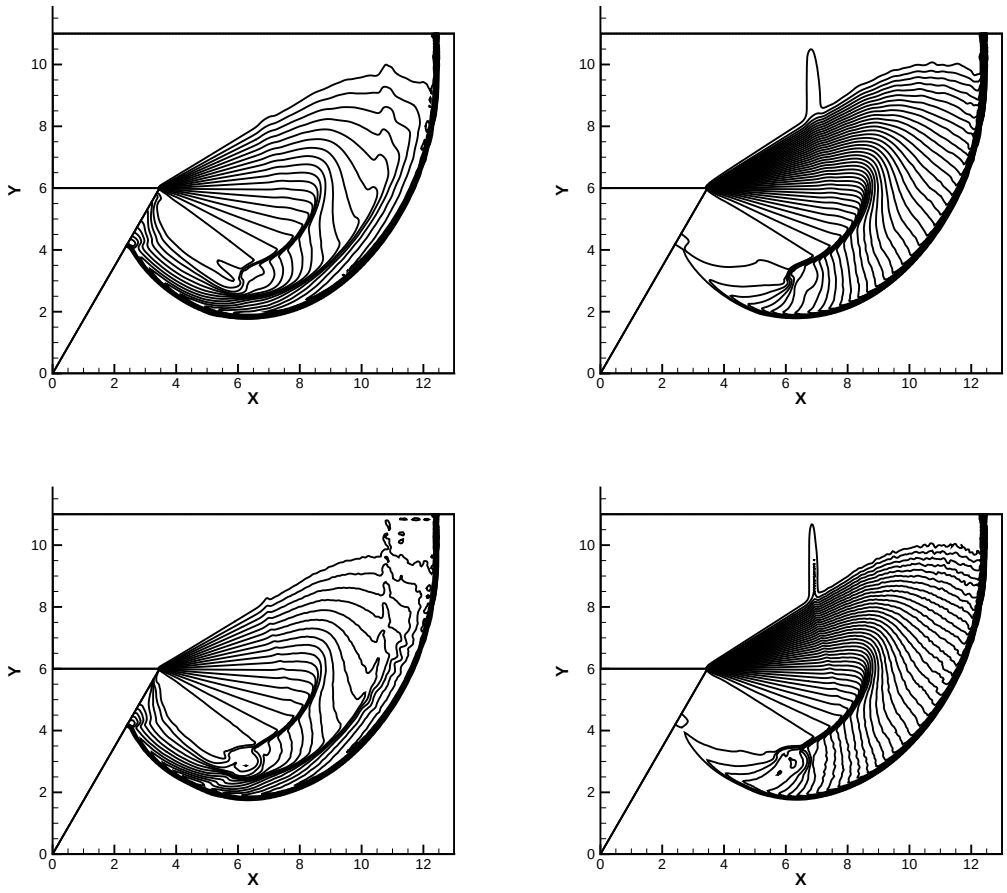


Figure 17: **Mach 10 Shock diffracting at a** 120° **corner. Top: second order** ($k = 1$)**; bottom: third order** ($k = 2$)**. Left: density, 20 equally spaced contour lines from 0.07 to 8.1; right: pressure, 40 equally spaced contour lines from 0.8 to 115. The mesh points on the boundary are uniformly distributed with cell length** $1/20$**.**

Example 8. We consider inviscid Euler transonic flow past a single NACA0012 airfoil configuration with Mach number $M = 0.8$, angle of attack $\alpha = 1.25$ and with $M = 0.85$, angle of attack $\alpha = 1$. The computational domain is $[-15, 15]\times[-15, 15]$. The mesh used in the computation is shown in Figure 18, consisting of 5780 elements with the maximum diameter of the circumcircle being 0.8857 and the minimum diameter being 0.00042 near the airfoil. The mesh uses curved cells near the airfoil. The second order and third order CPR schemes with the WENO limiter are used in the numerical experiments. Mach

number distributions are shown in Figure 19.

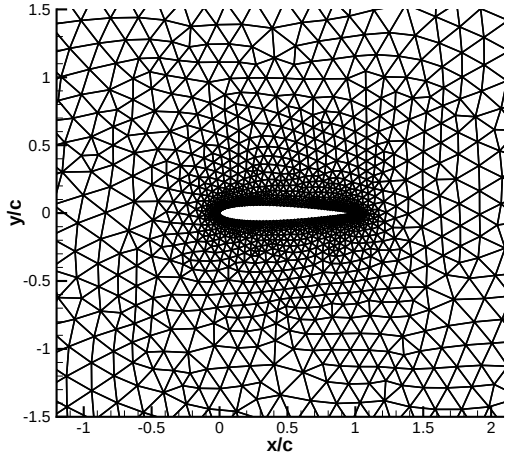


Figure 18: **NACA0012 airfoil mesh zoom in.**

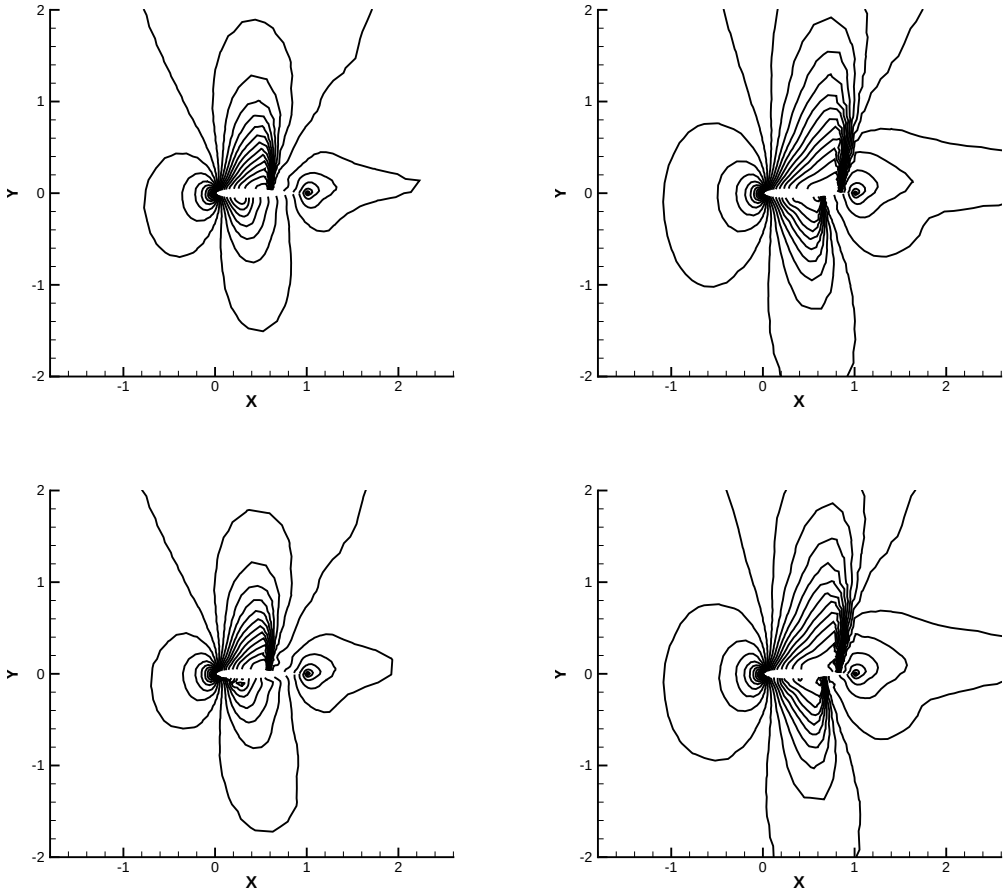


Figure 19: **NACA0012 airfoil. Mach number. Top: second order ($k = 1$); bottom: third order ($k = 2$). Left: $M_\infty = 0.8$, angle of attack $\alpha = 1.25°$, 30 equally spaced Mach number contours from 0.172 to 1.325; right: $M_\infty = 0.85$, angle of attack $\alpha = 1°$, 30 equally spaced Mach number contours from 0.158 to 1.357.**

6 Concluding remarks

In this paper, we adapt a WENO limiter [39] original designed for DG method to the CPR framework for solving hyperbolic conservation laws on two-dimensional unstructured triangular meshes with straight or curved edges to make it more robust for shocked flows and uniformly high order accurate. Also, we extend the limiters in [26, 36] to the CPR framework and get a positivity-preserving CPR framework. On each time level, we first use the WENO limiter to reconstruct the solutions on those troubled cells,

then use the positivity-preserving limiter to modify the solution polynomials in each cell if necessary. Finally, we update the numerical values at the solution points, and perform the normal CPR procedure to march to the next time level. Since the WENO limiter uses information only from immediate neighbors, it is very simple to implement and can maintain the compactness of the CPR framework. Also, we only perform this WENO limiter on the solution polynomials which can be discontinuous among adjacent cells, thus the conservativeness of the CPR framework will not be harmed. Numerical results are provided to show that this WENO limiting procedure can simultaneously maintain uniform high order accuracy of the CPR framework in smooth regions and control spurious numerical oscillations near discontinuities.

References

[1] D.S. Balsara and C.-W. Shu, *Monotonicity preserving weighted essentially non-oscillatory schemes with increasingly high order of accuracy*, Journal of Computational Physics, 160 (2000), 405–452.

[2] B. Cockburn, S. Hou and C.-W. Shu, *The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: the multidimensional case*, Mathematics of Computation, 54 (1990), 545–581.

[3] B. Cockburn and C.-W. Shu, *The Runge-Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems*, Journal of Computational Physics, 141 (1998), 199–224.

[4] J. Du, C.-W. Shu and M. Zhang, *A simple weighted essentially nonoscillatory limiter for the correction procedure via reconstruction (CPR) framework*, Applied Numerical Mathematics, to appear.

[5] H. Gao and Z.J. Wang, *A high-order lifting collocation penalty formulation for the Navier-Stokes equations on 2D mixed grids*, AIAA paper 2009–3784.

[6] H. Gao and Z.J. Wang, *A conservative correction procedure via reconstruction formulation with the Chain-Rule divergence evaluation*, Journal of Computational Physics, 232 (2013), 7–13.

[7] T. Haga, H. Gao and Z.J. Wang, *A high-order unifying discontinuous formulation for the Navier-Stokes equations on 3D mixed grids*. Mathematical Modelling of Natural Phenomena, 6 (2011), 28–56.

[8] J.S. Hesthaven and D. Gottlieb, *Stable spectral methods for conservation laws on triangles with unstructured grids*, Computer Methods in Applied Mechanics and Engineering, 175 (1999), 361–381.

[9] H.T. Huynh, *A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods*, AIAA Paper 2007–4079.

[10] H.T. Huynh, *A reconstruction approach to high-order schemes including discontinuous Galerkin for diffusion*, AIAA Paper 2009–403.

[11] C. Hu and C.-W. Shu, *Weighted essentially non-oscillatory schemes on triangular meshes*, Journal of Computational Physics, 150 (1999), 97–127.

[12] A. Jameson, P.E. Vincent and P. Castonguay, *On the non-linear stability of flux reconstruction schemes*, Journal of Scientific Computing, 50 (2012), 434–445.

[13] G. Jiang and C.-W. Shu, *Efficient implementation of weighted ENO schemes*, Journal of Computational Physics, 126 (1995), 202–228.

[14] G.E. Karniadakis and S.J. Sherwin, *Spectral/hp element methods for computational fluid dynamics*, Oxford University Press, 1999.

[15] L. Krivodonova, J. Xin, J.-F. Remacle, N. Chevaugeon and J.E. Flaherty, *Shock detection and limiting with discontinuous Galerkin methods for hyperbolic conservation laws*, Applied Numerical Mathematics, 48 (2004), 323–338.

[16] Y. Li and Z.J. Wang, *An optimized correction procedure via reconstruction formulation for broadband wave computation*, Communications in Computational Physics, 13 (2013), 1265–1291.

[17] X. Liu, S. Osher and T. Chan, *Weighted essentially non-oscillatory schemes*, Journal of Computational Physics, 115 (1994), 200–212.

[18] Y. Liu, M. Vinokur and Z.J. Wang, *Spectral difference method for unstructured grids I: Basic formulation*, Journal of Computational Physics, 216 (2006), 780–801.

[19] H. Luo, J.D. Baum and R. Lohner, *A Hermite WENO-based limiter for discontinuous Galerkin method on unstructured grids*, Journal of Computational Physics, 225 (2007), 686–713.

[20] G. May and A. Jameson, *A spectral difference method for the Euler and Navier-Stokes equations on unstructured meshes*, AIAA Paper 2006–304.

[21] J. Qiu and C.-W. Shu, *A comparison of troubled-cell indicators for Runge-Kutta discontinuous Galerkin methods using weighted essentially nonoscillatory limiters*, SIAM Journal on Scientific Computing, 27 (2005), 995–1013.

[22] W.H. Reed, T.R. Hill, *Triangular mesh methods for the neutron transport equation*, Technical Report LA-UR-73-479, 1973, Los Alamos Scientific Laboratory, Los Alamos.

[23] J. Shi, C. Hu and C.-W. Shu, *A technique of treating negative weights in WENO schemes*, Journal of Computational Physics, 175 (2002), 108–127.

[24] C.-W. Shu, *Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws*, in *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*, B. Cockburn, C. Johnson, C.-W. Shu and E. Tadmor (Editor: A. Quarteroni), Lecture Notes in Mathematics, volume 1697, Springer, Berlin, 1998, pp.325–432.

[25] C.-W. Shu and S. Osher, *Efficient implementation of essentially non-oscillatory shock-capturing schemes*, Journal of Computational Physics, 77 (1988), 439–471.

[26] C. Wang, X. Zhang, C.-W. Shu and J. Ning, *Robust high order discontinuous Galerkin schemes for two-dimensional gaseous detonations*, Journal of Computational Physics, 231 (2012), 653–665.

[27] Z.J. Wang, *Spectral (finite) volume method for conservation laws on unstructured grids: basic formulation*, Journal of Computational Physics, 178 (2002), 210–251.

[28] Z.J. Wang and H. Gao, *A unifying lifting collocation penalty formulation including the discontinuous Galerkin, spectral volume/difference methods for conservation laws on mixed grids*, Journal of Computational Physics, 228 (2009), 8161–8186.

[29] Z.J. Wang and H. Gao, *A unifying lifting collocation penalty formulation for the Euler equations on mixed grids*, AIAA Paper 2009–401.

[30] Z.J. Wang and Y. Liu, *Spectral (finite) volume method for conservation laws on unstructured grids II: extension to two-dimensional scalar equation*, Journal of Computational Physics, 179 (2002), 665–697.

[31] Z.J. Wang and Y. Liu, *Spectral (finite) volume method for conservation laws on unstructured grids III: one-dimensional systems and partition optimization*, Journal of Scientific Computing, 20 (2004), 137–157.

[32] Z.J. Wang, Y. Liu, G. May and A. Jameson, *Spectral difference method for unstructured grids II: Extension to the Euler equations*, Journal of Scientific Computing, 32 (2007), 45–71.

[33] Z.J. Wang, L. Shi, S. Fu, H. Zhang and L. Zhang, *A PNPM-CPR framework for hyperbolic conservation laws*, AIAA paper 2011–3227.

[34] Z.J. Wang, L. Zhang and Y. Liu, *Spectral (finite) volume method for conservation laws on unstructured grids IV: extension to two-dimensional Euler equations*, Journal of Computational Physics, 194 (2004), 716–741.

[35] P. Woodward and P. Colella, *The numerical simulation of two-dimensional fluid flow with strong shocks*, Journal of Computational Physics, 54 (1984), 115–173.

[36] X. Zhang, Y. Xia and C.-W. Shu, *Maximum-principle-satisfying and positivity-preserving high order discontinuous Galerkin schemes for conservation laws on triangular meshes*, Journal of Scientific Computing, 50 (2012), 29–62.

[37] X. Zhong and C.-W. Shu, *A simple weighted essentially nonoscillatory limiter for Runge-Kutta discontinuous Galerkin methods*, Journal of Computational Physics, 232 (2013), 397–415.

[38] J. Zhu, J. Qiu, C.-W. Shu and M. Dumbser, *Runge-Kutta discontinuous Galerkin method using WENO limiters II: unstructured meshes*, Journal of Computational Physics, 227 (2008), 4330–4353.

[39] J. Zhu, X. Zhong, C.-W. Shu and J.-X. Qiu, *Runge-Kutta discontinuous Galerkin method using a new type of WENO limiters on unstructured meshes*, Journal of Computational Physics, 248 (2013), 200–220.