



Entropy-stable schemes for relativistic hydrodynamics equations

Deepak Bhoriya and Harish Kumar

Abstract. In this article, we propose high-order finite difference schemes for the equations of relativistic hydrodynamics, which are entropy stable. The crucial components of these schemes are a computationally efficient entropy conservative flux and suitable high-order entropy dissipative operators. We first design a higher-order entropy conservative flux. For the construction of appropriate entropy dissipative operators, we derive entropy scaled right eigenvectors. This is then used with ENO-based sign-preserving reconstruction of scaled entropy variables, which results in higher-order entropy-stable schemes. Several numerical results are presented up to fourth order to demonstrate entropy stability and performance of these schemes.

Mathematics Subject Classification. 65M08, 65M12.

Keywords. Relativistic hydrodynamics, Symmetrization, Entropy stability, Finite difference scheme.

Contents

1. Introduction
2. Equations of relativistic hydrodynamics
3. Entropy-stable numerical schemes
 - 3.1. Entropy conservative schemes
 - 3.1.1. Entropy conservative flux
 - 3.1.2. Higher-order entropy conservative flux
 - 3.2. Entropy-stable schemes
 - 3.3. High-order diffusion operators
 - 3.4. Semi-discrete entropy stability
 - 3.5. Time discretization
4. Numerical results
 - 4.1. One-dimensional test cases
 - 4.1.1. Accuracy test
 - 4.1.2. Isentropic smooth flows
 - 4.1.3. Riemann problem 1
 - 4.1.4. Riemann problem 2
 - 4.1.5. Riemann problem 3
 - 4.1.6. Density perturbation test case
 - 4.1.7. Blast waves test case
 - 4.2. Two-dimensional test cases
 - 4.2.1. Two-dimensional Riemann problem 1
 - 4.2.2. Two-dimensional Riemann problem 2
 - 4.2.3. Two-dimensional Riemann problem 3
 - 4.2.4. Two-dimensional Riemann problem 4
 - 4.2.5. Two-dimensional Riemann problem 5

5. Conclusion

Acknowledgements

Appendix A. Computation of primitive variables

Appendix B. Entropy scaled right eigenvectors

References

1. Introduction

The equations of relativistic hydrodynamics are used to model astrophysical flows when the fluid is moving with speed comparable to the speed of light, or the flow is under the influence of large gravitational potentials such that relativistic effects cannot be ignored. Some examples are gamma-ray burst, relativistic jets from Galactic sources, core-collapse supernovae and extragalactic jets from active Galactic nuclei (see [5, 6, 20, 27, 39]).

In this article, we consider the equations of special relativistic hydrodynamics (RHD henceforth) with the ideal equation of state. The system of equations is hyperbolic conservation laws. So, the solutions of the system can exhibit discontinuities, even for the smooth initial data (see [15]). Therefore, we consider weak solutions of the systems which are characterized by the Rankine–Hugoniot jump condition across the discontinuities. However, the weak solutions of the system can be non-unique; hence, an additional criterion in the form of entropy inequality is imposed to choose the physically appropriate solution. However, more recently, it has been shown that even the entropy-stable solutions are not unique (see [9]). Still, entropy stability is one of the few nonlinear stability estimates available for the solutions.

Due to nonlinearity in the flux, the analytic study of the equations is difficult. Hence, we rely on computational methods for various applications. The computational methods for hyperbolic conservation laws are often based on the finite volume (or finite difference) methods. These methods are based on evolving the cell averages of the state variables by approximating the numerical fluxes across the cell interfaces. Higher order is achieved by reconstructing the solution at the cell edges, using TVD-, ENO- and WENO-based approximations (see [13, 17, 19, 32]). To prove the entropy stability of these schemes is, however, highly non-trivial, especially for the higher-order schemes.

For the RHD equations, one of the first attempts to solve the system computationally was made by [35]. Using an artificial viscosity technique to capture the shock solution, the author designed a Eulerian explicit finite difference scheme. However, the approach did not capture the flows accurately in the cases where the numerical value of the Lorentz factor was larger than 2 (see [7]). Another approach based on the van Leer schemes is presented in [11]. Schemes based on the piece-wise parabolic reconstruction methods are designed in [2, 22, 26]. In [10], authors designed a numerical scheme based on the TVD reconstruction. In [25], authors have designed an HLLC solver for the equations. More recently, in [36], authors have used the WENO reconstruction to obtain a high-order accurate physical-constraints-preserving finite difference scheme for RHD. Also, in [37], a physical-constraints-preserving DG scheme is designed. A comprehensive overview of the numerical schemes for RHD equations is presented in [14].

In this article, we aim to design higher-order entropy-stable numerical schemes for the RHD equations. We proceed as follows:

- First, we present entropy framework for the RHD equations at the continuous level.
- Following the work of [8], we then construct a second-order entropy conservative numerical flux. This can then be made arbitrary higher even-order accurate by following [21].
- We then design an appropriate higher-order entropy diffusion operator by following [12]. This is based on the higher-order sign-preserving reconstruction of the scaled entropy variables. The scaled entropy variables are obtained by deriving the expressions for the entropy scaled right eigenvectors (see Appendix B).

- Combining entropy conservative flux with the entropy diffusion operator, we get entropy-stable numerical schemes for the RHD equations.

The rest of the paper is organized as follows. In Sect. 2, we introduce the relativistic hydrodynamics equations and entropy framework. In the next section (Sect. 3), we present entropy-stable numerical schemes for the RHD equations. This includes the derivation of the entropy conservative numerical flux and the construction of the entropy diffusion operator. In Sect. 4, we present numerical experiments in one and two dimensions.

2. Equations of relativistic hydrodynamics

For an ideal fluid, the special relativistic hydrodynamics equations, in the laboratory frame of reference, can be written as a system of conservation laws for mass density, momentum density and the total energy density in the following form (see [3, 20, 33]):

$$\begin{aligned} \frac{\partial D}{\partial t} + \nabla \cdot (D\mathbf{u}) &= 0, \\ \frac{\partial \mathbf{m}}{\partial t} + \nabla \cdot (\mathbf{m}\mathbf{u} + p\mathbf{I}) &= 0, \\ \frac{\partial E}{\partial t} + \nabla \cdot \mathbf{m} &= 0, \end{aligned} \quad (1)$$

where the conserved quantities D , \mathbf{m} and E are the fluid mass density, momentum density and the total energy density, respectively. The primitive quantities ρ , \mathbf{u} and p are the proper mass density, fluid velocity vector and isotropic gas pressure, respectively. The vector of conservative variables, $\mathbf{U} = (D, \mathbf{m}, E)$, and primitive variables, $\mathbf{w} = (\rho, \mathbf{u}, p)$, are connected via relations

$$D = \Gamma\rho, \quad \mathbf{m} = \rho h\Gamma^2\mathbf{u}, \quad E = \rho h\Gamma^2 - p.$$

Here, h is the special enthalpy and Γ is the *Lorentz factor*, given by,

$$\Gamma = \frac{1}{\sqrt{1 - \mathbf{u}^2}}, \quad \text{with } |\mathbf{u}| < 1. \quad (2)$$

A key difficulty for the numerical schemes is the need to convert conservative variables to primitive variables and vice versa. In the case of RHD equations, for deriving velocity from the conservative variables, we need to solve a nonlinear equation. We describe the process in [Appendix A](#).

In conservative variables \mathbf{U} , the RHD system (1), for the two-dimensional case, can be written in the conservative form as,

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{f}^x}{\partial x} + \frac{\partial \mathbf{f}^y}{\partial y} = 0. \quad (3)$$

Here, the fluxes are given by,

$$\mathbf{f}^x = \begin{bmatrix} Du_x \\ m_x u_x + p \\ m_y u_x \\ m_z u_x \\ m_x \end{bmatrix}, \quad \text{and } \mathbf{f}^y = \begin{bmatrix} Du_y \\ m_x u_y \\ m_y u_y + p \\ m_z u_y \\ m_y \end{bmatrix}.$$

The system is closed using the ideal equation of state (EOS henceforth), for which the special enthalpy h is given by

$$h = 1 + \frac{\gamma}{\gamma - 1} \frac{p}{\rho}, \quad (4)$$

where γ is the specific heat ratio. For the solution to be physically meaningful, we need to have positive density, positive pressure and magnitude of velocity to be less than the speed of light, which is assumed to be unity. Therefore, we consider the following solution set of physical states,

$$\Omega = \{\mathbf{U} = (D, \mathbf{m}, E), \rho > 0, p > 0, |\mathbf{u}| < 1\}.$$

Following [3, 29], we have the following result:

Lemma 2.1. *The system (3) is hyperbolic for the states \mathbf{U} in Ω with real eigenvalues and a complete set of eigenvectors.*

The eigenvalues and a complete set of right eigenvectors are presented in Appendix B. We now introduce entropy function \mathcal{U} and associated entropy flux \mathcal{F}^i for the system (3) as follows:

$$\mathcal{U} = -\frac{\rho\Gamma s}{\gamma-1} \text{ and } \mathcal{F}^i = -\frac{\rho\Gamma s u_i}{\gamma-1}, \quad i = x, y; \quad (5)$$

where $s = \ln(p\rho^{-\gamma})$. The pair $(\mathcal{U}, \mathcal{F}^i)$, $\mathcal{U}, \mathcal{F}^i$ as in (5), is called an entropy–entropy flux pair. We now have the following result:

Proposition 2.2. *The smooth solutions of (3) satisfy the entropy equality,*

$$\partial_t s + u_x \partial_x s = 0. \quad (6)$$

As a corollary, for any smooth function $H(s)$, we have,

$$\partial_t(\rho\Gamma H(s)) + \partial_x(\rho\Gamma u_x H(s)) = 0. \quad (7)$$

In particular, smooth solutions will satisfy entropy equality,

$$\partial_t \mathcal{U} + \partial_x \mathcal{F}^x = 0. \quad (8)$$

To prove Proposition 2.2, we will first prove the following two lemmas:

Lemma 2.3. *The smooth solutions of (3) satisfy the identity*

$$\frac{1}{\Gamma^2} \partial_t p = \partial_t p + u_x \partial_x p + \left(\frac{\rho h}{\Gamma}\right) (\partial_t \Gamma + \partial_x(\Gamma u_x)) - \rho h \partial_x u_x. \quad (9)$$

Proof. For the one-dimensional case, the RHD system (3) simplifies to the following set of three equations

$$\partial_t(\rho\Gamma) + \partial_x(\rho\Gamma u_x) = 0, \quad (10)$$

$$\partial_t(\rho h \Gamma^2 u_x) + \partial_x(\rho h \Gamma^2 u_x^2) + \partial_x p = 0, \quad (11)$$

$$\partial_t(\rho h \Gamma^2) - \partial_t(p) + \partial_x(\rho h \Gamma^2 u_x) = 0. \quad (12)$$

Using the product rule, Eq. (11) simplifies to

$$\partial_t(\rho h \Gamma^2 u_x) + (\rho h \Gamma^2 u_x) \partial_x u_x + u_x \partial_x(\rho h \Gamma^2 u_x) + \partial_x p = 0.$$

Now, substituting the value of $\partial_x(\rho h \Gamma^2 u_x)$ from Eq. (12) to get:

$$\partial_t(\rho h \Gamma^2 u_x) + (\rho h \Gamma^2 u_x) \partial_x u_x + u_x \partial_t p - u_x \partial_t(\rho h \Gamma^2) + \partial_x p = 0.$$

Using the product rule, this simplifies to

$$(\rho h \Gamma^2) \partial_t u_x + (\rho h \Gamma^2 u_x) \partial_x u_x + u_x \partial_t p + \partial_x p = 0.$$

Multiplying this with u_x leads to,

$$(\rho h \Gamma^2 u_x) (\partial_t u_x + u_x \partial_x u_x) + u_x^2 \partial_t p + u_x \partial_x p = 0. \quad (13)$$

Note that $\Gamma^2 = \frac{1}{1-u_x^2}$; thus, we get

$$\partial_t \Gamma = \Gamma^3 u_x \partial_t u_x \text{ and } \partial_x \Gamma = \Gamma^3 u_x \partial_x u_x. \quad (14)$$

Substituting (14), along with the value of u_x^2 (in terms of Γ), in Eq. (13) results in the identity (9). \square

Lemma 2.4. *The smooth solutions of the system (3) satisfy the equation*

$$(\partial_t p + u_x \partial_x p) + \frac{p\gamma}{\Gamma} (\partial_t \Gamma + \partial_x (\Gamma u_x)) = 0. \quad (15)$$

Proof. Applying product rule on Eq. (12), we get

$$\partial_t p = (\partial_t \Gamma + \partial_x (\Gamma u_x)) (\rho h \Gamma) + \Gamma \partial_t (\rho h \Gamma) + \Gamma u_x \partial_x (\rho h \Gamma).$$

Using Eq. (10), we obtain,

$$\begin{aligned} \partial_t p &= (\partial_t \Gamma + \partial_x (\Gamma u_x)) (\rho \Gamma) + (\partial_t \Gamma + \partial_x (\Gamma u_x)) \frac{\gamma}{\gamma-1} p \Gamma - \Gamma \partial_x (\rho u_x \Gamma) \\ &\quad + \Gamma \left(\frac{\gamma}{\gamma-1} \right) \partial_t (p \Gamma) + \Gamma u_x \partial_x (\rho \Gamma) + \Gamma u_x \left(\frac{\gamma}{\gamma-1} \right) \partial_x (p \Gamma). \end{aligned}$$

Using the product rule, we get,

$$\begin{aligned} \partial_t p &= (\partial_t \Gamma + \partial_x (\Gamma u_x)) (\rho \Gamma) + 2(\partial_t \Gamma + \partial_x (\Gamma u_x)) \frac{\gamma}{\gamma-1} p \Gamma + \left(\frac{\gamma}{\gamma-1} \right) \Gamma^2 \partial_t p - \rho \Gamma^2 \partial_x u_x \\ &\quad + \left(\frac{\gamma}{\gamma-1} \right) u_x \Gamma^2 \partial_x p - \left(\frac{\gamma}{\gamma-1} \right) p \Gamma^2 \partial_x u_x. \end{aligned} \quad (16)$$

Multiplying both sides of Eq. (16) by $\left(\frac{\gamma-1}{\Gamma^2 \gamma} \right)$ results in,

$$\partial_t p + u_x \partial_x p = \frac{\gamma-1}{\Gamma^2 \gamma} \partial_t p - (\partial_t \Gamma + \partial_x (\Gamma u_x)) \frac{\rho}{\Gamma} \frac{\gamma-1}{\gamma} - 2(\partial_t \Gamma + \partial_x (\Gamma u_x)) \frac{p}{\Gamma} + \rho \frac{\gamma-1}{\gamma} \partial_x u_x + p \partial_x u_x.$$

Substituting the value of $\frac{1}{\Gamma^2} \partial_t p$, from Eq. (9), results in the identity

$$(\partial_t p + u_x \partial_x p) + \frac{p\gamma}{\Gamma} (\partial_t \Gamma + \partial_x (\Gamma u_x)) = 0. \quad \square$$

Proof of Proposition (2.2). Recall that $s = \ln(p\rho^{-\gamma})$, therefore

$$\partial_t s = \frac{1}{p} \partial_t p - \frac{\gamma}{\rho} \partial_t \rho \quad \text{and} \quad \partial_x s = \frac{1}{p} \partial_x p - \frac{\gamma}{\rho} \partial_x \rho. \quad (17)$$

We simplify Eq. (10) using the product rule to obtain

$$\frac{1}{\rho} (\partial_t \rho + u_x \partial_x \rho) = -\frac{1}{\Gamma} (\partial_t \Gamma + \partial_x (\Gamma u_x)). \quad (18)$$

Combining Eqs. (15) and (18), we get

$$\frac{1}{p} (\partial_t p + u_x \partial_x p) - \frac{\gamma}{\rho} (\partial_t \rho + u_x \partial_x \rho) = 0,$$

which further simplifies (using Eq. (17)) to $\partial_t s + u_x \partial_x s = 0$. Equation (7) follows immediately using (6). We substitute $H(s) = \frac{-s}{\gamma-1}$ in (7) to get the equality (8). \square

Remark 2.1. The entropy equality (8) gives rise to the entropy inequality

$$\partial_t \mathcal{U} + \partial_x \mathcal{F}^x \leq 0 \quad (19)$$

for the non-smooth solutions. Also, this estimate can be extended to the two-dimensional case.

In the next section, we will design numerical schemes which will satisfy the entropy inequality (19) at the semi-discrete level.

3. Entropy-stable numerical schemes

In this section, we present the semi-discrete schemes for the two-dimensional RHD equations. Let us consider the two-dimensional domain $D = (x_a, x_b) \times (y_a, y_b)$, which is discretized using uniform mesh of size $\Delta x \times \Delta y$ where $\Delta x = x_{i+1} - x_i$ and $\Delta y = y_{j+1} - y_j$, with $0 \leq i \leq N_x$ and $0 \leq j \leq N_y$. We define points $x_i = x_a + i\Delta x$ and $y_j = y_a + j\Delta y$. The cell centres are defined as $x_{i+1/2} := \frac{x_i + x_{i+1}}{2}$ and $y_{j+1/2} := \frac{y_j + y_{j+1}}{2}$, and the domain D is divided into cells I_{ij} . Then, a general semi-discrete finite difference scheme has the following form:

$$\frac{d}{dt} \mathbf{U}_{ij}(t) + \frac{1}{\Delta x} \left(\mathbf{F}_{i+\frac{1}{2},j}^x(t) - \mathbf{F}_{i-\frac{1}{2},j}^x(t) \right) + \frac{1}{\Delta y} \left(\mathbf{F}_{i,j+\frac{1}{2}}^y(t) - \mathbf{F}_{i,j-\frac{1}{2}}^y(t) \right) = 0, \quad (20)$$

where $\mathbf{F}_{i+\frac{1}{2},j}^x$ and $\mathbf{F}_{i,j+\frac{1}{2}}^y$ are the numerical fluxes consistent with the continuous fluxes \mathbf{f}^x and \mathbf{f}^y , respectively. Let us introduce notation $[[v]]$ for the jump and \bar{v} for the arithmetic average of a scalar v over the cell boundaries of the cell I_{ij} in the following way:

$$\begin{aligned} [[v]]_{i+\frac{1}{2},j} &= v_{i+1,j} - v_{i,j}, & \bar{v}_{i+\frac{1}{2},j} &= \frac{1}{2}(v_{i+1,j} + v_{i,j}), \\ [[v]]_{i,j+\frac{1}{2}} &= v_{i,j+1} - v_{i,j}, & \bar{v}_{i,j+\frac{1}{2}} &= \frac{1}{2}(v_{i,j+1} + v_{i,j}). \end{aligned}$$

3.1. Entropy conservative schemes

In the first step, to obtain an entropy-stable scheme, we construct an entropy conservative scheme of arbitrary higher even-order. We say a scheme is *entropy conservative scheme* if the computed solution satisfies a semi-discrete entropy equality,

$$\frac{d}{dt} \mathcal{U}(\mathbf{U}_{ij}) + \frac{1}{\Delta x} \left(\tilde{\mathcal{F}}_{i+\frac{1}{2},j}^x - \tilde{\mathcal{F}}_{i-\frac{1}{2},j}^x \right) + \frac{1}{\Delta y} \left(\tilde{\mathcal{F}}_{i,j+\frac{1}{2}}^y - \tilde{\mathcal{F}}_{i,j-\frac{1}{2}}^y \right) = 0$$

for some numerical entropy fluxes $\tilde{\mathcal{F}}^x$ and $\tilde{\mathcal{F}}^y$ consistent with the fluxes \mathcal{F}^x and \mathcal{F}^y , respectively.

Let us introduce the entropy variables, $\mathbf{v}(\mathbf{U}) = \partial_{\mathbf{U}} \mathcal{U}$, and the entropy potentials, $\psi^\alpha(\mathbf{U}) = \mathbf{v}^\top(\mathbf{U}) \cdot \mathbf{f}^\alpha(\mathbf{U}) - \mathcal{F}^\alpha(\mathbf{U})$ for $\alpha \in \{x, y\}$. A simple calculation results in

$$\mathbf{v} = \begin{pmatrix} \frac{\gamma-s}{\gamma-1} + \beta \\ u_x \Gamma \beta \\ u_y \Gamma \beta \\ -\Gamma \beta \end{pmatrix}, \text{ and } \psi^\alpha = \rho \Gamma u_\alpha \text{ for } \alpha \in \{x, y\}, \beta = \frac{\rho}{p}.$$

We now recall the following theorem due to [34], which provides a sufficient condition for the consistent fluxes, $\tilde{\mathbf{F}}^x$ and $\tilde{\mathbf{F}}^y$, to be entropy conservative.

Theorem 3.1. [34] *Let $\tilde{\mathbf{F}}^x$ and $\tilde{\mathbf{F}}^y$ be the consistent numerical fluxes, which satisfies,*

$$[[\mathbf{v}]]_{i+\frac{1}{2},j}^T \tilde{\mathbf{F}}_{i+\frac{1}{2},j}^x = [[\psi^x]]_{i+\frac{1}{2},j}, \quad [[\mathbf{v}]]_{i,j+\frac{1}{2}}^T \tilde{\mathbf{F}}_{i,j+\frac{1}{2}}^y = [[\psi^y]]_{i,j+\frac{1}{2}}, \quad (21)$$

then the scheme (20) with the numerical fluxes $\tilde{\mathbf{F}}^x$ and $\tilde{\mathbf{F}}^y$ is second-order accurate and entropy conservative, i.e. the computed solutions satisfy the discrete entropy equality

$$\frac{d}{dt} \mathcal{U}(\mathbf{U}_{ij}) + \frac{1}{\Delta x} \left(\tilde{\mathcal{F}}_{i+\frac{1}{2},j}^x - \tilde{\mathcal{F}}_{i-\frac{1}{2},j}^x \right) + \frac{1}{\Delta y} \left(\tilde{\mathcal{F}}_{i,j+\frac{1}{2}}^y - \tilde{\mathcal{F}}_{i,j-\frac{1}{2}}^y \right) = 0,$$

corresponding to the numerical entropy fluxes,

$$\tilde{\mathcal{F}}_{i+\frac{1}{2},j}^x = \bar{\mathbf{v}}_{i+\frac{1}{2},j}^T \tilde{\mathbf{F}}_{i+\frac{1}{2},j}^x - \bar{\psi}_{i+\frac{1}{2},j}^x, \text{ and } \tilde{\mathcal{F}}_{i,j+\frac{1}{2}}^y = \bar{\mathbf{v}}_{i,j+\frac{1}{2}}^T \tilde{\mathbf{F}}_{i,j+\frac{1}{2}}^y - \bar{\psi}_{i,j+\frac{1}{2}}^y.$$

Note that Eq. (21), for the x -directional flux with two velocity components, provides a single algebraic equation in the four unknowns $\tilde{\mathbf{F}}^x = (F_1^x, F_2^x, F_3^x, F_4^x)^\top$. For the Euler equations, in [18], authors have presented an affordable entropy conservative flux. More recently, Chandrashekar in [8] has proposed a new approach to find an entropy conservative flux for the Euler equations. Here, we follow [8], to find an entropy conservative flux.

3.1.1. Entropy conservative flux. For simplicity, we ignore the indices. Let us define $a^{\text{ln}} = \frac{[a]}{[\log a]}$ as the logarithmic average of scalar a (as in [12]). Using the jump identity $[[ab]] = \bar{a}[[b]] + \bar{b}[[a]]$ for Eq. (2), we obtain

$$[[\Gamma]] = \frac{1}{\bar{\Gamma}}(\bar{m}_x[[m_x]] + \bar{m}_y[[m_y]]),$$

where, $m_x = \Gamma u_x$ and $m_y = \Gamma u_y$. Using this, we can write the jump in \mathbf{v} in terms of the jump in ρ , β , m_x and m_y as follows:

$$[[\mathbf{v}]] = \begin{pmatrix} \frac{[[\rho]]}{\rho^{\text{ln}}} + k_1[[\beta]] \\ \bar{m}_x[[\beta]] + \bar{\beta}[[m_x]] \\ \bar{m}_y[[\beta]] + \bar{\beta}[[m_y]] \\ -\bar{\Gamma}[[\beta]] - \frac{\bar{\beta}\bar{m}_x}{\bar{\Gamma}}[[m_x]] - \frac{\bar{\beta}\bar{m}_y}{\bar{\Gamma}}[[m_y]] \end{pmatrix},$$

with $k_1 = \left(\frac{1}{\gamma-1}\frac{1}{\beta^{\text{ln}}} + 1\right)$. Furthermore, $[[\psi^x]]$ and $[[\psi^y]]$ can be written as

$$[[\psi^x]] = \bar{\rho}[[m_x]] + \bar{m}_x[[\rho]] \quad \text{and} \quad [[\psi^y]] = \bar{\rho}[[m_y]] + \bar{m}_y[[\rho]], \quad \text{respectively.}$$

Let $\tilde{\mathbf{F}}^x = (F_1^x, F_2^x, F_3^x, F_4^x)^\top$ be an entropy conservative flux and $\mathbf{v} = (v_1, v_2, v_3, v_4)^\top$, then Eq. (21) results in,

$$F_1^x[[v_1]] + F_2^x[[v_2]] + F_3^x[[v_3]] + F_4^x[[v_4]] = [[\psi^x]],$$

which simplifies to

$$\begin{aligned} & \left(\frac{F_1^x}{\rho^{\text{ln}}}\right) [[\rho]] + \left(\bar{\beta}F_2^x - \frac{\bar{\beta}\bar{m}_x}{\bar{\Gamma}}F_4^x\right) [[m_x]] + \left(\bar{\beta}F_3^x - \frac{\bar{\beta}\bar{m}_y}{\bar{\Gamma}}F_4^x\right) [[m_y]] + \left(k_1F_1^x + \bar{m}_x F_2^x + \bar{m}_y F_3^x - F_4^x \bar{\Gamma}\right) [[\beta]] \\ & = \bar{\rho}[[m_x]] + \bar{m}_x[[\rho]]. \end{aligned}$$

Comparing both sides the coefficients of independent jumps, i.e. coefficient of $[[\rho]]$, $[[\beta]]$, $[[m_x]]$ and $[[m_y]]$, we obtain $\tilde{\mathbf{F}}^x = (F_1^x, F_2^x, F_3^x, F_4^x)$, where the components are given by

$$\begin{aligned} F_1^x &= \rho^{\text{ln}}\bar{m}_x, \\ F_2^x &= \frac{1}{\bar{\beta}}\left(\frac{\bar{\beta}\bar{m}_x}{\bar{\Gamma}}F_4^x + \bar{\rho}\right), \\ F_3^x &= \frac{\bar{m}_y}{\bar{\Gamma}}F_4^x, \\ F_4^x &= \frac{-\bar{\Gamma}\left(k_1\rho^{\text{ln}}\bar{m}_x + \frac{\bar{m}_x\bar{\rho}}{\bar{\beta}}\right)}{(\bar{m}_x^2 + \bar{m}_y^2 - \bar{\Gamma}^2)}, \end{aligned}$$

with $k_1 = \left(\frac{1}{(\gamma-1)\beta^{\ln}} + 1 \right)$. It is straightforward to verify that $\tilde{\mathbf{F}}^x$ is consistent with the flux \mathbf{f}_x . Similarly, we can derive the expression for $\tilde{\mathbf{F}}^y$ given by,

$$\tilde{\mathbf{F}}^y(\mathbf{U}_l, \mathbf{U}_r) = \begin{pmatrix} \rho^{\ln} \overline{m}_y \\ \frac{-\overline{m}_x \left(k_1 \rho^{\ln} \overline{m}_y + \frac{\overline{m}_y \overline{\rho}}{\beta} \right)}{(\overline{m}_x^2 + \overline{m}_y^2 - \Gamma^2)} \\ \frac{-\overline{m}_y \left(k_1 \rho^{\ln} \overline{m}_y + \frac{\overline{m}_y \overline{\rho}}{\beta} \right)}{(\overline{m}_x^2 + \overline{m}_y^2 - \Gamma^2)} + \frac{\overline{\rho}}{\beta} \\ \frac{-\Gamma \left(k_1 \rho^{\ln} \overline{m}_y + \frac{\overline{m}_y \overline{\rho}}{\beta} \right)}{(\overline{m}_x^2 + \overline{m}_y^2 - \Gamma^2)} \end{pmatrix}.$$

3.1.2. Higher-order entropy conservative flux. The entropy conservative fluxes derived above are only second-order accurate. We can use second-order fluxes as the building blocks for the construction of $2p$ th-order accurate fluxes for any positive integer p . These $2p$ th-order fluxes consist of linear combinations of second-order accurate entropy conservative flux $\tilde{\mathbf{F}}^x$ by following [21]:

Theorem 3.2. [21, Theorem 4.4] For $p \in \mathbb{N}$, assume that $\alpha_1^p, \alpha_2^p, \dots, \alpha_p^p$ solves the p linear equations

$$\sum_{r=1}^p r \alpha_r^p = 1, \quad \sum_{i=1}^p i^{2s-1} \alpha_i^p \quad (s = 2, 3, \dots, p),$$

and defining $\tilde{\mathbf{F}}_{i+\frac{1}{2},j}^{x,2p} = \sum_{r=1}^p \alpha_r^p \sum_{s=0}^{r-1} \tilde{\mathbf{F}}^x(\mathbf{U}_{i-s,j}, \mathbf{U}_{i-s+r,j})$. Similarly, we have the expression for $\tilde{\mathbf{F}}_{i,j+\frac{1}{2}}^{y,2p}$. Then, the finite difference scheme with fluxes $\tilde{\mathbf{F}}^{x,2p}$ and $\tilde{\mathbf{F}}^{y,2p}$ is

- (i) $2p$ -th-order accurate
- (ii) entropy conservative, i.e. it satisfies the discrete entropy identity

$$\frac{d}{dt} \mathcal{U}(\mathbf{U}_{ij}) + \frac{1}{\Delta x} \left(\tilde{\mathcal{F}}_{i+\frac{1}{2},j}^{x,2p} - \tilde{\mathcal{F}}_{i-\frac{1}{2},j}^{x,2p} \right) + \frac{1}{\Delta y} \left(\tilde{\mathcal{F}}_{i,j+\frac{1}{2}}^{y,2p} - \tilde{\mathcal{F}}_{i,j-\frac{1}{2}}^{y,2p} \right) = 0$$

corresponding to the numerical entropy fluxes,

$$\begin{aligned} \tilde{\mathcal{F}}_{i+\frac{1}{2},j}^{x,2p} &= \sum_{r=1}^p \alpha_r^p \sum_{s=0}^{r-1} \tilde{\mathcal{F}}_{i+\frac{1}{2},j}^x(\mathbf{U}_{i-s,j}, \mathbf{U}_{i-s+r,j}), \\ \tilde{\mathcal{F}}_{i,j+\frac{1}{2}}^{y,2p} &= \sum_{r=1}^p \alpha_r^p \sum_{s=0}^{r-1} \tilde{\mathcal{F}}_{i,j+\frac{1}{2}}^y(\mathbf{U}_{i,j-s}, \mathbf{U}_{i,j-s+r}). \end{aligned}$$

In particular, the 4th-order ($p = 2$) x -directional entropy conservative flux is given by

$$\tilde{\mathbf{F}}_{i+\frac{1}{2},j}^{x,4} = \frac{4}{3} \tilde{\mathbf{F}}_{i+\frac{1}{2},j}^x(\mathbf{U}_{i,j}, \mathbf{U}_{i+1,j}) - \frac{1}{6} \left(\tilde{\mathbf{F}}_{i+\frac{1}{2},j}^x(\mathbf{U}_{i-1,j}, \mathbf{U}_{i+1,j}) + \tilde{\mathbf{F}}_{i+\frac{1}{2},j}^x(\mathbf{U}_{i,j}, \mathbf{U}_{i+2,j}) \right).$$

Similar expression can be derived for the y -directional fourth order flux.

Remark 3.1. Theorem 3.2 holds only for the even positive integers. So, for q th-order (q is an odd integer) accurate scheme, we use a $(q + 1)$ th-order (an even number) accurate flux.

3.2. Entropy-stable schemes

The entropy conservative schemes will produce high-frequency oscillations near the shocks as they do not dissipate entropy at the shocks. We follow [34] to introduce modified fluxes which will ensure entropy stability at the shocks. We consider

$$\begin{aligned}\mathbf{F}_{i+\frac{1}{2},j}^x &= \tilde{\mathbf{F}}_{i+\frac{1}{2},j}^x - \frac{1}{2} \mathbf{D}_{i+\frac{1}{2},j}^x \llbracket \mathbf{v} \rrbracket_{i+\frac{1}{2},j}, \\ \mathbf{F}_{i,j+\frac{1}{2}}^y &= \tilde{\mathbf{F}}_{i,j+\frac{1}{2}}^y - \frac{1}{2} \mathbf{D}_{i,j+\frac{1}{2}}^y \llbracket \mathbf{v} \rrbracket_{i,j+\frac{1}{2}},\end{aligned}\quad (22)$$

where $\mathbf{D}_{i+\frac{1}{2},j}^x$ and $\mathbf{D}_{i,j+\frac{1}{2}}^y$ are symmetric positive definite matrices. Then, we have the following lemma:

Lemma 3.3. [34] *The numerical scheme (20) with the modified numerical fluxes (22) is entropy stable, i.e. the computed solution satisfies*

$$\frac{d}{dt} \mathcal{U}(\mathbf{U}_{ij}) + \frac{1}{\Delta x} \left(\hat{\mathcal{F}}_{i+\frac{1}{2},j}^x - \hat{\mathcal{F}}_{i-\frac{1}{2},j}^x \right) + \frac{1}{\Delta y} \left(\hat{\mathcal{F}}_{i,j+\frac{1}{2}}^y - \hat{\mathcal{F}}_{i,j-\frac{1}{2}}^y \right) \leq 0$$

with the consistent numerical entropy flux functions,

$$\hat{\mathcal{F}}_{i+\frac{1}{2},j}^x = \tilde{\mathcal{F}}_{i+\frac{1}{2},j}^x + \frac{1}{2} \bar{\mathbf{v}}_{i+\frac{1}{2},j}^\top \mathbf{D}_{i+\frac{1}{2},j}^x \llbracket \mathbf{v} \rrbracket_{i+\frac{1}{2},j} \quad \text{and} \quad \hat{\mathcal{F}}_{i,j+\frac{1}{2}}^y = \tilde{\mathcal{F}}_{i,j+\frac{1}{2}}^y + \frac{1}{2} \bar{\mathbf{v}}_{i,j+\frac{1}{2}}^\top \mathbf{D}_{i,j+\frac{1}{2}}^y \llbracket \mathbf{v} \rrbracket_{i,j+\frac{1}{2}}.$$

Here, we use *Rusanov's type* diffusion operators for the matrix \mathbf{D} given by,

$$\mathbf{D}_{i+\frac{1}{2},j}^x = \mathbf{R}_{i+\frac{1}{2},j}^x \Lambda_{i+\frac{1}{2},j}^x \mathbf{R}_{i+\frac{1}{2},j}^{x\top}, \quad \text{and} \quad \mathbf{D}_{i,j+\frac{1}{2}}^y = \mathbf{R}_{i,j+\frac{1}{2}}^y \Lambda_{i,j+\frac{1}{2}}^y \mathbf{R}_{i,j+\frac{1}{2}}^{y\top} \quad (23)$$

where \mathbf{R}^i , $i \in \{x, y\}$, are matrices of the entropy scaled right eigenvectors (see Appendix B) of the Jacobian $\frac{\partial \mathbf{f}^i}{\partial \mathbf{U}}$, and Λ^i , $i \in \{x, y\}$, are 4×4 diagonal matrices of the form

$$\Lambda^i = \text{diag} \left\{ \left(\max_{1 \leq k \leq 4} |\lambda_k^i| \right) \mathbf{I}_{4 \times 4} \right\}, \quad i \in \{x, y\},$$

where $\{\lambda_k^i : 1 \leq k \leq 4\}$ is the set of eigenvalue of the Jacobian $\frac{\partial \mathbf{f}^i}{\partial \mathbf{U}}$. We also note that, following [4], $\mathbf{U}_\nu = \mathbf{R}^k (\mathbf{R}^k)^\top$ for $k \in \{x, y\}$ and hence,

$$\mathbf{R}^k \Lambda^k (\mathbf{R}^k)^{(-1)} \llbracket \mathbf{U} \rrbracket \approx \mathbf{R}^k \Lambda^k (\mathbf{R}^k)^{(-1)} \mathbf{U}_\nu \llbracket \mathbf{v} \rrbracket = \mathbf{R}^k \Lambda^k (\mathbf{R}^k)^\top \llbracket \mathbf{v} \rrbracket.$$

Hence, the entropy diffusion operator used here is similar to the *Roe* diffusion operator, when the entropy scaled right eigenvectors are used.

With the choice of diffusion operator (23), the numerical scheme (20) with the numerical flux (22) is entropy stable. However, the scheme is only first-order accurate due to the presence of the jump terms $\llbracket \mathbf{v} \rrbracket_{i+\frac{1}{2},j}$ and $\llbracket \mathbf{v} \rrbracket_{i,j+\frac{1}{2}}$.

3.3. High-order diffusion operators

A straightforward way to increase the order of accuracy is to approximate the jumps using higher-order reconstruction process. However, proving the entropy stability of the resulting scheme is not possible. Instead, we follow the process prescribed in [12] and introduce the *scaled entropy variables*,

$$\mathbf{W}_{ij}^{x,\pm} = \mathbf{R}_{i\pm\frac{1}{2},j}^{x\top} \mathbf{v}_{ij}.$$

If $\tilde{\mathbf{W}}_{ij}^{x,\pm}$ denotes the k -th-order reconstructed values of $\mathbf{W}_{i,j}^{x,\pm}$ in the x -direction, then,

$$\mathbf{p}_{ij}^{x,\pm} = \left\{ \tilde{\mathbf{R}}_{i\pm\frac{1}{2},j}^{x\top} \right\}^{(-1)} \tilde{\mathbf{W}}_{i,j}^{x,\pm},$$

TABLE 1. *Coefficients for Runge–Kutta time stepping*

Order	α_{il}		β_{il}		
2	1		1		
	1/2	1/2	0	1/2	
3	1		1		
	3/4	1/4	0	1/4	
	1/3	0	2/3	0	2/3

are the corresponding k -th-order reconstructed values for \mathbf{v}_{ij} . Hence, the modified numerical flux is given by,

$$\mathbf{F}_{i+\frac{1}{2},j}^{x,k} = \tilde{\mathbf{F}}_{i+\frac{1}{2},j}^{x,2p} - \frac{1}{2} \mathbf{D}_{i+\frac{1}{2},j}^x \llbracket \mathbf{P}^x \rrbracket_{i+\frac{1}{2},j} \tag{24}$$

where $\llbracket \mathbf{P}^x \rrbracket_{i+\frac{1}{2},j}$ stands for,

$$\llbracket \mathbf{P}^x \rrbracket_{i+\frac{1}{2},j} = \mathbf{P}_{i+1,j}^{x,-} - \mathbf{P}_{ij}^{x,+},$$

and $p \in \mathbb{N}$ is chosen as

- $p = k/2$ if k is even,
- $p = (k + 1)/2$ if k is odd.

Following [12], a sufficient condition for the numerical flux (24) to be entropy stable is that the reconstruction process for \mathbf{W} must satisfy the *sign-preserving property*. For the second-order reconstruction, we use the *minmod* reconstruction which satisfies this property. We denote this scheme by O2-ES. For the higher-order scheme, we follow [13] and use the ENO reconstruction. We use *third-* and *fourth-order* ENO-based reconstruction and denote these schemes by O3-ES and O4-ES, respectively.

3.4. Semi-discrete entropy stability

We now have the following result:

Theorem 3.4. *The semi-discrete schemes O2-ES, O3-ES and O4-ES designed above are entropy stable, i.e. they satisfy,*

$$\frac{d}{dt} \mathcal{U}(\mathbf{U}_{ij}) + \frac{1}{\Delta x} \left(\hat{\mathcal{F}}_{i+\frac{1}{2},j}^x - \hat{\mathcal{F}}_{i-\frac{1}{2},j}^x \right) + \frac{1}{\Delta y} \left(\hat{\mathcal{F}}_{i,j+\frac{1}{2}}^y - \hat{\mathcal{F}}_{i,j-\frac{1}{2}}^y \right) \leq 0$$

where $\hat{\mathcal{F}}^x$ and $\hat{\mathcal{F}}^y$ are the consistent numerical entropy fluxes.

3.5. Time discretization

For the time discretization, we use SSP Runge–Kutta methods [16]. The second- and third-order accurate SSP-RK methods rely on the following algorithm:

1. Define $\mathbf{U}^0 = \mathbf{U}^n$.
2. For $m = 1, \dots, k + 1$, evaluate,

$$\mathbf{U}_{i,j}^{(m)} = \sum_{l=0}^{m-1} \alpha_{ml} \mathbf{U}_{i,j}^{(l)} + \beta_{ml} \Delta t^n (\mathcal{L}_{i,j}(\mathbf{U}^{(l)})),$$

where, α_{ml} and β_{ml} follows from Table 1.

3. Set $\mathbf{U}_{i,j}^{n+1} = \mathbf{U}_{i,j}^{(k+1)}$.

TABLE 2. Accuracy test: rate of convergence of second-order (O2-ES), third-order (O3-ES) and fourth-order (O4-ES) entropy-stable schemes

Number of cells	O2-ES		O3-ES		O4-ES	
	L^1 error	Order	L^1 error	Order	L^1 error	Order
100	1.4397E-02	–	1.2232E-04	–	1.1444E-05	–
200	3.9752E-03	1.86	1.5290E-05	3.00	8.1805E-07	3.81
400	1.0807E-03	1.88	1.9115E-06	2.99	5.7077E-08	3.84
800	2.9050E-04	1.89	2.3896E-07	2.99	3.8740E-09	3.88
1600	7.6511E-05	1.92	2.9871E-08	2.99	2.5967E-10	3.90

For the fourth-order accuracy, we consider the following time update:

$$\mathbf{U}^1 = \mathbf{U}^n + 0.39175222700392(\Delta t \mathcal{L}(\mathbf{U}^n)),$$

$$\mathbf{U}^2 = 0.44437049406734(\mathbf{U}^n) + 0.55562950593266(\mathbf{U}^1) + 0.36841059262959(\Delta t \mathcal{L}(\mathbf{U}^1)),$$

$$\mathbf{U}^3 = 0.62010185138540(\mathbf{U}^n) + 0.37989814861460(\mathbf{U}^2) + 0.25189177424738(\Delta t \mathcal{L}(\mathbf{U}^2)),$$

$$\mathbf{U}^4 = 0.17807995410773(\mathbf{U}^n) + 0.82192004589227(\mathbf{U}^3) + 0.54497475021237(\Delta t \mathcal{L}(\mathbf{U}^3)),$$

and finally,

$$\begin{aligned} \mathbf{U}^{n+1} = & 0.00683325884039(\mathbf{U}^n) + 0.51723167208978(\mathbf{U}^2) + 0.12759831133288(\mathbf{U}^3) \\ & + 0.34833675773694(\mathbf{U}^4) + 0.08460416338212(\Delta t \mathcal{L}(\mathbf{U}^3)) + 0.22600748319395(\Delta t \mathcal{L}(\mathbf{U}^4)). \end{aligned}$$

4. Numerical results

We will now present numerical test cases for the above schemes. We present both one- and two-dimensional test cases and compute several Riemann problems. We use CFL number 0.2 for all the presented test cases.

4.1. One-dimensional test cases

4.1.1. Accuracy test. To test the accuracy of the proposed schemes, we first consider a problem which has smooth solution. We consider the computational domain $[0, 1]$ with periodic boundary conditions. The initial density profile is assumed to be $\rho(x, 0) = 2 + \sin(2\pi x)$, which is then advected with the velocity $\mathbf{u} = (0.5, 0)^\top$. The pressure is assumed to be constant with $p = 1.0$. The exact solution is transportation of the density profile in x -direction, i.e. $\rho(x, t) = 2 + \sin(2\pi(x - t))$, while the other variables remain unchanged. We use the adiabatic index, $\gamma = \frac{5}{3}$. The simulations are performed till time $t = 2.0$.

We present L^1 -errors of the density profile for the O2-ES, O3-ES, and O4-ES schemes at 100, 200, 400, 800 and 1600 cells in Table 2. We note that all the schemes preserve the predicted order of accuracy.

4.1.2. Isentropic smooth flows. In this test case, we consider a one-dimensional isentropic smooth pulse in a fixed reference state, $\mathbf{w}_{ref} = (\rho_{ref}, \mathbf{u}_{ref}, p_{ref})$. A detailed illustration of the problem is given in [23] and [40]. We set $\rho_{ref} = 1$, $\mathbf{u}_{ref} = \mathbf{0}$, $p_{ref} = 100$. For the initial pulse we consider $\alpha = 1$ and $L = 0.3$ and then set the density profile as,

$$\rho_0(x) = \rho_{ref}[1 + \alpha f(x)],$$

where f is given by

$$f(x) = \begin{cases} \left[\left(\frac{x}{L} \right)^2 - 1 \right]^4, & \text{if } |x| < L \\ 0, & \text{otherwise.} \end{cases}$$

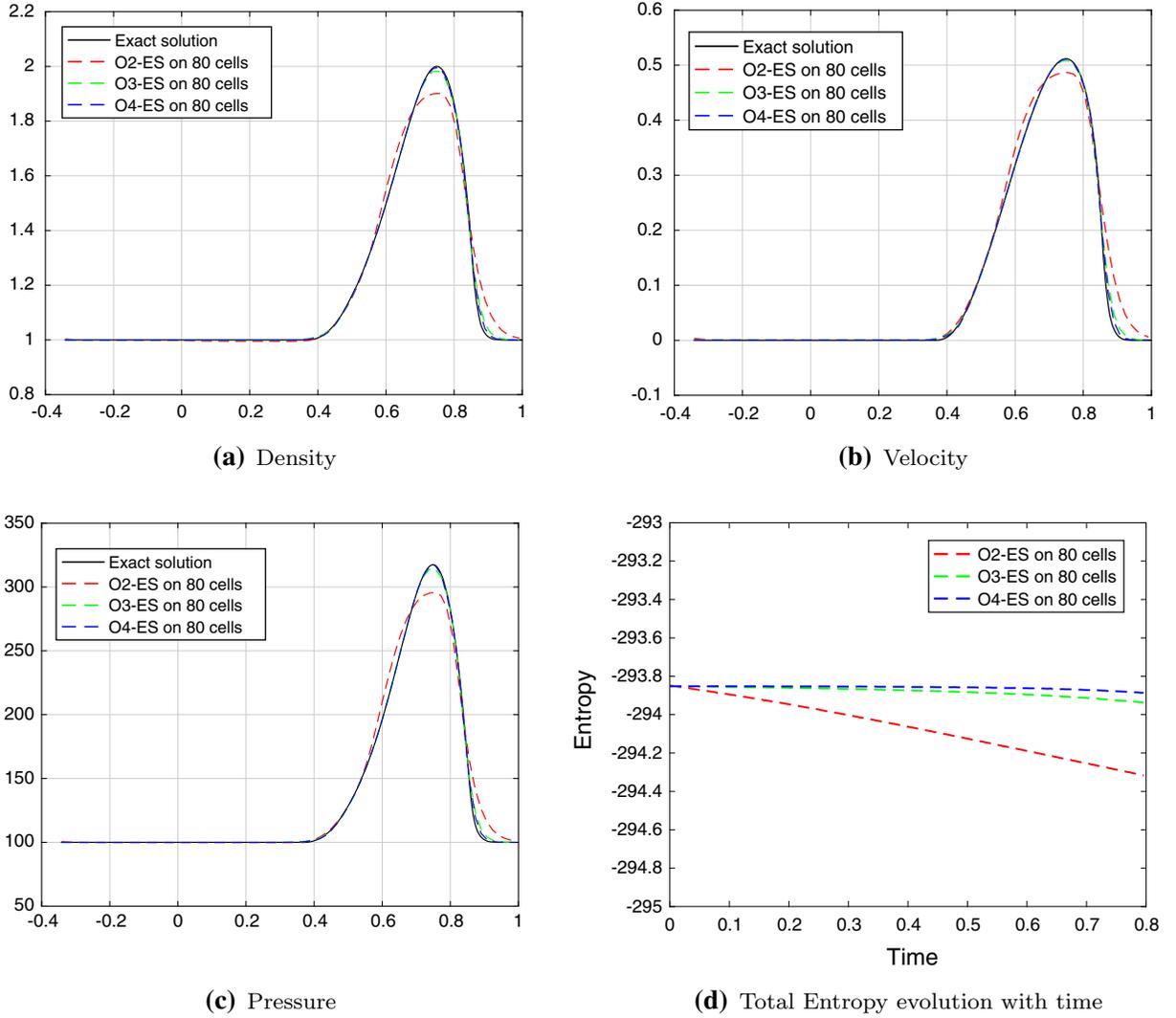


FIG. 1. Isentropic smooth flows: plots of density, velocity, pressure and total entropy evolution at 100 cells for O2-ES, O3-ES and O4-ES schemes

The initial pressure profile is given by $p = K\rho^\gamma$, where K is a constant. Throughout the domain, we assume a constant value to the Riemann invariant J , to obtain the initial velocity inside the pulse. Here, J is given by

$$J = \frac{1}{2} \ln \left(\frac{1 + u_x}{1 - u_x} \right) - \frac{1}{\sqrt{\gamma - 1}} \ln \left(\frac{\sqrt{\gamma - 1} + c}{\sqrt{\gamma - 1} - c} \right),$$

where c is the sound speed. We consider the periodic boundary condition for the test case. The adiabatic index is set to $\gamma = \frac{5}{3}$. We present the numerical results using 80 cells on the domain $[-0.35, 1]$. The computations are performed till time $t = 0.8$. The numerical results for the proposed schemes are presented in Fig. 1. The exact solution is using the characteristic analysis [40].

We observe that all schemes can capture the waves profile even at the coarse mesh of 80 cells. The second-order scheme O2-ES is most diffusive when compared to third- and fourth-order schemes. In

TABLE 3. *Isentropic smooth flows: rate of convergence of second (O2-ES), third (O3-ES) and fourth-order (O4-ES) entropy-stable schemes*

Number of cells	O2-ES		O3-ES		O4-ES	
	L^1 error	Order	L^1 error	Order	L^1 error	Order
80	2.0311E-02	–	3.8437E-03	–	1.9803E-03	–
160	6.7215E-03	1.60	7.4271E-04	2.37	3.0451E-04	2.70
320	2.0362E-03	1.72	1.1776E-04	2.66	6.3459E-05	2.26
640	5.8148E-04	1.81	1.7238E-05	2.77	2.4278E-05	1.39
1280	1.5773E-04	1.88	7.7320E-06	1.66	8.3571E-06	1.54
2560	4.1424E-05	1.93	4.6469E-06	0.73	5.1925E-06	0.69
5120	1.0693E-05	1.95	1.6906E-06	1.46	1.7269E-06	1.59

Fig. 1d, we plot the evolution of the total entropy with time. We observe that entropy remains almost conserved. Also, for the third- and fourth-order schemes we observe almost same entropy profile.

In Table 3, we have presented the L^1 -errors and order of convergence for the density at various resolutions. We note that the second-order schemes converge with the predicted rate of convergence; however, third- and fourth-order schemes do not converge with the desired accuracy. This is similar to the case in [40], where other higher-order schemes also have low accuracy, except WENO-based schemes.

4.1.3. Riemann problem 1. In this test case, we consider a Riemann problem from [25]. We take computational domain of $[0, 1]$ with the initial discontinuity centred at $x = 0.5$ with initial data:

$$(\rho, \mathbf{u}, p) = \begin{cases} (1, -0.6, 0, 10) & \text{if } x < 0.5 \\ (10, 0.5, 0, 20) & \text{if } x > 0.5. \end{cases}$$

We assume outflow boundary conditions at the boundary and consider adiabatic index, $\gamma = \frac{5}{3}$. The exact solution of the problem contains two rarefaction waves moving in the opposite direction separated by a contact wave.

The numerical results are presented in Fig. 2 for the O2-ES, O3-ES and O4-ES schemes using 100 and 500 cells. We have plotted density, velocity and pressure profile at time $t = 0.4$. We compare the obtained results with the exact solution achieved using [24]. We observe that at the resolution of 100 cells, O4-ES is more accurate than the O3-ES scheme, which further is more accurate than the O2-ES, as expected. At the refined mesh with 500 cells, all the schemes are relatively close to each other and resolve both rarefaction and the contact.

In Fig. 2d, we have plotted the evolution of total entropy with time. We note that at 100 cells O4-ES is least diffusive followed by O3-ES with O2-ES being most diffusive. At 500 cells, the entropy decay of the schemes is much closer to each other.

4.1.4. Riemann problem 2. In this test case, we consider a shock tube like test case from [24]. The problem will test wave capturing ability of the proposed schemes as all kinds of waves are present in the solution. Similar to previous Riemann problem, we consider the computational domain of $[0, 1]$ with outflow boundary conditions, and with initial discontinuity at $x = 0.5$ separating the states given as:

$$(\rho, \mathbf{u}, p) = \begin{cases} (1, \mathbf{0}, 10^3) & \text{if } x < 0.5 \\ (1, \mathbf{0}, 10^{-2}) & \text{if } x > 0.5 \end{cases}$$

In Fig. 3, we present the results at 100 and 500 cells for time $t = 0.4$ for O2-ES, O3-ES and O4-ES with $\gamma = \frac{5}{3}$. We have plotted density, velocity and pressure profiles. We observe that the second-order scheme O2-ES at 100 cells completely fails to capture the waves, while O3-ES and O4-ES perform better.

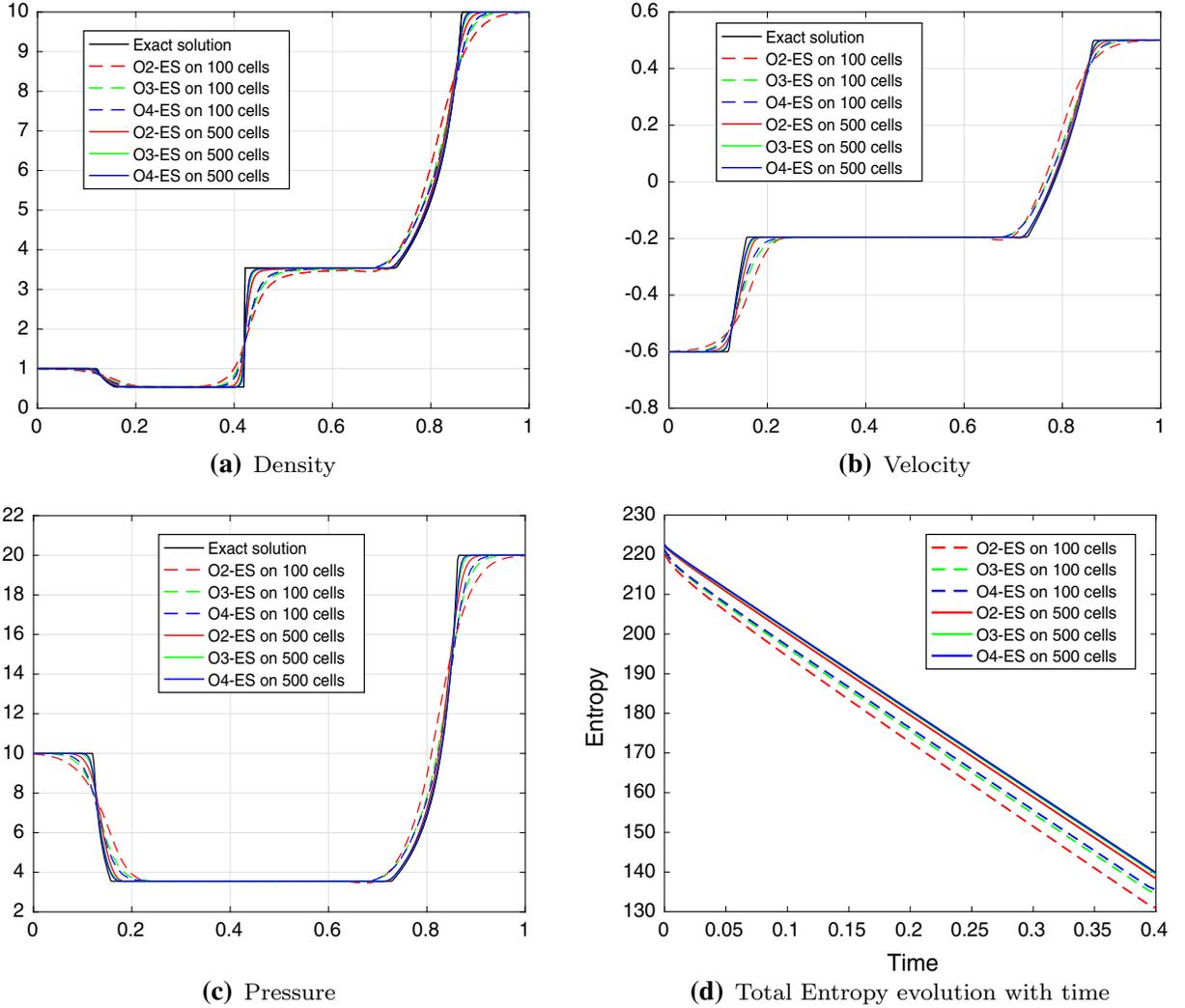


FIG. 2. Riemann problem 1: Plots of density, velocity, pressure and total entropy evolution at 100 and 500 cells for O2-ES, O3-ES and O4-ES schemes

However, at 500 cells all the schemes are able to capture the waves with O4-ES being most accurate. We have followed [24] to obtain the exact solution of Riemann Problem 2.

In Fig. 3d, we plot evolution of total entropy with time. At 100 cells, O2-ES scheme is most diffusive followed by O3-ES scheme. The fourth-order scheme O4-ES is most accurate. At the finer mesh of 500 cells, we have a similar performance of the schemes, with much closer decay rates.

4.1.5. Riemann problem 3. In this test case, we consider a Riemann problem which is similar to a test case in [25]. The computational domain of $[0, 1]$ contains an initial discontinuity separating the states:

$$(\rho, \mathbf{u}, p) = \begin{cases} (10, \mathbf{0}, \frac{40}{3}) & \text{if } x < 0.5, \\ (1, \mathbf{0}, \frac{2}{3} \times 10^{-5}) & \text{if } x > 0.5. \end{cases}$$

The problem is similar to the shock tube test case for Euler equation of compressible fluids. However, here the right state contains very low pressure, which will test robustness of the schemes. The numerical

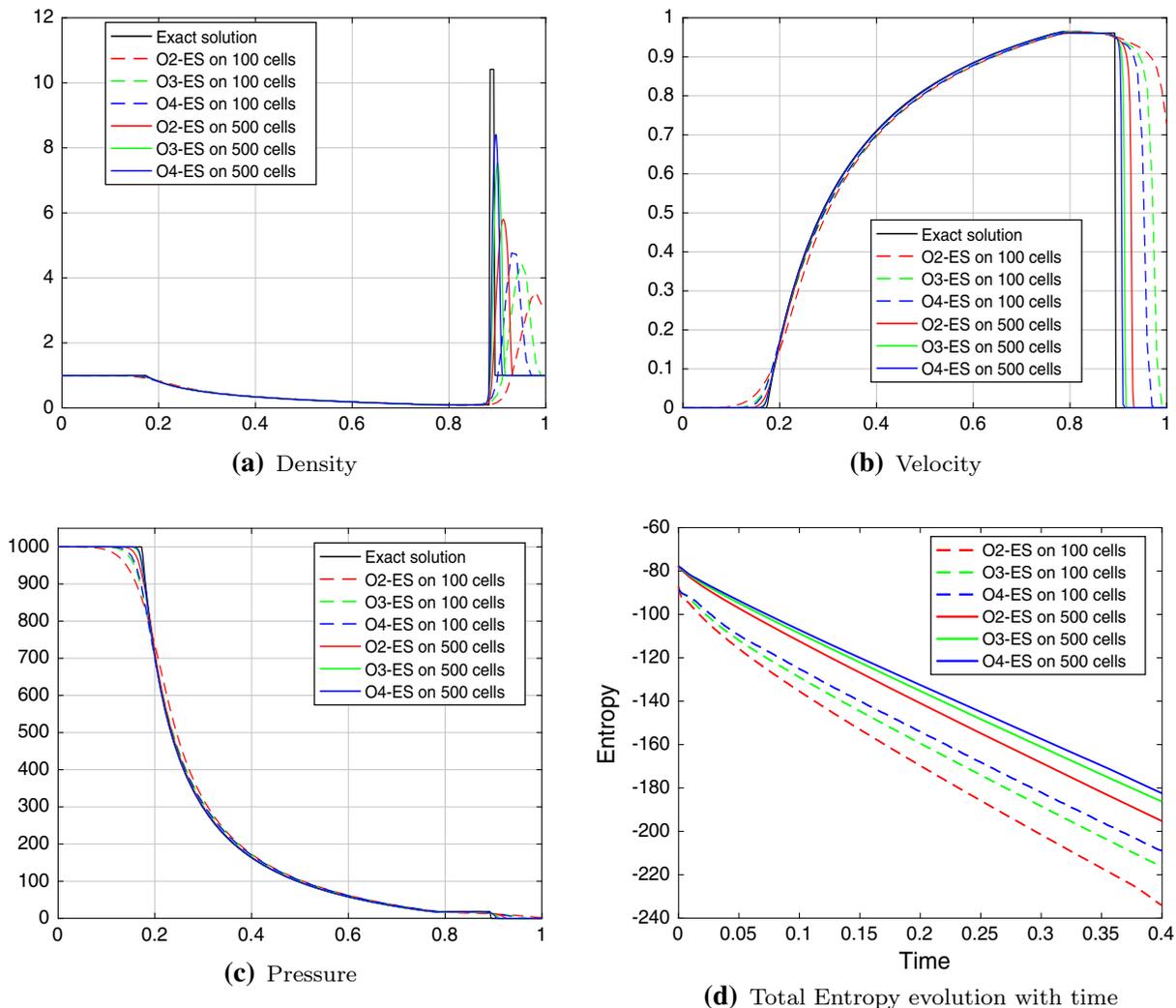


FIG. 3. Riemann problem 2: Plots of density, velocity, pressure and total entropy evolution at 100 and 500 cells for O2-ES, O3-ES and O4-ES schemes

results are presented in Fig. 4 at time $t = 0.4$. We use adiabatic index, $\gamma = \frac{5}{3}$. We proceed as in [24] to obtain the exact solution of Riemann Problem 3. Similar to test case above, at 100 cells, no scheme is able to capture the shock wave efficiently. However, at 500 cells, all the schemes resolve the shock wave with accuracy. For the entropy evolution, we observe that at 500 cells all the scheme have similar decay rate, whereas for the 100 cells, O3-ES and O4-ES are slightly better than the O2-ES.

4.1.6. Density perturbation test case. For this test case, we consider the test problem from [38]. We consider the computational domain of $[0, 1]$ with outflow boundary conditions. We assume adiabatic index to be $\gamma = \frac{5}{3}$. The initial conditions are given by,

$$(\rho, \mathbf{u}, p) = \begin{cases} (5, \mathbf{0}, 5) & \text{if } x < 0.5 \\ (2 + 0.3 \sin(50x), \mathbf{0}, 5) & \text{if } x > 0.5. \end{cases}$$

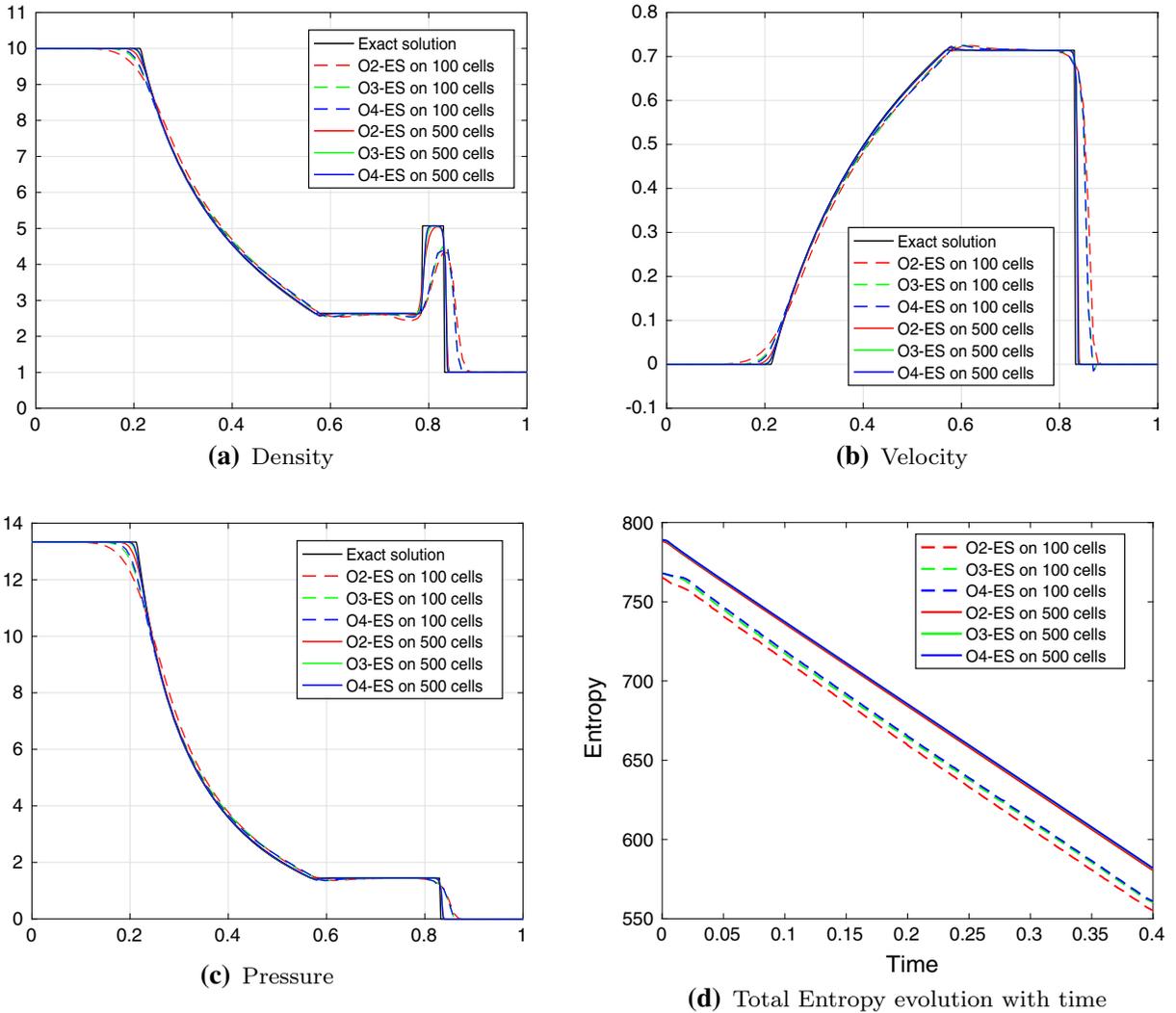


FIG. 4. Riemann problem 3: plots of density, velocity, pressure and total entropy evolution at 100 and 500 cells for O2-ES, O3-ES and O4-ES schemes

The solution is computed till time $t = 0.35$. The exact solution of the problem contains oscillatory density profile with both shocks and rarefaction waves. Hence, the problem will test accuracy of the schemes in capturing these oscillations and other waves.

We have plotted the numerical results for density, velocity and pressure in Fig. 5 for 100 and 500 cells using O2-ES, O3-ES, and O4-ES schemes. At coarse mesh of 100 cells, we again observe that the fourth-order scheme O4-ES is much more accurate than the third-order scheme O3-ES. The second-order scheme is the least accurate. For the fine mesh of 500 cells, O2-ES still fails to capture the peaks; however, O3-ES and O4-ES perform almost identically. This can be further observed from the entropy decay plot in Fig. 5d. We observe that the O3-ES and O4-ES schemes are much closer for 500 cells with O2-ES being more diffusive.

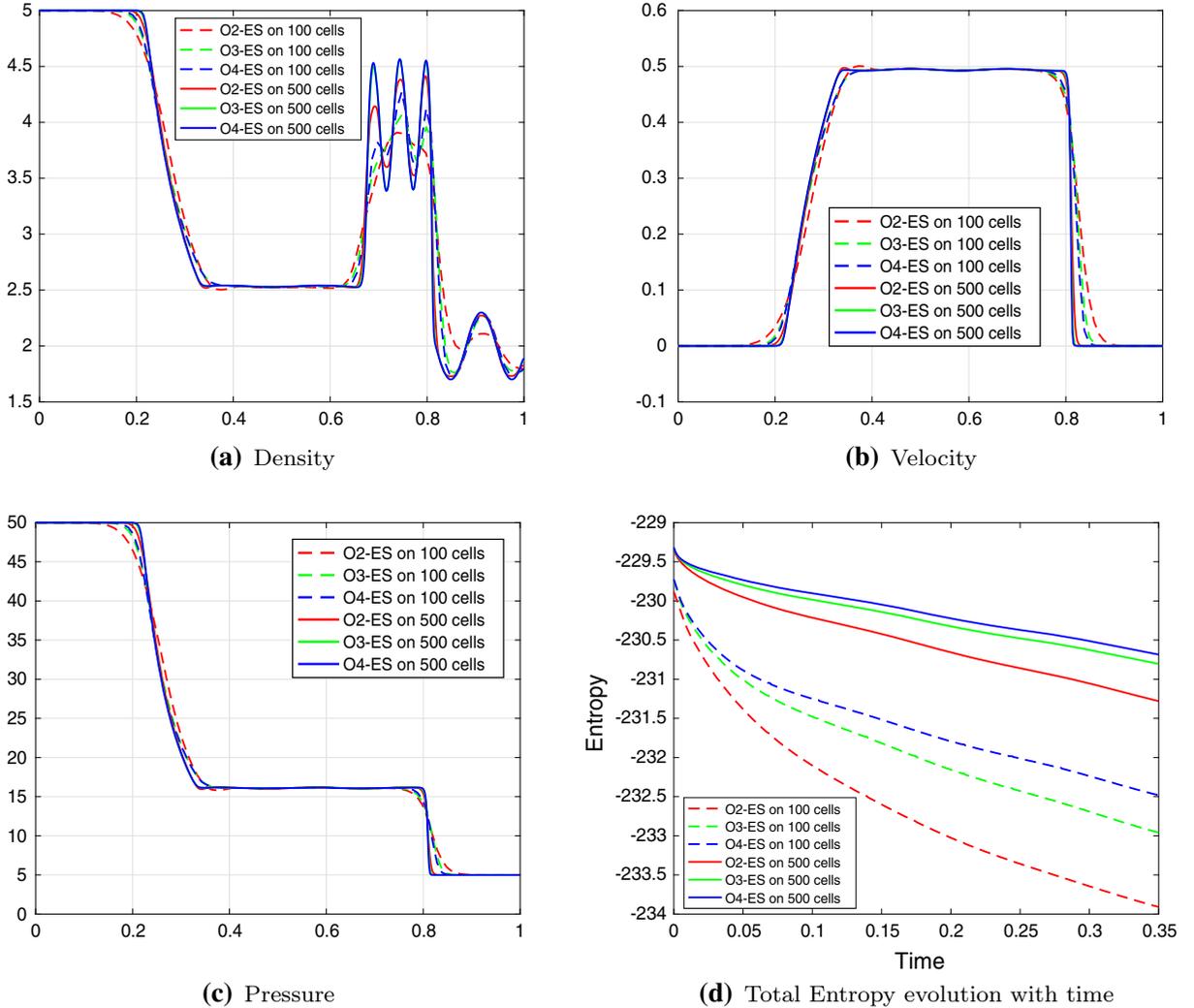


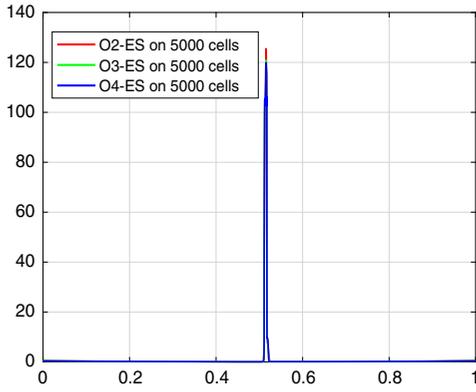
FIG. 5. Density perturbation test case: plots of density, velocity, pressure and total entropy evolution at 100 and 500 cells for O2-ES, O3-ES and O4-ES schemes

4.1.7. Blast waves test case. In this test case, we consider the blast wave interaction problem from [22]. The initial computational domain is considered to be $[0, 1]$ with outflow boundary conditions. The initial conditions in the domain are given by,

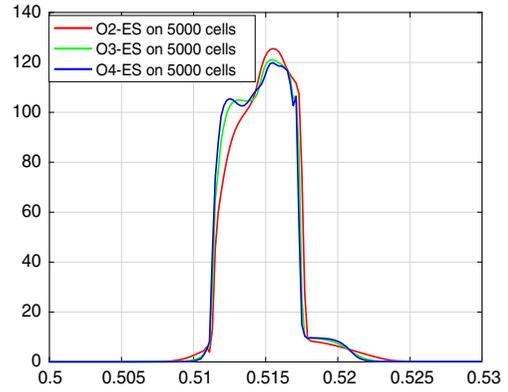
$$(\rho, \mathbf{u}, p) = \begin{cases} (1, \mathbf{0}, 1000) & \text{if } x < 0.1, \\ (1, \mathbf{0}, 0.01) & \text{if } 0.1 < x < 0.9, \\ (1, \mathbf{0}, 100) & \text{if } x > 0.9, \end{cases}$$

with adiabatic index $\gamma = 1.4$. We compute the solutions till time $t = 0.43$. As the test case is fairly extreme, to capture the waves, we use 5000 cells.

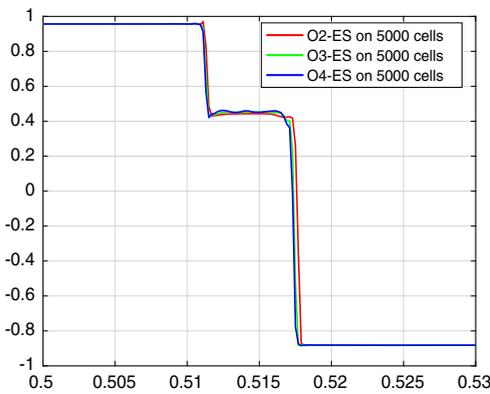
Numerical results for O2-ES, O3-ES and O4-ES schemes are plotted in Fig. 6. To show the results with more details, we plotted the variables in the interval $(0.50, 0.53)$. We again observe that O4-ES captures the waves more accurately, and O2-ES is most diffusive. We have also plotted the entropy evolution with time for these schemes.



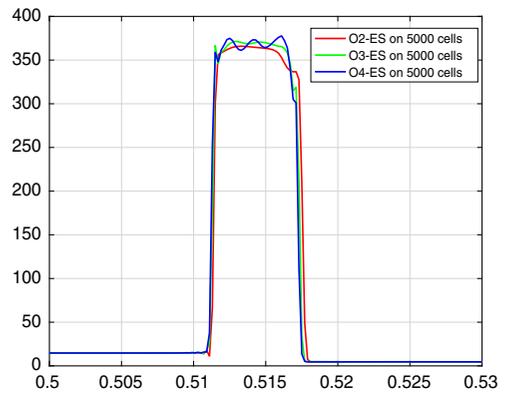
(a) Density



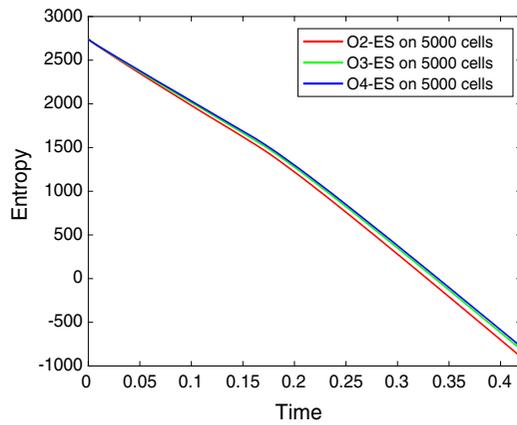
(b) Density in the interval (0.50, 0.53) at time $t=0.43$



(c) Velocity in the interval (0.50, 0.53) at time $t=0.43$



(d) Pressure in the interval (0.50, 0.53) at time $t=0.43$



(e) Total Entropy evolution with time

FIG. 6. Blast waves test case: plots of density, velocity, pressure and total entropy evolution with 5000 cells for O2-ES, O3-ES and O4-ES schemes

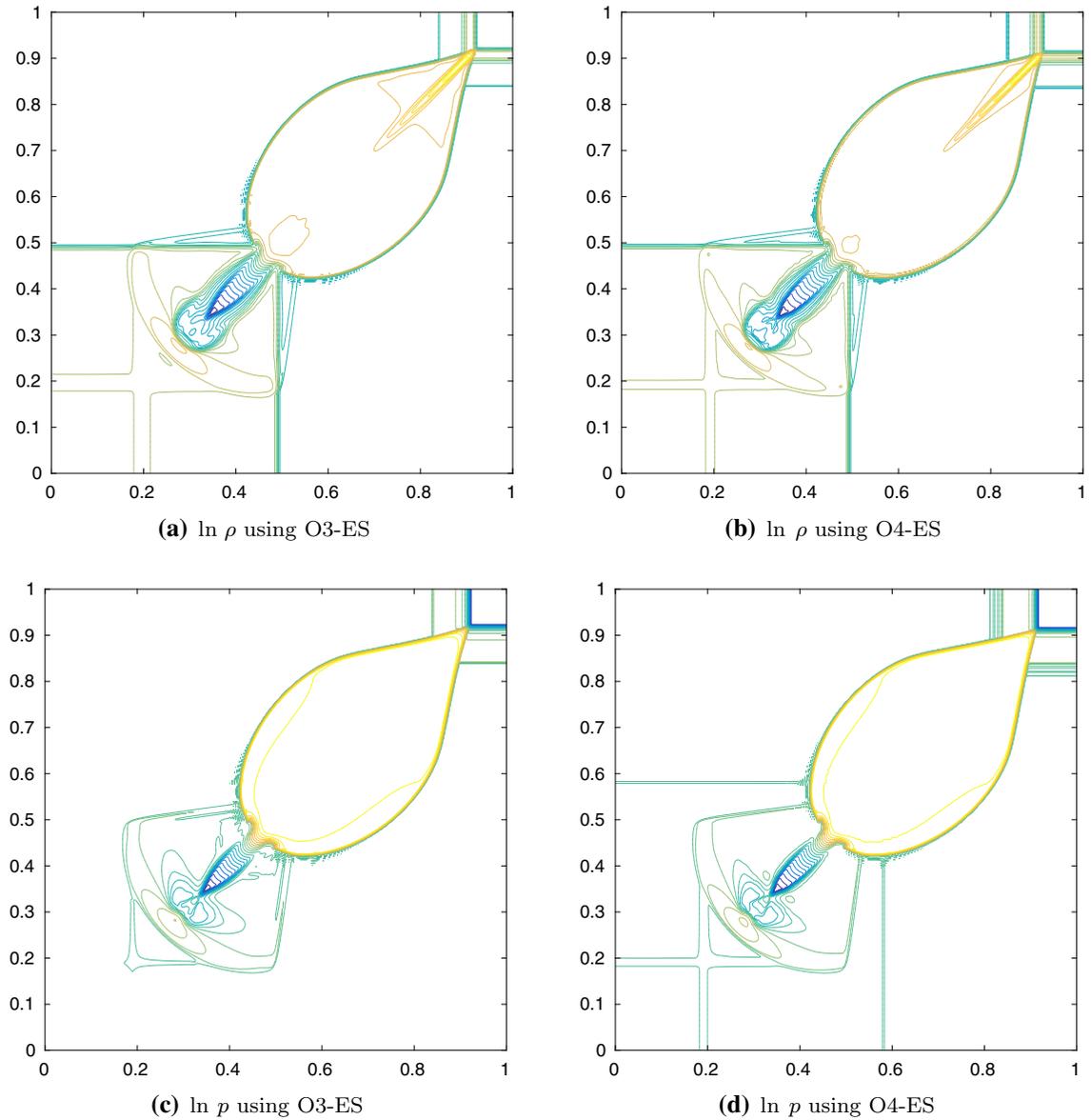


FIG. 7. Two-dimensional Riemann problem 1: plots for $\ln \rho$ and $\ln p$ for O3-ES and O4-ES using 25 contours

4.2. Two-dimensional test cases

For the two-dimensional test cases, we consider various two-dimensional Riemann problems. We assume a computational domain $[0, 1] \times [0, 1]$ which is divided into four parts with initial discontinuities along the lines $x = 0.5$ and $y = 0.5$. We use mesh of 400×400 cells and take $\gamma = \frac{5}{3}$. Outflow boundary conditions are used in all the test cases. To simplify the presentation, we present the numerical results for O3-ES and O4-ES schemes only, as O2-ES numerical results are more diffusion compare to O3-ES and O4-ES schemes.

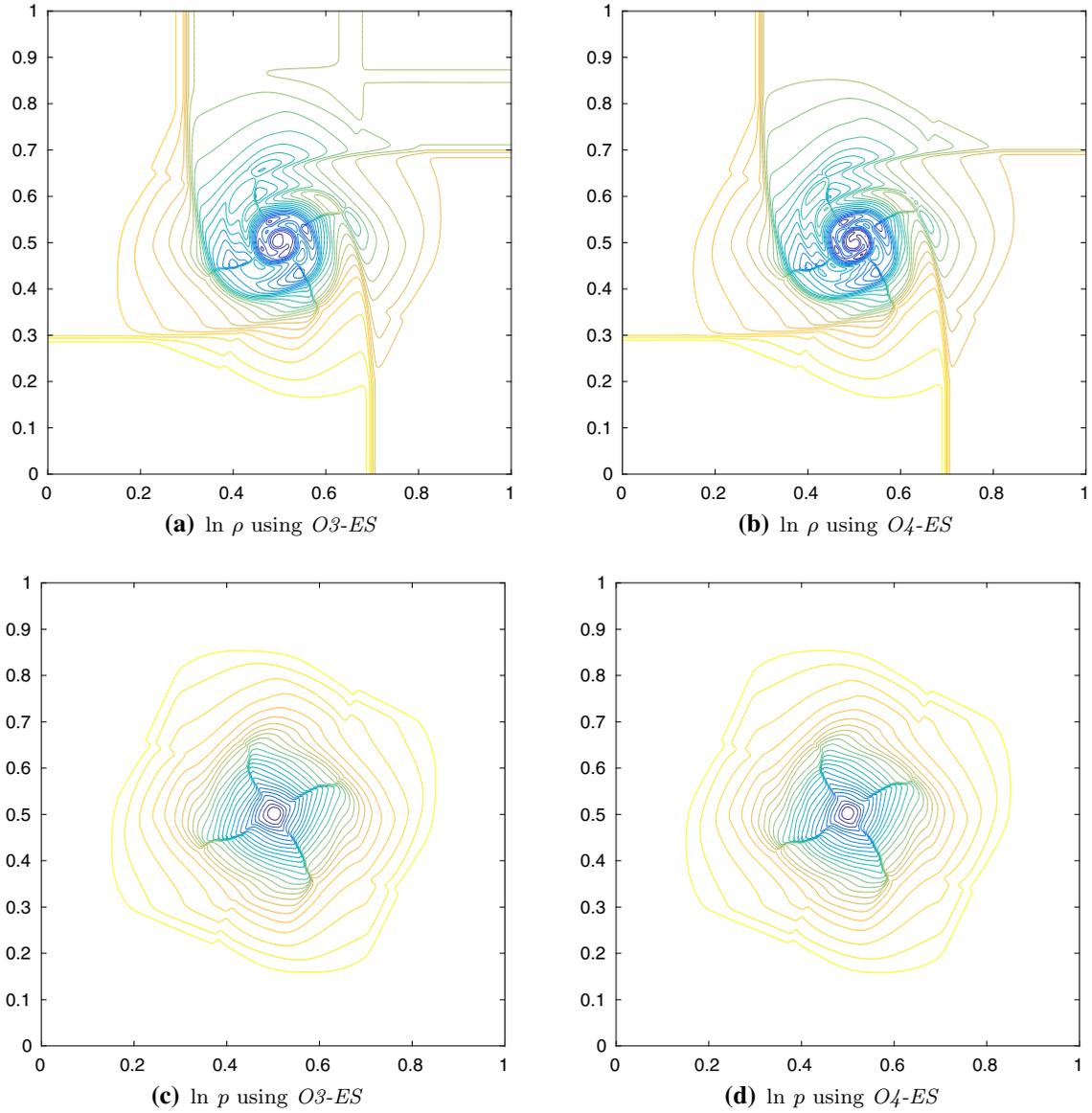


FIG. 8. Two-dimensional Riemann problem 2: plots for $\ln \rho$ and $\ln p$ for O3-ES and O4-ES using 25 contours

4.2.1. Two-dimensional Riemann problem 1. In this test case, we consider the Riemann problem from [38]. The domain is filled with four constant states given by,

$$(\rho, u_x, u_y, p) = \begin{cases} (0.1, 0, 0, 0.01), & \text{for } x, y > 0.5, \\ (0.1, 0.99, 0, 1.0), & \text{for } x < 0.5, y > 0.5, \\ (0.5, 0, 0, 1.0), & \text{for } x < 0.5, \\ (0.1, 0, 0.99, 1.0), & \text{for } x > 0.5, y < 0.5. \end{cases}$$

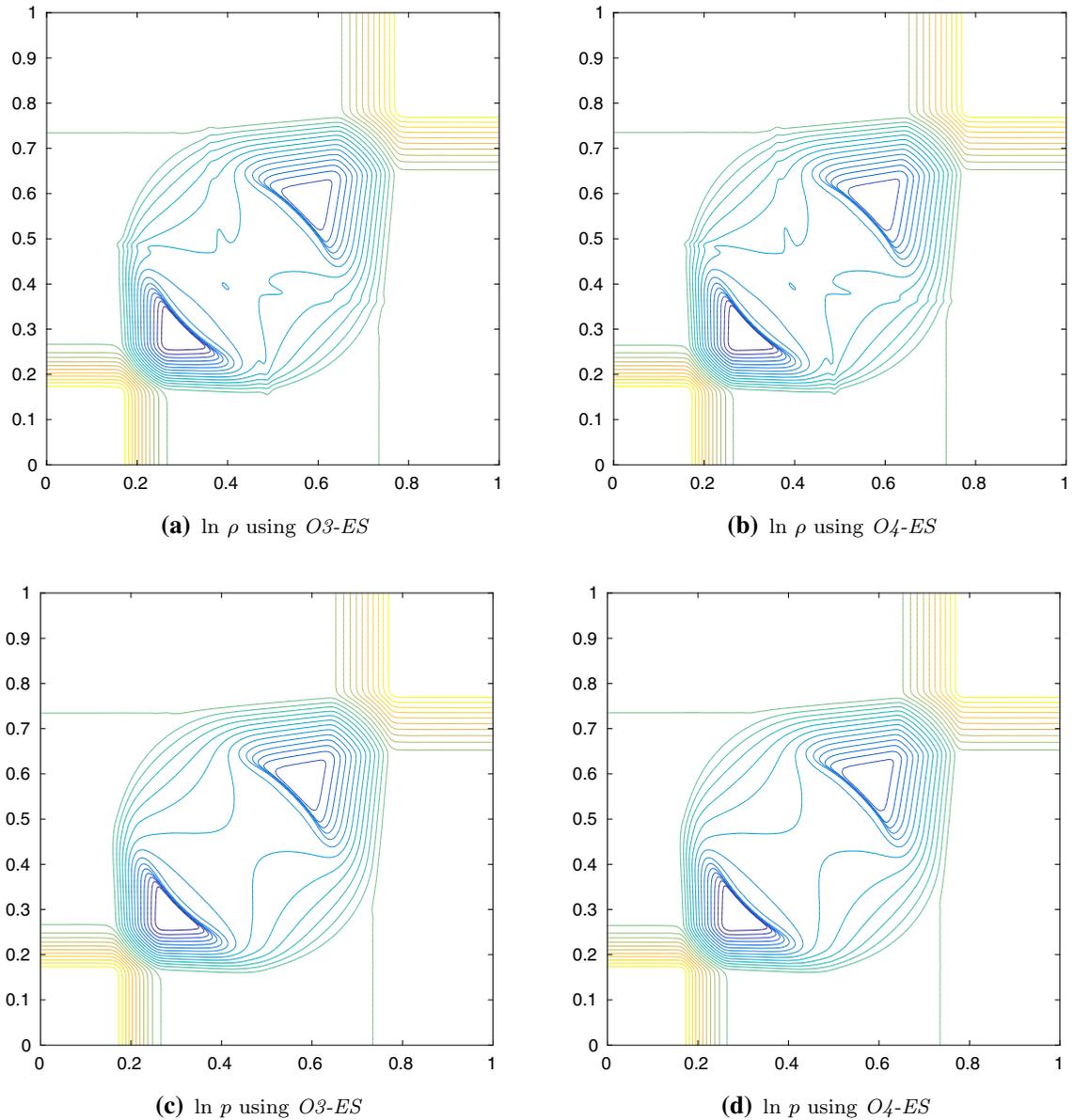


FIG. 9. Two-dimensional Riemann problem 3: plots for $\ln \rho$ and $\ln p$ for O3-ES and O4-ES using 25 contours

Numerical results for O3-ES and O4-ES schemes are presented in Fig. 7 at time $t = 0.4$. We have plotted 25 contours of $\ln \rho$ and $\ln p$. We note that both the schemes captures all the waves and have comparable accuracy with O4-ES being slightly more accurate.

4.2.2. Two-dimensional Riemann problem 2. For this test case, we consider a Riemann problem used in [28]. The four initial states are given by,

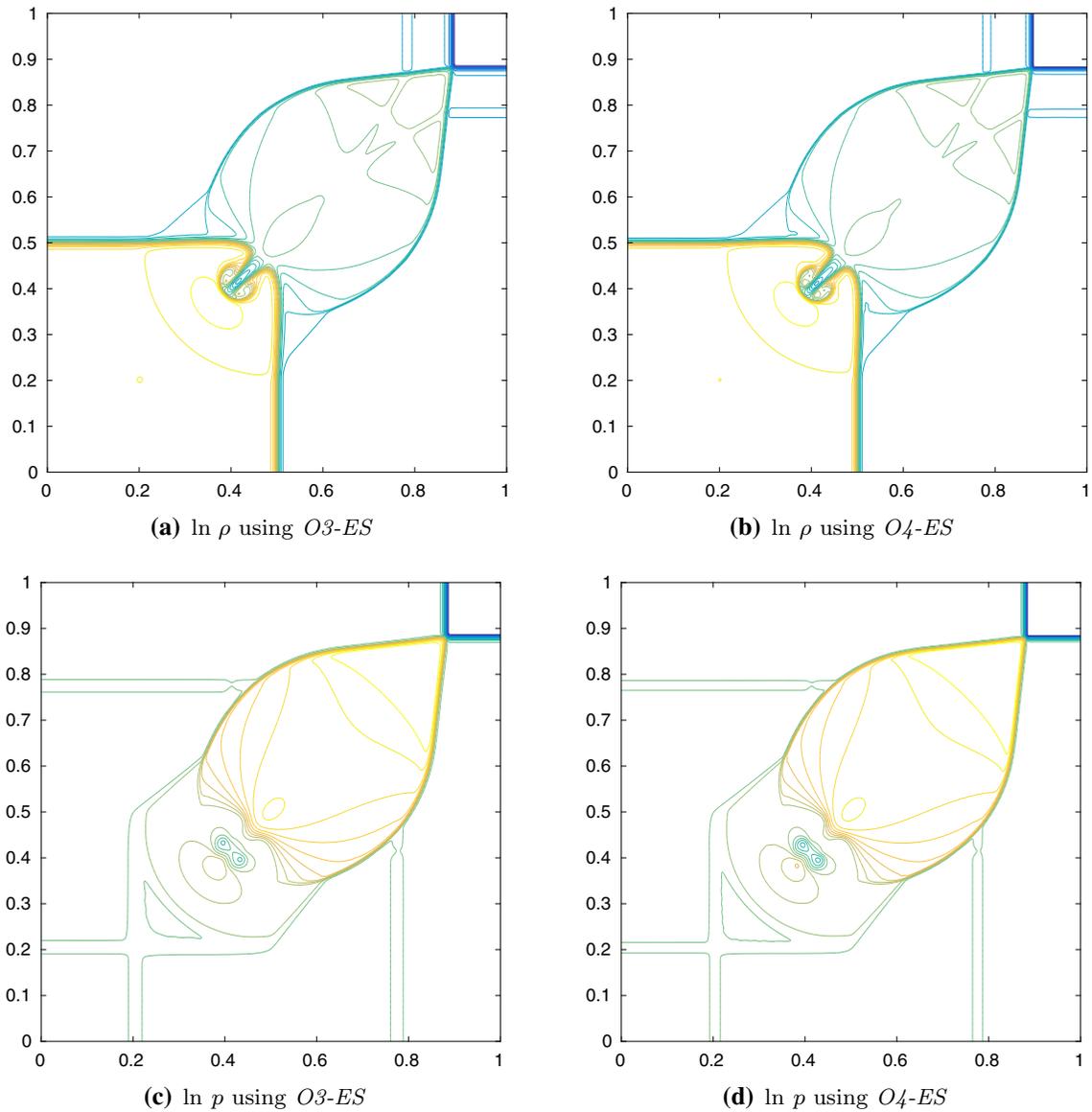


FIG. 10. Two-dimensional Riemann problem 4: plots for $\ln \rho$ and $\ln p$ for O3-ES and O4-ES using 25 contours

$$(\rho, \mathbf{u}, p) = \begin{cases} (0.5, 0.5, -0.5, 5), & \text{if } x, y > 0.5, \\ (1, 0.5, 0.5, 5), & \text{if } x < 0.5, y > 0.5, \\ (3, -0.5, 0.5, 5), & \text{if } x, y < 0.5, \\ (1.5, -0.5, -0.5, 5), & \text{if } x > 0.5, y < 0.5. \end{cases}$$

In Fig. 8, we have plotted $\ln \rho$ and $\ln p$ using 25 contours at $t = 0.4$ for O3-ES and O4-ES schemes. We again observe that both the schemes accurately capture all the waves in the vortex. Also, O4-ES produces more details compared to O3-ES.

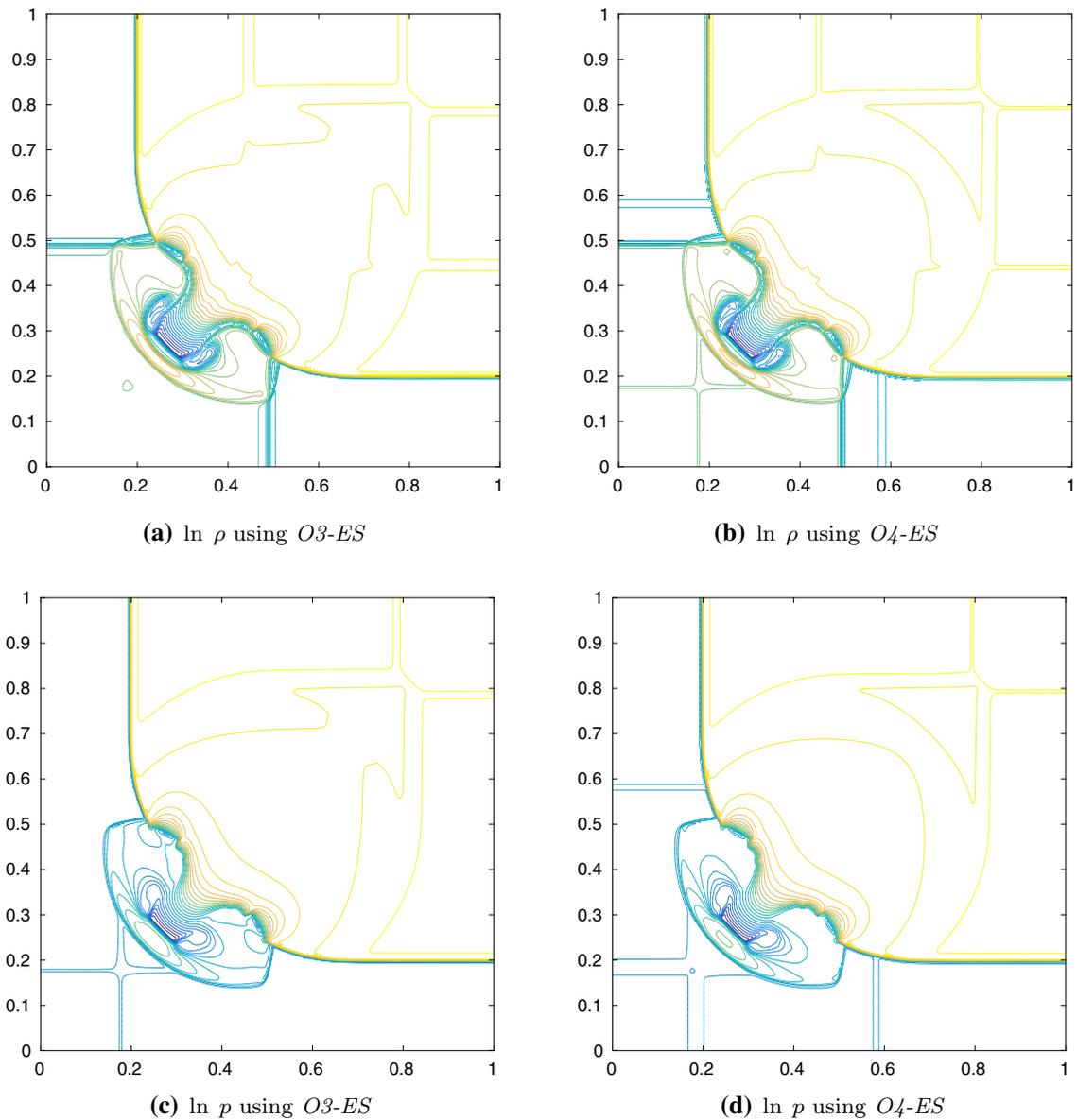


FIG. 11. Two-dimensional Riemann problem 5: plots for $\ln \rho$ and $\ln p$ for O3-ES and O4-ES using 25 contours

4.2.3. Two-dimensional Riemann problem 3. We consider another Riemann problem from [28]. The initial four states are given as follows:

$$(\rho, \mathbf{u}, p) = \begin{cases} (1, 0, 0, 1) & \text{if } x, y > 0.5, \\ (0.5771, -0.3529, 0, 0.4) & \text{if } x < 0.5, y > 0.5, \\ (1, -0.3529, -0.3529, 1) & \text{if } x, y < 0.5, \\ (0.5771, 0, -0.3529, 0.4) & \text{if } x > 0.5, y < 0.5. \end{cases}$$

The exact solution involves interactions of the planar rarefaction waves, which generates two symmetric shock waves. We present the numerical results using 25 contours for $\ln \rho$ and $\ln p$ at $t = 0.4$ in Fig. 9. We observe that both the schemes, O3-ES and O4-ES, capture both rarefaction and shock waves accurately and have similar performance.

4.2.4. Two-dimensional Riemann problem 4. We consider another test case from [28], for which initial states are given by,

$$(\rho, \mathbf{u}, p) = \begin{cases} (0.035145216124503, 0, 0, 0.162931056509027), & \text{if } x, y > 0.5, \\ (0.1, 0.7, 0, 1), & \text{if } x < 0.5, y > 0.5, \\ (0.5, 0, 0, 1), & \text{if } x, y < 0.5, \\ (0.1, 0, 0.7, 1), & \text{if } x > 0.5, y < 0.5. \end{cases}$$

In this test case, initial membranes break and two contact discontinuities appear on the left and bottom of the domain. Finally, we obtain two curved front shocks.

We have plotted $\ln \rho$ and $\ln p$ using 25 contours for O3-ES and O4-ES schemes at time $t = 0.4$ in Fig. 10. Both of the schemes resolve curved shocks and contact waves accurately.

4.2.5. Two-dimensional Riemann problem 5. In this test case from [37], the initial four states are given by,

$$(\rho, v, p) = \begin{cases} (0.1, 0, 0, 20) & \text{if } x, y > 0.5, \\ (0.00414329639576, 0.99, 0, 0.05) & \text{if } x < 0.5, y > 0.5, \\ (0.01, 0, 0, 0.05) & \text{if } x, y < 0.5, \\ (0.00414329639576, 0, 0.99, 0.05) & \text{if } x > 0.5, y < 0.5. \end{cases}$$

Here, the four initial discontinuities interact to give the distortion of initial shocks with a left-bottom going mushroom cloud, starting from the centre. The numerical results for O3-ES and O4-ES at time $t = 0.4$ are presented in Fig. 11. We use 25 contours to plot $\ln \rho$ and $\ln p$. We observe that both schemes resolve cloud accurately.

5. Conclusion

In this work, we consider the system of relativistic hydrodynamics. First, we discuss the entropy estimate for the continuous model. We then design second-, third- and fourth-order entropy-stable finite difference schemes. The schemes involve combining higher-order entropy conservative schemes with higher-order entropy diffusion operator. The design of higher-order diffusion operator is based on the entropic scaled eigenvectors and sign-preserving property of minmod and ENO reconstruction process. The resulting schemes are then tested on a variety of test cases, involving one- and two-dimensional Riemann problems. The schemes are shown to be stable and accurate in capturing various waves. We also quantify the total entropy decay for various test cases. Furthermore, using the component presented here, we can design arbitrary higher-order entropy-stable schemes for relativistic fluid flows.

Acknowledgements

We want to thank Prof. Praveen Chandrashekar (TIFR-CAM, Bengaluru, India) and Prof. Dinshaw Balsara (Notre-Dame University, IN, USA) for several useful discussions. We also want to thank Prof.

José María Martí (Universidad de Valencia, Spain) for providing us the codes for the exact solutions of the Riemann problems.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Appendix A. Computation of primitive variables

The RHD equations involve the conserved quantities, D , \mathbf{m} and E ; but to solve the RHD equations numerically, we need to know the explicit values of the primitive variables, ρ , \mathbf{u} and p . The primitive variables can be extracted by inverting the RHD equations (1). For the ideal equation of state (4), [30] showed that (using Eq. (1)) the variable $u(=|\mathbf{u}|)$ satisfies

$$u^4 + a_3 u^3 + a_2 u^2 + a_1 u + a_0 = 0, \quad (25)$$

where

$$a_3 = -\frac{2\gamma(\gamma-1)ME}{(\gamma-1)^2(M^2+D^2)}, \quad a_2 = \frac{(\gamma^2 E^2 + 2(\gamma-1)M^2 - (\gamma-1)^2 D^2)}{(\gamma-1)^2(M^2+D^2)},$$

$$a_1 = \frac{-2\gamma ME}{(\gamma-1)^2(M^2+D^2)}, \quad a_0 = \frac{M^2}{(\gamma-1)^2(M^2+D^2)},$$

with $M = (m_x^2 + m_y^2 + m_z^2)^{1/2}$. Equation (25) could either be solved using the analytical methods or using an approximate root solver. In [30], authors have suggested the following expressions for the lower bound u_{lb} and upper bound u_{ub} :

$$u_{\text{lb}} = \frac{1}{2M(\gamma-1)}(\gamma E - \sqrt{\gamma^2 E^2 - 4(\gamma-1)M^2})$$

and

$$u_{\text{ub}} = \min\left(1, \frac{M}{E} + \delta\right),$$

where $\delta \approx 10^{-6}$. Furthermore, to find the root of Eq. (25) using the *Newton-Raphson* method, [30] suggested the initial guess u_o as,

$$u_o = \frac{1}{2}(u_{\text{lb}} + u_{\text{ub}}) + z,$$

where $z = \frac{1}{2}\left(1 - \frac{D}{E}\right)(u_{\text{lb}} - u_{\text{ub}})$ if $u_{\text{lb}} > 10^{-9}$ and $z = 0$ otherwise. We then obtain u_o between u_{lb} and u_{ub} . Also, an accuracy of nine digits can be achieved in at most five iterations of the *Newton-Raphson* root solver.

For the analytic solution, we can use the direct formulation of the roots of the quartic equations. The general form of analytical roots is available in [1]. We note that Eq. (25) has two complex roots and two real roots. Among two real roots, only one root lies between the lower and upper bounds suggested by [30], which is given by,

$$u = \frac{-B + \sqrt{B^2 - 4C}}{2},$$

where,

$$B = \frac{a_3}{2} + \sqrt{\frac{a_3^2}{4} + z_1 - a_2}, \quad C = \frac{z_1}{2} - \sqrt{\left(\frac{z_1}{2}\right)^2 - a_0}, \quad z_1 = (s_1 + s_2) - \frac{b_2}{3}, \quad s_1 = (r + \sqrt{q^3 + r^2})^{1/3},$$

$$s_2 = (r - \sqrt{q^3 + r^2})^{1/3}, \quad q = \frac{b_1}{3} - \frac{b_2^2}{9}, \quad r = \frac{1}{6}(b_1 b_2 - 3b_0) - \frac{1}{27}b_2^3, \quad b_0 = -(a_1^2 + a_0 a_3^2 - 4a_0 a_2),$$

$$b_1 = a_1 a_3 - 4a_0 \quad \text{and} \quad b_2 = -a_2.$$

Once we have the value of u , we can find all the primitive variables, i.e. ρ , u_i , p , using the expressions

$$\begin{aligned}\rho &= \frac{D}{\Gamma}, \\ u_x &= \frac{m_x}{M}u, \quad u_y = \frac{m_y}{M}u, \quad u_z = \frac{m_z}{M}u, \\ p &= (\gamma - 1)(E - m_x u_x - m_y u_y - m_z u_z - \rho).\end{aligned}$$

In this article, we use numerical approach to find the primitive variables from the conservative variables.

Appendix B. Entropy scaled right eigenvectors

In this section, we present the entropy scaled right eigenvectors. First, we consider the case of x -direction right eigenvectors. We rewrite the RHD system (3),

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}^x \frac{\partial \mathbf{U}}{\partial x} = 0, \quad \text{where } \mathbf{A}^x = \frac{\partial \mathbf{f}^x}{\partial \mathbf{U}}.$$

Following [29], the eigenvalues of the Jacobian matrix \mathbf{A}^x are given by:

$$\lambda_1^x = \frac{(1 - c^2)u_x - (c/\Gamma)\sqrt{Q^x}}{1 - c^2 \mathbf{u}^2}, \quad \lambda_2^x = \lambda_3^x = u_x \quad \text{and} \quad \lambda_4^x = \frac{(1 - c^2)u_x + (c/\Gamma)\sqrt{Q^x}}{1 - c^2 \mathbf{u}^2}.$$

Here, $Q^x = 1 - u_x^2 - c^2 u_y^2$ and c is the sound speed given by the expression

$$c^2 = -\frac{\rho}{nh} \frac{\partial h}{\partial \rho},$$

where $n = k - 1$ with $k = \frac{\gamma}{\gamma - 1}$, for all $\mathbf{U} \in \Omega$. Note that $Q^x = 1 - u_x^2 - c^2 u_y^2 > 0$; therefore, all the eigenvalues are real. A straightforward calculation results in the following expressions for the eigenvectors in primitive variables:

$$e_1 = \begin{pmatrix} \frac{1}{c^2 h} \\ \frac{-\sqrt{Q^x}}{ch\Gamma\rho} \\ \frac{(c - \Gamma\sqrt{Q^x}u_x)u_y}{ch\Gamma^2\rho(u_x^2 - 1)} \\ 1 \end{pmatrix}, \quad e_2 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad e_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \quad \text{and} \quad e_4 = \begin{pmatrix} \frac{1}{c^2 h} \\ \frac{\sqrt{Q^x}}{ch\Gamma\rho} \\ \frac{(c + \Gamma\sqrt{Q^x}u_x)u_y}{ch\Gamma^2\rho(u_x^2 - 1)} \\ 1 \end{pmatrix}.$$

The eigenvector matrix for the conservative system, $\tilde{\mathbf{R}}^x$, can be derived using,

$$\tilde{\mathbf{R}}^x = \frac{\partial \mathbf{U}}{\partial \mathbf{w}} \tilde{\mathbf{R}}_w^x, \quad (26)$$

where $\tilde{\mathbf{R}}_w^x = (e_1, e_2, e_3, e_4)$, i.e. the matrix of right eigenvectors for the primitive system. To obtain the scaled right eigenvector matrix \mathbf{R}^x , we need to find a scaling matrix \mathbf{T}^x such that the scaled right eigenvector matrix, $\mathbf{R}^x = \tilde{\mathbf{R}}^x \mathbf{T}^x$, satisfies

$$\frac{\partial \mathbf{U}}{\partial \mathbf{v}} = \mathbf{R}^x \mathbf{R}^{x\top}. \quad (27)$$

We first derive the expression of the entropy scaled right eigenvector for the primitive variable system, \mathbf{R}_w^x . Following the procedure of [4, 31], the scaling matrix \mathbf{T}^x is the square root of the matrix \mathbf{Y}^x , where

$$\mathbf{Y}^x = \tilde{\mathbf{R}}_w^x{}^{-1} \frac{\partial \mathbf{w}}{\partial \mathbf{v}} \frac{\partial \mathbf{U}}{\partial \mathbf{w}}{}^{-\top} \tilde{\mathbf{R}}_w^x{}^{-\top},$$

which on a long calculation results in,

$$\mathbf{Y}^x = \begin{pmatrix} \frac{c^2 h p \left(1 + \frac{c u_x}{\Gamma \sqrt{Q^x}}\right)}{2\Gamma(1-c^2 u^2)} & 0 & 0 & 0 \\ 0 & \frac{\rho}{k\Gamma} & 0 & 0 \\ 0 & 0 & \frac{p}{h\Gamma^5 \rho^2 (1-u_x^2)} & 0 \\ 0 & 0 & 0 & \frac{c^2 h p \left(1 - \frac{c u_x}{\Gamma \sqrt{Q^x}}\right)}{2\Gamma(1-c^2 u^2)} \end{pmatrix}.$$

Consequently,

$$\mathbf{T}^x = \sqrt{\mathbf{Y}^x} = \begin{pmatrix} \sqrt{\frac{c^2 h p \left(1 + \frac{c u_x}{\Gamma \sqrt{Q^x}}\right)}{2\Gamma(1-c^2 u^2)}} & 0 & 0 & 0 \\ 0 & \sqrt{\frac{\rho}{k\Gamma}} & 0 & 0 \\ 0 & 0 & \sqrt{\frac{p}{h\Gamma^5 \rho^2 (1-u_x^2)}} & 0 \\ 0 & 0 & 0 & \sqrt{\frac{c^2 h p \left(1 - \frac{c u_x}{\Gamma \sqrt{Q^x}}\right)}{2\Gamma(1-c^2 u^2)}} \end{pmatrix}.$$

So, the entropy scaled right eigenvector matrix for the primitive system is given by

$$\begin{aligned} \mathbf{R}_w^x &= \tilde{\mathbf{R}}_w^x \mathbf{T}^x \\ &= \begin{pmatrix} \frac{A_1^x}{c^2 h} & A_2^x & 0 & \frac{A_4^x}{c^2 h} \\ -\frac{A_1^x \sqrt{Q^x}}{ch\Gamma\rho} & 0 & 0 & \frac{A_4^x \sqrt{Q^x}}{ch\Gamma\rho} \\ \frac{A_1^x (c - \Gamma \sqrt{Q^x} u_x) u_y}{ch\Gamma^2 \rho (u_x^2 - 1)} & 0 & A_3^x & \frac{A_4^x (c + \Gamma \sqrt{Q^x} u_x) u_y}{ch\Gamma^2 \rho (u_x^2 - 1)} \\ A_1^x & 0 & 0 & A_4^x \end{pmatrix}, \end{aligned}$$

where

$$A_1^x = \sqrt{\frac{c^2 h p \left(1 + \frac{c u_x}{\Gamma \sqrt{Q^x}}\right)}{2\Gamma(1-c^2 u^2)}}, \quad A_2^x = \sqrt{\frac{\rho}{k\Gamma}}, \quad A_3^x = \sqrt{\frac{p}{h\Gamma^5 \rho^2 (1-u_x^2)}} \quad \text{and} \quad A_4^x = \sqrt{\frac{c^2 h p \left(1 - \frac{c u_x}{\Gamma \sqrt{Q^x}}\right)}{2\Gamma(1-c^2 u^2)}}.$$

Using Eq. (26), we can calculate the entropy scaled right eigenvectors for the conservative system satisfying (27).

Remark B.1. Proceeding similarly in the y -direction, we get the eigenvalues of the Jacobian matrix as

$$\lambda_1^y = \frac{(1-c^2)u_y - (c/\Gamma)\sqrt{Q^y}}{1-c^2 u^2}, \quad \lambda_2^y = u_y, \quad \lambda_3^y = u_y \quad \text{and} \quad \lambda_4^y = \frac{(1-c^2)u_y + (c/\Gamma)\sqrt{Q^y}}{1-c^2 u^2},$$

and the right eigenvector matrix, for the system in primitive variables, as

$$\tilde{\mathbf{R}}_w^y = \begin{pmatrix} \frac{1}{c^2 h} & 1 & 0 & \frac{1}{c^2 h} \\ \frac{(c - \Gamma \sqrt{Q^y} u_y) u_x}{ch\Gamma^2 \rho (u_y^2 - 1)} & 0 & 1 & \frac{(c + \Gamma \sqrt{Q^y} u_y) u_x}{ch\Gamma^2 \rho (u_y^2 - 1)} \\ \frac{-\sqrt{Q^y}}{ch\Gamma\rho} & 0 & 0 & \frac{\sqrt{Q^y}}{ch\Gamma\rho} \\ 1 & 0 & 0 & 1 \end{pmatrix}$$

where $Q^y = 1 - u_y^2 - c^2 u_x^2$. Consequently, the entropy scaled right eigenvector matrix is given by

$$\mathbf{R}_w^y = \begin{pmatrix} \frac{A_1^y}{c^2 h} & A_2^y & 0 & \frac{A_4^y}{c^2 h} \\ \frac{A_1^y (c - \Gamma \sqrt{Q^y} u_y) u_x}{ch\Gamma^2 \rho (u_y^2 - 1)} & 0 & A_3^y & \frac{A_4^y (c + \Gamma \sqrt{Q^y} u_y) u_x}{ch\Gamma^2 \rho (u_y^2 - 1)} \\ -\frac{A_1^y \sqrt{Q^y}}{ch\Gamma\rho} & 0 & 0 & \frac{A_4^y \sqrt{Q^y}}{ch\Gamma\rho} \\ A_1^y & 0 & 0 & A_4^y \end{pmatrix},$$

where

$$A_1^y = \sqrt{\frac{c^2 h \frac{\rho}{\beta} \left(1 + \frac{cu_y}{\Gamma \sqrt{Q^y}}\right)}{2\Gamma(1 - c^2 \mathbf{u}^2)}}, \quad A_2^y = \sqrt{\frac{\rho}{k\Gamma}}, \quad A_3^y = \sqrt{\frac{\rho}{\beta h \Gamma^5 \rho^2 (1 - u_y^2)}}, \quad \text{and} \quad A_4^y = \sqrt{\frac{c^2 h \frac{\rho}{\beta} \left(1 - \frac{cu_y}{\Gamma \sqrt{Q^y}}\right)}{2\Gamma(1 - c^2 \mathbf{u}^2)}}.$$

References

- [1] Abramowitz, M., Stegun, I.A.: Handbook of Mathematical Functions: With Formulas, Graphs, and Mathematical Tables. Dovers Publications, Inc., New York (1972)
- [2] Aloy, M.A., Ibáñez, J.M., Martí, J.M., Müller, E.: GENESIS: a high-resolution code for three-dimensional relativistic hydrodynamics. *Astrophys. J. Suppl. Ser.* **122**(1), 151–166 (1999)
- [3] Anile, A.M.: Relativistic Fluids and Magneto-Fluids: With Applications in Astrophysics and Plasma Physics. Cambridge University Press, Cambridge (1989)
- [4] Barth, T.J.: Numerical methods for gas-dynamics systems on unstructured meshes. In: Kroner, D., Ohlberger, M., Rohde, C. (eds.) *An Introduction to Recent Developments in Theory and Numerics of Conservation Laws. Lecture Notes in Computational Science*, vol. 5, pp. 195–285. Springer, Berlin (1999)
- [5] Begelman, M.C., Blandford, R.D., Rees, M.J.: Theory of extragalactic radio sources. *Rev. Mod. Phys.* **56**(2), 255–351 (1984)
- [6] Boettcher, M., Harris, D.E., Krawczynski, H.: Relativistic Jets from Active Galactic Nuclei. Wiley, Hoboken (2012)
- [7] Centrella, J., Wilson, J.R.: Planar numerical cosmology. II. The difference equations and numerical tests. *Astrophys. J. Suppl. Ser.* **54**(2), 229–249 (1984)
- [8] Chandrasekhar, P.: Kinetic energy preserving and entropy stable finite volume schemes for compressible Euler and Navier–Stokes equations. *Commun. Comput. Phys.* **14**(5), 1252–1286 (2013)
- [9] Chiodaroli, E., Lellis, C.D., Kreml, O.: Global ill-posedness of the isentropic system of gas dynamics. *Commun. Pure Appl. Math.* **68**(7), 1157–1190 (2015)
- [10] Choi, E., Ryu, D.: Numerical relativistic hydrodynamics based on the total variation diminishing scheme. *New Astron.* **11**(2), 116–129 (2005)
- [11] Falle, S.A.E.G., Komissarov, S.S.: An upwind numerical scheme for relativistic hydrodynamics with a general equation of state. *Mon. Not. R. Astron. Soc.* **278**(2), 586–602 (1996)
- [12] Fjordholm, U.S., Mishra, S., Tadmor, E.: Arbitrarily high-order accurate entropy stable essentially nonoscillatory schemes for systems of conservation laws. *SIAM J. Numer. Anal.* **50**(2), 544–573 (2012)
- [13] Fjordholm, U.S., Mishra, S., Tadmor, E.: ENO reconstruction and ENO interpolation are stable. *Found. Comput. Math.* **13**(2), 139–159 (2013)
- [14] Font, J.A.: Numerical hydrodynamics and magnetohydrodynamics in general relativity. *Living Rev. Relativ.* **11**(1), 7 (2008)
- [15] Godlewski, E., Raviart, P.-A.: Hyperbolic Systems of Conservation Laws. Ellipses, Paris (1991)
- [16] Gottlieb, S., Shu, C.-W., Tadmor, E.: Strong stability-preserving high-order time discretization methods. *SIAM Rev.* **43**(1), 89–112 (2001)
- [17] Harten, A.: High resolution schemes for hyperbolic conservation laws. *J. Comput. Phys.* **49**(3), 357–393 (1983)
- [18] Ismail, F., Roe, P.L.: Affordable, entropy-consistent euler flux functions II: entropy production at shocks. *J. Comput. Phys.* **228**(15), 5410–5436 (2009)
- [19] Jiang, G.-S., Shu, C.-W.: Efficient implementation of weighted ENO schemes. *J. Comput. Phys.* **126**(1), 202–228 (1996)
- [20] Landau, L.D., Lifschitz, E.M.: Fluid Mechanics. Chapter XV - Relativistic Fluid Dynamics, 2nd edn. Pergamon, New York (1987)
- [21] LeFloch, P.G., Mercier, J.M., Rohde, C.: Fully discrete entropy conservative schemes of arbitrary order. *SIAM J. Numer. Anal.* **40**(5), 1968–1992 (2002)
- [22] Martí, J.M., Müller, E.: Extension of the piecewise parabolic method to one-dimensional relativistic hydrodynamics. *J. Comput. Phys.* **123**(1), 1–14 (1996)
- [23] Martí, M.J., Müller, E.: Grid-based methods in relativistic hydrodynamics and magnetohydrodynamics. *Living Rev. Comput. Astrophys.* **1**(1), 3 (2015)
- [24] Martí, M.J., Müller, E.: Numerical hydrodynamics in special relativity. *Living Rev. Relativ.* **6**(1), 7 (2003)
- [25] Mignone, A., Bodo, G.: An HLLC Riemann solver for relativistic flows I. Hydrodynamics. *Mon. Not. R. Astron. Soc.* **364**(1), 126–136 (2005)
- [26] Mignone, A., Plewa, T., Bodo, G.: The piecewise parabolic method for multidimensional relativistic fluid dynamics. *Astrophys. J. Suppl. Ser.* **160**(1), 199–219 (2005)
- [27] Mirabel, I.F., Rodríguez, L.F.: Sources of relativistic jets in the galaxy. *Ann. Rev. Astron. Astrophys.* **37**, 409–443 (1999)

- [28] Rosa, J.N.D.L., Munz, C.-D.: XTROEM-FV: a new code for computational astrophysics based on very high order finite-volume methods-II. Relativistic hydro- and magnetohydrodynamics. *Mon. Not. R. Astron. Soc.* **460**(1), 535–559 (2016)
- [29] Ryu, D., Chattopadhyay, I., Choi, E.: Equation of state in numerical relativistic hydrodynamics. *Astrophys. J. Suppl. Ser.* **166**(1), 410–420 (2006)
- [30] Schneider, V., Katscher, U., Rischke, D.H., Waldhauser, B., Maruhn, J.A., Munz, C.-D.: New algorithms for ultra-relativistic numerical hydrodynamics. *J. Comput. Phys.* **105**(1), 92–107 (1993)
- [31] Sen, C., Kumar, H.: Entropy stable schemes for ten-moment Gaussian closure equations. *J. Sci. Comput.* **75**(2), 1128–1155 (2018)
- [32] Sweby, P.K.: High-resolution schemes using flux limiters for hyperbolic conservation laws. *SIAM J. Numer. Anal.* **21**(5), 995–1011 (1984)
- [33] Synge, J.L.: *Relativity: The Special Theory*. North-Holland Pub. Co., Amsterdam (1956)
- [34] Tadmor, E.: The numerical viscosity of entropy stable schemes for systems of conservation laws. I. *Math. Comput.* **49**(179), 91–103 (1987)
- [35] Wilson, J.R.: A numerical method for relativistic hydrodynamics. In: Smarr, L.L. (ed.) *Sources of Gravitational Radiation*. Proceedings of the Battelle Seattle Workshop, July 24–August 4, 1978, pp. 423–445. Cambridge University Press, Cambridge (1979)
- [36] Wu, K., Tang, H.Z.: High-order accurate physical-constraints-preserving finite difference WENO schemes for special relativistic hydrodynamics. *J. Comput. Phys.* **298**, 539–564 (2015)
- [37] Wu, K., Tang, H.Z.: Physical-constraints-preserving central discontinuous Galerkin methods for special relativistic hydrodynamics with a general equation of state. *Astrophys. J. Suppl. Ser.* **228**(3), 23 (2016)
- [38] Zanna, L.D., Bucciantini, N.: An efficient shock-capturing central-type scheme for multidimensional relativistic flows I. *Hydrodynamics. Ann. Rev. Astron. Astrophys.* **390**(3), 1177–1186 (2002)
- [39] Zensus, J.A.: Parsec-scale jets in extragalactic radio sources. *Ann. Rev. Astron. Astrophys.* **35**(1), 607–636 (1997)
- [40] Zhang, W., MacFadyen, A.I.: RAM: a relativistic adaptive mesh refinement hydrodynamics code. *Astrophys. J. Suppl. Ser.* **164**(1), 255–279 (2006)

Deepak Bhojriya and Harish Kumar
Department of Mathematics
IIT Delhi
Hauz Khas, New Delhi-16
India
e-mail: hkumar@maths.iitd.ac.in

Deepak Bhojriya
e-mail: dkbhojriya@gmail.com

(Received: June 6, 2019; revised: December 15, 2019)