Computers and Fluids 300 (2025) 106738

Contents lists available at ScienceDirect

Computers and Fluids

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

High-order flux splitting schemes for the Euler equations of gas dynamics

Shaoshuai Chu [a,*], Michael Herty [a,b], Eleuterio F. Toro [c]

[a] Department of Mathematics, RWTH Aachen University, Aachen, 52056, Germany
[b] Department of Mathematics and Applied Mathematics, University of Pretoria, Private Bag X20, Hatfield, 0028, South Africa
[c] Laboratory of Applied Mathematics, DICAM, University of Trento, Trento, 14-38122, Italy

ARTICLE INFO

MSC:
35L65
65M06
65M08
76M12
76M20
76L05
76N15

Keywords:
Flux splitting schemes
Euler equations of gas dynamics
Advection system
Pressure system
A-WENO schemes

ABSTRACT

We develop high-order flux splitting schemes for the one- and two-dimensional Euler equations of gas dynamics. The proposed schemes are high-order extensions of the existing first-order flux splitting schemes introduced in Toro and Vázquez-Cendón (2012) where the Euler equations of gas dynamics are split into two subsystems: the advection and pressure systems. In this paper, we formulate the TV splitting within the semi-discrete framework to extend it to higher orders of accuracy for the first time. The second-order extension is obtained by using piecewise linear interpolant to reconstruct the one-sided point values of the unknowns. The third- and fifth-order schemes are developed using the finite-difference alternative weighted essentially non-oscillatory (A-WENO) framework, which is particularly effective in handling multidimensional problems and provides a more straightforward approach to constructing higher-order WENO schemes. These extensions significantly improve the resolution of discontinuities and the accuracy of numerical solutions, as demonstrated by a series of numerical experiments of both the one- and two-dimensional Euler equations of gas dynamics.

1. Introduction

This paper focuses on numerical solutions of the Euler equations of gas dynamics, which in the one-dimensional (1-D) case, read as

$$\boldsymbol{U}_t + \boldsymbol{F}(\boldsymbol{U})_x = \boldsymbol{0}, \tag{1}$$

with

$$\boldsymbol{U} := (\rho, \rho u, E)^\top, \quad \text{and} \quad \boldsymbol{F}(\boldsymbol{U}) = (\rho u, \rho u^2 + p, u(E+p))^\top, \tag{2}$$

where t is the time, x is the spatial variable, ρ, u, p, and E are the density, velocity, pressure, and total energy, respectively. The system is completed through the following equations of state (EOS):

$$p = (\gamma - 1)\left[E - \frac{1}{2}\rho u^2\right], \tag{3}$$

where the parameter γ represents the specific heat ratio.

It is well-known that equations (1)–(3) model a wide range of physical phenomena such as shock waves, turbulence, and compressible flows. These systems often involve complex wave patterns, including discontinuities like shocks and rarefactions, even when the initial data are very smooth, which presents significant challenges for numerical methods. Traditional approaches, particularly first-order methods, may struggle to resolve these discontinuities accurately, often requiring very fine meshes to achieve satisfactory results.

Since the pioneering works of [1–3], numerous methods have been developed to solve hyperbolic systems like the one in (1); see, e.g., the monographs and review papers [4–9] and the references therein. Here, we focus on flux splitting methods, which are fundamental in computational fluid dynamics due to their ability to decompose fluxes into components corresponding to different wave families, such as shock, contact, and expansion waves. This decomposition enhances the resolution of shock waves and discontinuities, which is crucial for accurate simulations of compressible flows; see, e.g., [10–23]. While classical flux splitting schemes (e.g., [10–14]) are effective, they often struggle to resolve intermediate characteristic fields, leading to excessive dissipation or numerical artifacts. To overcome these limitations, more advanced methods, such as the advection upstream splitting method (AUSM), have been developed (see, e.g., [15]) offering better handling of contact waves and improved resolution of wave patterns. AUSM has since garnered considerable attention and undergone refinement, with further developments [16–18]. Additionally, a flux splitting approach similar to AUSM was proposed in [23], and subsequent advancements have been recorded in works such as [19–22], further enriching the field of computational fluid dynamics.

Recently, a new flux splitting method, known as the TV splitting scheme, was introduced for the 1-D Euler equations (1)–(3) in [24] and

* Corresponding author.
E-mail addresses: chu@igpm.rwth-aachen.de (S. Chu), herty@igpm.rwth-aachen.de (M. Herty), toro@ing.unitn.it (E.F. Toro).

https://doi.org/10.1016/j.compfluid.2025.106738
Received 2 April 2025; Received in revised form 26 June 2025; Accepted 29 June 2025
Available online 15 July 2025

later extended to higher dimensions in [25], where the system (1)–(3) has been divided into two subsystems: the advection and pressure systems. These subsystems are analyzed, and corresponding Godunov-type discretization schemes are formulated. The proposed schemes are characterized by their simplicity, robustness, and accuracy, offering significant improvements over existing flux splitting methods. In particular, they effectively capture contact and shear waves while precisely preserving isolated stationary contacts. Beyond its application to high-dimensional Euler equations, it has also been extended to other systems, including magnetohydrodynamics and shallow water equations; see, e.g., [25–29].

The TV splitting schemes introduced in [24,25] offer a robust and accurate first-order approach for solving the Euler equations of gas dynamics. However, their accuracy is inherently limited. These methods typically require very fine meshes to achieve acceptable resolution for shock waves and other important wave structures. As the complexity increases, higher-order methods are necessary to improve the resolution and reduce the computational cost associated with fine grids. In this paper, we extend these schemes to higher orders of accuracy — second, third, and fifth — in the semi-discrete framework for the first time. The second-order extension is achieved through applying piecewise linear interpolant to reconstruct the one-sided point values of the unknowns, while third- and fifth-order schemes are developed within the finite-difference (FD) alternative essentially non-oscillatory (A-WENO) framework. This framework has been proven to be a powerful tool for generalizing low-order finite-volume (FV) schemes to higher-order FD ones, particularly in multidimensional cases. Its "dimension-by-dimension" reconstruction process simplifies the development of higher-order WENO schemes; see, e.g., [30–35].

The proposed TV splitting schemes are applied to both 1-D and two-dimensional (2-D) Euler equations of gas dynamics. We test the first-, second-, third-, and fifth-order versions of these schemes on various numerical examples. As expected, the resolution improves with higher-order schemes used, especially when transiting from the first-order scheme to the second-order one. Additionally, we compare the studied TV splitting schemes with the Central-Upwind (CU) schemes (see, e.g.,[36–38]), or HLL schemes (see, e.g., [39]), and the HLLC schemes (see, e.g., [40]), demonstrating the high efficiency of the proposed methods.

The rest of this paper is organized as follows. In Section 2, beginning with a brief overview of the existing first-order TV splitting scheme for the 1-D Euler equations of gas dynamics, we extend it to second-, third-, and fifth-order accuracy. In Section 3, we first introduce the 2-D first-order TV splitting scheme and similarly extend it to high orders. Finally, in Section 4, we present a number of the 1-D and 2-D numerical results to demonstrate their performances.

2. One-dimensional TV splitting schemes

In this section, we first describe the first-order TV splitting scheme for the 1-D Euler equations (1)–(3) from [24] and then extend it to second-, third-, and fifth-order accuracy.

2.1. 1-D first-order TV splitting schemes: A brief overview

Supposing the computational domain is covered with uniform cells $C_j := [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$ with $x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}} \equiv \Delta x$ centered at $x_j = (x_{j-\frac{1}{2}} + x_{j+\frac{1}{2}})/2$, $j = 1, \ldots, N$, we assume that the cell average values

$$\overline{\boldsymbol{U}}_j(t) :\approx \frac{1}{\Delta x}\int_{C_j} \boldsymbol{U}(x,t)\,\mathrm{d}x$$

are available at a certain time level t. Note that all of the indexed quantities are time-dependent, but from here on, we suppress the time-dependence of all of the indexed quantities for the sake of brevity.

The computed cell averages $\overline{\boldsymbol{U}}_j$ of the 1-D system (1)–(3) are evolved in time by solving the following semi-discrete system of ordinary differential equations (ODEs):

$$\frac{\mathrm{d}\overline{\boldsymbol{U}}_j}{\mathrm{d}t} = -\frac{\boldsymbol{\mathcal{F}}^{\mathrm{FV}}_{j+\frac{1}{2}} - \boldsymbol{\mathcal{F}}^{\mathrm{FV}}_{j-\frac{1}{2}}}{\Delta x}, \tag{4}$$

where $\boldsymbol{\mathcal{F}}^{\mathrm{FV}}_{j+\frac{1}{2}} = \boldsymbol{\mathcal{F}}^{\mathrm{FV}}_{j+\frac{1}{2}}\left(\boldsymbol{U}^-_{j+\frac{1}{2}}, \boldsymbol{U}^+_{j+\frac{1}{2}}\right)$ is the numerical flux, defined by

$$\boldsymbol{\mathcal{F}}^{\mathrm{FV}}_{j+\frac{1}{2}}\left(\boldsymbol{U}^-_{j+\frac{1}{2}}, \boldsymbol{U}^+_{j+\frac{1}{2}}\right) = \boldsymbol{\mathcal{F}}^{A}_{j+\frac{1}{2}}\left(\boldsymbol{U}^-_{j+\frac{1}{2}}, \boldsymbol{U}^+_{j+\frac{1}{2}}\right) + \boldsymbol{\mathcal{F}}^{P}_{j+\frac{1}{2}}\left(\boldsymbol{U}^-_{j+\frac{1}{2}}, \boldsymbol{U}^+_{j+\frac{1}{2}}\right). \tag{5}$$

with $\boldsymbol{U}^{\pm}_{j+\frac{1}{2}}$ being the left/right-sided point values of $\boldsymbol{U}$ at the cell interfaces $x_{j+\frac{1}{2}}$. For the first order scheme, we take $\boldsymbol{U}^+_{j+\frac{1}{2}} = \boldsymbol{U}_{j+1}$ and $\boldsymbol{U}^-_{j+\frac{1}{2}} = \boldsymbol{U}_j$.

Here, $\boldsymbol{\mathcal{F}}^{A}_{j+\frac{1}{2}}\left(\boldsymbol{U}^-_{j+\frac{1}{2}}, \boldsymbol{U}^+_{j+\frac{1}{2}}\right)$ is the advection flux given by

$$\boldsymbol{\mathcal{F}}^{A}_{j+\frac{1}{2}}\left(\boldsymbol{U}^-_{j+\frac{1}{2}}, \boldsymbol{U}^+_{j+\frac{1}{2}}\right) = \begin{cases} u^*_{j+\frac{1}{2}} \begin{pmatrix} \rho^-_{j+\frac{1}{2}} \\ (\rho u)^-_{j+\frac{1}{2}} \\ \frac{1}{2}\rho^-_{j+\frac{1}{2}}(u^-_{j+\frac{1}{2}})^2 \end{pmatrix}, & \text{if } u^*_{j+\frac{1}{2}} \geq 0, \\ u^*_{j+\frac{1}{2}} \begin{pmatrix} \rho^+_{j+\frac{1}{2}} \\ (\rho u)^+_{j+\frac{1}{2}} \\ \frac{1}{2}\rho^+_{j+\frac{1}{2}}(u^+_{j+\frac{1}{2}})^2 \end{pmatrix}, & \text{otherwise,} \end{cases}$$

and $\boldsymbol{\mathcal{F}}^{P}_{j+\frac{1}{2}}\left(\boldsymbol{U}^-_{j+\frac{1}{2}}, \boldsymbol{U}^+_{j+\frac{1}{2}}\right)$ is the pressure flux given by

$$\boldsymbol{\mathcal{F}}^{P}_{j+\frac{1}{2}}\left(\boldsymbol{U}^-_{j+\frac{1}{2}}, \boldsymbol{U}^+_{j+\frac{1}{2}}\right) = \begin{pmatrix} 0 \\ p^*_{j+\frac{1}{2}} \\ \gamma u^*_{j+\frac{1}{2}} p^*_{j+\frac{1}{2}} \\ \hline \gamma - 1 \end{pmatrix},$$

where

$$\begin{aligned} u^*_{j+\frac{1}{2}} &= \frac{C^+_{j+\frac{1}{2}} u^+_{j+\frac{1}{2}} - C^-_{j+\frac{1}{2}} u^-_{j+\frac{1}{2}}}{C^+_{j+\frac{1}{2}} - C^-_{j+\frac{1}{2}}} - \frac{2}{C^+_{j+\frac{1}{2}} - C^-_{j+\frac{1}{2}}}(p^+_{j+\frac{1}{2}} - p^-_{j+\frac{1}{2}}), \\ p^*_{j+\frac{1}{2}} &= \frac{C^+_{j+\frac{1}{2}} p^-_{j+\frac{1}{2}} - C^-_{j+\frac{1}{2}} p^+_{j+\frac{1}{2}}}{C^+_{j+\frac{1}{2}} - C^-_{j+\frac{1}{2}}} + \frac{C^+_{j+\frac{1}{2}} C^-_{j+\frac{1}{2}}}{2(C^+_{j+\frac{1}{2}} - C^-_{j+\frac{1}{2}})}(u^+_{j+\frac{1}{2}} - u^-_{j+\frac{1}{2}}), \\ C^{\pm}_{j+\frac{1}{2}} &= \rho^{\pm}_{j+\frac{1}{2}}\left(u^{\pm}_{j+\frac{1}{2}} \pm \sqrt{\left(u^{\pm}_{j+\frac{1}{2}}\right)^2 + 4\left(c^+_{j+\frac{1}{2}}\right)^2}\right), \end{aligned} \tag{6}$$

with

$$u^{\pm}_{j+\frac{1}{2}} = \frac{(\rho u)^{\pm}_{j+\frac{1}{2}}}{\rho^{\pm}_{j+\frac{1}{2}}}, \quad p^{\pm}_{j+\frac{1}{2}} = (\gamma - 1)\left(E^{\pm}_{j+\frac{1}{2}} - \frac{1}{2}\rho^{\pm}_{j+\frac{1}{2}}(u^{\pm}_{j+\frac{1}{2}})^2\right),$$

and

$$c^{\pm}_{j+\frac{1}{2}} = \sqrt{\gamma p^{\pm}_{j+\frac{1}{2}} / \rho^{\pm}_{j+\frac{1}{2}}}.$$

2.2. 1-D second-order TV splitting scheme

We now extend the first-order TV splitting scheme introduced in Section 2.1 to the second order of accuracy. The resulting scheme (4)–(6) achieves second-order accuracy provided that the one-sided point values $\boldsymbol{U}^{\pm}_{j+\frac{1}{2}}$, used to compute the numerical flux $\boldsymbol{\mathcal{F}}^{\mathrm{FV}}_{j+\frac{1}{2}}$ in (5), are second-order of accurate. To this end, we approximate $\boldsymbol{U}^{\pm}_{j+\frac{1}{2}}$ using a piecewise linear interpolant

$$\widetilde{\boldsymbol{U}}(x) = \overline{\boldsymbol{U}}_j + (\boldsymbol{U}_x)_j(x - x_j), \quad x \in C_j, \tag{7}$$

which leads to

$$\boldsymbol{U}^{-}_{j+\frac{1}{2}} = \overline{\boldsymbol{U}}_j + \frac{\Delta x}{2}(\boldsymbol{U}_x)_j, \quad \boldsymbol{U}^{+}_{j+\frac{1}{2}} = \overline{\boldsymbol{U}}_{j+1} - \frac{\Delta x}{2}(\boldsymbol{U}_x)_{j+1}. \tag{8}$$

In order to ensure the reconstruction (7)–(8) is non-oscillatory, one needs to compute the slopes $(\boldsymbol{U}_x)_j$ in (7) with the help of a nonlinear limiter. In all of the numerical experiments reported in Section 4, we have used a generalized minmod limiter [41–43]:

$$(\boldsymbol{U}_x)_j = \text{minmod}\left(\theta\frac{\overline{\boldsymbol{U}}_j - \overline{\boldsymbol{U}}_{j-1}}{\Delta x}, \frac{\overline{\boldsymbol{U}}_{j+1} - \overline{\boldsymbol{U}}_{j-1}}{2\Delta x}, \theta\frac{\overline{\boldsymbol{U}}_{j+1} - \overline{\boldsymbol{U}}_j}{\Delta x}\right), \tag{9}$$

applied in a component-wise manner. Here, the minmod function is defined as

$$\text{minmod}(z_1, z_2, \ldots) := \begin{cases} \min_j\{z_j\} & \text{if } z_j > 0 \quad \forall j, \\ \max_j\{z_j\} & \text{if } z_j < 0 \quad \forall j, \\ 0 & \text{otherwise.} \end{cases} \tag{10}$$

The parameter $\theta \in [1, 2]$ in (9) is used to control the amount of numerical viscosity present in the resulting scheme, and larger values of θ correspond to sharper but, in general, more oscillatory reconstructions. In this paper, we use $\theta = 1.3$.

2.3. 1-D third-order TV splitting schemes

In this section, we extend the second-order TV splitting scheme introduced in Section 2.2 to the third-order of accuracy in the framework of the finite-difference A-WENO scheme introduced in [34] (see also [30–33]), which has been proven to be a powerful tool for generalizing low-order FV schemes to higher-order FD ones.

Following [34], the point values $\boldsymbol{U}_j$ are evolved in time by solving the following system of ODEs:

$$\frac{\mathrm{d}\boldsymbol{U}_j}{\mathrm{d}t} = -\frac{\boldsymbol{H}_{j+\frac{1}{2}} - \boldsymbol{H}_{j-\frac{1}{2}}}{\Delta x}, \tag{11}$$

where $\boldsymbol{H}_{j+\frac{1}{2}}$ is the (third-order accurate) numerical flux defined by

$$\boldsymbol{H}_{j+\frac{1}{2}} = \boldsymbol{\mathcal{F}}^{\text{FV}}_{j+\frac{1}{2}} - \frac{1}{24}(\Delta x)^2(\boldsymbol{F}_{xx})_{j+\frac{1}{2}}.$$

Here, $\boldsymbol{\mathcal{F}}^{\text{FV}}_{j+\frac{1}{2}}$ is the FV numerical flux as in (5) and $(\boldsymbol{F}_{xx})_{j+\frac{1}{2}}$ is the higher-order correction term used to increase the order of the numerical flux. The correction term $(\boldsymbol{F}_{xx})_{j+\frac{1}{2}}$ can be approximated using the standard central FDs:

$$(\boldsymbol{F}_{xx})_{j+\frac{1}{2}} = \frac{1}{2(\Delta x)^2}\left[\boldsymbol{F}_{j-1} - \boldsymbol{F}_j - \boldsymbol{F}_{j+1} + \boldsymbol{F}_{j+2}\right].$$

We stress that the resulting scheme is third-order once the one-sided point values $\boldsymbol{U}^{\pm}_{j+\frac{1}{2}}$ employed to compute the numerical flux $\boldsymbol{\mathcal{F}}^{\text{FV}}_{j+\frac{1}{2}}$ are third-order accurate. This can be done by implementing a certain nonlinear limiting procedure like the third-order WENO-type interpolation (see, e.g, [31,32,44]) applied to the local characteristic variables; see Appendix A for a detailed explanation.

2.4. 1-D fifth-order TV splitting schemes

According to [34], to achieve fifth-order accuracy, the point values $\boldsymbol{U}_j$ are evolved in time by solving the system (11) with the (fifth-order accurate) numerical flux

$$\boldsymbol{H}_{j+\frac{1}{2}} = \boldsymbol{\mathcal{F}}^{\text{FV}}_{j+\frac{1}{2}} - \frac{1}{24}(\Delta x)^2(\boldsymbol{F}_{xx})_{j+\frac{1}{2}} + \frac{7}{5760}(\Delta x)^4(\boldsymbol{F}_{xxxx})_{j+\frac{1}{2}},$$

where $\boldsymbol{\mathcal{F}}^{\text{FV}}_{j+\frac{1}{2}}$ is the FV numerical flux as in (5), $(\boldsymbol{F}_{xx})_{j+\frac{1}{2}}$ and $(\boldsymbol{F}_{xxxx})_{j+\frac{1}{2}}$ are the higher-order correction terms computed by the fourth- and second-order accurate FDs, respectively; see, e.g., [31,33,45–47]. Here, we have used the following higher-order correction terms:

$$(\boldsymbol{F}_{xx})_{j+\frac{1}{2}} = \frac{1}{48(\Delta x)^2}\left[-5\boldsymbol{F}_{j-2} + 39\boldsymbol{F}_{j-1} - 34\boldsymbol{F}_j - 34\boldsymbol{F}_{j+1} + 39\boldsymbol{F}_{j+2} - 5\boldsymbol{F}_{j+3}\right],$$
$$(\boldsymbol{F}_{xxxx})_{j+\frac{1}{2}} = \frac{1}{2(\Delta x)^4}\left[\boldsymbol{F}_{j-2} - 3\boldsymbol{F}_{j-1} + 2\boldsymbol{F}_j + 2\boldsymbol{F}_{j+1} - 3\boldsymbol{F}_{j+2} + \boldsymbol{F}_{j+3}\right],$$

where $\boldsymbol{F}_j := \boldsymbol{F}(\boldsymbol{U}_j)$. In order to ensure the resulting scheme is fifth order, the one-sided point values $\boldsymbol{U}^{\pm}_{j+\frac{1}{2}}$ employed to compute the numerical flux $\boldsymbol{\mathcal{F}}^{\text{FV}}_{j+\frac{1}{2}}$ need to be at least fifth-order accurate. This can be done by using a certain nonlinear limiting procedure like the fifth-order WENO-Z interpolation from [30,31,34,48,49] applied to the local characteristic variables; see Appendix B for details.

Remark 2.1. Note that we extend the first-order TV splitting schemes to third- and fifth-order accuracy in the framework of FD A-WENO methods. It is also easy to extend it to even higher-order accuracy in this framework; see, e.g. [49].

3. Two-dimensional schemes

In this section, we first briefly overview the first-order TV splitting scheme for the 2-D Euler equations of gas dynamics and then extend it to the higher order of accuracy.

The 2-D Euler equations of gas dynamics read as

$$\boldsymbol{U}_t + \boldsymbol{F}(\boldsymbol{U})_x + \boldsymbol{G}(\boldsymbol{U})_y = \boldsymbol{0}, \tag{12}$$

with $\boldsymbol{U} := (\rho, \rho u, \rho v, E)^\top$, $\boldsymbol{F}(\boldsymbol{U}) = (\rho u, \rho u^2 + p, \rho uv, u(E + p))^\top$, and $\boldsymbol{G}(\boldsymbol{U}) = (\rho v, \rho uv, \rho v^2 + p, v(E + p))^\top$. Here v is the y-velocity and the rest of the notations are the same as in the 1-D case (2)–(3). The system is completed through the following EOS:

$$p = (\gamma - 1)\left[E - \frac{\rho}{2}(u^2 + v^2)\right]. \tag{13}$$

3.1. 2-D first-order TV splitting schemes

Supposing that the computational domain is covered with uniform cells $C_{j,k} := [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}] \times [y_{k-\frac{1}{2}}, y_{k+\frac{1}{2}}]$ centered at $(x_j, y_k) = \big((x_{j-\frac{1}{2}} + x_{j+\frac{1}{2}})/2, (y_{k+\frac{1}{2}} + y_{k-\frac{1}{2}})/2\big)$ with $x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}} \equiv \Delta x$ and $y_{k+\frac{1}{2}} - y_{k-\frac{1}{2}} \equiv \Delta y$ for all j, k, we assume that the cell averages

$$\overline{\boldsymbol{U}}_{j,k}(t) := \approx \frac{1}{\Delta x \Delta y}\int_{C_{j,k}} \boldsymbol{U}(x, y, t)\,\mathrm{d}x\mathrm{d}y$$

are available at a certain time level t. The cell averages $\overline{\boldsymbol{U}}_{j,k}$ are then evolved in time by numerically solving the following system of ODEs:

$$\frac{\mathrm{d}\overline{\boldsymbol{U}}_{j,k}}{\mathrm{d}t} = -\frac{\boldsymbol{\mathcal{F}}^{\text{FV}}_{j+\frac{1}{2},k} - \boldsymbol{\mathcal{F}}^{\text{FV}}_{j-\frac{1}{2},k}}{\Delta x} - \frac{\boldsymbol{\mathcal{G}}^{\text{FV}}_{j,k+\frac{1}{2}} - \boldsymbol{\mathcal{G}}^{\text{FV}}_{j,k-\frac{1}{2}}}{\Delta y}. \tag{14}$$

Here, $\boldsymbol{\mathcal{F}}^{\text{FV}}_{j+\frac{1}{2},k}\left(\boldsymbol{U}^{-}_{j+\frac{1}{2},k}, \boldsymbol{U}^{+}_{j+\frac{1}{2},k}\right)$ and $\boldsymbol{\mathcal{G}}^{\text{FV}}_{j,k+\frac{1}{2}}\left(\boldsymbol{U}^{-}_{j,k+\frac{1}{2}}, \boldsymbol{U}^{+}_{j,k+\frac{1}{2}}\right)$ are the numerical fluxes, defined by

$$\begin{aligned}
&\boldsymbol{\mathcal{F}}^{\text{FV}}_{j+\frac{1}{2},k}\left(\boldsymbol{U}^{-}_{j+\frac{1}{2},k}, \boldsymbol{U}^{+}_{j+\frac{1}{2},k}\right) = \boldsymbol{\mathcal{F}}^{A}_{j+\frac{1}{2},k}\left(\boldsymbol{U}^{-}_{j+\frac{1}{2},k}, \boldsymbol{U}^{+}_{j+\frac{1}{2},k}\right) + \boldsymbol{\mathcal{F}}^{P}_{j+\frac{1}{2},k}\left(\boldsymbol{U}^{-}_{j+\frac{1}{2},k}, \boldsymbol{U}^{+}_{j+\frac{1}{2},k}\right), \\
&\boldsymbol{\mathcal{G}}^{\text{FV}}_{j,k+\frac{1}{2}}\left(\boldsymbol{U}^{-}_{j,k+\frac{1}{2}}, \boldsymbol{U}^{+}_{j,k+\frac{1}{2}}\right) = \boldsymbol{\mathcal{G}}^{A}_{j,k+\frac{1}{2}}\left(\boldsymbol{U}^{-}_{j,k+\frac{1}{2}}, \boldsymbol{U}^{+}_{j,k+\frac{1}{2}}\right) + \boldsymbol{\mathcal{G}}^{P}_{j,k+\frac{1}{2}}\left(\boldsymbol{U}^{-}_{j,k+\frac{1}{2}}, \boldsymbol{U}^{+}_{j,k+\frac{1}{2}}\right),
\end{aligned} \tag{15}$$

where $\boldsymbol{U}^{\pm}_{j+\frac{1}{2},k}$ and $\boldsymbol{U}^{\pm}_{j,k+\frac{1}{2}}$ are the left/right-sided point values of $\boldsymbol{U}$ at the cell interfaces $(x_{j+\frac{1}{2}}, y_k)$ and $(x_j, y_{k+\frac{1}{2}})$, respectively. In the first-order scheme, we take $\boldsymbol{U}^{+}_{j+\frac{1}{2},k} = \boldsymbol{U}_{j+1,k}$, $\boldsymbol{U}^{+}_{j,k+\frac{1}{2}} = \boldsymbol{U}_{j,k+1}$, and $\boldsymbol{U}^{-}_{j+\frac{1}{2},k} = \boldsymbol{U}^{-}_{j,k+\frac{1}{2}} = \boldsymbol{U}_{j,k}$.

Here, $\mathcal{F}^A_{j+\frac{1}{2},k}\left(\boldsymbol{U}^-_{j+\frac{1}{2},k}, \boldsymbol{U}^+_{j+\frac{1}{2},k}\right)$ is the x-direction advection flux given by

$$\mathcal{F}^A_{j+\frac{1}{2},k}\left(\boldsymbol{U}^-_{j+\frac{1}{2},k}, \boldsymbol{U}^+_{j+\frac{1}{2},k}\right) = \begin{cases} u^*_{j+\frac{1}{2},k}\begin{pmatrix} \rho^-_{j+\frac{1}{2},k} \\ (\rho u)^-_{j+\frac{1}{2},k} \\ (\rho v)^-_{j+\frac{1}{2},k} \\ \frac{1}{2}\rho^-_{j+\frac{1}{2}}\left[(u^-_{j+\frac{1}{2},k})^2 + (v^-_{j+\frac{1}{2},k})^2\right] \end{pmatrix}, & \text{if } u^*_{j+\frac{1}{2},k} \geq 0, \\ u^*_{j+\frac{1}{2},k}\begin{pmatrix} \rho^+_{j+\frac{1}{2},k} \\ (\rho u)^+_{j+\frac{1}{2},k} \\ (\rho v)^+_{j+\frac{1}{2},k} \\ \frac{1}{2}\rho^+_{j+\frac{1}{2}}\left[(u^+_{j+\frac{1}{2},k})^2 + (v^+_{j+\frac{1}{2},k})^2\right] \end{pmatrix}, & \text{otherwise,} \end{cases}$$

and $\mathcal{F}^P_{j+\frac{1}{2},k}\left(\boldsymbol{U}^-_{j+\frac{1}{2},k}, \boldsymbol{U}^+_{j+\frac{1}{2},k}\right)$ is the x-direction pressure flux given by

$$\mathcal{F}^P_{j+\frac{1}{2},k}\left(\boldsymbol{U}^-_{j+\frac{1}{2},k}, \boldsymbol{U}^+_{j+\frac{1}{2},k}\right) = \begin{pmatrix} 0 \\ p^*_{j+\frac{1}{2},k} \\ 0 \\ \dfrac{\gamma u^*_{j+\frac{1}{2},k} p^*_{j+\frac{1}{2},k}}{\gamma - 1} \end{pmatrix},$$

where

$$\begin{aligned} u^*_{j+\frac{1}{2},k} &= \frac{C^+_{j+\frac{1}{2},k} u^+_{j+\frac{1}{2},k} - C^-_{j+\frac{1}{2},k} u^-_{j+\frac{1}{2},k}}{C^+_{j+\frac{1}{2},k} - C^-_{j+\frac{1}{2},k}} - \frac{2}{C^+_{j+\frac{1}{2},k} - C^-_{j+\frac{1}{2},k}}\left(p^+_{j+\frac{1}{2},k} - p^-_{j+\frac{1}{2},k}\right), \\ p^*_{j+\frac{1}{2},k} &= \frac{C^+_{j+\frac{1}{2},k} p^-_{j+\frac{1}{2},k} - C^-_{j+\frac{1}{2},k} p^+_{j+\frac{1}{2},k}}{C^+_{j+\frac{1}{2},k} - C^-_{j+\frac{1}{2},k}} + \frac{C^+_{j+\frac{1}{2},k} C^-_{j+\frac{1}{2},k}}{2(C^+_{j+\frac{1}{2},k} - C^-_{j+\frac{1}{2},k})}\left(u^+_{j+\frac{1}{2},k} - u^-_{j+\frac{1}{2},k}\right), \\ C^\pm_{j+\frac{1}{2},k} &= \rho^\pm_{j+\frac{1}{2},k}\left(u^\pm_{j+\frac{1}{2},k} \pm \sqrt{\left(u^\pm_{j+\frac{1}{2},k}\right)^2 + 4\left(c^\pm_{j+\frac{1}{2},k}\right)^2}\right), \end{aligned} \tag{16}$$

with

$$\begin{aligned} u^\pm_{j+\frac{1}{2},k} &= \frac{(\rho u)^\pm_{j+\frac{1}{2},k}}{\rho^\pm_{j+\frac{1}{2},k}}, \quad v^\pm_{j+\frac{1}{2},k} = \frac{(\rho v)^\pm_{j+\frac{1}{2},k}}{\rho^\pm_{j+\frac{1}{2},k}}, \\ p^\pm_{j+\frac{1}{2},k} &= (\gamma - 1)\left(E^\pm_{j+\frac{1}{2},k} - \frac{1}{2}\rho^\pm_{j+\frac{1}{2},k}[(u^\pm_{j+\frac{1}{2},k})^2 + (v^\pm_{j+\frac{1}{2},k})^2]\right), \end{aligned}$$

and

$$c^\pm_{j+\frac{1}{2},k} = \sqrt{\gamma p^\pm_{j+\frac{1}{2},k} / \rho^\pm_{j+\frac{1}{2},k}}.$$

Similarly, $\mathcal{G}^A_{j,k+\frac{1}{2}}\left(\boldsymbol{U}^-_{j,k+\frac{1}{2}}, \boldsymbol{U}^+_{j+\frac{1}{2},k}\right)$ is the y-direction advection flux given by

$$\mathcal{G}^A_{j,k+\frac{1}{2}}\left(\boldsymbol{U}^-_{j,k+\frac{1}{2}}, \boldsymbol{U}^+_{j+\frac{1}{2},k}\right) = \begin{cases} v^*_{j,k+\frac{1}{2}}\begin{pmatrix} \rho^-_{j,k+\frac{1}{2}} \\ (\rho u)^-_{j,k+\frac{1}{2}} \\ (\rho v)^-_{j,k+\frac{1}{2}} \\ \frac{1}{2}\rho^-_{j+\frac{1}{2}}\left[(u^-_{j,k+\frac{1}{2}})^2 + (v^-_{j,k+\frac{1}{2}})^2\right] \end{pmatrix}, & \text{if } v^*_{j,k+\frac{1}{2}} \geq 0, \\ v^*_{j,k+\frac{1}{2}}\begin{pmatrix} \rho^+_{j,k+\frac{1}{2}} \\ (\rho u)^+_{j,k+\frac{1}{2}} \\ (\rho v)^+_{j,k+\frac{1}{2}} \\ \frac{1}{2}\rho^+_{j,k+\frac{1}{2}}\left[(u^+_{j,k+\frac{1}{2}})^2 + (v^+_{j,k+\frac{1}{2}})^2\right] \end{pmatrix}, & \text{otherwise,} \end{cases}$$

and $\mathcal{G}^P_{j,k+\frac{1}{2}}\left(\boldsymbol{U}^-_{j,k+\frac{1}{2}}, \boldsymbol{U}^+_{j,k+\frac{1}{2}}\right)$ is the y-direction pressure flux given by

$$\mathcal{G}^P_{j,k+\frac{1}{2}}\left(\boldsymbol{U}^-_{j,k+\frac{1}{2}}, \boldsymbol{U}^+_{j,k+\frac{1}{2}}\right) = \begin{pmatrix} 0 \\ 0 \\ p^*_{j,k+\frac{1}{2}} \\ \dfrac{\gamma v^*_{j,k+\frac{1}{2}} p^*_{j,k+\frac{1}{2}}}{\gamma - 1} \end{pmatrix},$$

where

$$\begin{aligned} v^*_{j,k+\frac{1}{2}} &= \frac{C^+_{j,k+\frac{1}{2}} v^+_{j,k+\frac{1}{2}} - C^-_{j,k+\frac{1}{2}} v^-_{j,k+\frac{1}{2}}}{C^+_{j,k+\frac{1}{2}} - C^-_{j,k+\frac{1}{2}}} - \frac{2}{C^+_{j,k+\frac{1}{2}} - C^-_{j,k+\frac{1}{2}}}\left(p^+_{j,k+\frac{1}{2}} - p^-_{j,k+\frac{1}{2}}\right), \\ p^*_{j,k+\frac{1}{2}} &= \frac{C^+_{j,k+\frac{1}{2}} p^-_{j,k+\frac{1}{2}} - C^-_{j,k+\frac{1}{2}} p^+_{j,k+\frac{1}{2}}}{C^+_{j,k+\frac{1}{2}} - C^-_{j,k+\frac{1}{2}}} + \frac{C^+_{j,k+\frac{1}{2}} C^-_{j,k+\frac{1}{2}}}{2(C^+_{j,k+\frac{1}{2}} - C^-_{j,k+\frac{1}{2}})}\left(v^+_{j,k+\frac{1}{2}} - v^-_{j,k+\frac{1}{2}}\right), \\ C^\pm_{j,k+\frac{1}{2}} &= \rho^\pm_{j,k+\frac{1}{2}}\left(v^\pm_{j,k+\frac{1}{2}} \pm \sqrt{\left(v^\pm_{j,k+\frac{1}{2}}\right)^2 + 4\left(c^\pm_{j,k+\frac{1}{2}}\right)^2}\right), \end{aligned} \tag{17}$$

with

$$\begin{aligned} u^\pm_{j,k+\frac{1}{2}} &= \frac{(\rho u)^\pm_{j,k+\frac{1}{2}}}{\rho^\pm_{j,k+\frac{1}{2}}}, \quad v^\pm_{j,k+\frac{1}{2}} = \frac{(\rho v)^\pm_{j,k+\frac{1}{2}}}{\rho^\pm_{j,k+\frac{1}{2}}}, \\ p^\pm_{j,k+\frac{1}{2}} &= (\gamma - 1)\left(E^\pm_{j,k+\frac{1}{2}} - \frac{1}{2}\rho^\pm_{j,k+\frac{1}{2}}[(u^\pm_{j,k+\frac{1}{2}})^2 + (v^\pm_{j,k+\frac{1}{2}})^2]\right), \end{aligned}$$

and

$$c^\pm_{j,k+\frac{1}{2}} = \sqrt{\gamma p^\pm_{j,k+\frac{1}{2}} / \rho^\pm_{j,k+\frac{1}{2}}}.$$

3.2. 2-D second-order TV splitting scheme

As in the 1-D case, the resulting scheme (14)–(15) is second-order accurate once the one-sided point values $\boldsymbol{U}^\pm_{j+\frac{1}{2},k}$ and $\boldsymbol{U}^\pm_{j,k+\frac{1}{2}}$ employed to compute the numerical fluxes (15) are second order. To this end, we approximate $\boldsymbol{U}^\pm_{j+\frac{1}{2},k}$ and $\boldsymbol{U}^\pm_{j,k+\frac{1}{2}}$ by

$$\widetilde{\boldsymbol{U}}(x,y) = \overline{\boldsymbol{U}}_{j,k} + (\boldsymbol{U}_x)_{j,k}(x - x_j) + (\boldsymbol{U}_y)_{j,k}(y - y_k), \quad x \in C_{j,k},$$

which leads to

$$\begin{aligned} \boldsymbol{U}^-_{j+\frac{1}{2},k} &= \overline{\boldsymbol{U}}_{j,k} + \frac{\Delta x}{2}(\boldsymbol{U}_x)_{j,k}, \quad \boldsymbol{U}^+_{j+\frac{1}{2},k} = \overline{\boldsymbol{U}}_{j+1,k} - \frac{\Delta x}{2}(\boldsymbol{U}_x)_{j+1,k}, \\ \boldsymbol{U}^-_{j,k+\frac{1}{2}} &= \overline{\boldsymbol{U}}_{j,k} + \frac{\Delta y}{2}(\boldsymbol{U}_y)_{j,k}, \quad \boldsymbol{U}^+_{j,k+\frac{1}{2}} = \overline{\boldsymbol{U}}_{j+1,k} - \frac{\Delta y}{2}(\boldsymbol{U}_y)_{j+1,k}, \end{aligned}$$

where

$$\begin{aligned} (\boldsymbol{U}_x)_{j,k} &= \text{minmod}\left(\theta\frac{\overline{\boldsymbol{U}}_{j,k} - \overline{\boldsymbol{U}}_{j-1,k}}{\Delta x}, \frac{\overline{\boldsymbol{U}}_{j+1,k} - \overline{\boldsymbol{U}}_{j-1,k}}{2\Delta x}, \theta\frac{\overline{\boldsymbol{U}}_{j+1,k} - \overline{\boldsymbol{U}}_{j,k}}{\Delta x}\right), \\ (\boldsymbol{U}_y)_{j,k} &= \text{minmod}\left(\theta\frac{\overline{\boldsymbol{U}}_{j,k} - \overline{\boldsymbol{U}}_{j,k-1}}{\Delta y}, \frac{\overline{\boldsymbol{U}}_{j,k+1} - \overline{\boldsymbol{U}}_{j,k-1}}{2\Delta y}, \theta\frac{\overline{\boldsymbol{U}}_{j,k+1} - \overline{\boldsymbol{U}}_{j,k}}{\Delta y}\right). \end{aligned}$$

Here, the minmod function is defined by (10).

3.3. 2-D third-order TV splitting schemes

Following [34], the point values $\boldsymbol{U}_{j,k}$ are evolved in time by solving the following system of ODEs:

$$\frac{\mathrm{d}\boldsymbol{U}_{j,k}}{\mathrm{d}t} = -\frac{\boldsymbol{H}_{j+\frac{1}{2},k} - \boldsymbol{H}_{j-\frac{1}{2},k}}{\Delta x} - \frac{\boldsymbol{H}_{j,k+\frac{1}{2}} - \boldsymbol{H}_{j,k+\frac{1}{2}}}{\Delta y}, \tag{18}$$

where the numerical fluxes $\boldsymbol{H}_{j+\frac{1}{2},k}$ and $\boldsymbol{H}_{j,k+\frac{1}{2}}$ are defined by

$$\boldsymbol{H}_{j+\frac{1}{2},k} = \mathcal{F}^{\text{FV}}_{j+\frac{1}{2},k} - \frac{1}{24}(\Delta x)^2(\boldsymbol{F}_{xx})_{j+\frac{1}{2},k}, \quad \boldsymbol{H}_{j,k+\frac{1}{2}} = \mathcal{G}^{\text{FV}}_{j,k+\frac{1}{2}} - \frac{1}{24}(\Delta y)^2(\boldsymbol{G}_{yy})_{j,k+\frac{1}{2}}.$$

Here, $\mathcal{F}^{\text{FV}}_{j+\frac{1}{2},k}$ and $\mathcal{G}^{\text{FV}}_{j,k+\frac{1}{2}}$ are the FV numerical fluxes as in (15), $(\boldsymbol{F}_{xx})_{j+\frac{1}{2},k}$ and $(\boldsymbol{G}_{yy})_{j,k+\frac{1}{2}}$ are the higher-order correction terms computed by the standard central FDs:

$$(\boldsymbol{F}_{xx})_{j+\frac{1}{2},k} = \frac{1}{12(\Delta x)^2}\left[\boldsymbol{F}_{j-1,k} - \boldsymbol{F}_{j,k} - \boldsymbol{F}_{j+1,k} + \boldsymbol{F}_{j+2,k}\right],$$
$$(\boldsymbol{G}_{yy})_{j,k+\frac{1}{2}} = \frac{1}{12(\Delta y)^2}\left[\boldsymbol{G}_{j,k-1} - \boldsymbol{G}_{j,k} - \boldsymbol{G}_{j,k+1} + \boldsymbol{G}_{j,k+2}\right].$$

To ensure the resulting scheme is third-order accuracy, the one-sided point values $\boldsymbol{U}^{\pm}_{j+\frac{1}{2},k}$ and $\boldsymbol{U}^{\pm}_{j,k+\frac{1}{2}}$ are also computed using third-order WENO-type interpolation applied to the local characteristic variables. Note that this can be done in a "dimension-by-dimension" manner as in the 1-D case, we therefore omit the details for the sake of brevity.

3.4. 2-D fifth-order TV splitting schemes

According to [34], the point values $\boldsymbol{U}_j$ are evolved in time by solving the system of ODEs (18) with the following numerical fluxes $\boldsymbol{H}_{j+\frac{1}{2},k}$ and $\boldsymbol{H}_{j,k+\frac{1}{2}}$:

$$\boldsymbol{H}_{j+\frac{1}{2},k} = \mathcal{F}^{\text{FV}}_{j+\frac{1}{2},k} - \frac{1}{24}(\Delta x)^2(\boldsymbol{F}_{xx})_{j+\frac{1}{2},k} + \frac{7}{5760}(\Delta x)^4(\boldsymbol{F}_{xxxx})_{j+\frac{1}{2},k},$$
$$\boldsymbol{G}_{j,k+\frac{1}{2}} = \mathcal{G}^{\text{FV}}_{j,k+\frac{1}{2}} - \frac{1}{24}(\Delta y)^2(\boldsymbol{G}_{yy})_{j,k+\frac{1}{2}} + \frac{7}{5760}(\Delta y)^4(\boldsymbol{G}_{yyyy})_{j,k+\frac{1}{2}}.$$

Here, $\mathcal{F}^{\text{FV}}_{j+\frac{1}{2},k}$ and $\mathcal{G}^{\text{FV}}_{j,k+\frac{1}{2}}$ are the FV numerical fluxes as in (15), $(\boldsymbol{F}_{xx})_{j+\frac{1}{2},k}$, $(\boldsymbol{F}_{xxxx})_{j+\frac{1}{2},k}$, $(\boldsymbol{G}_{yy})_{j,k+\frac{1}{2}}$, $(\boldsymbol{G}_{yyyy})_{j,k+\frac{1}{2}}$ are approximations of the second- and fourth-order spatial derivatives of $\boldsymbol{F}$ at $(x,y) = (x_{j+\frac{1}{2}}, y_k)$ and $\boldsymbol{G}$ at $(x,y) = (x_j, y_{k+\frac{1}{2}})$, respectively. In this paper, we have used the following higher-order correction terms from [46]:

$$(\boldsymbol{F}_{xx})_{j+\frac{1}{2},k} = \frac{1}{48(\Delta x)^2}\left(-5\boldsymbol{F}_{j-2,k} + 39\boldsymbol{F}_{j-1,k} - 34\boldsymbol{F}_{j,k} - 34\boldsymbol{F}_{j+1,k} + 39\boldsymbol{F}_{j+2,k} - 5\boldsymbol{F}_{j+3,k}\right),$$
$$(\boldsymbol{F}_{xxxx})_{j+\frac{1}{2},k} = \frac{1}{2(\Delta x)^4}\left(\boldsymbol{F}_{j-2,k} - 3\boldsymbol{F}_{j-1,k} + 2\boldsymbol{F}_{j,k} + 2\boldsymbol{F}_{j+1,k} - 3\boldsymbol{F}_{j+2,k} + \boldsymbol{F}_{j+3,k}\right),$$
$$(\boldsymbol{G}_{yy})_{j,k+\frac{1}{2}} = \frac{1}{48(\Delta y)^2}\left(-5\boldsymbol{G}_{j,k-2} + 39\boldsymbol{G}_{j,k-1} - 34\boldsymbol{G}_{j,k} - 34\boldsymbol{G}_{j,k+1} + 39\boldsymbol{G}_{j,k+2} - 5\boldsymbol{G}_{j,k+3}\right),$$
$$(\boldsymbol{G}_{yyyy})_{j,k+\frac{1}{2}} = \frac{1}{2(\Delta y)^4}\left(\boldsymbol{G}_{j,k-2} - 3\boldsymbol{G}_{j,k-1} + 2\boldsymbol{G}_{j,k} + 2\boldsymbol{G}_{j,k+1} - 3\boldsymbol{G}_{j,k+2} + \boldsymbol{G}_{j,k+3}\right),$$

where $\boldsymbol{F}_{j,k} := \boldsymbol{F}(\boldsymbol{U}_{j,k})$ and $\boldsymbol{G}_{j,k} := \boldsymbol{G}(\boldsymbol{U}_{j,k})$. To achieve fifth-order accuracy, the one-sided point values $\boldsymbol{U}^{\pm}_{j+\frac{1}{2},k}$ and $\boldsymbol{U}^{\pm}_{j,k+\frac{1}{2}}$ employed to compute the numerical flux $\mathcal{F}^{\text{FV}}_{j+\frac{1}{2},k}$ and $\mathcal{G}^{\text{FV}}_{j,k+\frac{1}{2}}$ need to be at least fifth-order accurate. This can also be done in a "dimension-by-dimension" manner as in the 1-D case, we therefore omit the details for the sake of brevity.

4. Numerical examples

In this section, [2] we test the studied first-, second-, third-, and fifth-order schemes on several numerical examples and compare their performances. For the sake of brevity, these schemes will be referred to as the 1-, 2-, 3-, and 5-Order schemes, respectively.

We numerically integrate the ODE systems (4), (11), (14), and (18) by the three-stage third-order strong stability preserving (SSP) Runge–Kutta method (see, e.g., [50,51]) and use the CFL number 0.45.

4.1. One-dimensional examples

We begin with the 1-D Euler equations of gas dynamics (1)–(3). In all of the Examples 1–5, we take the specific heat ratio $\gamma = 1.4$.

Example 1—1-D accuracy test

In the first example, we consider the system (1)–(3) subject to the following periodic initial conditions,

$$\rho(x,0) = 1 + \frac{1}{10}\sin(2\pi x), \quad u(x,0) \equiv 1, \quad p(x,0) \equiv 1.$$

The exact solution of this initial value problem is given by

$$\rho(x,t) = 1 + \frac{1}{10}\sin\left[2\pi(x-t)\right], \quad u(x,t) \equiv 1, \quad p(x,0) \equiv 1.$$

We first compute the numerical solution on the computational domain $[-1,1]$ until the final time $t = 0.1$ by the 1-, 2-, 3-, and 5-Order schemes on a sequence of uniform meshes: 100, 200, 400, and 800, measure the L^1-errors, and then compute the corresponding experimental convergence rates for the density. The obtained results are presented in Table 1, where one can clearly see that the expected order of accuracy is achieved for the studied schemes.

Remark 4.1. We stress that in order to achieve the fifth order of accuracy for the 5-Order scheme, we use smaller time steps with $\Delta t \sim (\Delta x)^{\frac{5}{3}}$ to balance the spatial and temporal errors.

Example 2—Advection of smooth density

In the second example taken from [8], we evaluate the efficiency of the four studied schemes. We consider the following initial conditions:

$$(\rho, u, p)(x,0) = (2 + \sin^4(\pi x), 1, 1),$$

subject to the period boundary conditions at both ends on the computational interval $[-1,1]$. The exact solution of this initial value problem can be easily obtained and is given by

$$(\rho, u, p)(x,t) = (2 + \sin^4(\pi(x-t)), 1, 1).$$

In Fig. 1, we present the computational cost (bars) versus the order of accuracy for three cases, each corresponding to prescribed L^2-errors (10^{-7}, 10^{-8}, and 10^{-9}). As one can see, the computational cost decreases significantly with increasing order of accuracy, demonstrating the superior efficiency of high-order methods. In fact, high-order methods can achieve the same accuracy as low-order methods while requiring orders of magnitude less CPU time, making them substantially more efficient

Example 3—Shock–density wave interaction problem

In this example taken from [52], we consider the shock–density wave interaction problem with the following initial data,

$$(\rho, u, p)\big|_{(x,0)} = \begin{cases} \left(\frac{27}{7}, \frac{4\sqrt{35}}{9}, \frac{31}{3}\right), & x < -4, \\ (1 + 0.2\sin(5x), 0, 1), & x > -4, \end{cases}$$

prescribed in the computational domain $[-5,5]$ subject to the free boundary conditions.

We compute the numerical solutions until the final time $t = 5$ by the 1-, 2-, 3-, and 5-Order schemes on a uniform mesh of 400 cells, and present the obtained numerical results in Fig. 2 together with the reference solution computed by the 5-Order scheme on a much finer mesh of 8000 cells. One can clearly see that the resolution of the computed density improves significantly when high-order schemes are used.

Example 4—Titarev-Toro problem

In this example, we consider the Titarev–Toro problem taken from [53]; see also [54,55]. The initial conditions,

$$(\rho, u, p)(x,0) = \begin{cases} (1.51695, 0.523346, 1.805), & x < -4.5, \\ (1 + 0.1\sin(20x), 0, 1), & x > -4.5, \end{cases}$$

correspond to a forward-facing shock wave of Mach 1.1 interacting with high-frequency density perturbations, that is, as the shock wave moves, the perturbations spread ahead. In this example, we set free boundary conditions at both ends of the computational domain $[-10,5]$.

We compute the solutions until the final time $t = 5$ by the 1-, 2-, 3-, and 5-Order schemes on a uniform mesh of 1200 cells. The numerical results are shown in Fig. 3 along with the reference solution

Table 1
Example 1: The L^1-errors and experimental convergence rates for the density ρ computed by the 1-, 2-, 3-, and 5-Order schemes.

Mesh	1-Order		2-Order		3-Order		5-Order	
	Error	Rate	Error	Rate	Error	Rate	Error	Rate
100	4.93e−03	–	4.70e−04	–	1.02e−05	–	1.33e−07	–
200	2.49e−03	0.985	1.12e−04	2.07	1.24e−06	3.04	4.40e−09	4.92
400	1.25e−03	0.993	2.76e−05	2.03	1.55e−07	3.00	1.42e−10	4.95
800	6.27e−04	0.996	6.30e−06	2.13	1.94e−08	3.00	4.55e−12	5.00

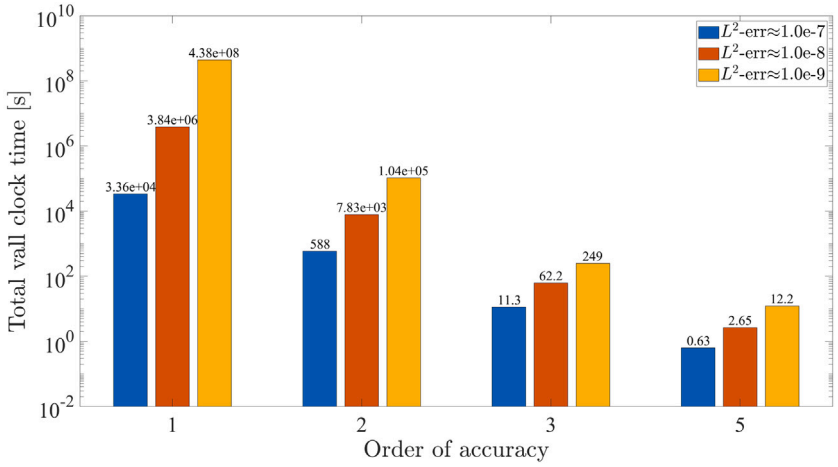

Fig. 1. Example 2: Computational cost (bars) against order of accuracy in space and time for three prescribed errors: 10^{-7}, 10^{-8}, and 10^{-9}.

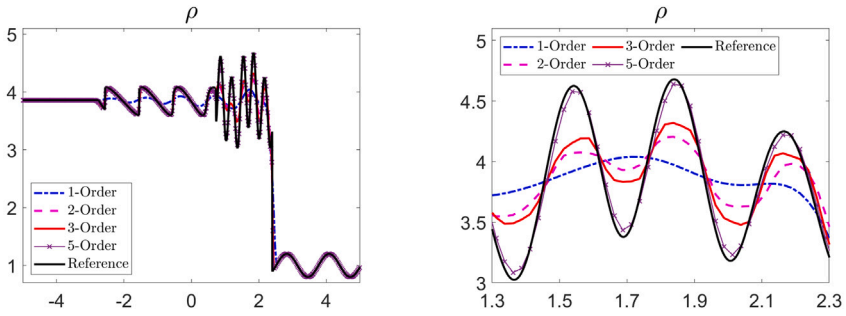

Fig. 2. Example 3: Density ρ computed by the 1-, 2-, 3-, and 5-Order schemes (left) and zoom at $[1.3, 2.3]$ (right).

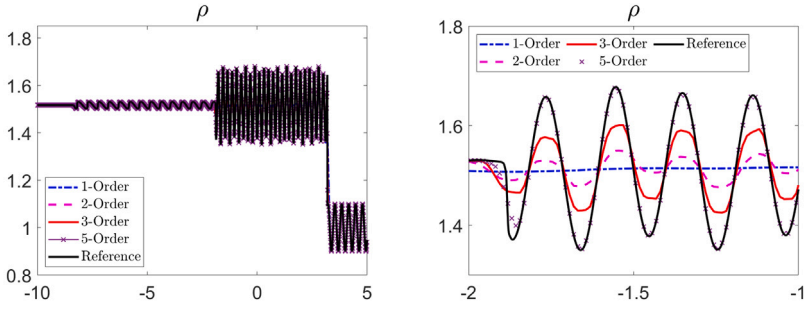

Fig. 3. Example 4: Density ρ computed by the 1-, 2-, 3-, and 5-Order schemes (left) and zoom at $[-2, -1]$ (right).

computed by the 5-Order scheme on a much finer mesh of 12000 cells. The obtained results clearly demonstrate a substantial difference in the resolution computed by schemes with different orders.

In this example, we compare the performances of the studied TV splitting scheme with the CU scheme (see, e.g.,[37,38]) or HLL scheme (see, e.g.,[39]), and HLLC scheme (see e.g.,[40]. We compute the solutions by the corresponding CU (HLL) and HLLC schemes and present the obtained numerical densities in Fig. 4, where one can both TV splitting and HLLC schemes achieve better resolutions than the CU (HLL) scheme. At the same time, HLLC scheme contains slightly less numerical dissipations than the studied TV splitting scheme; see the first two rows in Fig. 4, and the results computed by TV splitting and HLLC schemes are almost identical after extended to higher orders; see the last two rows in Fig. 4.

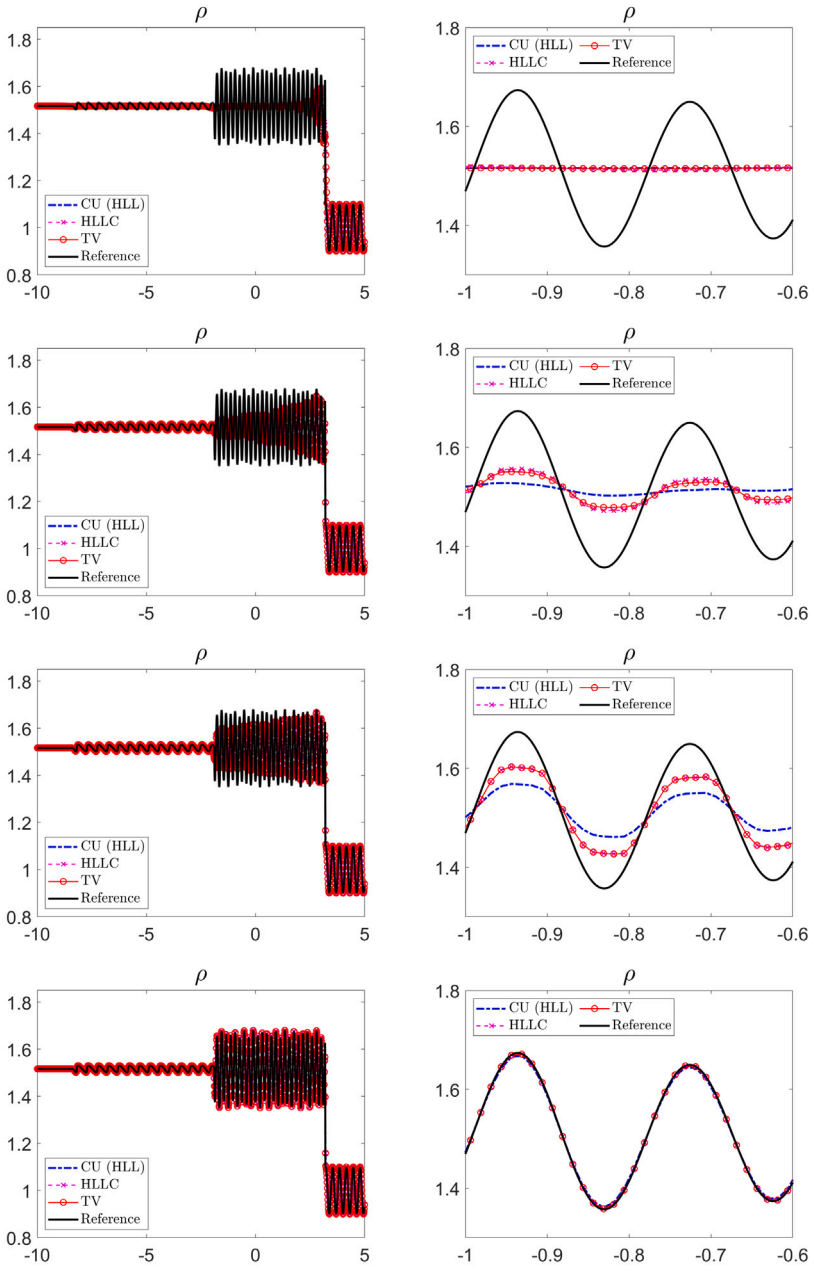

Fig. 4. Example 4: Density ρ computed by the 1- (top row), 2- (second row), 3- (third-row) and 5-Order (bottom row) TV splitting, CU (HLL), and HLLC schemes (left) and zoom at $[-1, -0.6]$ (right).

Example 5—Blast wave problem

In the last 1-D example, we consider the strong shocks interaction problem from [56] with the following initial conditions:

$$(\rho, u, p)(x, 0) = \begin{cases} (1, 0, 1000), & x < 0.1, \\ (1, 0, 0.01), & 0.1 \le x \le 0.9, \\ (1, 0, 100), & x > 0.9, \end{cases}$$

prescribed in the computational domain $[0, 1]$ subject to the solid wall boundary conditions at both ends.

We compute the numerical solutions until the final time $t = 0.038$ by the 1-, 2-, 3-, and 5-Order schemes on a uniform mesh of 400 cells. The obtained results are presented in Fig. 5 together with the reference solution computed by the 5-Order scheme on a much finer mesh of 4000 cells, demonstrating that the resolution of the computed density improves significantly with the use of high-order schemes, especially when transiting from the 1-Order scheme to the 2-Order one.

4.2. Two-dimensional examples

In this section, we consider the 2-D Euler equations of gas dynamics (12)–(13). In Examples 6–10, we take the specific heat ratio $\gamma = 1.4$, while in Example 11, we take $\gamma = 5/3$.

Example 6—2-D accuracy test

In the first 2-D example taken from [46,57,58], we consider the 2-D Euler equations of gas dynamics subject to the periodic initial conditions,

$$\rho(x, y, 0) = 1 + \frac{1}{5}\sin(\pi(x + y)), \quad u(x, y, 0) \equiv 1,$$

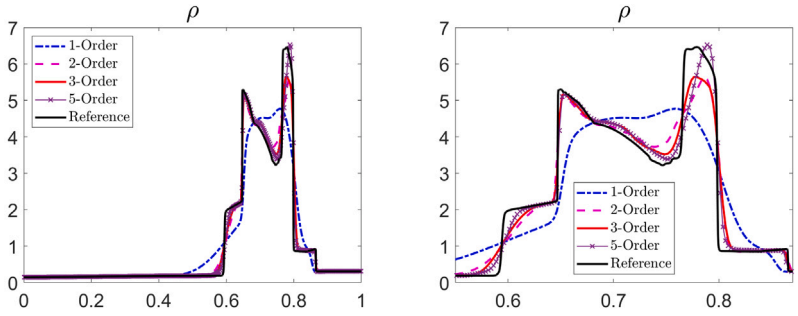

Fig. 5. Example 5: Density ρ computed by the 1-, 2-, 3-, and 5-Order schemes (left) and zoom at $[0.55, 0.87]$ (right).

Table 2
Example 6: The L^1-errors and experimental convergence rates for the density ρ computed by the 1-, 2-, 3-, and 5-Order schemes.

Mesh	1-Order		2-Order		3-Order		5-Order	
	Error	Rate	Error	Rate	Error	Rate	Error	Rate
50 × 50	1.68e−02	–	1.08e−03	–	3.36e−05	–	2.49e−07	–
100 × 100	8.47e−03	0.986	2.64e−04	2.03	4.21e−06	2.99	7.80e−09	5.00
200 × 200	4.25e−03	0.994	6.16e−05	2.10	5.27e−07	3.00	2.44e−10	5.00
400 × 400	2.13e−03	0.997	1.47e−05	2.06	6.59e−08	3.00	7.83e−12	4.96

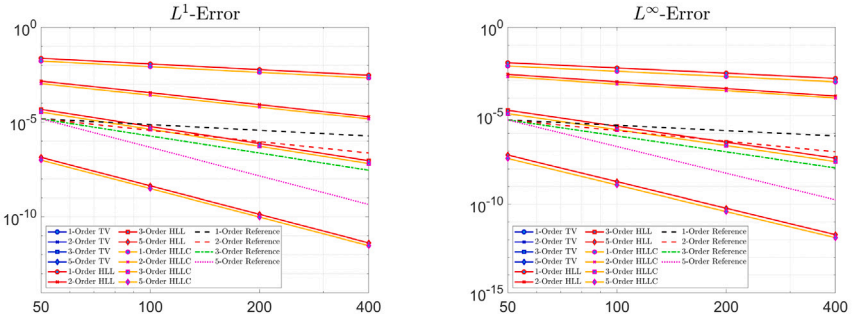

Fig. 6. Example 6: L^1- (left) and L^∞- (right) errors computed by the 1-, 2-, 3-, and 5-Orders TV splitting, CU (HLL), and HLLC schemes.

$$v(x,y,0) \equiv -0.7, \quad p(x,y,0) \equiv 1,$$

prescribed on $[-1,1] \times [-1,1]$. The exact solution of this initial value problem can be easily obtained and is given by

$$\rho(x,y,t) = 1 + \frac{1}{5}\sin\left[\pi(x+y-0.3t)\right], \; u(x,y,t) \equiv 1,$$
$$v(x,y,t) \equiv -0.7, \; p(x,y,0) \equiv 1.$$

We first compute the numerical solution until the final time $t = 0.1$ using the 1-, 2-, 3-, and 5-Order schemes on a sequence of uniform meshes: 50 × 50, 100 × 100, 200 × 200, and 400 × 400. We then measure the L^1-errors and the corresponding experimental convergence rates for the density. The obtained results are presented in Table 2, where one can see that the studied 1-, 2-, 3-, and 5-Order schemes achieve the expected order of accuracy. Similar to Example 1, we had to use smaller time steps with $\Delta t \sim \min\left\{(\Delta x)^{\frac{5}{3}}, (\Delta y)^{\frac{5}{3}}\right\}$ to achieve the fifth order of accuracy.

We also compute the numerical results using the CU (HLL) and HLLC schemes on the same meshes and present the obtained L^1- and L^∞-errors in Fig. 6. As observed, all the studied schemes obtain the expected order of accuracy. Moreover, the TV-splitting schemes demonstrate accuracy comparable to that of the HLLC schemes and exhibit higher accuracy than the corresponding CU (HLL) schemes.

Example 7—2-D vortex evolution problem

In this example taken from [59]; see also [60], we consider the 2-D vortex evolution problem with the following initial conditions

$$(\rho,u,v,p)(x,y,0) = \left(T^{\frac{1}{\gamma-1}}, 1 - \frac{\varepsilon}{2\pi}e^{\frac{1}{2}(1-r^2)}y, 1 + \frac{\varepsilon}{2\pi}e^{\frac{1}{2}(1-r^2)}x, \rho^\gamma\right),$$

where $T = 1 - \frac{(\gamma-1)\varepsilon^2}{8\gamma\pi^2}e^{(1-r^2)}$, $r^2 = x^2 + y^2$, and $\varepsilon = 5$ is the vortex strength. The initial data, which is prescribed in the computational domain $[-5,5] \times [-5,5]$ subject to the periodic boundary conditions, corresponds to a smooth vortex placed at the origin and is defined as the isentropic perturbation to the uniform flow of unit values of primitive variables and the exact solution is a vortex moving with a constant velocity at 45° to the Cartesian mesh lines.

We compute the numerical solution until the final time $t = 10$ using the 1-, 2-, 3-, and 5-Order TV splitting schemes on a sequence of uniform meshes: 25 × 25, 50 × 50, 100 × 100, 200 × 200, and 400 × 400, and then measure the L^1- and L^∞-errors between the computed solutions and the exact solutions and the corresponding experimental convergence rates for the density. We also compare the performances of the TV splitting schemes with the CU (HLL) and HLLC schemes and present the obtained results in Tables 3 and 4. One can see that for the 1- and 2-Order schemes, both first- and second-order convergence rates are observed only after significant mesh refinement, while the 3- and 5-Order TV splitting, CU (HLL), and HLLC schemes

Table 3
Example 7: L^1-errors and experimental convergence rates for the density ρ computed by the 1-, 2-, 3-, and 5-Order TV splitting, CU (HLL), and HLLC schemes.

Method	Mesh	1-Order		2-Order		3-Order		5-Order	
		Error	Rate	Error	Rate	Error	Rate	Error	Rate
TV splitting	25 × 25	2.89	–	2.19e−00	–	8.89e−01	–	1.62e−01	–
	50 × 50	2.84	2.09e−02	6.24e−01	1.81	1.68e−01	2.40	1.01e−02	4.00
	100 × 100	2.49	1.90e−01	1.24e−01	2.33	2.41e−02	2.80	3.73e−04	4.76
	200 × 200	1.91	3.84e−01	3.00e−02	2.05	3.12e−03	2.95	1.20e−05	4.96
	400 × 400	1.26	5.97e−01	6.75e−03	2.15	3.93e−04	2.99	4.65e−07	4.70
CU (HLL)	25 × 25	2.60	–	1.81e−00	–	5.76e−01	–	1.58e−01	–
	50 × 50	2.34	1.52e−01	5.69e−01	1.67	1.18e−01	2.29	1.12e−02	3.83
	100 × 100	1.86	3.30e−01	1.48e−01	1.94	1.83e−02	2.69	4.42e−04	4.66
	200 × 200	1.33	4.84e−01	4.54e−02	1.71	2.37e−03	2.95	1.44e−05	4.94
	400 × 400	0.84	6.71e−01	1.00e−02	2.18	2.98e−04	2.99	4.73e−07	4.93
HLLC	25 × 25	2.53	–	1.68e−00	–	5.39e−01	–	1.53e−01	–
	50 × 50	2.22	1.87e−01	5.36e−01	1.64	1.07e−01	2.33	9.90e−03	3.95
	100 × 100	1.74	3.49e−01	1.44e−01	1.90	1.61e−02	2.74	3.82e−04	4.70
	200 × 200	1.21	5.26e−01	4.67e−02	1.62	2.07e−03	2.95	1.24e−05	4.94
	400 × 400	0.74	7.04e−01	9.88e−03	2.24	2.60e−04	2.99	4.11e−07	4.92

Table 4
Example 7: L^∞-errors and experimental convergence rates for the density ρ computed by the 1-, 2-, 3-, and 5-Order TV splitting, CU (HLL), and HLLC schemes.

Method	Mesh	1-Order		2-Order		3-Order		5-Order	
		Error	Rate	Error	Rate	Error	Rate	Error	Rate
TV splitting	25 × 25	4.81e−01	–	4.38e−01	–	1.96e−01	–	2.61e−02	–
	50 × 50	4.75e−01	1.80e−02	1.78e−01	1.30	3.05e−02	2.68	2.66e−03	3.30
	100 × 100	4.58e−01	5.44e−02	7.02e−02	1.34	4.20e−03	2.86	9.86e−05	4.75
	200 × 200	3.79e−01	2.74e−01	2.35e−02	1.58	5.43e−04	2.95	3.17e−06	4.96
	400 × 400	2.52e−01	5.86e−01	7.13e−03	1.72	6.78e−05	3.00	9.99e−08	4.99
CU (HLL)	25 × 25	4.60e−01	–	3.50e−01	–	1.20e−01	–	4.48e−02	–
	50 × 50	4.26e−01	1.08e−01	1.21e−01	1.53	2.10e−02	2.51	2.18e−03	4.36
	100 × 100	3.70e−01	2.05e−01	6.44e−02	0.91	3.03e−03	2.79	7.99e−05	4.77
	200 × 200	2.67e−01	4.69e−01	3.05e−02	1.08	3.93e−04	2.95	2.64e−06	4.92
	400 × 400	1.58e−01	7.61e−01	8.02e−03	1.93	4.93e−05	2.99	8.14e−08	5.02
HLLC	25 × 25	4.54e−01	–	3.27e−01	–	1.14e−01	–	4.04e−02	–
	50 × 50	4.15e−01	1.30e−01	1.07e−01	1.62	2.02e−02	2.49	1.82e−03	4.47
	100 × 100	3.49e−01	2.47e−01	6.39e−02	0.74	2.86e−03	2.82	7.02e−05	4.70
	200 × 200	2.42e−01	5.30e−01	2.97e−02	1.11	3.65e−04	2.97	2.31e−06	4.93
	400 × 400	1.39e−01	7.97e−01	7.88e−03	1.91	4.57e−05	3.00	7.18e−08	5.00

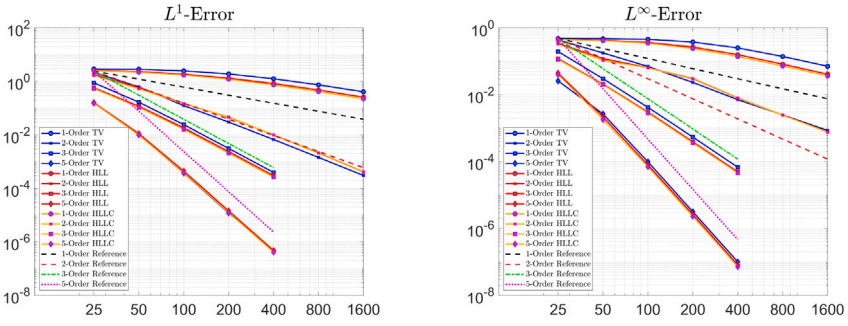

Fig. 7. Example 7: L^1- (left) and L^∞- (right) errors computed by the 1-, 2-, 3-, and 5-Orders TV splitting, HLL, and HLLC schemes.

achieve the expected order of accuracy. In order to have a better view, we also show the L^1- and L^∞-errors in Fig. 7, where we also show the results computed by the 1- and 2-Order schemes on finer meshes 800 × 800 and 1600 × 1600 to show that the 1- and 2-Order schemes achieve expected convergence rates after mesh refinement. Note that the L^1- and L^∞-errors computed by the TV splitting schemes are slightly larger than the ones computed by the CU (HLL) and HLLC schemes, while HLLC schemes are the most accurate ones among the studied schemes. However, it can be observed that the TV-splitting schemes achieve accuracy comparable to that of the HLLC schemes and are more accurate than the corresponding CU (HLL) schemes; see Examples 4, 6, and 9.

Example 8—Explosion problem

In this example, we consider the explosion problem studied in [8]. We take the following initial conditions,

$$(\rho,u,v,p)(x,y,0)=\begin{cases}(1,0,0,1), & x^2+y^2<0.16,\\ (0.125,0,0,0.1), & \text{otherwise},\end{cases}$$

prescribed in the computational domain $[-1,1]\times[-1,1]$ subject to free boundary conditions at all the four sides.

We apply the studied 1-, 2-, 3-, and 5-Order schemes and compute the numerical solutions until the final time $t=0.25$ on a uniform mesh with 50 × 50 cells. The obtained results are presented in Figs. 8 and 9. In Fig. 9, we show the slices of the densities along the diagonal $y=x$ for

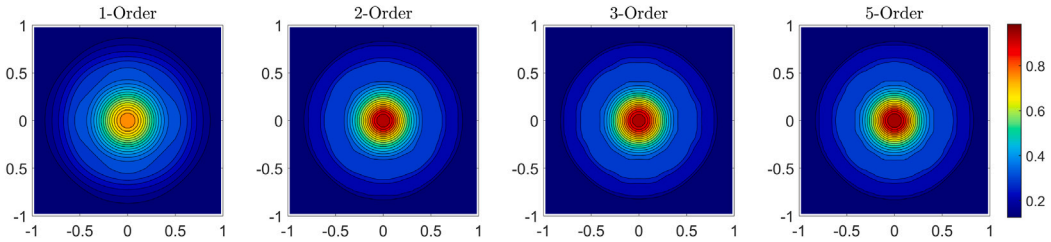

Fig. 8. Example 8: Density ρ computed by the 1-, 2-, 3-, and 5-Order schemes.

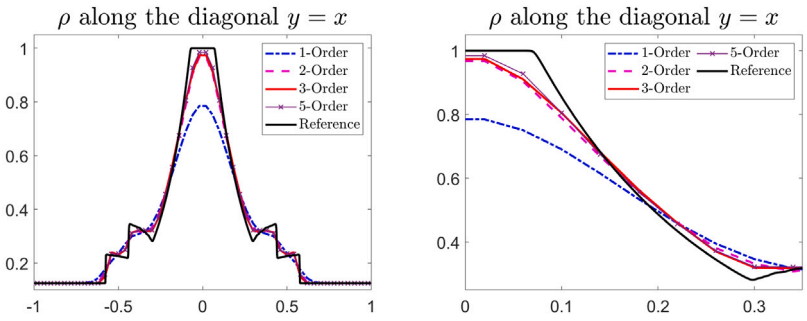

Fig. 9. Example 8: Diagonal slices of the density ρ computed by the 1-, 2-, 3-, and 5-Order schemes (left) and zoom at $x \in [0, 0.35]$.

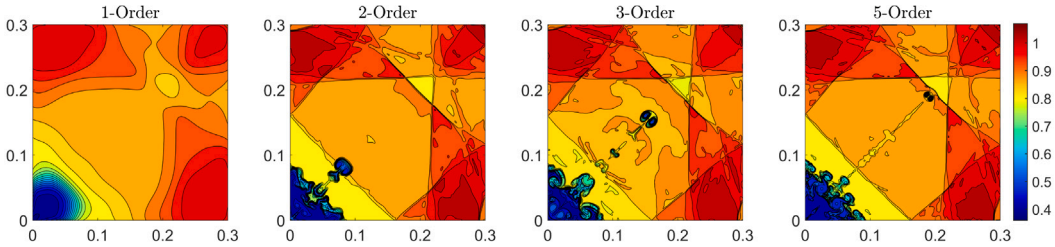

Fig. 10. Example 9: Density (ρ) computed by the 1-, 2, 3-, and 5-Order schemes.

different schemes. As one can clearly see, using higher-order numerical schemes can achieve better resolution, with a particularly significant improvement observed when transiting from the 1-Order scheme to the 2-Order one.

Example 9—Implosion problem

In this example, we consider the implosion problem taken from [61] (see also [38,58,61,62]). The initial conditions,

$$(\rho, u, v, p)(x, y, 0) = \begin{cases} (0.125, 0, 0, 0.14), & |x| + |y| < 0.15, \\ (1, 0, 0, 1), & \text{otherwise}, \end{cases}$$

are prescribed in the computational domain $[0, 0.3] \times [0, 0.3]$ with solid boundary conditions imposed at all four sides. This example was designed to test the amount of numerical diffusion present in different schemes as there is a jet forming near the origin and propagating along the diagonal $y = x$ direction, and schemes containing large numerical diffusion may not resolve the jet at all or the jet propagation velocity may be affected by the numerical diffusion.

We compute the numerical solutions until the final time $t = 2.5$ by the 1-, 2-, 3-, and 5-Order schemes on a uniform mesh of 400×400 cells. The obtained results are depicted in Fig. 10, where one can clearly observe that the jet propagates much further in the direction of $y = x$ when using higher-order schemes, clearly indicating that the high-order schemes are substantially less dissipative than the low-order ones.

In order to compare the studied TV splitting schemes with the CU, or HLL, and HLLC schemes, we compute the numerical results by the corresponding CU (HLL) and HLLC schemes and plot the corresponding results in Fig. 11. One can see that, the jets produced by the HLLC and TV splitting schemes move further than the corresponding CU (HLL) scheme. At the same time, the positions of the jets produced by the HLLC and TV splitting schemes are close, indicating that there is much less dissipation in the TV splitting schemes compared with the CU (HLL) ones.

It is instructive to check whether the studied TV splitting schemes are more efficient than the CU (HLL) and HLLC schemes. To this end, we measure the CPU time consumed during the CU (HLL) schemes and refine the mesh used by the HLLC and TV splitting scheme to the level that exactly the same CPU time is consumed to compute all three numerical solutions. The corresponding meshes are 400×400 for the 1-, 2-, 3-, and 5-Order CU (HLL) schemes, 425×425, 425×425, 405×405, and 408×408 for the HLLC schemes, and 455×455, 455×455, 420×420, and 418×418 for the TV splitting scheme. The obtained numerical results, presented in Fig. 12, indicate that the TV splitting schemes still achieve a much higher resolution than the CU (HLL) schemes. At the same time, the TV splitting schemes are slightly less dissipative than the HLLC schemes; see, e.g., the 2-Order results.

Example 10—Kelvin–Helmholtz (KH) instability

In this example, we study the KH instability taken from [63,64] (see also [58,62,65]). We take the following initial data:

$$(\rho(x, y, 0), u(x, y, 0)) = \begin{cases} (1, -0.5 + 0.5e^{(y+0.25)/L}), & y < -0.25, \\ (2, 0.5 - 0.5e^{(-y-0.25)/L}), & -0.25 < y < 0, \\ (2, 0.5 - 0.5e^{(y-0.25)/L}), & 0 < y < 0.25, \\ (1, -0.5 + 0.5e^{(0.25-y)/L}), & y > 0.25, \end{cases}$$

$$v(x, y, 0) = 0.01 \sin(4\pi x), \qquad p(x, y, 0) \equiv 1.5,$$

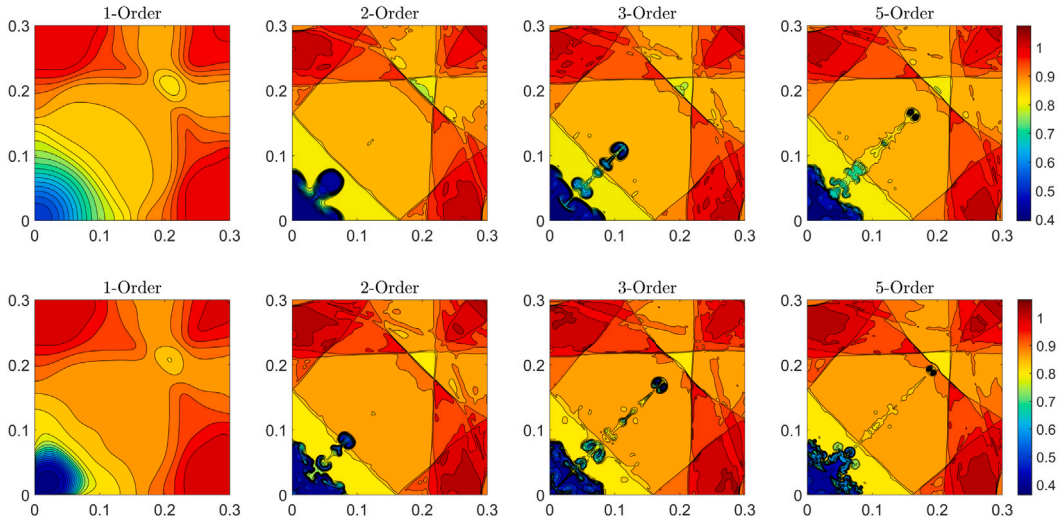

Fig. 11. Example 9: Density ρ computed by the 1-, 2-, 3-, and 5-Order CU (top row) and HLLC (bottom row) schemes.

where L is a smoothing parameter (we take $L = 0.00625$), which corresponds to a thin shear interface with a perturbed vertical velocity field v in the conducted simulations. The periodic boundary conditions are imposed on all four sides of the computational domain $[-0.5, 0.5] \times [-0.5, 0.5]$.

We compute the numerical solutions until the final time $t = 4$ by the 1-, 2-, 3-, and 5-Order schemes on a uniform mesh of 1024×1024 cells, and plot the numerical results at times $t = 1$, 2.5, and 4 in Fig. 13. One can observe that at the early time $t = 1$, the vortex streets generated by the high-order schemes are more pronounced. These structures grow exponentially over time, leading to increasingly complex turbulent mixing, particularly evident at later times $t = 2.5$ and 4, clearly indicating that high-order schemes exhibit significantly less dissipation compared to low-order ones.

Example 11—Rayleigh–Taylor (RT) instability

In the last example taken from [66] (see also [33,58,62,67]), we investigate the RT instability, which is a physical phenomenon occurring when a layer of heavier fluid is placed on top of a layer of lighter fluid. The model is governed by the 2-D Euler equations (12)–(13) with added gravitational source terms and the modified system reads as

$$\begin{aligned}
&\rho_t + (\rho u)_x + (\rho v)_y = 0,\\
&(\rho u)_t + (\rho u^2 + p)_x + (\rho uv)_y = 0,\\
&(\rho v)_t + (\rho uv)_x + (\rho v^2 + p)_y = \rho,\\
&E_t + [u(E+p)]_x + [v(E+p)]_y = \rho v.
\end{aligned}$$

We consider the following initial conditions:

$$(\rho, u, v, p)(x, y, 0) = \begin{cases} (2, 0, -0.025\, c \cos(8\pi x), 2y + 1), & y < 0.5,\\ (1, 0, -0.025\, c \cos(8\pi x), y + 1.5), & \text{otherwise}, \end{cases}$$

where $c := \sqrt{\gamma p/\rho}$ is the speed of sound, prescribed in the computational domain $[0, 0.25] \times [0, 1]$ with the solid wall boundary conditions imposed at $x = 0$ and $x = 0.25$, and the following Dirichlet boundary conditions imposed at the top and bottom boundaries:

$$(\rho, u, v, p)|_{y=1} = (1, 0, 0, 2.5), \qquad (\rho, u, v, p)|_{y=0} = (2, 0, 0, 1).$$

We compute the numerical solutions until the final time $t = 2.95$ by 1-, 2-, 3-, and 5-Order schemes on the uniform mesh of 256×1024 cells and present the numerical results at times $t = 1.95$ and 2.95 in Fig. 14. One can see that, there are pronounced differences between the solutions computed by different schemes and the structures captured by the high-order schemes are much more complicated, which again demonstrates that the high-order schemes can capture more details and are less dissipative than the low-order ones.

CRediT authorship contribution statement

Shaoshuai Chu: Writing – review & editing, Writing – original draft, Visualization, Methodology, Investigation, Formal analysis, Data curation. **Michael Herty:** Writing – review & editing, Writing – original draft, Supervision, Project administration, Methodology, Funding acquisition, Formal analysis. **Eleuterio F. Toro:** Writing – review & editing, Supervision, Investigation, Funding acquisition, Formal analysis.

Declaration of competing interest

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

Acknowledgments

The work of S. Chu was supported in part by the DFG (German Research Foundation) through HE5386/19-3, 27-1. The work of M. Herty was funded by the DFG–SPP 2183: Eigenschaftsgeregelte Umformprozesse, Germany with the Project(s) HE5386/19-2,19-3 Entwicklung eines flexiblen isothermen Reckschmiedeprozesses für die eigenschaftsgeregelte Herstellung von Turbinenschaufeln aus Hochtemperaturwerkstoffen, Germany (424334423) and by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation), Germany–SPP 2410 Hyperbolic Balance Laws in Fluid Mechanics: Complexity, Scales, Randomness (CoScaRa) within the Project(s) HE5386/26-1 (Numerische Verfahren für gekoppelte Mehrskalenprobleme,525842915) and (Zufällige kompressible Euler Gleichungen: Numerik und ihre Analysis, 525853336) HE5386/27-1. E. F. Toro acknowledges partial support from project number P130625149, entitled Development of a Numerical Multiphase Flow Tool for Applications to Petroleum Industry, funded by Repsol S.A. Spain.

Appendix A. One-dimensional local characteristic decomposition based third-order WENO-type interpolant

In this appendix, we briefly describe the 1-D local characteristic decomposition (LCD) based third-order WENO-Type interpolant.

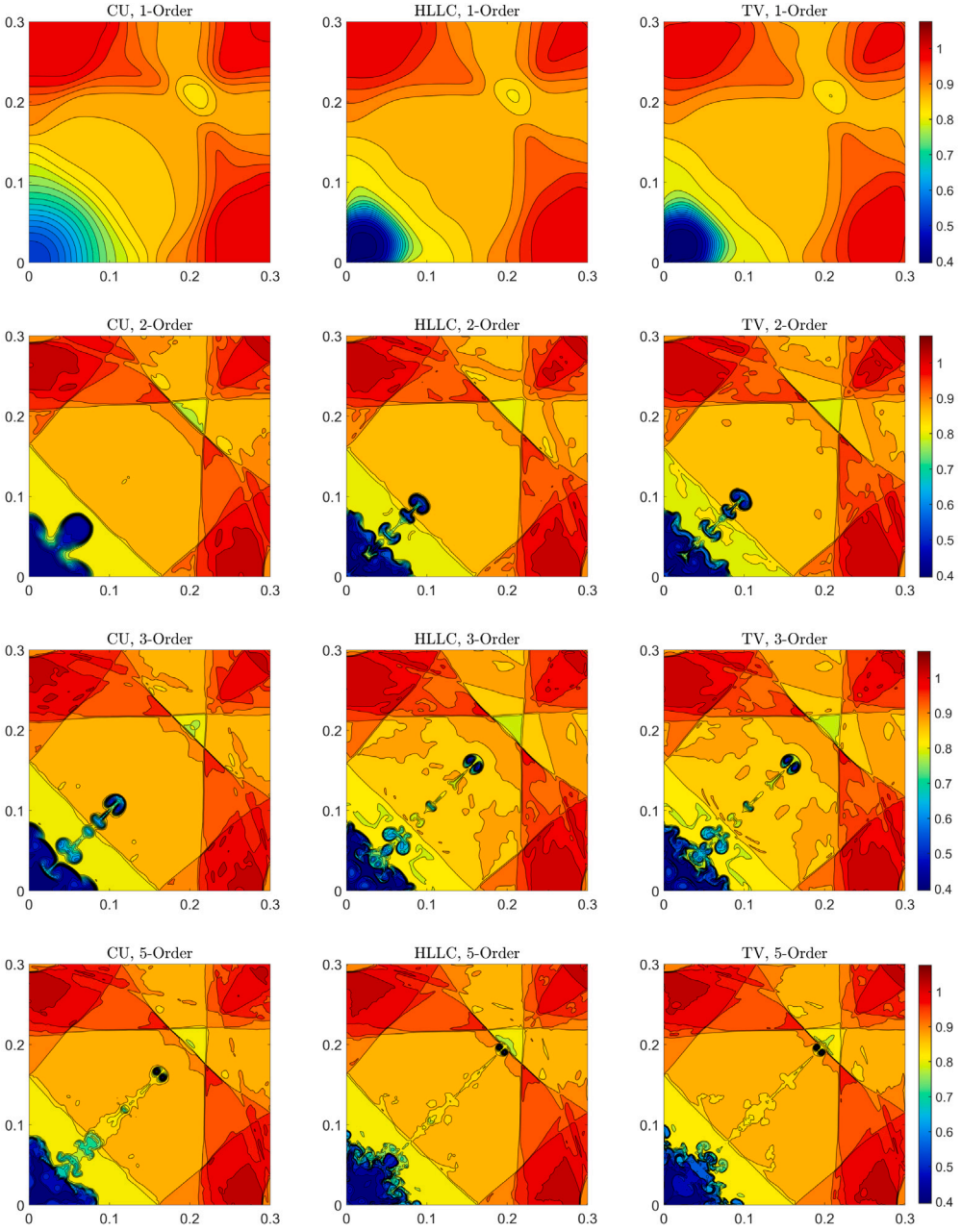

Fig. 12. Example 9: Density ρ computed by the CU (left column), HLLC (middle column), and TV splitting (right column) schemes.

Supposing the point values $\boldsymbol{U}_j$ are given at uniform grid points $x = x_j$, we now demonstrate how to compute the interpolated left-sided value of $\boldsymbol{U}$ at $x = x_{j+\frac{1}{2}}$, denoted as $\boldsymbol{U}^-_{j+\frac{1}{2}}$. The corresponding right-sided value $\boldsymbol{U}^+_{j+\frac{1}{2}}$, can be computed in a mirror-symmetric manner.

The value of $(U^{(i)}_{j+\frac{1}{2}})^-$ is computed using a weighted average of the two linear interpolants $\boldsymbol{\mathcal{P}}_0(x)$ and $\boldsymbol{\mathcal{P}}_1(x)$, which are obtained using the stencils $[x_{j-1}, x_j]$ and $[x_j, x_{j+1}]$, respectively:

$$(U^{(i)}_{j+\frac{1}{2}})^- = \omega^{(i)}_0 \mathcal{P}^{(i)}_0(x_{j+\frac{1}{2}}) + \omega^{(i)}_1 \mathcal{P}^{(i)}_1(x_{j+\frac{1}{2}}), \tag{A.1}$$

where

$$\mathcal{P}^{(i)}_0(x_{j+\frac{1}{2}}) = -\frac{1}{2}U^{(i)}_{j-1} + \frac{3}{2}U^{(i)}_j, \quad \text{and} \quad \mathcal{P}^{(i)}_1(x_{j+\frac{1}{2}}) = \frac{1}{2}U^{(i)}_j + \frac{1}{2}U^{(i)}_{j+1}. \tag{A.2}$$

By performing a straightforward Taylor expansion, one can demonstrate that (A.1)–(A.2) achieve third-order accuracy if the weights ω_k in (A.1) are chosen as:

$$\omega^{(i)}_k = \frac{\alpha^{(i)}_k}{\alpha^{(i)}_0 + \alpha^{(i)}_1}, \quad \alpha^{(i)}_k = d_k \left[1 + \frac{\tau^{(i)}_3}{\beta^{(i)}_k + \varepsilon}\right], \tag{A.3}$$

where $d_0 = \frac{1}{4}$ and $d_1 = \frac{3}{4}$,

$$\tau^{(i)}_3 = \left|\beta^{(i)}_2 - \beta^{(i)}_3\right|^p,$$

and the smoothness indicators $\beta^{(i)}_k$ are given by (see e.g., [44])

$$\beta^{(i)}_0 = (U^{(i)}_{j-1} - U^{(i)}_j)^2, \quad \beta^{(i)}_1 = (U^{(i)}_j - U^{(i)}_{j+1})^2, \tag{A.4}$$

and

$$\begin{aligned} \beta_2 &= \frac{13}{12}(U^{(i)}_{j-1} - 2U^{(i)}_j + U^{(i)}_{j+1})^2 + \frac{1}{4}(U^{(i)}_{j+1} - U^{(i)}_{j-1})^2, \\ \beta_3 &= \frac{13}{12}(U^{(i)}_j - 2U^{(i)}_{j+1} + U^{(i)}_{j+2})^2 + \frac{1}{4}(3U^{(i)}_j - 4U^{(i)}_{j+1} + U^{(i)}_{j+2})^2. \end{aligned} \tag{A.5}$$

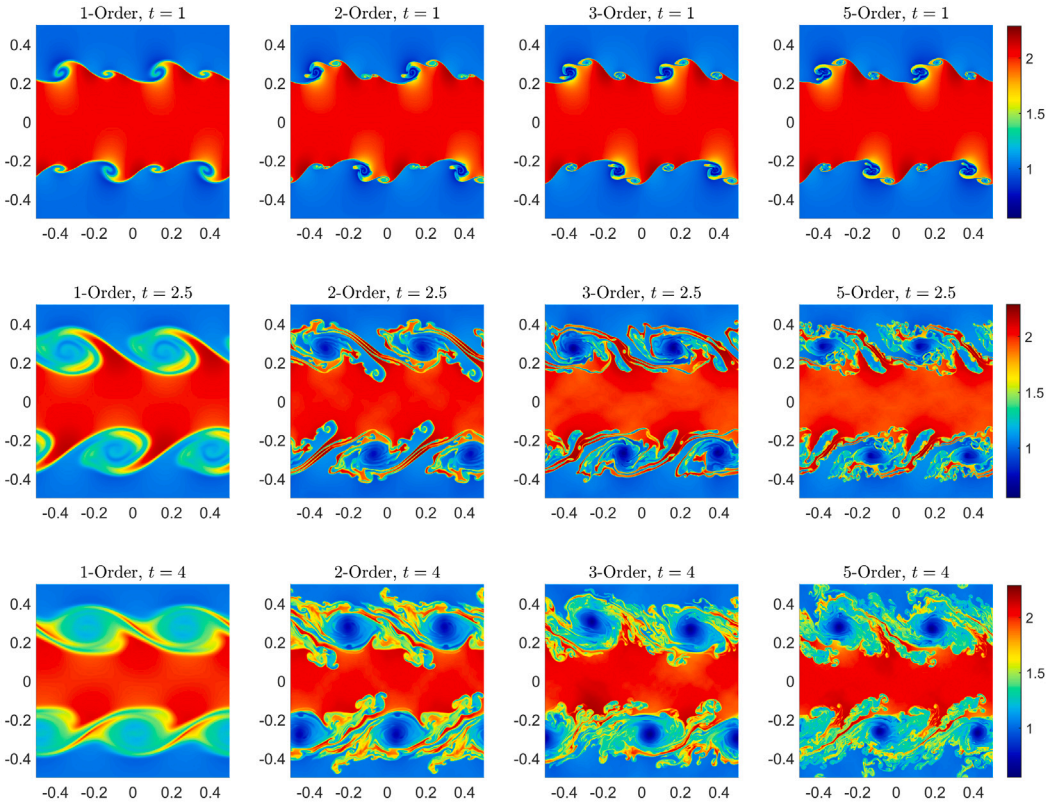

Fig. 13. Example 10: Time snapshots of density ρ computed by the 1- (first column), 2- (second column), 3- (third column), and 5- (fourth column) Order schemes at $t = 1$ (top row), $t = 2.5$ (middle row), and $t = 4$ (bottom row).

Finally, in all of the numerical examples reported in Section 4, we have chosen $p = 1.4$ and $\varepsilon = 10^{-12}$.

Although the third-order interpolant (A.1)–(A.5) is essentially non-oscillatory, it is well-known that applying it to the conservative variables $\boldsymbol{U}$ in a componentwise manner can result in spurious oscillations in the computed solution. To address this, we adopt the reconstruction procedure within the LCD framework.

To this end, we first introduce the matrix $\widehat{A}_{j+\frac{1}{2}} := A(\widehat{\boldsymbol{U}}_{j+\frac{1}{2}})$, where $A = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{U}}$ and $\widehat{\boldsymbol{U}}_{j+\frac{1}{2}}$ is either a simple average $(\boldsymbol{U}_j + \boldsymbol{U}_{j+1})/2$ or another type of average of the $\boldsymbol{U}_j$ and $\boldsymbol{U}_{j+1}$ states (in the numerical examples reported in Section 4, we have used the simple average). We then compute the matrices $R_{j+\frac{1}{2}}$ and $R^{-1}_{j+\frac{1}{2}}$ such that $R^{-1}_{j+\frac{1}{2}} \widehat{A}_{j+\frac{1}{2}} R_{j+\frac{1}{2}}$ is a diagonal matrix and introduce the local characteristic variables in the neighborhood of $x = x_{j+\frac{1}{2}}$:

$$\boldsymbol{\Gamma}_m = R^{-1}_{j+\frac{1}{2}} \boldsymbol{U}_m, \quad m = j-1, \dots, j+2.$$

Equipped with the values $\boldsymbol{\Gamma}_{j-1}$, $\boldsymbol{\Gamma}_j$, $\boldsymbol{\Gamma}_{j+1}$, and $\boldsymbol{\Gamma}_{j+2}$, we apply the interpolation procedure (A.1)–(A.5) to each of the components $\Gamma^{(i)}$, $i = 1, \dots, d$ of $\boldsymbol{\Gamma}$ to obtain $\boldsymbol{\Gamma}^-_{j+\frac{1}{2}}$ (the values of $\boldsymbol{\Gamma}^+_{j+\frac{1}{2}}$ are computed, as mentioned above, in the mirror-symmetric way). Finally, the corresponding point values of $\boldsymbol{U}$ are given by

$$\boldsymbol{U}^{\pm}_{j+\frac{1}{2}} = R_{j+\frac{1}{2}} \boldsymbol{\Gamma}^{\pm}_{j+\frac{1}{2}}.$$

Remark A.1. A detailed explanation of how the average matrix $\widehat{A}_{j+\frac{1}{2}}$ and the corresponding matrices $R_{j+\frac{1}{2}}$ and $R^{-1}_{j+\frac{1}{2}}$ of the 1-D Euler equation of gas dynamics can be found in, e.g., [58].

Appendix B. 1-D local characteristic decomposition based fifth-order WENO-Z interpolant

In this appendix, we briefly describe the 1-D LCD based fifth-order WENO-Z interpolant.

Given the point values $\boldsymbol{U}_j$ at uniform grid points $x = x_j$, the value $(U^{(i)})^-_{j+\frac{1}{2}}$ is computed using a weighted average of the three parabolic interpolants $\mathcal{P}_0(x)$, $\mathcal{P}_1(x)$ and $\mathcal{P}_2(x)$ obtained using the stencils $[x_{j-2}, x_{j-1}, x_j]$, $[x_{j-1}, x_j, x_{j+1}]$, and $[x_j, x_{j+1}, x_{j+2}]$, respectively:

$$(U^{(i)})^-_{j+\frac{1}{2}} = \sum_{k=0}^{2} \omega_k^{(i)} \mathcal{P}_k^{(i)}(x_{j+\frac{1}{2}}), \tag{B.1}$$

where

$$\begin{aligned}
\mathcal{P}_0^{(i)}(x_{j+\frac{1}{2}}) &= \frac{3}{8}U^{(i)}_{j-2} - \frac{5}{4}U^{(i)}_{j-1} + \frac{15}{8}U^{(i)}_j, \\
\mathcal{P}_1^{(i)}(x_{j+\frac{1}{2}}) &= -\frac{1}{8}U^{(i)}_{j-1} + \frac{3}{4}U^{(i)}_j + \frac{3}{8}U^{(i)}_{j+1}, \\
\mathcal{P}_2^{(i)}(x_{j+\frac{1}{2}}) &= \frac{3}{8}U^{(i)}_j + \frac{3}{4}U^{(i)}_{j+1} - \frac{1}{8}U^{(i)}_{j+2}.
\end{aligned} \tag{B.2}$$

To ensure (B.1)–(B.2) is fifth-order accurate and nonoscillatory, one can take the weights ω_k in (B.1) to be

$$\omega_k^{(i)} := \frac{\alpha_k^{(i)}}{\alpha_0^{(i)} + \alpha_1^{(i)} + \alpha_2^{(i)}}, \quad \alpha_k^{(i)} = d_k \left[1 + \left(\frac{\tau_5^{(i)}}{\beta_k^{(i)} + \varepsilon} \right)^p \right], \quad \tau_5 = |\beta_2^{(i)} - \beta_0^{(i)}|, \tag{B.3}$$

where $d_0 = \frac{1}{16}$, $d_1 = \frac{5}{8}$, and $d_2 = \frac{5}{16}$, and the smoothness indicators $\beta_k^{(i)}$ are given by

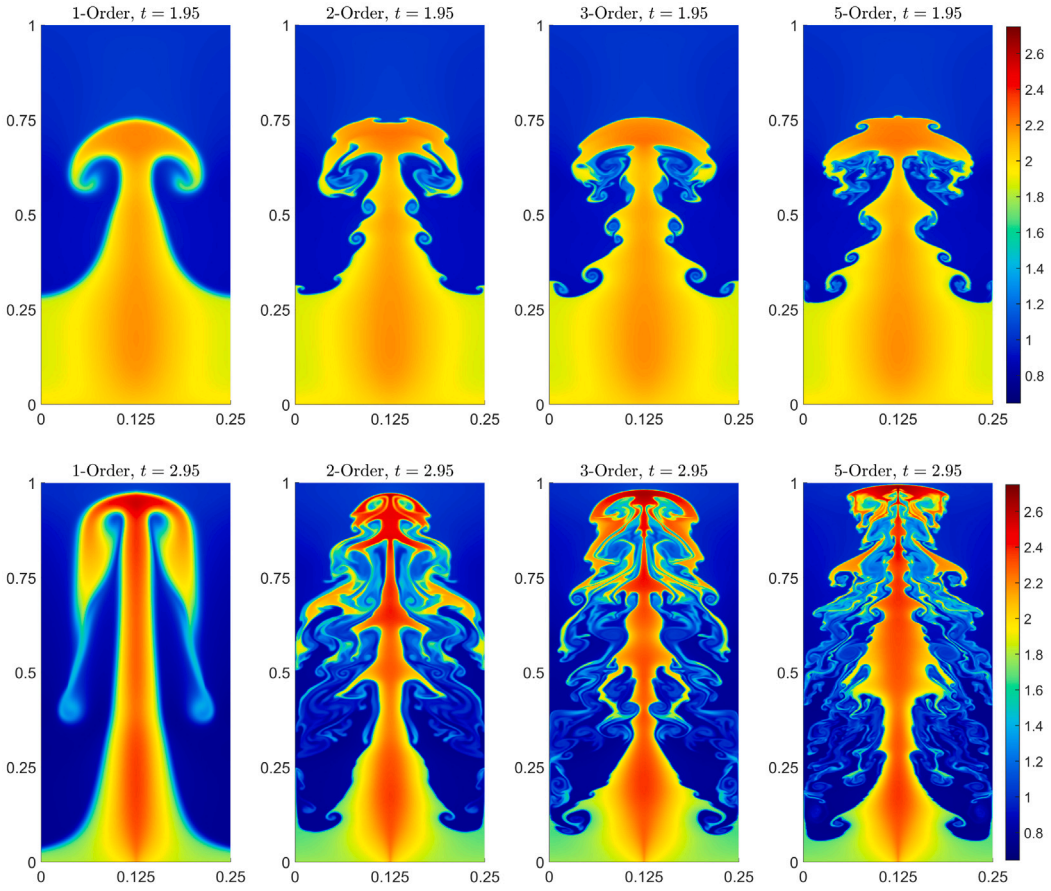

Fig. 14. Example 11: Density ρ computed by the 1- (first column), 2- (second column), 3- (third column), and 5- (fourth column) Order schemes at $t = 1.95$ (top row) and $t = 2.95$ (bottom row).

$$\begin{aligned}
\beta_0^{(i)} &= \frac{13}{12}\left(U_{j-2}^{(i)} - 2U_{j-1}^{(i)} + U_j^{(i)}\right)^2 + \frac{1}{4}\left(U_{j-2}^{(i)} - 4U_{j-1}^{(i)} + 3U_j^{(i)}\right)^2,\\
\beta_1^{(i)} &= \frac{13}{12}\left(U_{j-1}^{(i)} - 2U_j^{(i)} + U_{j+1}^{(i)}\right)^2 + \frac{1}{4}\left(U_{j-1}^{(i)} - U_{j+1}^{(i)}\right)^2,\\
\beta_2^{(i)} &= \frac{13}{12}\left(U_j^{(i)} - 2U_{j+1}^{(i)} + U_{j+2}^{(i)}\right)^2 + \frac{1}{4}\left(3U_j^{(i)} - 4U_{j+1}^{(i)} + U_{j+2}^{(i)}\right)^2.
\end{aligned} \tag{B.4}$$

We have used $p = 2$ and $\epsilon = 10^{-12}$ in all of the numerical examples reported in this paper. The corresponding right-sided value, $(U^{(i)}_{j+\frac{1}{2}})^-$, can also be derived using a mirror-symmetric approach.

As in Appendix A, to ensure the nonoscillatory nature of the reconstruction (B.1)–(B.4), we also need to adopt the reconstruction procedure within the LCD framework. To this end, we first introduce the local characteristic variables in the neighborhood of $x = x_{j+\frac{1}{2}}$:

$$\boldsymbol{\Gamma}_m = R^{-1}_{j+\frac{1}{2}} \boldsymbol{U}_m, \quad m = j-2, \ldots, j+3.$$

Equipped with the values $\boldsymbol{\Gamma}_{j-2}$, $\boldsymbol{\Gamma}_{j-1}$, $\boldsymbol{\Gamma}_j$, $\boldsymbol{\Gamma}_{j+1}$, $\boldsymbol{\Gamma}_{j+2}$, and $\boldsymbol{\Gamma}_{j+3}$, we then apply the interpolation procedure (B.1)–(B.4) to each of the components $\Gamma^{(i)}$, $i = 1, \ldots, d$ of $\boldsymbol{\Gamma}$ to obtain $\boldsymbol{\Gamma}^-_{j+\frac{1}{2}}$ (the values of $\boldsymbol{\Gamma}^+_{j+\frac{1}{2}}$ are computed in the mirror-symmetric way). Finally, the corresponding one-sided point values of $\boldsymbol{U}$ are given by

$$\boldsymbol{U}^{\pm}_{j+\frac{1}{2}} = R_{j+\frac{1}{2}} \boldsymbol{\Gamma}^{\pm}_{j+\frac{1}{2}}.$$

Data availability

Data will be made available on request.

References

[1] Friedrichs KO. Symmetric hyperbolic linear differential equations. Comm Pure Appl Math 1954;7:345–92.
[2] Lax PD. Weak solutions of nonlinear hyperbolic equations and their numerical computation. Comm Pure Appl Math 1954;7:159–93.
[3] Godunov SK. A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics. Mat Sb (NS) 1959;47:271–306.
[4] Hesthaven JS. Numerical methods for conservation laws: From analysis to algorithms. In: Comput Sci Eng 18. Philadelphia: SIAM; 2018.
[5] LeVeque RJ. Finite volume methods for hyperbolic problems. In: Cambridge texts in appl math. Cambridge, UK: Cambridge University Press; 2002.
[6] Ketcheson DI, LeVeque RJ, del Razo MJ. Riemann problems and jupyter solutions. In: Fundamentals of algorithms. Philadelphia: SIAM; 2020.
[7] Shu C-W. Essentially non-oscillatory and weighted essentially non-oscillatory schemes. Acta Numer 2020;5:701–62.
[8] Toro EF. Riemann solvers and numerical methods for fluid dynamics: A practical introduction. third ed.. Berlin, Heidelberg: Springer-Verlag; 2009.
[9] Ben-Artzi M, Falcovitz J. Generalized Riemann problems in computational fluid dynamics. In: Cambridge monographs on applied and computational mathematics, Cambridge, UK: Cambridge University Press; 2003.
[10] Anderson WK, Thomas JL, Leer Bvan. Comparison of finite volume flux vector splittings for the Euler equations. AIAA J 1986;24:1453–60.
[11] Anderson WK, Thomas JL, Rumsey L. Extension and application of flux–vector splitting to calculations on dynamic meshes. AIAA J 1989;27:673–4.
[12] Steger JL, Warming RF. Flux vector splitting of the inviscid gas dynamic equations with applications to finite difference methods. J Comput Phys 1981;40:263–93.
[13] van Leer B. Flux–vector splitting for the Euler equations. Technical report ICASE 82-30 1982.
[14] van Leer B. Flux-vector splitting for the Euler equations. In: Eighth international conference on numerical methods in fluid dynamics. Berlin, Heidelberg: Springer; 1982.
[15] Liou MS. A new flux splitting scheme. J Comput Phys 1993;107:23–39.
[16] Liou MS. A sequel to AUSM: AUSM+. J Comput Phys 1996;129:364–82.

[17] Liou MS. Recent progress and applications of AUSM+. In: Lecture notes in physics. Sixteenth international conference on numerical methods in fluid dynamics, Berlin, Verlag: Springer; 1998.
[18] Liou MS. A Sequel to AUSM, Part II: AUSM+-up for all speeds. J Comput Phys 2006;214:137–70.
[19] Dumbser M, Toro EF. On universal osher-type schemes for general nonlinear hyperbolic conservation laws. Commun Comput Phys 2011;10:635–71.
[20] Luo H, Luo LQ, Nourgaliev R. A reconstructed discontinuous Galerkin method for the Euler equations on arbitrary grids. Commun Comput Phys 2012;12:1495–519.
[21] Gressier J, Villedieu P, Moschetta JM. Positivity of flux vector splitting schemes. J Comput Phys 1999;155:199–220.
[22] Kitamura K, Shima E, Fujimoto K, Wang ZJ. Performance of low-dissipation Euler fluxes and preconditioned LU-SGS at low speeds. Commun Comput Phys 2011;10:90–119.
[23] Zha G-C, Bilgen E. Numerical solution of Euler equations by a new flux vector splitting scheme. Internat J Numer Methods Fluids 1993;17:115–44.
[24] Toro EF, Vázquez-Cendón ME. Flux splitting schemes for the Euler equations. Comput & Fluids 2012;70:1–12.
[25] Toro EF, Castro CE, Lee BJ. A novel numerical flux for the 3D Euler equations with general equation of state. J Comput Phys 2015;303:80–94.
[26] Balsara DS, Montecinos GI, Toro EF. Exploring various flux vector splittings for the magnetohydrodynamic system. J Comput Phys 2016;311:1–21.
[27] Dumbser M, Balsara DS, Tavelli M, Fambri F. A divergence-free semi-implicit finite volume scheme for ideal, viscous, and resistive magnetohydrodynamics. Internat J Numer Methods Fluids 2019;89:16–42.
[28] Toro EF, Castro CE, Vanzo D, Siviglia A. A flux-vector splitting scheme for the shallow water equations extended to high-order on unstructured meshes. Internat J Numer Methods Fluids 2022;94:1679–705.
[29] Tokareva S, Toro EF. A flux splitting method for the Baer-Nunziato equations of compressible two-phase flow. In: Finite volumes for complex applications VIII—hyperbolic, elliptic and parabolic problems. Cham: Springer; 2017.
[30] Liu H. A numerical study of the performance of alternative weighted ENO methods based on various numerical fluxes for conservation law. Appl Math Comput 2017;296:182–97.
[31] Wang B-S, Li P, Gao Z, Don WS. An improved fifth order alternative WENO-Z finite difference scheme for hyperbolic conservation laws. J Comput Phys 2018;374:469–77.
[32] Wang B-S, Garg NK. Third-order numerical scheme for Euler equations of gas dynamics using Jordan canonical based splitting flux. Comput & Fluids 2024;281:106370.
[33] Wang B-S, Don WS, Garg NK, Kurganov A. Fifth-order A-WENO finite-difference schemes based on a new adaptive diffusion central numerical flux. SIAM J Sci Comput 2020;42:A3932–56.
[34] Jiang Y, Shu C-W, Zhang MP. An alternative formulation of finite difference weighted ENO schemes with Lax-Wendroff time discretization for conservation laws. SIAM J Sci Comput 2013;35:A1137–60.
[35] Jiang G-S, Shu C-W. Efficient implementation of weighted ENO schemes. J Comput Phys 1996;126:202–28.
[36] Kurganov A, Tadmor E. Solution of two-dimensional Riemann problems for gas dynamics without Riemann problem solvers. Numer Methods Partial Differential Equations 2002;18:584–608.
[37] Kurganov A, Noelle P, Petrova G. Semi-discrete central-upwind schemes for hyperbolic conservation laws and Hamilton-Jacobi equations. SIAM J Sci Comput 2001;23:707–40.
[38] Kurganov A, Lin C-T. On the reduction of numerical dissipation in central-upwind schemes. Commun Comput Phys 2007;2:141–63.
[39] Harten A, Lax PD, Leer Bvan. On upstream differencing and godunov-type schemes for hyperbolic conservation laws. SIAM Rev 1983;25:35–61.
[40] Toro EF, Spruce M, Speares W. Restoration of the contact surface in the HLL-Riemann solver. Shock Waves 1994;4:25–34.
[41] Lie K-A, Noelle S. On the artificial compression method for second-order nonoscillatory central difference schemes for systems of conservation laws. SIAM J Sci Comput 2003;24:1157–74.
[42] Nessyahu H, Tadmor E. Nonoscillatory central differencing for hyperbolic conservation laws. J Comput Phys 1990;87:408–63.
[43] Sweby PK. High resolution schemes using flux limiters for hyperbolic conservation laws. SIAM J Numer Anal 1984;21:995–1011.
[44] Chu S, Fu Q, Herty M, Kuaganov A. Novel and efficient third-order WENO schemes (in preparation).
[45] Chu S, Kurganov A, Xin R. New more efficient A-WENO schemes. J Sci Comput 2025;104:53.
[46] Chertock A, Chu S, Kurganov A. Adaptive high-order A-WENO schemes based on a new local smoothness indicator. E Asian J Appl Math 2023;13:576–609.
[47] Chu S, Kurganov A, Na M. Fifth-order A-WENO schemes based on the path-conservative central-upwind method. J Comput Phys 2022;469:111508.
[48] Don WS, Li DM, Gao Z, Wang B-S. A characteristic-wise alternative WENO-Z finite difference scheme for solving the compressible multicomponent non-reactive flows in the overestimated quasi-conservative form. J Sci Comput 2020;82:27.
[49] Gao Z, Fang L-L, Wang B-S, Wang Y, Don WS. Seventh and ninth orders alternative WENO finite difference schemes for hyperbolic conservation laws. Comput & Fluids 2020;202:104519.
[50] Gottlieb S, Ketcheson D, Shu C-W. Strong stability preserving Runge-Kutta and multistep time discretizations. Hackensack, NJ: World Scientific Publishing Co Pte Ltd; 2011.
[51] Gottlieb S, Shu C-W, Tadmor E. Strong stability-preserving high-order time discretization methods. SIAM Rev 2001;43:89–112.
[52] Shu C-W, Osher S. Efficient implementation of essentially nonoscillatory shock-capturing schemes. II. J Comput Phys 1989;83:32–78.
[53] Titarev VA, Toro EF. WENO schemes based on upwind and centred TVD fluxes. Comput & Fluids 2005;34:705–20.
[54] Shu C-W, Osher S. Efficient implementation of essentially non-oscillatory shock-capturing schemes. J Comput Phys 1988;77:439–71.
[55] Toro EF, Titarev VA. TVD fluxes for the high-order ADER schemes. J Sci Comput 2005;24:285–309.
[56] Woodward P, Colella P. The numerical solution of two-dimensional fluid flow with strong shocks. J Comput Phys 1988;54:115–73.
[57] Kuganov A, Prugger M, Wu T. Second-order fully discrete central-upwind scheme for two-dimensional hyperbolic systems of conservation laws. SIAM J Sci Comput 2017;39:A947–65.
[58] Chertock A, Chu S, Herty M, Kurganov A, Lukáčová-Medviďová M. Local characteristic decomposition based central-upwind scheme. J Comput Phys 2023;473:111718.
[59] Titarev VA, Toro EF. ADER schemes for three-dimensional non-linear hyperbolic systems. J Comput Phys 2005;204:715–36.
[60] Hu C, Shu C-W. Weighted essentially non-oscillatory schemes on triangular meshes. J Comput Phys 1999;150:97–127.
[61] Liska R, Wendrof B. Comparison of several diference schemes on 1D and 2D test problems for the Euler equations. SIAM J Sci Comput 2003;25:995–1017.
[62] Chu S, Kurganov A, Xin R. A fifth-order A-WENO scheme based on the low-dissipation central-upwind fluxes. In: Hyperbolic problems: Theory, numerics, applications, vol. II, Cham: Springer; 2024.
[63] Fjordholm US, Mishra S, Tadmor E. On the computation of measure-valued solutions. Acta Numer 2016;25:567–679.
[64] Panuelos J, Wadsley J, Kevlahan N. Low shear difusion central schemes for particle methods. J Comput Phys 2020;414:109454.
[65] Feireisl E, Lukáčová-Medviďová M, She B, Wang Y. Computing oscillatory solutions of the Euler system via $\mathcal{K}$-convergence. Math Models Methods Appl Sci 2021;31:537–76.
[66] Shi J, Zhang Y-T, Shu C-W. Resolution of high order WENO schemes for complicated flow structures. J Comput Phys 2003;186:690–6.
[67] Chu S, Kurganov A, Menshov I. New adaptive low-dissipation central-upwind schemes. Appl Numer Math 2025;209:155–70.