

A Sweeping Positivity-Preserving High Order Finite Difference WENO Scheme for Euler Equations

D. Chloe Griffin¹ and Chi-Wang Shu^{1*}

¹Division of Applied Mathematics, Brown University, Providence, RI
02906, USA.

*Corresponding author(s). E-mail(s): chi-wang_shu@brown.edu;
Contributing authors: danielle_griffin@brown.edu;

Abstract

We develop a simple, high-order, conservative and robust positivity-preserving sweeping procedure for the density and the nonlinear pressure function in the compressible Euler equations. Using the scaling limiter in Zhang and Shu (2010) [33], we obtain a nontrivial extension of the scalar sweeping technique in Liu, Cheng, and Shu (2016) [16] for the positivity of pressure. The sweeping procedure developed in this paper is a post-processing technique, which can be applied to any concave functions of the conserved variables in hyperbolic conservation law systems. Thus, it has applications beyond the Euler equations. This procedure preserves positivity and conservation of physical quantities without destroying the accuracy of the underlying scheme. The algorithm works for general schemes including finite difference, finite-volume and discontinuous Galerkin methods; however, in this paper we focus on finite-difference weighted essentially non-oscillatory (WENO) methods. We provide numerical tests of the fifth order finite difference WENO scheme to demonstrate the accuracy and robustness of the technique.

Keywords: Positivity preserving, Compressible Euler equations, Hyperbolic conservation laws, Gas dynamics, High order accuracy, Finite difference scheme, Weighted essentially non-oscillatory scheme

1 Introduction

Positivity preservation for physically meaningful quantities, particularly density and pressure in Euler equations, is essential for computing numerical solutions to hyperbolic conservation laws. Failure to maintain positivity leads to non-physical distortions, ill-posedness, and numerical blow-ups. Similar concerns arise in other systems, where positivity must be preserved for the density in Vlasov–Boltzmann transport equations or water height in shallow water equations. Consequently, positivity-preserving methods are crucial in diverse applications, including computational fluid dynamics (CFD), plasma simulations, computational astrophysics, traffic flow, and population models [35]. Many numerical methods used to solve hyperbolic conservation laws do not inherently satisfy the positivity requirements of key quantities.

Finite volume (FV) and finite difference (FD) essentially non-oscillatory (ENO) schemes [8, 22] and weighted ENO (WENO) schemes [10, 15, 25], and discontinuous Galerkin (DG) methods [6, 7], are among the most widely used approaches for solving hyperbolic conservation laws and are valued for their high-order accuracy and robust shock-capturing capabilities. Finite-difference WENO schemes are appealing because they are simple to implement and computationally efficient, particularly for multidimensional problems. Their main limitation is that their application is restricted to smooth, structured, curvilinear meshes, which limits their application to more complex domains. Although finite-volume WENO and DG schemes are more costly, they provide greater flexibility because they can be applied to arbitrary triangulations and non-smooth meshes. We refer to [24] for a thorough description and comparison of each of these methods. In some cases, particularly for multidimensional systems, finite difference methods are preferred, whenever applicable, because of their low computational and memory costs when compared to other methods. In comparison with the finite-volume WENO and DG schemes, rigorously high order positivity-preserving techniques for finite-difference schemes are less mature.

One of the most successful strategies for ensuring the positivity of high-order schemes is the scaling limiter introduced in [33]. The idea in [33] is to assume that the cell averages of the high order numerical solution is positive (which has to be proved separately), then the application of the scaling limiter can bring the whole polynomial in each cell to be positive at designated quadrature points, without losing the high order accuracy. This limiter has been applied to many PDEs in both finite volume WENO [4, 32] and DG [28, 29, 34, 36, 38–40] setting. For these methods, the scaling limiter is used in the reconstruction and DG polynomials. It is extended in [37] for positivity preservation of density and pressure in the Euler equations for high order finite difference WENO schemes. Simplified implementations of the scaling limiter are provided in [35] and [27]. The scaling limiter provably maintains positivity and accuracy of the underlying scheme but requires strict theoretical CFL restrictions in order to prove the positivity of the cell averages before the limiter can be applied. This framework can be used on finite difference schemes as well [37], but it is more restrictive and less popular than for finite volume and DG schemes.

Another common approach to enforce positivity preserving for finite difference methods is via flux correction. Parameterized flux limiters use a combination of high- and low-order fluxes, similar to those in classic flux-corrected transport (FCT) methods [1–3]. Examples of these flux limiters include those developed in [14, 30] which are generalized and applied to compressible Euler equations in [5, 31]. Notably, [14] has provable accuracy for certain schemes with some restrictions on the CFL. There are other flux limiters with direct applications to Euler equations in a finite-difference setting, including [9, 12, 20]. These limiters provide positivity control for both density and pressure. Some methods, like those in [13, 18] and [16], provide simplified and inexpensive scalar modifications for the positivity preservation of the water height in shallow water equations and density in Euler equations, respectively.

Our method seeks to combine the accuracy of the scaling limiter in [33] with the simplicity of [16] as an alternative positivity-preserving procedure for density, and especially for nonlinear pressure, in the Euler systems. The sweeping procedure in [16] provides a bound-preserving technique for conserved scalar quantities that is independent of the underlying scheme. It is used as a post-processing step, with examples including one- and two-dimensional high-order finite-difference and finite-volume methods and a Fourier spectral method. The method guarantees the enforcement of bounds after a single forward and backward sweep. The procedure maintains the accuracy of the underlying schemes in numerical tests. In [11], Laiu and Hauck compare several nodal and modal limiters, including the sweeping limiter. They prove the optimal discrete L^1 convergence rate of the sweeping limiter for the spectral approximation of kinetic transport equations in certain geometries, as well as optimal numerical convergence results.

For the Euler systems, even though the sweeping procedure in [16] can be directly applied to density, which is one of the conserved variables, there is a significant challenge to extend it for positivity-preserving of pressure, or internal energy, or any nonlinear concave function of the conserved variables. In this paper, we provide a simple, conservative, positivity-preserving sweeping procedure for any concave functions of the conserved variables in hyperbolic conservation laws. The limiter is used as a post-processing step and can be applied to many functions in general schemes including finite difference, finite-volume and discontinuous Galerkin methods, however we will focus on the fifth order WENO finite difference scheme in this paper. We emphasize again that the extension of the sweeping limiter in [16] to the nonlinear pressure function in the Euler equations (or to any nonlinear concave function of the conserved variables) is nontrivial. The scaling limiter in [33] provides a robust adjustment mechanism for negative points; however, it relies on the existence of nearby points with positive pressure. We provide strategies for the adjustments if the local points have

negative pressure during the sweeping. Unlike for the scalar case, we can no longer prove that the sweeping procedure will terminate after a single forward and backward sweep. Nevertheless, we observe numerically that, even for highly demanding test cases, only one or two forward and backward sweeps are required at any given Runge-Kutta step. We apply the pressure sweeping technique for many of the same test cases as in [37]. The procedure maintains fifth order accuracy and also maintains positivity and prevents blow-ups for robust cases, including those with high CFL numbers at the instability threshold.

The remainder of this paper is organized as follows. In Section 2, we provide the necessary description for the Euler equations, finite-difference WENO, scalar sweeping procedure (which can be directly applied to density), and scaling limiter. We also describe the intuition behind the sweeping extension. In Section 3, we outline the algorithm in detail, including relevant properties of the algorithm and multidimensional extension. In Section 4, we provide numerical results for the fifth order finite difference WENO scheme with initial conditions similar to those in [37]. In Section 5, we provide final remarks and discuss future work.

2 Preliminaries

2.1 Euler Equations

In this study, we propose a positivity-preserving sweeping technique for the Euler equations. The one-dimensional version for the perfect gas is given by

$$\mathbf{u}_t + \mathbf{f}(\mathbf{u})_x = 0, \quad t \geq 0, \quad x \in \mathbb{R}, \quad (1)$$

$$\mathbf{u} = \begin{pmatrix} \rho \\ m \\ E \end{pmatrix}, \quad \mathbf{f}(\mathbf{u}) = \begin{pmatrix} m \\ \rho v^2 + p, \\ (E + p)v \end{pmatrix}$$

and

$$m = \rho v, \quad p = (\gamma - 1)\rho e, \quad E = \frac{1}{2}\rho v^2 + \rho e,$$

where ρ is density, p is the pressure, m is the momentum, v is the velocity, E is total energy, and e is the internal energy. Here, $\gamma > 1$ is a constant, where $\gamma = 1.4$ is chosen for air. We consider the pressure function $p(\mathbf{u}) = (\gamma - 1)(E - \frac{1}{2}\frac{m^2}{\rho})$. If $\rho > 0$, $p(\mathbf{u})$ is a concave function due to its negative semi-definite Hessian matrix. Let us consider a sequence of discrete points, $\{\mathbf{u}_j\}_{j=1}^n$. We define the average conserved quantity $\bar{\mathbf{u}} = \sum_{j=1}^n \frac{1}{n} \mathbf{u}_j$. Let us consider two arbitrary points $\mathbf{u}_1$ and $\mathbf{u}_2$. From Jensen's inequality, we know that, for $0 \leq t \leq 1$,

$$p(t\mathbf{u}_1 + (1-t)\mathbf{u}_2) \geq tp(\mathbf{u}_1) + (1-t)p(\mathbf{u}_2), \quad \text{if } \rho_1, \rho_2 > 0. \quad (2)$$

The pressure sweeping algorithm is designed to ensure that all discrete point values have positive pressure. For robustness, we use an epsilon threshold for positivity, where $\epsilon \ll 1$. We use $\epsilon = 10^{-13}$ in our experiments. Following the notation in [37], we aim to ensure that all points are in the admissible set

$$\mathcal{G} = \left\{ \mathbf{u} = \begin{pmatrix} \rho \\ m \\ E \end{pmatrix} \left| \rho > \epsilon \quad \text{and} \quad p = (\gamma - 1)(E - \frac{1}{2}\frac{m^2}{\rho}) > \epsilon \right. \right\}. \quad (3)$$

We note that $\mathcal{G}$ is convex. For initial value problems, we have a natural assumption that $p(\mathbf{u}_j^0) > \epsilon$ for $j = 1, \dots, n$, where the superscript denotes the time level. From Jensen's inequality, this implies that for a conservative scheme and periodic or compact

boundary condition, we have

$$p(\bar{\mathbf{u}}^k) = p\left(\sum_{j=1}^n \frac{1}{n} \mathbf{u}_j^k\right) = p\left(\sum_{j=1}^n \frac{1}{n} \mathbf{u}_j^0\right) \geq \sum_{j=1}^n \frac{1}{n} p(\mathbf{u}_j^0) > \epsilon \quad (4)$$

at arbitrary time level k . Thus, it is a natural assumption that $p(\bar{\mathbf{u}})$ is positive for all time steps. Without positivity treatment, we cannot assume that that $\overline{p(\mathbf{u})} = \sum_{j=1}^n \frac{1}{n} p(\mathbf{u}_j) > \epsilon$ once time stepping begins.

Positivity of density and pressure is vital in maintaining the stability of numerical discretizations of the conservation law in (1). Without positivity, the linearized system becomes ill-posed, and blow-ups may occur in the computation. Positivity is particularly important in preventing the computation of the sound speed, $c = \sqrt{\gamma p / \rho}$ from producing imaginary values. In this study, we focus on the fifth order finite difference WENO as our discretization method, although our algorithm can be applied to finite-volume, finite-difference, and DG methods.

2.2 Finite difference WENO

Here, we summarize the fifth order finite difference WENO scheme for (1) used in our applications. We summarize the scheme as given in [23]. We use the same implementation as [37] with a different positivity treatment. We first define a uniform one-dimensional mesh with nodes x_i and mesh size $\Delta x = x_{i+1} - x_i$, and half-points $x_{i+\frac{1}{2}} = \frac{1}{2}(x_{i+1} + x_i)$. We write $\mathbf{u}_i(t)$ as the numerical approximation of the exact solution $\mathbf{u}(x_i, t)$. To discretize (1), we use a conservative approximation

$$\frac{d\mathbf{u}_i(t)}{dt} = L(\mathbf{u})_i = -\frac{1}{\Delta x} \left(\hat{\mathbf{f}}_{i+\frac{1}{2}} - \hat{\mathbf{f}}_{i-\frac{1}{2}} \right). \quad (5)$$

For the case of Euler equations, we use Lax-Friedrichs (Rusanov) splitting to ensure upwind bias,

$$\mathbf{f}^\pm(\mathbf{u}) = \frac{1}{2} \left(\mathbf{u} \pm \frac{\mathbf{f}(\mathbf{u})}{\alpha} \right).$$

We compute $\alpha = \max(|v| + c)$ where v is the velocity and c is the sound speed. We take the maximum globally as a constant in each Runge-Kutta (RK) stage. Alternatively, the α computation can be performed locally in the WENO reconstruction stencil. At each half-point $x_{i+\frac{1}{2}}$, we

1. Apply flux splitting:

$$\mathbf{f}_i^\pm = \frac{1}{2} \left(\mathbf{u}_i \pm \frac{\mathbf{f}(\mathbf{u}_i)}{\alpha} \right)$$

2. Compute the Roe average matrix: Let $\mathbf{A}_{i+\frac{1}{2}}$ be the Roe average matrix [19]. Compute the left and right eigenvector matrices $\mathbf{L}_{i+\frac{1}{2}}$ and $\mathbf{R}_{i+\frac{1}{2}}$ such that $\mathbf{A} = \mathbf{R}\mathbf{\Lambda}\mathbf{L}$ where $\mathbf{\Lambda}$ is the diagonal matrix such that diagonal entries are eigenvalues of $\mathbf{A}$.
3. Project into characteristic fields: For each j in a neighborhood of i , find

$$\mathbf{F}_j^\pm = \mathbf{L}_{i+\frac{1}{2}} \mathbf{f}_j^\pm.$$

4. Reconstruct with fifth order WENO: Apply fifth order WENO reconstruction to each component of $\mathbf{F}_j^\pm$ to obtain nodal values $\hat{\mathbf{F}}_{i+\frac{1}{2}}^\pm$,

$$\hat{\mathbf{F}}_{i+1/2}^\pm = \text{WENO}(\mathbf{F}_{i-r+1}^\pm, \dots, \mathbf{F}_{i+r}^\pm),$$

where $r = 3$ for the fifth-order WENO reconstruction.

5. Project back into physical space:

$$\hat{\mathbf{f}}_{i+\frac{1}{2}}^\pm = \mathbf{R}_{i+\frac{1}{2}} \hat{\mathbf{F}}_{i+\frac{1}{2}}^\pm.$$

6. Form fluxes and plug into conservative scheme:

$$\hat{\mathbf{f}}_{i+\frac{1}{2}} = \alpha(\hat{\mathbf{f}}_{i+\frac{1}{2}}^+ - \hat{\mathbf{f}}_{i+\frac{1}{2}}^-)$$

and obtain $L(\mathbf{u}_i)$ as in (5).

For the fully discrete method, we use third-order Runge-Kutta time-stepping:

$$\begin{aligned}\mathbf{u}^{(1)} &= \mathbf{u}^n + \Delta t L(\mathbf{u}^n), \\ \mathbf{u}^{(2)} &= \frac{3}{4}\mathbf{u}^n + \frac{1}{4}(\mathbf{u}^{(1)} + \Delta t L(\mathbf{u}^{(1)})), \\ \mathbf{u}^{n+1} &= \frac{1}{3}\mathbf{u}^n + \frac{2}{3}(\mathbf{u}^{(2)} + \Delta t L(\mathbf{u}^{(2)})).\end{aligned}\tag{6}$$

We apply the sweeping limiter as a post-processing step after each RK stage. The limiter is only applied whenever there is negative pressure or density.

2.3 Density sweeping procedure

We use the simple density sweeping procedure in [16] and the scaling limiter in [33] as the primary building blocks of the pressure-sweeping algorithm. In [16], a sweeping algorithm is provided for a generic, conserved, scalar quantity in either a finite difference or finite volume implementation. To modify density in the Euler equations in a finite-difference WENO setting, we consider a collection of point values $\{\rho_j\}_{j=1}^n$, with the average density $\bar{\rho} = \sum_{j=1}^n \frac{1}{n}\rho_j$. Here, we rewrite the density-sweeping algorithm for clarity.

Algorithm 1 Positivity-preserving density sweep (Algorithm 2.1 of [16])

```
 $\epsilon \leftarrow 10^{-13}$   
for  $j = 1 : n - 1$  do ▷ Forward Sweep  
  if  $\rho_j < \epsilon$  then  
     $d \leftarrow \rho_j - \epsilon$   
     $\rho_{j+1} \leftarrow \rho_{j+1} + d$   
     $\rho_j \leftarrow \epsilon$   
  end if  
end for  
for  $j = n : 2$  do ▷ Backward Sweep  
  if  $\rho_j < \epsilon$  then  
     $d \leftarrow \rho_j - \epsilon$   
     $\rho_{j-1} \leftarrow \rho_{j-1} + d$   
     $\rho_j \leftarrow \epsilon$   
  end if  
end for
```

The sweeping algorithm has several advantages. It is simple to implement and can be used as a post-processing step in many numerical methods where positivity is required. For example, the sweeping technique is used in [26] for the numerical solutions of the 3D-Vlasov-Maxwell system. As a global serial process, a disadvantage is that it cannot be parallelized. Nevertheless, only a single forward and backward sweep is required to preserve positivity. Thus, this approach is relatively inexpensive. In our application of the density sweeping procedure, and our extension for pressure, we use parallel computation for spatial discretization and only apply the serial function as a post-processing step in the Runge-Kutta time discretization. In the numerical tests provided in [16], the sweeping technique preserves the accuracy of the underlying scheme. Recent theoretical convergence results have also been obtained. In [11], Laiu

and Hauck evaluate the sweeping limiter's use in the spectral approximation of kinetic transport equations. Specifically, they apply sweeping to the values of the spectral approximation at certain quadrature points. They compare several nodal and modal limiters, finding that sweeping outperformed both modal limiters, as well as the clipping limiter in [21] in terms of accuracy. They introduce a new hierarchical version of the clipping limiter with similar accuracy. Although hierarchical clipping has the benefit of parallel implementation, it suffers from more numerical artifacts in certain test cases. Laiu and Hauck provide theoretical and numerical convergence results for the limiters. They provide optimal discrete L^1 convergence rates for the sweeping limiter on slab geometries, which are sharp in numerical applications. They also provide sub-optimal discrete L^1 consistency estimates on $\mathbb{S}^2$, although numerical results indicate that the results are not sharp.

We choose to extend the sweeping limiter owing to its accuracy and simple implementation. However, the extension to the nonlinear pressure function is not straightforward. The density sweeping algorithm relies on the fact that $\bar{\rho} > \epsilon$ is conserved. Thus, upon gathering all negative density in the forward sweep and redistributing the negative density in the backward sweep, there must be nothing left over at the end. Otherwise, this would violate $\bar{\rho} > \epsilon$. This is the non-technical description of the proof given in [16] that only one forward and backward sweep is needed. We have no guarantee that the average pressure, $\overline{p(\mathbf{u})} = \sum_{i=1}^n p(\mathbf{u}_j) > \epsilon$. Combining the results underlying assumption (4) and Jensen's inequality (2), concavity of pressure and conservation only ensures that

$$\overline{p(\mathbf{u})} \leq p(\bar{\mathbf{u}}) \text{ and } p(\bar{\mathbf{u}}) > \epsilon.$$

It is possible for $\bar{\mathbf{u}} \in \mathcal{G}$ while $\mathbf{u}_j \notin \mathcal{G}$ for $j = 1, \dots, n$. Thus, even if we can easily project any negative value $\mathbf{u}_j$ to the admissible set $\mathcal{G}$ and make conservative exchanges as

in the original sweeping procedure, we cannot guarantee that only one forward and backward sweep is needed. Furthermore, projecting onto a convex set is expensive. Thus, we provide an alternative adjustment style using the scaling limiter in [33].

2.4 Scaling limiter

Zhang and Shu introduce the scaling limiter in [33] as a bound preserving technique where reconstruction or DG polynomials are scaled linearly around the cell average (which is proved to be positive separately) while maintaining uniform high order accuracy under certain CFL conditions. This strategy is an extension of the approach in [17] and is applied in a finite difference setting for Euler equations in [37]. We consider the scaling limiter for the treatment of pressure in the Euler equations. Assuming $\rho_1, \rho_2 > \epsilon$, let us consider $\mathbf{u}_1, \mathbf{u}_2 \in C$ such that $p(\mathbf{u}_1) \leq \epsilon$ and $p(\mathbf{u}_2) > \epsilon$. Thus, $\mathbf{u}_2 \in \mathcal{G}$ and $\mathbf{u}_1 \notin \mathcal{G}$. Motivated by the sweeping limiter in [16], we make a nodal adjustment of $\mathbf{u}_1$ with its neighbor such that $\mathbf{u}_1^* \in \mathcal{G}$. Following the approach in [34], one nodal adjustment choice is to set

$$\mathbf{u}_1^* = t(\mathbf{u}_2 - \mathbf{u}_1) + \mathbf{u}_1, \quad (7)$$

where t satisfies

$$p((1-t)\mathbf{u}_1 + t\mathbf{u}_2) = \epsilon. \quad (8)$$

Thus, t provides the parameterized amount to move along the line between $\mathbf{u}_1$ and $\mathbf{u}_2$ for $\mathbf{u}_1^*$ to be in the admissible set $\mathcal{G}$, but requires a quadratic solver. A proof is also provided in [34] that verifies that the limiter does not destroy the accuracy of smooth solutions in its application to discontinuous Galerkin methods. This result extends to a simplified and more robust method to find t given in [27]. We rewrite it here in terms of our applications. Using Jensen's inequality, we solve for t which satisfies

$$\epsilon = (1-t)p(\mathbf{u}_1) + tp(\mathbf{u}_2) \leq p((1-t)\mathbf{u}_1 + t\mathbf{u}_2) = p(\mathbf{u}_1^*).$$

Thus, for $\rho_1, \rho_2 > \epsilon$ and $p(\mathbf{u}_1) \leq \epsilon$, we ensure $p(\mathbf{u}_1^*) \geq \epsilon$ by setting

$$t = \frac{p(\mathbf{u}_1) - \epsilon}{p(\mathbf{u}_1) - p(\mathbf{u}_2)}. \quad (9)$$

We note that this more robust definition is necessary for problems involving blast waves. Thus far, the sweeping limiter only provides an adjustment protocol for $p(\mathbf{u}_1) \leq \epsilon$ whenever we also have $p(\mathbf{u}_2) > \epsilon$. We restrict exchanges to neighboring nodes in hopes of improving accuracy. Furthermore, we are not guaranteed that there is another positive point to exchange with, even if we consider all points. Therefore, we must provide an alternative mechanism to ensure that all points are in $\mathcal{G}$. For this, we use the scaling limiter to obtain an estimate of the distance to the convex set $\mathcal{G}$ from a negative point. The only guaranteed point in the set is $\bar{\mathbf{u}}$. Thus, we estimate that distance by solving

$$t_1 = \frac{p(\mathbf{u}_1) - \epsilon}{p(\mathbf{u}_1) - p(\bar{\mathbf{u}})} \quad (10)$$

and scale along the line between $\mathbf{u}_1$ and $\mathbf{u}_2$. Thus, we aim to move the necessary distance to reach the set $\mathcal{G}$ if the line between $\mathbf{u}_1$ and $\mathbf{u}_2$ intersects with the set. If the line does not intersect, there is no way for this nodal adjustment to indicate $\mathbf{u}_1^* \in \mathcal{G}$. In this case, the best we can do is to ensure that the two points move closer together, thereby increasing their combined pressure. Thus, we include a minimum of $1/4$. In this way, the distance between negative points will at most be halved. If they are already close to $\mathcal{G}$, a smaller distance may be provided by the scaling limiter estimate. The full definition for t in this case is

$$t = \min \left(t_1 \left(\frac{\|\mathbf{u}_1 - \bar{\mathbf{u}}\|}{\|\mathbf{u}_1 - \mathbf{u}_2\|} \right), \frac{1}{4} \right).$$

We provide a full description of the algorithm in the next section.

3 Conservative positivity-preserving sweeping procedure for pressure

Here, we describe the pressure sweeping algorithm with positivity-preserving properties. Given $\{\mathbf{u}_j\}_{j=1}^n$ with $p(\bar{\mathbf{u}}) = p(\sum_{j=1}^n \frac{1}{n} \mathbf{u}_j) > \epsilon$, the pressure sweeping procedure continues until positivity of all $\rho_j \in \{\rho_j\}_{j=1}^n$ and $p(\mathbf{u}_j) \in \{p(\mathbf{u}_j)\}_{j=1}^n$ is achieved. In other words, it provides a method to adjust all points towards admissible set $\mathcal{G}$ given in (3) under the natural assumption in (4). Furthermore, it maintains the conservation of $\mathbf{u}$ such that $\bar{\mathbf{u}} = \sum_{j=1}^n \frac{1}{n} \mathbf{u}_j$ is unchanged. To correct a “negative point” $\mathbf{u}_j$, or a point where $p(\mathbf{u}_j) < \epsilon$, we find an adjusted point $\mathbf{u}_j^*$ using a convex combination of $\mathbf{u}_j$ with its neighbor. As in the original sweeping algorithm, we make a nodal adjustment with the next point in the direction of sweep. We only adjust the negative points, and the nodal adjustment differs if the neighboring point is positive or negative.

If the neighboring point is positive, $\mathbf{u}_j^*$ is adjusted using the scaling limiter as in (7) with t from (9). If the neighboring point is negative, we use a similar estimate to find the distance to the convex set $\mathcal{G}$ using $\bar{\mathbf{u}}$. We use this distance as an estimate of how much to move along a line between the negative point and its neighbor. We take a minimum of this scaled distance with $1/4$. In this way, the distance between negative points is at most halved. When points are close to the admissible set, the estimate by the scaling limiter may provide a better estimate of the distance needed for $\mathbf{u}_j^* \in \mathcal{G}$.

In practice, $\{\mathbf{u}_j\}$ may be a sequence of vectors with point values or cell averages, but we provide the algorithm for point values in a finite difference scheme. Throughout this work, we denote by $\|\cdot\|$ the discrete L^2 norm in $\mathbb{R}^3$, defined for any vector $\mathbf{x} = (x_1, x_2, x_3) \in \mathbb{R}^3$ as

$$\|\mathbf{x}\| = \sqrt{x_1^2 + x_2^2 + x_3^2}.$$

The algorithm is as follows:

Procedure 1 (*A conservative positivity-preserving sweeping procedure*)

1. Apply Algorithm 2.3 to $\{\rho_j\}_{j=1}^n$,
2. Apply the positivity preserving sweep in Algorithm 2 using the nodal adjustment in Algorithm 3.

Algorithm 2 Positivity Preserving Sweep

```

 $\epsilon \leftarrow 10^{-13}$ 

while  $\min(p(\mathbf{u})) < \epsilon$  do
  for  $j = 1 : n - 1$  do                                     ▷ Forward Sweep
    if  $p(\mathbf{u}_j) < \epsilon$  then
       $\mathbf{u}_j^* \leftarrow \text{NodeAdjust}(\mathbf{u}_j, \mathbf{u}_{j+1}, \bar{\mathbf{u}}, \epsilon)$ 
       $\mathbf{d} \leftarrow \mathbf{u}_j - \mathbf{u}_j^*$ 
       $\mathbf{u}_{j+1}^* \leftarrow \mathbf{u}_{j+1} + \mathbf{d}$ 
       $\mathbf{u}_j \leftarrow \mathbf{u}_j^*$ 
       $\mathbf{u}_{j+1} \leftarrow \mathbf{u}_{j+1}^*$ 
    end if
  end for
  for  $j = n : 2$  do                                         ▷ Backward Sweep
    if  $p(\mathbf{u}_j) < \epsilon$  then
       $\mathbf{u}_j^* \leftarrow \text{NodeAdjust}(\mathbf{u}_j, \mathbf{u}_{j-1}, \bar{\mathbf{u}}, \epsilon)$ 
       $\mathbf{d} \leftarrow \mathbf{u}_j - \mathbf{u}_j^*$ 
       $\mathbf{u}_{j-1}^* \leftarrow \mathbf{u}_{j-1} + \mathbf{d}$ 
       $\mathbf{u}_j \leftarrow \mathbf{u}_j^*$ 
       $\mathbf{u}_{j-1} \leftarrow \mathbf{u}_{j-1}^*$ 
    end if
  end for
end while

```

The following function provides a node adjustment protocol.

Algorithm 3 $\mathbf{u}_1^* = \text{NodeAdjust}(\mathbf{u}_1, \mathbf{u}_2, \bar{\mathbf{u}}, \epsilon)$

if $p(\mathbf{u}_2) > \epsilon$ **then**

$$t \leftarrow \left(\frac{p(\mathbf{u}_1) - \epsilon}{p(\mathbf{u}_1) - p(\mathbf{u}_2)} \right)$$

else

$$t_1 \leftarrow \frac{p(\mathbf{u}_1) - \epsilon}{(p(\mathbf{u}_1) - p(\bar{\mathbf{u}}))}$$

$$t \leftarrow \min \left(t_1 \left(\frac{\|\mathbf{u}_1 - \bar{\mathbf{u}}\|}{\|\mathbf{u}_1 - \mathbf{u}_2\|} \right), 1/4 \right)$$

end if

$$\mathbf{u}_1^* \leftarrow (1 - t)\mathbf{u}_1 + t\mathbf{u}_2$$

We describe several useful properties of the algorithm.

Property 1 (Conservation) *First, we note that none of the operations in Algorithm 1 change $\bar{\mathbf{u}}$. Denoting $\mathbf{u}_n$ as a node where $p(\mathbf{u}_n) < \epsilon$ and $\mathbf{u}_p$ as the neighboring node, we have*

$$\mathbf{u}_n^* + \mathbf{u}_p^* = \mathbf{u}_n^* + (\mathbf{u}_p + (\mathbf{u}_n - \mathbf{u}_n^*)) = \mathbf{u}_n + \mathbf{u}_p.$$

This implies that $\bar{\mathbf{u}}$ is not changed and hence we still have $p(\bar{\mathbf{u}}) > \epsilon$.

Property 2 (Monotonic Improvement) *Any operation will only decrease the distance between points and increase the average of pressure, $\overline{p(\mathbf{u})} = \sum_{i=1}^n \frac{1}{n} p(\mathbf{u}_i)$. We first rewrite*

$$\begin{aligned} \mathbf{u}_p^* &= \mathbf{u}_p + (\mathbf{u}_n - \mathbf{u}_n^*) \\ &= \mathbf{u}_p + \mathbf{u}_n - ((1 - t)\mathbf{u}_n + t\mathbf{u}_p) \\ &= t\mathbf{u}_n + (1 - t)\mathbf{u}_p. \end{aligned}$$

Using the definition on $\mathbf{u}_p^$ and Cauchy-Schwarz, we find a bound on the adjusted distance between points as*

$$\begin{aligned} \|\mathbf{u}_p^* - \mathbf{u}_n^*\| &= \|\mathbf{u}_p + (\mathbf{u}_n - \mathbf{u}_n^*) - \mathbf{u}_n^*\| \\ &= \|\mathbf{u}_p + (\mathbf{u}_n - ((1 - t)\mathbf{u}_n + t\mathbf{u}_p)) - ((1 - t)\mathbf{u}_n + t\mathbf{u}_p)\| \\ &= \|(1 - 2t)\mathbf{u}_p - (1 - 2t)\mathbf{u}_n\| \end{aligned}$$

$$= \|(1-2t)(\mathbf{u}_p - \mathbf{u}_n)\|$$

$$\leq |1-2t| \|\mathbf{u}_p - \mathbf{u}_n\|.$$

Letting $a = p(\mathbf{u}_1) - \epsilon$ and $b = p(\mathbf{u}_2) + \epsilon$ or $b = p(\bar{\mathbf{u}}) + \epsilon$, it is easy to see that $t \in (0, 1)$ from the simple fact that

$$a < 0 < b \implies 0 < \frac{a}{a-b} < 1.$$

Thus,

$$|1-2t| \leq 1 \implies \|\mathbf{u}_p^* - \mathbf{u}_n^*\| \leq \|\mathbf{u}_p - \mathbf{u}_n\|.$$

Furthermore, the concavity of the pressure implies that as points move closer together, their combined pressure increases. Using Jensen's inequality, we find that

$$\begin{aligned} p(\mathbf{u}_p^*) + p(\mathbf{u}_n^*) &= p(t\mathbf{u}_n + (1-t)\mathbf{u}_p) + p((1-t)\mathbf{u}_n + t\mathbf{u}_p) \\ &\geq tp(\mathbf{u}_n) + p(\mathbf{u}_p) - tp(\mathbf{u}_p) + p(\mathbf{u}_n) - tp(\mathbf{u}_n) + tp(\mathbf{u}_p) \\ &= p(\mathbf{u}_p) + p(\mathbf{u}_n). \end{aligned}$$

It is clear that with successive application of the sweeping limiter, the average value of pressure, that is, $\frac{1}{n} \sum_{i=1}^n p(\mathbf{u}_j)$ is monotonically increasing and bounded above by $p(\bar{\mathbf{u}})$. Thus, sweeping may only improve the conditions for pressure positivity.

Property 3 (Variance Reduction) *As the points get closer together, the total variance decreases. As the sweeping procedure is continuously applied, all elements monotonically move toward the set $\mathcal{G}$ and expected value, $\bar{\mathbf{u}}$. To show this, we consider how the total variance $\mathbb{V} = \sum_{j=1}^n \|\mathbf{u}_j - \bar{\mathbf{u}}\|^2$ changes upon each combined nodal adjustment. We consider the local variance for $\mathbf{u}_p$ and $\mathbf{u}_n$ defined as*

$$\mathbb{V}_{n,p} := \|\mathbb{V}_n\|^2 + \|\mathbb{V}_p\|^2 := \|\mathbf{u}_n - \bar{\mathbf{u}}\|^2 + \|\mathbf{u}_p - \bar{\mathbf{u}}\|^2,$$

and how it changes in relation to

$$\mathbb{V}_{n,p}^* := \|\mathbb{V}_n^*\|^2 + \|\mathbb{V}_p^*\|^2 := \|\mathbf{u}_n^* - \bar{\mathbf{u}}\|^2 + \|\mathbf{u}_p^* - \bar{\mathbf{u}}\|^2.$$

Through a simple calculation, we see that

$$\mathbb{V}_n^* = \mathbf{u}_n^* - \bar{\mathbf{u}} = (1-t)\mathbf{u}_n + t\mathbf{u}_p - \bar{\mathbf{u}} = (1-t)(\mathbf{u}_n - \bar{\mathbf{u}}) + t(\mathbf{u}_p - \bar{\mathbf{u}}) = (1-t)\mathbb{V}_n + t\mathbb{V}_p.$$

Through the same calculation, we find $\mathbb{V}_p^* = \mathbf{u}_p^* - \bar{\mathbf{u}} = t\mathbb{V}_n + (1-t)\mathbb{V}_p$. We expand

$$\begin{aligned} \mathbb{V}_{n,p}^* &= \|\mathbb{V}_n^*\|^2 + \|\mathbb{V}_p^*\|^2 \\ &= \|(1-t)\mathbb{V}_n + t\mathbb{V}_p\|^2 + \|t\mathbb{V}_n + (1-t)\mathbb{V}_p\|^2 \\ &= (1-t)\|\mathbb{V}_n\|^2 + t^2\|\mathbb{V}_p\|^2 + 2t(1-t)\mathbb{V}_n\mathbb{V}_p + \\ &\quad + t^2\|\mathbb{V}_n\|^2 + (1-t)^2\|\mathbb{V}_p\|^2 + 2t(1-t)\mathbb{V}_n\mathbb{V}_p \\ &= [(1-t)^2 + t^2]\|\mathbb{V}_n\|^2 + [t^2 + (1-t)^2]\|\mathbb{V}_p\|^2 + 4t(1-t)\mathbb{V}_n\mathbb{V}_p \\ &= (2t(t-1) + 1)(\|\mathbb{V}_n\|^2 + \|\mathbb{V}_p\|^2) + 4t(1-t)\mathbb{V}_n\mathbb{V}_p. \end{aligned}$$

We use the fact that

$$\|\mathbb{V}_p - \mathbb{V}_n\|^2 = \|\mathbb{V}_p\|^2 + \|\mathbb{V}_n\|^2 - 2\mathbb{V}_n\mathbb{V}_p \implies 2\mathbb{V}_n\mathbb{V}_p = \|\mathbb{V}_p\|^2 + \|\mathbb{V}_n\|^2 - \|\mathbb{V}_p - \mathbb{V}_n\|^2.$$

Thus,

$$4t(1-t)\mathbb{V}_n\mathbb{V}_p = 2t(1-t) \left[\|\mathbb{V}_p\|^2 + \|\mathbb{V}_n\|^2 - \|\mathbb{V}_p - \mathbb{V}_n\|^2 \right].$$

Combining these results, we obtain the estimate

$$\begin{aligned} \mathbb{V}_{n,p}^* &= \|\mathbb{V}_p\|^2 + \|\mathbb{V}_n\|^2 - 2t(1-t)\|\mathbb{V}_p - \mathbb{V}_n\| \\ &= \|\mathbb{V}_p\|^2 + \|\mathbb{V}_n\|^2 - 2t(1-t)\|\mathbf{u}_p - \mathbf{u}_n\|. \end{aligned}$$

For $t \in (0, 1)$, $0 < 2t(1-t) < 1$. Thus, the local variance $\mathbb{V}_{p,n}^* < \mathbb{V}_{p,n}$ as long as $\mathbf{u}_p \neq \mathbf{u}_n$. Other point values are not affected. This implies that the total variance $\mathbb{V}$ is monotonically decreasing.

Remark 1 While we do not provide rigorous convergence results to the set $\mathcal{G}$, we justify that pressure is increasing and variance decreases while conservation is maintained. Thus, points can only move closer to the admissible set, and negative pressure values are improved. In practice, only a few points have negative pressure, and they are likely to have positive neighbors nearby. Thus, we conjecture that the combination of the robust scaling limiter for

pressure correction and the mass-redistribution technique of the sweeping limiter explains the need for only one or two forward and backward sweeps to achieve positivity in our numerical results in Section 4.

3.1 Extension to multiple dimensions

In two spatial dimensions, the Euler equations can be written in conservative form as

$$\frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{f}(\mathbf{w})}{\partial x} + \frac{\partial \mathbf{g}(\mathbf{w})}{\partial y} = 0, (x, y) \in \mathbb{R}^2, t \geq 0, \quad (11)$$

where the vector of conserved variables is $\mathbf{w} = (\rho, m, n, E)^T$. Here, ρ is the density, $m = \rho u$ and $n = \rho v$ are the momentum components in the x - and y -directions, respectively, and E is the total energy. The flux vectors are

$$\mathbf{f}(\mathbf{w}) = \begin{pmatrix} m \\ \frac{m^2}{\rho} + p \\ \frac{mn}{\rho} \\ \frac{m}{\rho}(E + p) \end{pmatrix}, \quad \mathbf{g}(\mathbf{w}) = \begin{pmatrix} n \\ \frac{mn}{\rho} \\ \frac{n^2}{\rho} + p \\ \frac{n}{\rho}(E + p) \end{pmatrix}. \quad (12)$$

The pressure is determined from the ideal-gas law

$$p = (\gamma - 1) \left(E - \frac{1}{2} \frac{m^2 + n^2}{\rho} \right),$$

with $\gamma > 1$ describing the ratio of the specific heats. As before, one can easily verify that p is a concave function for $\rho > 0$. The admissible set now becomes

$$\mathcal{G} = \left\{ \mathbf{w} = \begin{pmatrix} \rho \\ m \\ n \\ E \end{pmatrix} \left| \rho > \epsilon \quad \text{and} \quad p = (\gamma - 1) \left(E - \frac{1}{2} \frac{m^2 + n^2}{\rho} \right) > \epsilon \right. \right\}. \quad (13)$$

which is still convex. For a two-dimensional uniform mesh with nodes (x_i, y_j) , we use a conservative approximation

$$\frac{d\mathbf{w}_{i,j}(t)}{dt} = L(\mathbf{w})_{i,j} = -\frac{1}{\Delta x} \left(\hat{\mathbf{f}}_{i+\frac{1}{2},j} - \hat{\mathbf{f}}_{i-\frac{1}{2},j} \right) - \frac{1}{\Delta y} \left(\hat{\mathbf{g}}_{i,j+\frac{1}{2}} - \hat{\mathbf{g}}_{i,j-\frac{1}{2}} \right).$$

As in the one-dimensional case, we use Lax-Friedrichs (Rusanov) splitting to ensure upwind bias,

$$\mathbf{f}^\pm(\mathbf{w}) = \frac{1}{2} \left(\mathbf{w} \pm \frac{\mathbf{f}(\mathbf{w})}{\alpha_1} \right) \quad \text{and} \quad \mathbf{g}^\pm(\mathbf{w}) = \frac{1}{2} \left(\mathbf{w} \pm \frac{\mathbf{g}(\mathbf{w})}{\alpha_2} \right), \quad (14)$$

where $\alpha_1 = \max(|u| + c)$ and $\alpha_2 = \max(|v| + c)$. We take the maximum globally as a constant in each RK stage. Alternatively, the computation can be performed locally in the WENO reconstruction stencil.

The extension of the pressure sweeping algorithm to two-dimensions is simple. We generate indexing arrays much like in [16]. We maintain conservation over the entire computation domain. In other words, we maintain $\bar{\mathbf{w}} = \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \frac{1}{n_x n_y} \mathbf{w}_{i,j}$. We treat $\{\mathbf{w}_k\}_{k=1}^{n_x n_y}$ as a one-dimensional matrix. There are many ways to create this matrix while maintaining neighboring relations of the 2D array. We alternate between the two orderings given in [16]. The orderings are written as:

- Sweep I: Connect mesh cells column by column: For each column, $j = 1, \dots, n_y$, use order $i = 1, 2, \dots, n_x$ for odd j , and order $i = n_x, n_x - 1, \dots, 1$ for even j .
- Sweep II: Connect mesh cells row by row: For each row, $i = 1, 2, \dots, n_x$, use order $j = 1, \dots, n_y$ for odd i , and order $j = n_y, n_y - 1, \dots, 1$ for even i .

We alternate between Sweep I and Sweep II in our numerical tests. We count a forward and backward sweep as one full pressure sweep. Thus, at a given RK step, Sweep I is used for odd sweeps and Sweep II is used for even sweeps. This choice ensures access to neighbors in both the x and y directions. Nevertheless, most of the test cases in Section 4 require a single sweep at any Runge–Kutta step. Most of our test cases need only one sweep. Thus, we expect that using only Sweep I or Sweep II would be sufficient. There is also a possibility of parallel implementation in the two-dimensional case because we allow more than a single forward and backward sweep.

4 Numerical results

We apply the sweeping procedure to maintain the positivity of density and pressure for both the one- and two-dimensional cases. The numerical experiments in this study are designed to mirror those in [37], demonstrating the accuracy and positivity-preserving properties of the sweeping method. We present seven numerical examples in which negative pressure occurs. In most of these cases, negative pressure causes the WENO algorithm to blow up without positivity treatment. In cases where negative values did not occur in a stable CFL range, we increased the CFL beyond the standard limits. Applying sweeping at the instability threshold, we still see key features of the numerical solution which demonstrates the robustness of the algorithm. For detailed accounts of the number of sweeps used (both in total and average/max quantities for each RK stage), see Table 2 and Table 3 for pressure and density respectively.

Example 1 (1D double rarefaction) We first present a double rarefaction problem in one dimension. The exact solution contains a vacuum; thus, negative values may appear. We solve the 1D Euler equations given by (1) with the Riemann initial condition

$$\begin{cases} (\rho, v, p) = (7, -1, 0.2) & \text{if } x \leq 0, \\ (\rho, v, p) = (7, 1, 0.2) & \text{if } x > 0, \end{cases}$$

and $\gamma = 1.4$. The boundaries are outflow. We run our numerical test on $x = [-0.5, 0.5]$ with 200 points until the final time $T = 0.3$. We use $\text{CFL} = 0.9$ to produce negative pressure values. Even with a larger CFL, our results in Figure 1 are comparable to those in [37].

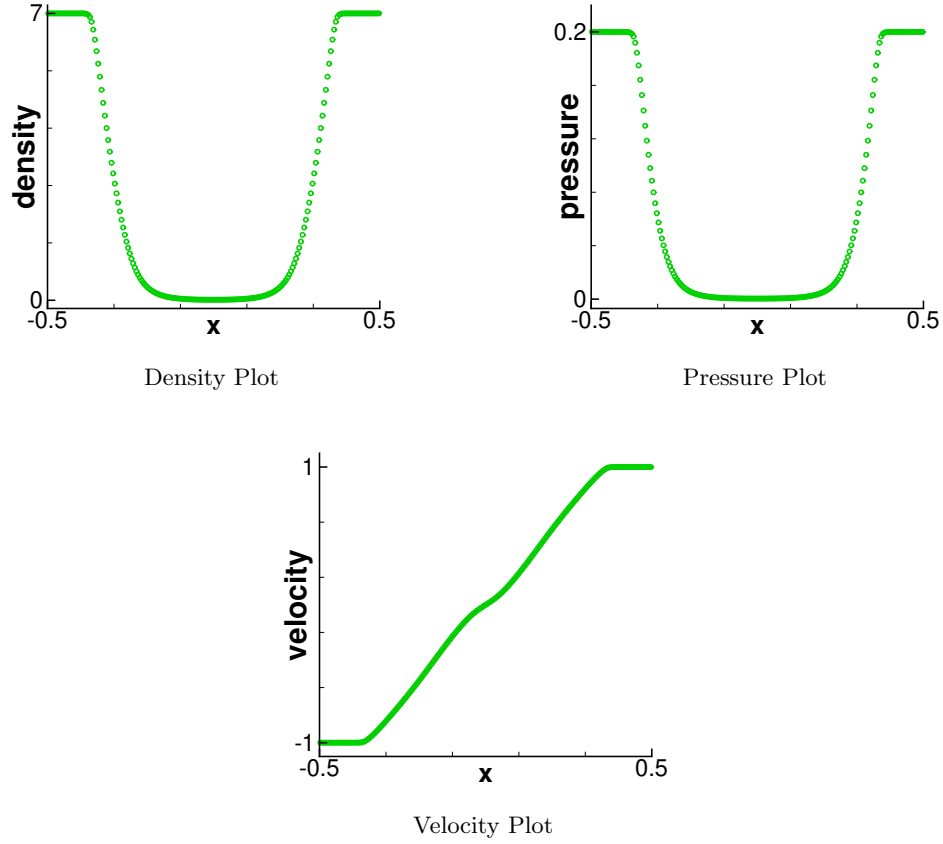


Fig. 1 Example 1: Approximate solution of the 1D double rarefaction problem using fifth order WENO with the sweeping limiter. Green symbols are approximate results.

Example 2 (1D Sedov blast) We consider the one-dimensional Sedov blast wave of (1), which results in low density and pressure. The initial condition resembles a delta function, where density is 1, velocity is 0, and total energy is 10^{-12} everywhere except the center cell containing energy constant $3,200,000/\Delta x$. The boundary conditions are outflow. We run our numerical test on $x = [-2, 2]$ with 800 points until final time $T = 0.001$. We use $\text{CFL} = 1.2$ to produce negative pressure values. Even with a larger CFL, our results in Figure 2 closely match the exact solution.

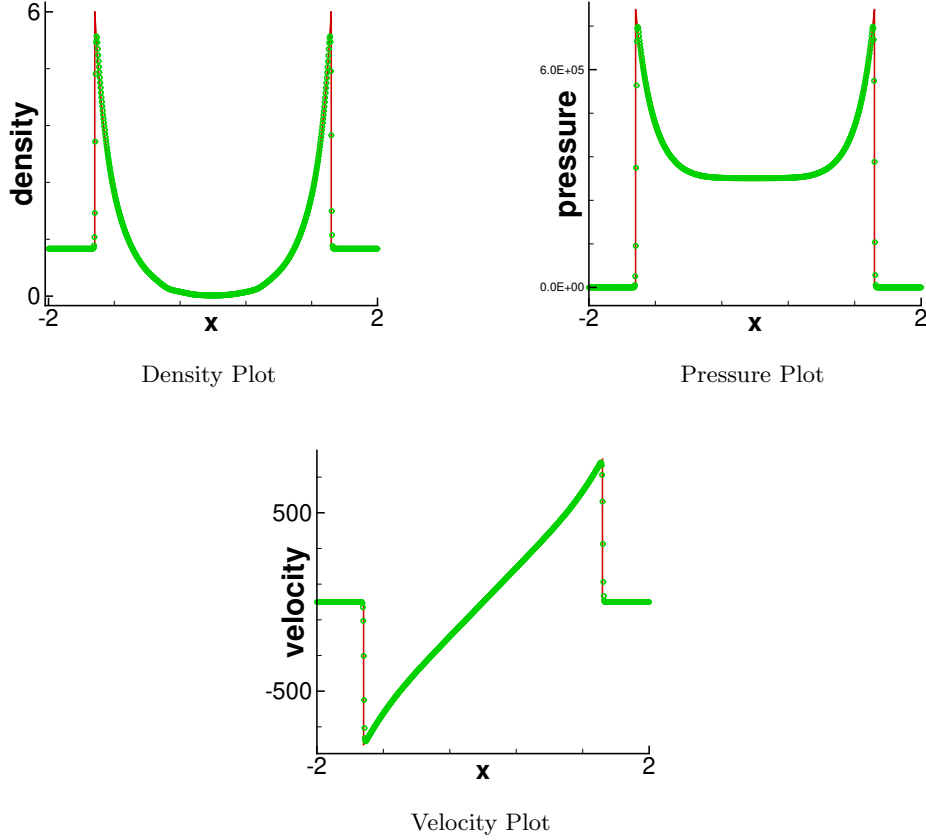


Fig. 2 Example 2: Numerical results 1D Sedov blast test case using fifth order WENO with sweeping limiter. Green symbols are approximate results. Red line is exact solution.

Example 3 (2D vortex evolution accuracy) For an accuracy test, we solve the 2D Euler equations (11) while considering an isentropic vortex perturbation in (u, v) centered at $(x_0, y_0) = (5, 5)$ on $\Omega = [0, 10] \times [0, 10]$. The mean flow $\rho = u = v = p = 1$ is summed with a vortex perturbation given by

$$(\delta u, \delta v) = -\frac{\Gamma}{2\pi} \exp\left(\frac{1-r^2}{2}\right) (-\bar{y}, \bar{x}),$$

$$\delta T = \frac{(\gamma-1)\Gamma^2}{8\pi^2\gamma} \exp(1-r^2),$$

and

$$r^2 = \bar{x}^2 + \bar{y}^2$$

where Γ is the vortex strength and $(\bar{x}, \bar{y}) = (x - x_0, y - y_0)$. $S = p/\rho^\gamma$ remains unperturbed. This is the same test as [37], where $\Gamma = 10.0828$. The exact solution is the advection of the vortex by the mean flow. Using periodic boundaries, we run our numerical test with increasing points, as shown in Table 1 until the final time $T = 0.01$ with $\text{CFL} = 0.5$. We can clearly see the fifth-order accuracy. For a visual representation of the results, we plot a slice through $x = 5$ for the pressure and density with 180×180 points in space, as shown in Figure 3. We also include the sweeping data for this case in Table 2.

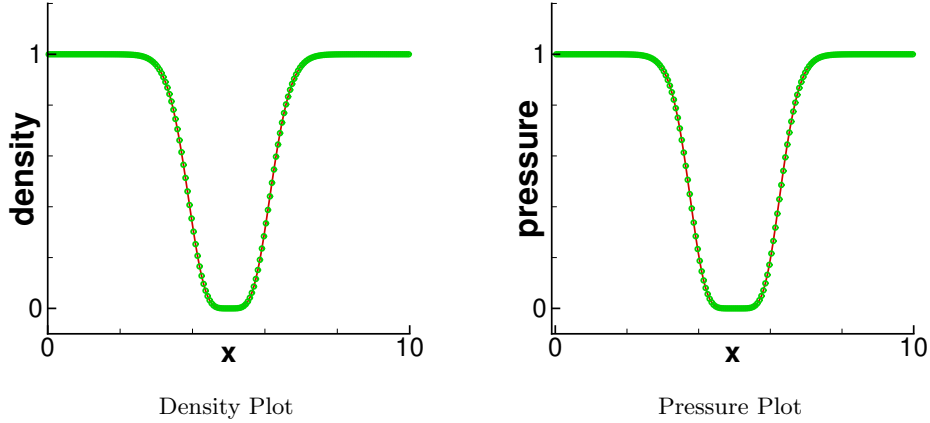


Fig. 3 Example 3: Numerical results for 2D vortex evolution test case using fifth order WENO with sweeping limiter. Green symbols are approximate results. Red line is exact solution.

Table 1 Convergence Data

Number of points	L_1 Error	L_1 Order	L_∞ Error	L_∞ Order
45×45	4.0672×10^{-5}	0.0000	2.4255×10^{-3}	0.0000
90×90	4.6604×10^{-6}	3.1255	4.2309×10^{-4}	2.5192
180×180	1.7786×10^{-7}	4.7117	2.2275×10^{-5}	4.2475
360×360	4.1597×10^{-9}	5.4181	4.9210×10^{-7}	5.5003

Example 4 (2D Sedov blast) We use the upper-right-quarter computational domain $[0, 1.3] \times [0, 1.3]$ to solve equations (11). The initial condition resembles a delta function, where density is 1, velocity is 0, total energy is 10^{-12} , and the lower left corner cell contains an energy constant $0.244816/(\Delta x \Delta y)$. The left and bottom boundaries are reflective, and the top and right boundaries are outflow. Using sweeping, we obtain results that are comparable to the exact solution and positivity-preserving method of [37]. With 640×640 points in space and CFL = 0.5, we compute the solution at the final time $T = 1$. The results are shown in Figure 4.

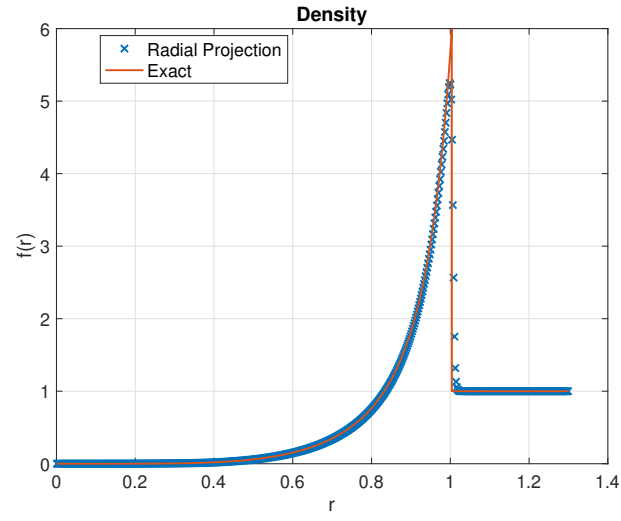
Example 5 (Mach 2000 robustness test) To demonstrate the robustness of the sweeping algorithm, we run the same high Mach problem as [34] and [37]. We solve (11) with $\gamma = 5/3$. The initial condition is

$$(\rho, u, v, p) = (0.5, 0, 0, 0.4127).$$

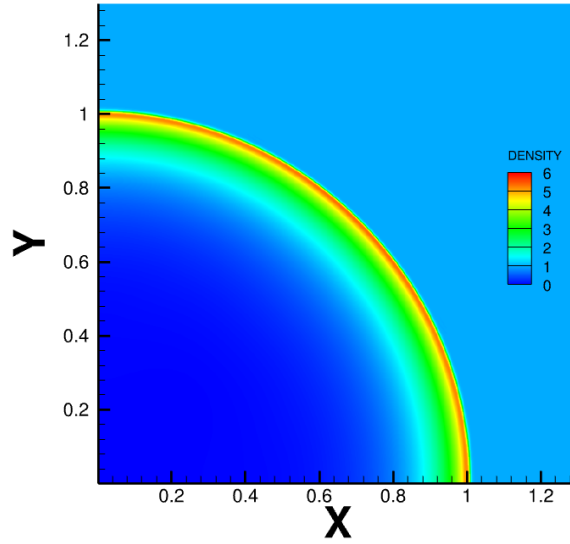
For the left boundary condition along $x = 0$, we have

$$\begin{cases} (\rho, u, v, p) = (5, 800, 0, 0.4127) & \text{if } y \in [-0.05, 0.05], \\ (\rho, u, v, p) = (5, 0, 0, 0.4127) & \text{if } y \notin [-0.05, 0.05]. \end{cases}$$

All other boundaries are outflow. We run our numerical test on $\Omega = [0, 1] \times [0, 0.25]$ with 800×400 points in space until the final time $T = 0.001$ with CFL = 0.25. Using a jet speed of 800, the high Mach number results in many negative pressure and density values that are corrected by sweeping. The high number of total sweeps is presented in Table 2. However, only a maximum of two sweeps were required at any given RK step. The results for density and pressure are shown in Figure 5.



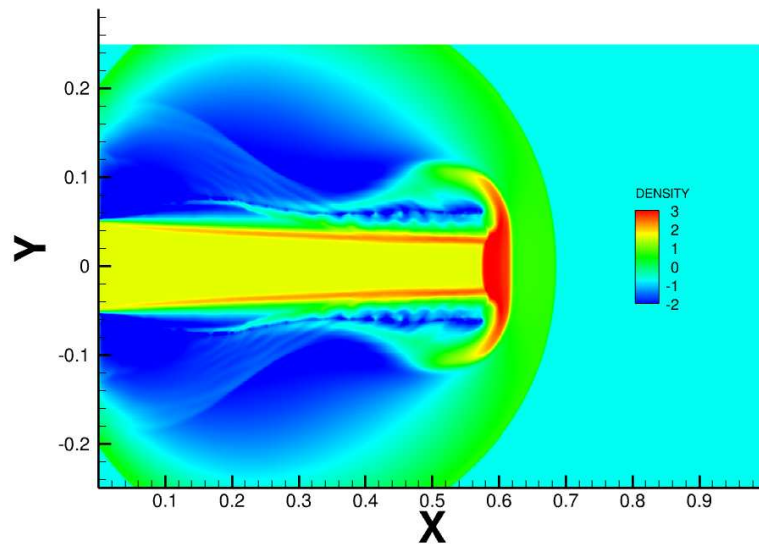
Density Radial Projection



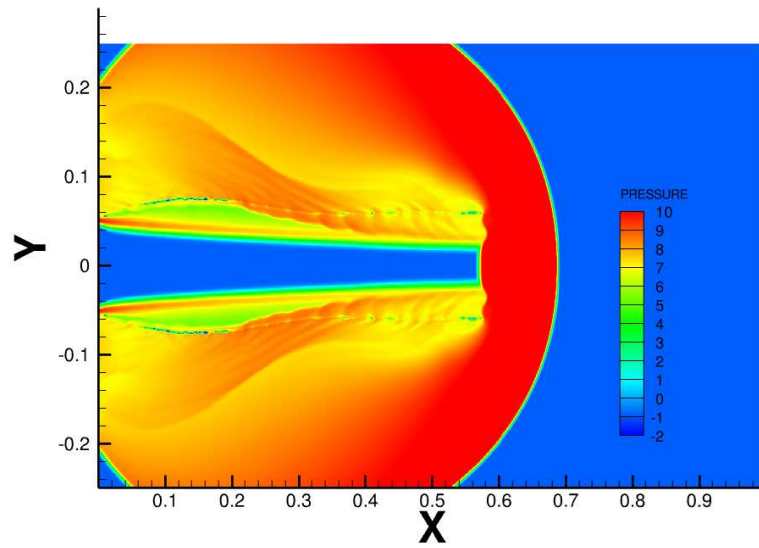
Density Contour

Fig. 4 Example 4: Numerical results for the 2D Sedov blast wave using fifth order WENO with sweeping limiter.

Example 6 (Shock diffraction: High CFL) We consider the low-density and low-pressure shock diffraction problem of (11). The domain is the union of $[0, 1] \times [6, 11]$ which contains the



Density Plot



Pressure Plot

Fig. 5 Example 5: Color contour of results on a logarithmic scale for Mach 2000 test case using fifth order WENO with sweeping limiter.

pure, right-moving Mach 5.09 shock along $x = 0.5$ and $[1, 13] \times [0, 11]$ which contains ambient gas $(\rho, u, v, p) = (1.4, 0, 0, 1)$. The boundary condition is outflow along the top, right, and bottom of $[1, 13] \times [0, 11]$. The boundary condition is inflow on the left and reflective on the bottom of $[0, 1] \times [6, 11]$. Along the wall $\{0\} \times [0, 6]$, reflective boundaries are used. We run our numerical test on $\Omega = [0, 13] \times [0, 11]$ with 1040×880 points until the final time $T = 2.3$. We use a high CFL = 0.9 to produce negative pressure values. Due to the high CFL, we see small fluctuations in the solution. Nevertheless, the plot shows key features, even at the instability threshold. The results are shown in Figure 6.

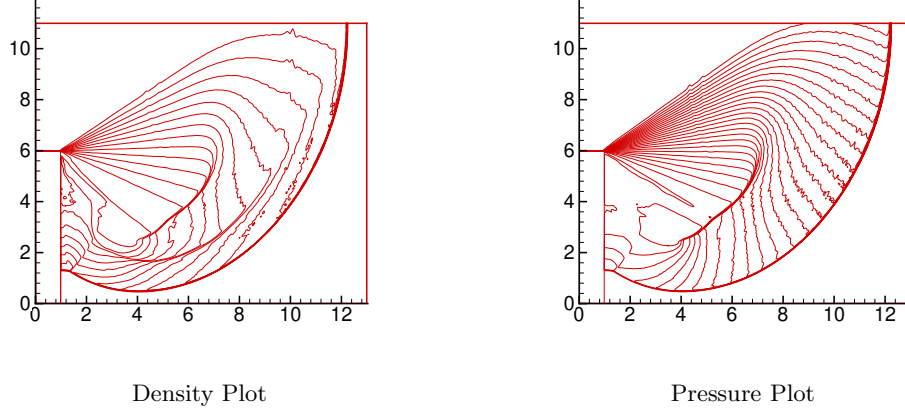


Fig. 6 Example 6: Contour plot of results for shock diffraction test case using fifth order WENO with sweeping limiter. Left plot: 20 equally spaced contour lines between 0.066227 and 7.0668. Right plot: 40 contour lines between 0.091 and 42.

Example 7 (Reactive Euler detonation: High CFL) The reactive Euler equations are an extension of the standard Euler equations for modeling chemical reactions or detonations. The governing equations are written as

$$\mathbf{w}_t + \mathbf{f}(\mathbf{w})_x + \mathbf{g}(\mathbf{w})_y = \mathbf{s}(\mathbf{w}), \quad t \geq 0, \quad (x, y) \in \mathbb{R}^2, \quad (15)$$

$$\mathbf{w} = \begin{pmatrix} \rho \\ m \\ n \\ E \\ \rho Y \end{pmatrix}, \quad \mathbf{f}(\mathbf{w}) = \begin{pmatrix} m \\ \rho u^2 + p \\ \rho uv \\ (E + p)u \\ \rho u Y \end{pmatrix}, \quad \mathbf{g}(\mathbf{w}) = \begin{pmatrix} n \\ \rho uv \\ \rho v^2 + p \\ (E + p)v \\ \rho v Y \end{pmatrix}, \quad \mathbf{s}(\mathbf{w}) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \omega \end{pmatrix},$$

where

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

$$m = \rho u, \quad n = \rho v, \quad \omega = -K\rho Y \exp(-E_a/\tilde{T}), \tilde{T} = p/\rho.$$

Here, ω is in the Arrhenius kinetic form, γ is the heat ratio, $\tilde{T}$ is the temperature, Y is the reactant mass fraction, and Q is the heat release of the reaction. Solving the energy equation for pressure, we obtain

$$p = (\gamma - 1) \left(\frac{m^2 + n^2}{2\rho} - Q(\rho Y) \right).$$

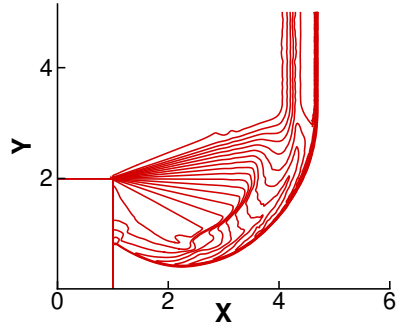
The sum of concave functions is concave. For $\rho > 0$, $\frac{m^2 + n^2}{2\rho}$ is clearly concave, and $Q(\rho Y)$ is linear for a constant Q . Thus, sweeping is applied to the modified pressure function. The domain is $[0, 1] \times [2, 5] \cup [1, 5] \times [0, 5]$. The initial condition is given by

$$\begin{cases} (\rho, u, v, E, Y) = (11, 6.18, 0, 970, 1), & \text{if } x < 0.5, \\ (\rho, u, v, E, Y) = (1, 0, 0, 55, 1), & \text{if } x \geq 0.5 \end{cases}$$

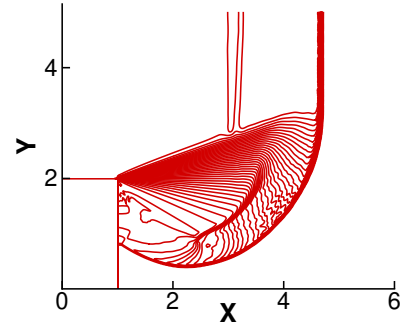
At $x = 0$, we have the boundary condition

$$(\rho, u, v, E, Y) = (11, 6.18, 0, 970, 1),$$

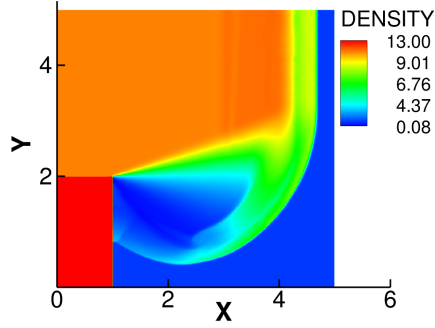
while the remaining boundaries are reflective. We run our numerical test on $\Omega = [0, 5] \times [0, 5]$ with 400×400 points until the final time $T = 0.6$. We use $\text{CFL} = 0.89$ to produce negative pressure values. We use constants $K = 2566.4$, $Q = 50$, $E_a = 50$, and $\gamma = 1.2$. Even with a high CFL number at the instability threshold of the underlying scheme, the sweeping method prevents blow-ups of the numerical solution and preserves key features. The results are shown in Figure 7.



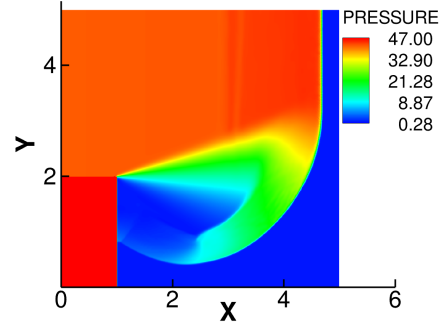
Density Contour Lines



Pressure Contour Lines



Density Color Contour



Pressure Color Contour

Fig. 7 Example 7: Approximate solution of detonation diffraction test case of reactive Euler equations using fifth order WENO with sweeping limiter. Upper left plot: 20 contour lines between 0 and 13. Upper right plot: 40 contour lines between 0 and 43.

Here, we provide a table of the maximum number of sweeps required at any given RK step. We also provide the average and total number of sweeps required by the final time. A full sweep includes the forward and backward direction. In most cases, only one forward and backward sweep is necessary for positivity preservation. We reiterate that the density sweep requires at most one forward and backward sweep, thus we only provide the total number of density sweeps until final time.

Table 2 Pressure Sweeping Data

	1D Rarefaction	1D Sedov	Vortex Accuracy	2D Sedov
Max Sweep	1	1	1	1
Average Sweep	1	1	1	1
Total Sweep	2	1	8	3
	Mach 2000	Shock Diffraction	Reactive Euler	
Max Sweep	2	2	1	
Average Sweep	1.9106	1.0004	1	
Total Sweep	18185	2376	286	

Table 3 Density Sweeping Data

	1D Rarefaction	1D Sedov	Vortex Accuracy	2D Sedov
Total Sweep	0	1	0	0
	Mach 2000	Shock Diffraction	Reactive Euler	
Total Sweep	0	890	22	

5 Concluding Remarks

In this work, we provide a conservative, positivity-preserving sweeping procedure for the pressure function in the Euler equations of gas dynamics as a post-processing step for both finite difference and finite volume methods. We show that the procedure is conservative and positivity-preserving. Our results for fifth order finite difference WENO demonstrate that the sweeping procedure is high-order and robust with only one or two sweeps needed at any given Runge-Kutta time-stepping stage, and high order accuracy is always maintained.

We aim to extend the sweeping procedure to the Navier-Stokes equations and other applications where positivity for concave functions of the conserved variables is required. Other applications include magnetohydrodynamics using implicit algorithms.

6 Acknowledgements

The research of the first author is partially supported by the NSF Graduate Research Fellowship Program. The research of the second author is partially supported by NSF grant DMS-2309249.

References

- [1] Book, D.L., Boris, J.P., and Hain, K. Flux-corrected transport II: Generalizations of the method. *Journal of Computational Physics*, 18(3), 248–283. (1975). [https://doi.org/10.1016/0021-9991\(75\)90002-9](https://doi.org/10.1016/0021-9991(75)90002-9).
- [2] Boris, J.P. and Book, D.L. Flux-corrected transport. I. SHASTA, a fluid transport algorithm that works. *Journal of Computational Physics*, 11(1), 38–69. (1973). [https://doi.org/10.1016/0021-9991\(73\)90147-2](https://doi.org/10.1016/0021-9991(73)90147-2).
- [3] Boris, J.P. and Book, D.L. Flux-corrected transport. III. Minimal-error FCT algorithms. *Journal of Computational Physics*, 20(4), 397–431. (1976). [https://doi.org/10.1016/0021-9991\(76\)90091-7](https://doi.org/10.1016/0021-9991(76)90091-7).
- [4] Cai, X., Zhang, X., and Qiu, J. Positivity-preserving high order finite volume HWENO schemes for compressible Euler equations. *Journal of Scientific Computing*, 68, 464–483. (2016). <https://doi.org/10.1007/s10915-015-0147-8>.
- [5] Christlieb, A.J., Liu, Y., Tang, Q., and Xu, Z. High order parametrized maximum-principle-preserving and positivity-preserving WENO schemes on unstructured meshes. *Journal of Computational Physics*, 281, 334–351. (2015). <https://doi.org/10.1016/j.jcp.2014.10.029>.

- [6] Cockburn, B., Lin, S.Y., and Shu, C.-W. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: One-dimensional systems. *Journal of Computational Physics*, 84(1), 90–113. (1989). [https://doi.org/10.1016/0021-9991\(89\)90183-6](https://doi.org/10.1016/0021-9991(89)90183-6).
- [7] Cockburn, B. and Shu, C.-W. Runge–Kutta discontinuous Galerkin methods for convection-dominated problems. *Journal of Scientific Computing*, 16, 173–261. (2001). <https://doi.org/10.1023/A:1012873910884>.
- [8] Harten, A., Engquist, B., Osher, S., and Chakravarty, S.R. Uniformly high order accurate essentially non-oscillatory schemes, III. *Journal of Computational Physics*, 131(1), 3–47. (1997). <https://doi.org/10.1006/jcph.1996.5632>.
- [9] Hu, X.Y., Adams, N.A., and Shu, C.-W. Positivity-preserving method for high-order conservative schemes solving compressible Euler equations. *Journal of Computational Physics*, 242, 169–180. (2013). <https://doi.org/10.1016/j.jcp.2013.01.024>.
- [10] Jiang, G.-S. and Shu, C.-W. Efficient implementation of weighted ENO schemes. *Journal of Computational Physics*, 126(1), 202–228. (1996). <https://doi.org/10.1006/jcph.1996.0130>.
- [11] Laiu, M.P. and Hauck, C.D. Positivity limiters for filtered spectral approximations of linear kinetic transport equations. *Journal of Scientific Computing*, 78, 918–950. (2019). <https://doi.org/10.1007/s10915-018-0790-y>.
- [12] Li, J.-L., Don, W.-S., Wang, C.-F., and Wang, B.-S. Spatial-temporal adaptive-order positivity-preserving WENO finite difference scheme with relaxed CFL condition for Euler equations with extreme conditions. *Advances in Applied Mathematics and Mechanics*, 17(3), 804–839. (2025).

- [13] Li, H., Xie, S., and Zhang, X. A high order accurate bound-preserving compact finite difference scheme for scalar convection diffusion equations. *SIAM Journal on Numerical Analysis*, 56(6), 3308–3345. (2018). <https://doi.org/10.1137/18M1208551>.
- [14] Liang, C. and Xu, Z. Parametrized maximum principle preserving flux limiters for high order schemes solving multi-dimensional scalar hyperbolic conservation laws. *Journal of Scientific Computing*, 58, 41–60. (2014). <https://doi.org/10.1007/s10915-013-9724-x>.
- [15] Liu, X.-D., Osher, S., and Chan, T. Weighted essentially non-oscillatory schemes. *Journal of Computational Physics*, 115(1), 200–212. (1994). <https://doi.org/10.1006/jcph.1994.1187>.
- [16] Liu, Y., Cheng, Y., and Shu, C.-W. A simple bound-preserving sweeping technique for conservative numerical approximations. *Journal of Scientific Computing*, 73, 1028–1071. (2017). <https://doi.org/10.1007/s10915-017-0395-x>.
- [17] Perthame, B. and Shu, C.-W. On positivity preserving finite volume schemes for Euler equations. *Numerische Mathematik*, 73, 119–130. (1996). <https://doi.org/10.1007/s002110050187>.
- [18] Ren, B., Wang, B.-S., Zhang, X., and Gao, Z. Positivity and bound preserving well-balanced high order compact finite difference scheme for Ripa and pollutant transport model. *Computers & Mathematics with Applications*, 176, 545–563. (2024). <https://doi.org/10.1016/J.CAMWA.2024.11.012>.
- [19] Roe, P. L. Approximate Riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics*, 43(2), 357–372. (1981).

- [20] Seal, D.C., Tang, Q., Xu, Z., and Christlieb, A.J. An explicit high-order single-stage single-step positivity-preserving finite difference WENO method for the compressible Euler equations. *Journal of Scientific Computing*, 68, 171–190. (2014). <https://doi.org/10.1007/s10915-015-0134-0>.
- [21] Shashkov, M. and Wendroff, B. The repair paradigm and application to conservation laws. *Journal of Computational Physics*, 198(1), 265–277. (2004). <https://doi.org/10.1016/j.jcp.2004.01.014>.
- [22] Shu, C.-W. and Osher, S. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of Computational Physics*, 77(2), 439–471. (1988). [https://doi.org/10.1016/0021-9991\(88\)90177-5](https://doi.org/10.1016/0021-9991(88)90177-5).
- [23] Shu, C.-W. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. In Cockburn, B., Johnson, C., Shu, C.-W., and Tadmor, E., editors, *Advanced numerical approximation of nonlinear hyperbolic equations*, volume 1697 of *Lecture Notes in Mathematics*, pages 325–432. Springer, Berlin. (1998).
- [24] Shu, C.-W. High-order finite difference and finite volume WENO schemes and discontinuous Galerkin methods for CFD. *International Journal of Computational Fluid Dynamics*, 17(2), 107–118. (2003). <https://doi.org/10.1080/1061856031000104851>.
- [25] Shu, C.-W. High order weighted essentially nonoscillatory schemes for convection dominated problems. *SIAM Review*, 51(1), 82–126. (2009). <https://doi.org/10.1137/070679065>.

- [26] Tsybulnik, E., Zhu, X., and Zhang, Y.-T. Efficient sparse-grid implementation of a fifth-order multi-resolution WENO scheme for hyperbolic equations. *Communications on Applied Mathematics and Computation*, 5, 1339–1364. (2023). <https://doi.org/10.1007/s42967-022-00202-4>.
- [27] Wang, C., Zhang, X., Shu, C.-W., and Ning, J. Robust high order discontinuous Galerkin schemes for two-dimensional gaseous detonations. *Journal of Computational Physics*, 231(2), 653–665. (2012). <https://doi.org/10.1016/j.jcp.2011.10.002>.
- [28] Xing, Y. and Zhang, X. Positivity-preserving well-balanced discontinuous Galerkin methods for the shallow water equations on unstructured triangular meshes. *Journal of Scientific Computing*, 57, 19–41. (2013). <https://doi.org/10.1007/s10915-013-9695-y>.
- [29] Xing, Y., Zhang, X., and Shu, C.-W. Positivity-preserving high order well-balanced discontinuous Galerkin methods for the shallow water equations. *Advances in Water Resources*, 33(12), 1476–1493. (2010). <https://doi.org/10.1016/j.advwatres.2010.08.005>.
- [30] Xiong, T., Qiu, J.-M., and Xu, Z. A parametrized maximum principle preserving flux limiter for finite difference RK-WENO schemes with applications in incompressible flows. *Journal of Computational Physics*, 252, 310–331. (2013). <https://doi.org/10.1016/j.jcp.2013.06.026>.
- [31] Xiong, T., Qiu, J.-M., and Xu, Z. Parametrized positivity preserving flux limiters for the high order finite difference WENO scheme solving compressible Euler equations. *Journal of Scientific Computing*, 67, 1066–1088. (2016). <https://doi.org/10.1007/s10915-015-0118-0>.

- [32] Zhang, X., Liu, Y., and Shu, C.-W. Maximum-principle-satisfying high order finite volume weighted essentially nonoscillatory schemes for convection-diffusion equations. *SIAM Journal on Scientific Computing*, 34(2), 627–658. (2012). <https://doi.org/10.1137/110839230>.
- [33] Zhang, X. and Shu, C.-W. On maximum-principle-satisfying high order schemes for scalar conservation laws. *Journal of Computational Physics*, 229(9), 3091–3120. (2010a). <https://doi.org/10.1016/j.jcp.2009.12.030>.
- [34] Zhang, X. and Shu, C.-W. On positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations on rectangular meshes. *Journal of Computational Physics*, 229(23), 8918–8934. (2010b). <https://doi.org/10.1016/j.jcp.2010.08.016>.
- [35] Zhang, X. and Shu, C.-W. Maximum-principle-satisfying and positivity-preserving high-order schemes for conservation laws: Survey and new developments. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 467(2134), 2752–2776. (2011a). <https://doi.org/10.1098/RSPA.2011.0153>.
- [36] Zhang, X. and Shu, C.-W. Positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations with source terms. *Journal of Computational Physics*, 230(4), 1238–1248. (2011b). <https://doi.org/10.1016/j.jcp.2010.10.036>.
- [37] Zhang, X. and Shu, C.-W. Positivity-preserving high order finite difference WENO schemes for compressible Euler equations. *Journal of Computational Physics*, 231(5), 2245–2258. (2012). <https://doi.org/10.1016/j.jcp.2011.11.020>.

- [38] Zhang, X., Xia, Y., and Shu, C.-W. Maximum-principle-satisfying and positivity-preserving high order discontinuous Galerkin schemes for conservation laws on triangular meshes. *Journal of Scientific Computing*, 50, 29–62. (2012). <https://doi.org/10.1007/s10915-011-9472-8>.
- [39] Zhang, Y., Zhang, X., and Shu, C.-W. Maximum-principle-satisfying second order discontinuous Galerkin schemes for convection–diffusion equations on triangular meshes. *Journal of Computational Physics*, 234, 295–316. (2013). <https://doi.org/10.1016/j.jcp.2012.09.032>.
- [40] Zhang, X. On positivity-preserving high order discontinuous Galerkin schemes for compressible Navier–Stokes equations. *Journal of Computational Physics*, 328, 301–343. (2017). <https://doi.org/10.1016/j.jcp.2016.10.002>.