CrossMark

A Simulation Study of Ultra-relativistic Jets–I. A New Code for Relativistic Hydrodynamics

Jeongbhin Seo[1] , Hyesung Kang[1] , Dongsu Ryu[2] , Seungwoo Ha[2], and Indranil Chattopadhyay[3]
[1] Department of Earth Sciences, Pusan National University, Busan 46241, Republic of Korea
[2] Department of Physics, College of Natural Sciences, UNIST, Ulsan 44919, Republic of Korea; ryu@sirius.unist.ac.kr
[3] Aryabhatta Research Institute of Observational Sciences, Nainital, India
Received 2021 June 7; revised 2021 July 24; accepted 2021 July 26; published 2021 October 25

Abstract

In an attempt to investigate the structures of ultra-relativistic jets injected into the intracluster medium (ICM) and the associated flow dynamics, such as shocks, velocity shear, and turbulence, we have developed a new special relativistic hydrodynamic (RHD) code in the Cartesian coordinates, based on the weighted essentially non-oscillatory (WENO) scheme. It is a finite difference scheme of high spatial accuracy, which has been widely employed for solving hyperbolic systems of conservation equations. The code is equipped with different WENO versions, such as the fifth-order accurate WENO-JS, WENO-Z, and WENO-ZA, and different time-integration methods, such as the fourth-order accurate Runge–Kutta (RK4) and strong stability preserving RK (SSPRK), as well as the implementation of the equations of state (EOSs) that closely approximate the EOS of the single-component perfect gas in relativistic regimes. In addition, it incorporates a high-order accurate averaging of fluxes along the transverse directions to enhance the accuracy of multidimensional problems, and a modification of eigenvalues for the acoustic modes to effectively control the carbuncle instability. Through extensive numerical tests, we assess the accuracy and robustness of the code, and choose WENO-Z, SSPRK, and the EOS suggested in Ryu et al. as the fiducial setup for simulations of ultra-relativistic jets. The results of our study of ultra-relativistic jets using the code is reported in an accompanying paper.

Unified Astronomy Thesaurus concepts: Relativistic jets (1390); Hydrodynamical simulations (767); Relativistic fluid dynamics (1389); Radio galaxies (1343)

1. Introduction

High-energy astrophysical phenomena, such as pulsar wind nebulae, gamma-ray bursts (GRBs), and radio-loud active galactic nuclei (AGNs), often involve relativistic jets and flows (see, e.g., Piran 2005; Gaensler & Slane 2006; Hardcastle & Croston 2020, for reviews). Relativistically beamed jets from blazers are inferred to have a bulk Lorentz factor of up to $\Gamma \sim 50$ (e.g., Savolainen et al. 2010; Lister et al. 2013). In the most widely accepted model for GRBs, a highly focused explosion associated with the formation of a black hole drives a pair of collimated relativistic jets with $\Gamma \lesssim 400$ (e.g., Nava et al. 2017).

For studies of relativistic hydrodynamic (RHD) problems, numerical codes have been built by adopting various schemes originally developed for non-relativistic Newtonian hydrodynamics. A partial list of RHD codes includes the following: Falle & Komissarov (1996) based on the van Leer scheme; Martı & Müller (1996), Aloy et al. (1999), and Mignone et al. (2005) based on the PPM scheme; Zhang & MacFadyen (2006), Radice & Rezzolla (2012), and Núñez-de la Rosa & Munz (2016) implementing high-order reconstructions including WENO; Schneider et al. (1993) and Mignone & Bodo (2005) based on the HLLE and HLLC schemes; Ryu et al. (2006) based on the TVD scheme; Dolezal & Wong (1995) based on the ENO scheme; Del Zanna & Bucciantini (2002) based on the CENO scheme; Qin et al. (2016) based on the DG scheme; and Duffell & MacFadyen (2011) and He & Tang (2012) based on moving-mesh methods. Reviews of RHD codes can be found, e.g., in Martí & Müller (2003, 2015).

In an accompanying paper (Seo et al. 2021, Paper II), we report the findings of our recent study on the structures of ultra-relativistic jets injected into the intracluster medium (ICM) and the associated flow dynamics, such as shocks, velocity shear,

and turbulence, through high-resolution RHD simulations. In this paper, we describe the special RHD code newly developed for that study. Our code is based on the weighted essentially non-oscillatory (WENO) scheme for hyperbolic systems of conservation equations, and high-order accurate time-integration methods. It also incorporates a number of features that make the code suitable for the study of ultra-relativistic jets.

The WENO scheme is one of upwind methods, designed to achieve a high-order accuracy in smooth regions and keep the essentially non-oscillatory property near discontinuities; hence, it can accurately reproduce the nonlinear dynamics in complex flows. The basic idea of the WENO scheme lies in an adaptive interpolation or reconstruction procedure (see Shu 2009 for a review). Liu et al. (1994) first introduced a third-order accurate finite volume (FV) WENO scheme, in which the cell averages integrated over the cell volume are used to obtain the reconstructed solutions at the cell interfaces. Later, Jiang & Shu (1996) proposed a fifth-order accurate finite difference (FD) WENO scheme, in which the point values of the physical fluxes are used to produce the reconstructed fluxes at the cell interfaces with weight functions. Since the work of Jiang & Shu (1996), different versions of FD WENO scheme with improved weight functions have been proposed to achieve higher resolution for smooth flows and/or smaller dissipation near discontinuities. Our code is built by implementing the three versions that have been widely applied to various computational fluid dynamic codes: (1) the WENO scheme presented in Jiang & Shu (1996) and Jiang & Wu (1999) (WENO-JS, hereafter), (2) WENO-Z by Borges et al. (2008), and (3) WENO-ZA by Liu et al. (2018), all of which have fifth-order spatial accuracy. A partial list of other FD WENO varieties includes WENO-M by Henrick et al. (2005), WENO-CU by Hu et al. (2010), and WENO-NS by Ha et al. (2013).

For the time integration of hyperbolic equations, the classical Runge–Kutta (RK) methods have been widely employed along with the WENO scheme (e.g., Shu & Osher 1988, 1989; Jiang & Shu 1996). An improved approach, called the strong stability preserving Runge–Kutta (SSPRK) method, was proposed to ensure nonlinear stability properties and treat discontinuous structures without spurious oscillations and smearing (e.g., Spiteri & Ruuth 2002, 2003; Gottlieb 2005). Since shocks and contact discontinuities are ubiquitous in the flows associated with relativistic jets, our code is built with the fourth-order accurate, five-stage SSPRK time-integration method described in Spiteri & Ruuth (2002), as well as the fourth-order accurate Runge–Kutta (RK4) method for comparison.

For multidimensional problems, FD schemes can achieve high-order accuracies with the so-called dimension-by-dimension method, in which the one-dimensional (1D) flux is used along each direction (e.g., Shu & Osher 1989). On the other hand, FV schemes equipped with the dimension-by-dimension method normally retain only the second-order accuracy in nonlinear problems (e.g., Zhang et al. 2011). Buchmüller & Helzel (2014) and Buchmüller et al. (2016) proposed a modified dimension-by-dimension method for FV WENO schemes, which leads to high-order accuracies for smooth solutions. Although our code employs an FD WENO scheme, it borrows the idea of the modified dimension-by-dimension method and implements a high-order accurate averaging of fluxes along the transverse directions in smooth-flow regions to improve the performance of multidimensional problems.

It is known that high-accuracy, shock-capturing, upwind codes could be prone to the so-called carbuncle instability, resulting in the deformation of shock front (e.g., Peery & Imlay 1988; Dumbser et al. 2004). The instability develops when a shock wave propagates almost aligned with the computation grid. It arises owing to the insufficient numerical dissipation of upwind schemes. WENO codes like ours are expected to be subject to this instability. In fact, in test simulations of relativistic jets with our code, the instability often appears at the bow shock surface that encompasses the cocoon and shocked ICM (see Section 3.3.2). Solutions for this problem normally introduce additional numerical dissipations at shocks, but at the cost of losing the accuracy of original schemes (e.g., Pandolfi & D'Ambrosio 2001; Hanawa et al. 2008). Recently, Fleischmann et al. (2020) demonstrated that a modification of eigenvalues for the acoustic modes can suppress the carbuncle instability, while keeping the numerical dissipation low. Our code implements this modification to prevent this carbuncle phenomenon.

The description of RHD fluids requires, along with the governing equations for the dynamics, the equation of state (EOS) that relates the pressure to the internal energy density, that is, the thermodynamics. For the sake of convenience, we adopt the following short names for the EOSs used here: RP, ID, TM, and RC. RP stands for the EOS of the "single-component"[4] perfect gas in a relativistic regime (e.g., Synge 1957). In RP, the relation between the pressure and the internal energy density is

given with the modified Bessel functions, which is computationally expensive to manage in numerical codes. Hence, the so-called ideal EOS (ID for short), assuming a constant adiabatic index γ, has been widely used in RHD codes. But with ID, the transition from $\gamma = 5/3$ for subrelativistic temperature to $\gamma = 4/3$ for fully relativistic temperature or vice versa cannot be properly reproduced. On the other hand, EOSs that closely approximate RP and yet are simple enough to be used in numerical codes have been suggested. For instance, an EOS that follows the lower bound of the Taub's inequality (Taub 1948) was first used by Mathews (1971) and later proposed for a RHD code by Mignone et al. (2005) (TM for short). Another EOS that gives a better fit to RP was suggested by Ryu et al. (2006) (RC for short). Our code implements ID, TM, and RC in order to examine the effects of different EOSs.

In this paper, we describe in detail the different elements of our newly developed RHD code and present an extensive set of canonical tests, which are devised to demonstrate the accuracy and robustness of the code. We then choose the fifth-order accurate WENO-Z, the fourth-order accurate SSPRK, and RC as the fiducial setup for the simulation study of ultra-relativistic jets, which is reported in Paper II.

The paper is organized as follows. In Section 2 we present the RHD equations and EOSs, and describe the components of the code, such as the WENO schemes and the time-integration methods along with additional features. The tests are presented in Section 3. Summary and discussion follow in Section 4.

2. Code Description

2.1. Basic Equations

The conservation equations that govern the special RHDs in the laboratory frame can be written as a hyperbolic system of equations

$$\frac{\partial D}{\partial t} + \boldsymbol{\nabla} \cdot (D\boldsymbol{v}) = 0, \tag{1}$$

$$\frac{\partial \boldsymbol{M}}{\partial t} + \boldsymbol{\nabla} \cdot (\boldsymbol{M}\boldsymbol{v} + p) = 0, \tag{2}$$

$$\frac{\partial E}{\partial t} + \boldsymbol{\nabla} \cdot [(E + p)\boldsymbol{v}] = 0, \tag{3}$$

where the conserved quantities, D, $\boldsymbol{M}$, and E, are the mass, momentum, and total energy densities, respectively (e.g., Landau & Lifshitz 1959). The conserved quantities are related to the primitive variables, the rest-mass density, ρ, the fluid three-velocity, $\boldsymbol{v}$, and the isotropic gas pressure, p, as follows:

$$D = \Gamma \rho, \tag{4}$$

$$\boldsymbol{M} = \Gamma^2 \rho h \boldsymbol{v}, \tag{5}$$

$$E = \Gamma^2 \rho h - p, \tag{6}$$

where $\Gamma = 1/\sqrt{1 - v^2}$ with $v = |\boldsymbol{v}|$ is the Lorentz factor, and $h = (e + p)/\rho$ is the specific enthalpy. Here, e is the sum of the internal and rest-mass energy densities. The velocity is normalized by the speed of light (i.e., $c \equiv 1$).

---

[4] Here, by "single-component," fluids are composed of particles of same mass. However, relativistic fluids may consist of electrons, positrons, protons, and ions, which become thermally relativistic at different temperatures. Hence, in general, whether a fluid is in thermally relativistic regime or not should depend not only on the temperature but also on its composition (i.e., either electronpositron pairs, or electrons and protons, or some other combinations). Our single-component EOSs below do not include the composition dependence of relativistic behaviors. An EOS for multicomponent relativistic fluids is discussed, for instance, in Chattopadhyay & Ryu (2009), and its effects on shocks in relativistic jets are discussed, for instance, in Joshi et al. (2021).

2.2. Equation of State

The above equations for RHDs is "closed" with an EOS. In the RP EOS, the specific enthalpy is given as

$$h(p, \rho) = \frac{K_3(1/\Theta)}{K_2(1/\Theta)}, \qquad (7)$$

where K_2 and K_3 are the modified Bessel functions of the second kind of order two and three, respectively (e.g., Synge 1957). Here, $\Theta = p/\rho$ is a temperature-like variable.

As mentioned in the introduction, RP is not a practical EOS for RHD codes, because the calculation of the primitive variables from the conserved quantities is computationally expensive due to the modified Bessel functions. Hence, most RHD codes have adopted either computationally manageable EOSs or EOSs that approximate RP, including the following:

(1) the ID EOS with a constant adiabatic index γ,

$$p = (\gamma - 1)(e - \rho)$$
$$\text{or } h = 1 + \frac{\gamma\Theta}{\gamma - 1}, \qquad (8)$$

(2) the TM EOS suggested by Mignone et al. (2005),

$$\frac{p}{\rho} = \frac{1}{3}\left(\frac{e}{\rho} - \frac{\rho}{e}\right)$$
$$\text{or } h = \frac{5}{2}\Theta + \frac{3}{2}\sqrt{\Theta^2 + \frac{4}{9}}, \qquad (9)$$

and (3) the RC EOS suggested by Ryu et al. (2006),

$$\frac{p}{e - \rho} = \frac{3p + 2\rho}{9p + 3\rho}$$
$$\text{or } h = 2\frac{6\Theta^2 + 4\Theta + 1}{3\Theta + 2}. \qquad (10)$$

With the relativistic kinetic theory, Taub (1948) showed that the EOS for RHDs should satisfy the inequality

$$(h - \Theta)(h - 4\Theta) \geqslant 1. \qquad (11)$$

While TM follows the lower bound of the inequality, RC locates above it satisfying the inequality for all Θ.

The primitive variables, which are used in the calculation of numerical fluxes (see Section 2.3), can be obtained through the inversion of Equations (4)–(6), following the procedures presented in Ryu et al. (2006). The three EOSs, ID, TM, and RC, are described in detail and compared to RP in Ryu et al. (2006). We point that while both RC and TM approximate RP, RC gives a slightly better fit to RP.

2.3. Finite Difference WENO Scheme

In Cartesian geometry, Equations (1)–(3) can be written as

$$\frac{\partial \boldsymbol{q}}{\partial t} + \frac{\partial \boldsymbol{F}}{\partial x} + \frac{\partial \boldsymbol{G}}{\partial y} + \frac{\partial \boldsymbol{H}}{\partial z} = 0, \qquad (12)$$

with the state vector, $\boldsymbol{q}$, and the flux vectors, $\boldsymbol{F}$, $\boldsymbol{G}$, and $\boldsymbol{H}$. The explicit forms of the state and flux vectors can be found, for example, in Ryu et al. (2006). Our code updates the state vector, $\boldsymbol{q}_{i,j,k}$, defined at the cell center of three-dimensional (3D), uniform

Cartesian grids, using the dimension-by-dimension method:

$$\boldsymbol{q}_{i,j,k}' = \boldsymbol{q}_{i,j,k} - \frac{\Delta t}{\Delta x}(\boldsymbol{F}_{i+\frac{1}{2},j,k} - \boldsymbol{F}_{i-\frac{1}{2},j,k})$$
$$- \frac{\Delta t}{\Delta y}(\boldsymbol{G}_{i,j+\frac{1}{2},k} - \boldsymbol{G}_{i,j-\frac{1}{2},k})$$
$$- \frac{\Delta t}{\Delta z}(\boldsymbol{H}_{i,j,k+\frac{1}{2}} - \boldsymbol{H}_{i,j,k-\frac{1}{2}}), \qquad (13)$$

where Δx, Δy, and Δz are the cell sizes along the x, y, and z-directions, and Δt is the time step (see the Section 2.4). The numerical fluxes, $\boldsymbol{F}_{i\pm\frac{1}{2},j,k}$, $\boldsymbol{G}_{i,j\pm\frac{1}{2},k}$, and $\boldsymbol{H}_{i,j,k\pm\frac{1}{2}}$, defined at the cell interfaces, are estimated with the fifth-order accurate FD WENO scheme.

We here describe the reconstructions of the x-flux, $\boldsymbol{F}_{i+\frac{1}{2}}$ (the subscripts j and k are dropped for simplicity), while the reconstructions of y- and z-fluxes can be done by alternating the coordinates. The numerical flux, $\boldsymbol{F}_{i+\frac{1}{2}}$, is obtained using the cell-center fluxes, $\boldsymbol{F}_m$ at $m = i - 2,...,i + 3$, (Jiang & Shu 1996; Jiang & Wu 1999) as

$$\boldsymbol{F}_{i+\frac{1}{2}} = \frac{1}{12}(-\boldsymbol{F}_{i-1} + 7\boldsymbol{F}_i + 7\boldsymbol{F}_{i+1} - \boldsymbol{F}_{i+2})$$
$$+ \sum_{s=1}^{5}[-\varphi_N(\Delta\boldsymbol{F}_{i-\frac{3}{2}}^{s+}, \Delta\boldsymbol{F}_{i-\frac{1}{2}}^{s+}, \Delta\boldsymbol{F}_{i+\frac{1}{2}}^{s+}, \Delta\boldsymbol{F}_{i+\frac{3}{2}}^{s+})$$
$$+ \varphi_N(\Delta\boldsymbol{F}_{i+\frac{5}{2}}^{s-}, \Delta\boldsymbol{F}_{i+\frac{3}{2}}^{s-}, \Delta\boldsymbol{F}_{i+\frac{1}{2}}^{s-}, \Delta\boldsymbol{F}_{i-\frac{1}{2}}^{s-})]\boldsymbol{R}_{i+\frac{1}{2}}^{s}, \qquad (14)$$

where s denotes the five characteristic modes of RHDs and $\boldsymbol{R}_{i+\frac{1}{2}}^{s}$ is the right eigenvector. Here, $\Delta\boldsymbol{F}_{m+\frac{1}{2}}^{s\pm}$ with $m = i - 2,...,$ $i + 2$ is calculated using the local Lax-Friedrichs flux splitting as follows:

$$\boldsymbol{F}_m^s = \boldsymbol{L}_{i+\frac{1}{2}}^s\boldsymbol{F}_m, \quad \boldsymbol{q}_m^s = \boldsymbol{L}_{i+\frac{1}{2}}^s\boldsymbol{q}_m, \qquad (15)$$

$$\Delta\boldsymbol{F}_{m+\frac{1}{2}}^s = \boldsymbol{F}_{m+1}^s - \boldsymbol{F}_m^s,$$
$$\Delta\boldsymbol{q}_{m+\frac{1}{2}}^s = \boldsymbol{q}_{m+1}^s - \boldsymbol{q}_m^s, \qquad (16)$$

$$\Delta\boldsymbol{F}_{m+\frac{1}{2}}^{s\pm} = \frac{1}{2}\left(\Delta\boldsymbol{F}_{m+\frac{1}{2}}^s \pm \delta_{i+\frac{1}{2}}^s\Delta\boldsymbol{q}_{m+\frac{1}{2}}^s\right), \qquad (17)$$

where $\delta_{i+\frac{1}{2}}^s = \max_{i-2\leqslant m\leqslant i+3}|\lambda_m^s|$ with the sth eigenvalue λ_m^s, and $\boldsymbol{L}_{i+\frac{1}{2}}^s$ is the left eigenvector.

The above steps require the eigenvalues and eigenvectors of characteristic modes. We use the formulae presented in Ryu et al. (2006); they are expressed in generic forms with the primitive variables, specific enthalpy, polytropic index, and sound speed, and hence can be used independent of EOSs. The eigenvalues, λ_i^s, are calculated using the primitive variables at the cell center, while the eigenvectors, $\boldsymbol{R}_{i+\frac{1}{2}}^s$ and $\boldsymbol{L}_{i+\frac{1}{2}}^s$, at the cell interfaces are calculated using the arithmetic averages of the fluid three-velocity and specific enthalpy at the two neighboring grids of i and $i + 1$.

The function φ_N is defined as

$$\varphi_N(a, b, c, d) = \frac{1}{3}\omega_0(a - 2b + c)$$
$$+ \frac{1}{6}\left(\omega_2 - \frac{1}{2}\right)(b - 2c + d). \qquad (18)$$

Here, ω_0 and ω_2 are the weight functions, which are designed to achieve high-order accuracies in smooth flows, while keeping the essentially non-oscillatory property near discontinuities. Since the widely used WENO-JS (Jiang & Shu 1996), different

versions of the WENO scheme with different weight functions have been suggested to improve performance, such as WENO-Z by Borges et al. (2008) and WENO-ZA by Liu et al. (2018).

The weight functions are given as[5]

$$\omega_0 = \frac{\delta_0}{\delta_0 + \delta_1 + \delta_2},$$
$$\omega_2 = \frac{\delta_2}{\delta_0 + \delta_1 + \delta_2}. \tag{19}$$

In WENO-JS,

$$\delta_r^{JS} = \frac{C_r}{(\epsilon + IS_r)^2},$$
$$r = 0, 1, 2, \tag{20}$$

in WENO-Z,

$$\delta_r^{Z} = C_r\left(1 + \left(\frac{\tau_5}{\epsilon + IS_r}\right)^2\right),$$
$$r = 0, 1, 2, \tag{21}$$

and in WENO-ZA,

$$\delta_r^{ZA} = C_r\left(1 + \frac{A \cdot \tau_6}{\epsilon + IS_r}\right), \quad r = 0, 1, 2, \tag{22}$$

where $C_0 = 1$, $C_1 = 6$, and $C_2 = 3$, respectively. Here, IS_r's are the local smoothness indicators, which are given as

$$IS_0 = 13(a - b)^2 + 3(a - 3b)^2,$$
$$IS_1 = 13(b - c)^2 + 3(b + c)^2,$$
$$IS_2 = 13(c - d)^2 + 3(3c - d)^2. \tag{23}$$

In WENO-Z, $\tau_5 = |IS_0 - IS_2|$, and in WENO-ZA,

$$A = \frac{\tau_6}{\epsilon + IS_0 + IS_2 - \tau_6}, \tag{24}$$

$$\tau_6 = 3(|a - 3b| - |3c - d|)^2 + 13(|a - b| - |c - d|)^2. \tag{25}$$

The parameter ϵ is included to avoid the zero denominator, and $\epsilon = 10^{-6}$ is used in our code.

WENO-Z and WENO-ZA, which include high-order global smoothness indicators, were developed to achieve higher resolution and smaller dissipation than WENO-JS, but at the same time, they could be prone to numerical artifacts for a wider range of problems.

2.4. Time Integration

Numerical codes based on the WENO scheme commonly employ high-order RK-type methods for the time integration. Our code implements the fourth-order accurate, five-stage SSPRK method (e.g., Spiteri & Ruuth 2002, 2003; Gottlieb 2005), as well as RK4, the classical fourth-order accurate RK method (e.g., Shu & Osher 1988, 1989; Jiang & Shu 1996).

---

[5] Note that δ's here denote quantities different from δ in Equation (17); δ is used in both Equations (17) and (20) to keep the original notations of Jiang & Shu (1996).

In RK4, the time-stepping from $q_{i,j,k}^n$ to $q_{i,j,k}^{n+1}$ proceeds with the following four stages:

$$q^{(0)} = q^n,$$
$$q^{(1)} = q^{(0)} + \frac{\Delta t}{2}\mathcal{L}^{(0)},$$
$$q^{(2)} = q^{(0)} + \frac{\Delta t}{2}\mathcal{L}^{(1)},$$
$$q^{(3)} = q^{(0)} + \Delta t\mathcal{L}^{(2)},$$
$$q^{n+1} = \frac{1}{3}(-q^{(0)} + q^{(1)} + 2q^{(2)} + q^{(3)})$$
$$+ \frac{\Delta t}{6}\mathcal{L}^{(3)}. \tag{26}$$

In SSPRK, the time-stepping is given as

$$q^{(0)} = q^n,$$
$$q^{(l)} = \sum_{m=0}^{l-1}(\chi_{lm}q^{(m)} + \Delta t\beta_{lm}\mathcal{L}^{(m)}),$$
$$l = 1, 2, \cdots, 5,$$
$$q^{n+1} = q^{(5)}. \tag{27}$$

The coefficients χ_{lm} and β_{lm} can be found, for instance, in Spiteri & Ruuth (2002). Here

$$\mathcal{L}_{i,j,k}^{(l)} = -\frac{F_{i+\frac{1}{2},j,k}^{(l)} - F_{i-\frac{1}{2},j,k}^{(l)}}{\Delta x}$$
$$- \frac{G_{i,j+\frac{1}{2},k}^{(l)} - G_{i,j-\frac{1}{2},k}^{(l)}}{\Delta y}$$
$$- \frac{H_{i,j,k+\frac{1}{2}}^{(l)} - H_{i,j,k-\frac{1}{2}}^{(l)}}{\Delta z}, \tag{28}$$

where $\mathcal{L}_{i,j,k}^{(l)}$ is calculated with $q_{i,j,k}^{(l)}$. In Equations (26) and (27), the subscripts i, j, and k are dropped for simplicity.

The time step, Δt, should be restricted by the CFL (Courant-Friedrichs-Levy) condition for the numerical stability,

$$\Delta t = \text{CFL} \bigg/ \left[\frac{\lambda_x^{\max}}{\Delta x} + \frac{\lambda_y^{\max}}{\Delta y} + \frac{\lambda_z^{\max}}{\Delta z}\right], \tag{29}$$

where $\lambda^{\max}$'s are the maxima of the eigenvalues at the cell center, $\lambda_{i,j,k}^s$, along the x-, y-, and z-directions, respectively. While normally $\text{CFL} \leqslant 1$ is required for RK methods, $\text{CFL} > 1$ is allowed with SSPRK (e.g., Spiteri & Ruuth 2002, 2003). In our code, $\text{CFL} = 0.8$ is used as the default value, unless otherwise specified.

2.5. Averaging of Fluxes along Transverse Directions

As mentioned in the introduction, borrowing the idea of the modified dimension-by-dimension method proposed by Buchmüller & Helzel (2014) and Buchmüller et al. (2016) for FV WENO schemes, our code implements an averaging of fluxes along the transverse directions as follows. Here, the averaging of the x-flux is described, and those of the y- and z-fluxes can be done similarly.

For the calculation of the numerical fluxes described in Section 2.3, instead of $q_{i,j,k}$, the averaged state vector,

$$\bar{q}_{i,j,k} = \frac{1}{\Delta y \Delta z} \int_{\sigma_{yz}} q_{i,j,k} dy dz + \mathcal{O}(\Delta^p), \tag{30}$$

is used. Here, $\sigma_{yz} = [y_i - \Delta y/2, \; y_i + \Delta y/2] \times [z_i - \Delta z/2, z_i + \Delta z/2]$, and $\mathcal{O}(\Delta^p)$ denotes the order of the averaging. Then, in the time-integration step to update the state vectors, which is described in Section 2.4, instead of $F_{i\pm\frac{1}{2},j,k}$, the averaged flux,

$$\bar{F}_{i\pm\frac{1}{2},j,k} = \frac{1}{\Delta y \Delta z} \int_{\sigma_{yz}} F_{i\pm\frac{1}{2},j,k} dy dz - \mathcal{O}(\Delta^p) \tag{31}$$

is used. To avoid possible problems of degrading shocks and contact discontinuities and generating spurious oscillations there, the above averagings are applied only to the regions of smooth flows (see Buchmüller et al. 2016 for details).

Considering the accuracies of the WENO scheme and the SSPRK method in our code, the fourth-order accurate averaging with $p = 4$ is employed, which is given as,

$$\bar{q}_{i,j,k} = q_{i,j,k} - \frac{1}{24}(q_{i,j-1,k} - 2q_{i,j,k} + q_{i,j+1,k})$$
$$- \frac{1}{24}(q_{i,j,k-1} - 2q_{i,j,k} + q_{i,j,k+1}), \tag{32}$$

and

$$\bar{F}_{i\pm\frac{1}{2},j,k} = F_{i\pm\frac{1}{2},j,k}$$
$$+ \frac{1}{24}(F_{i\pm\frac{1}{2},j-1,k} - 2F_{i\pm\frac{1}{2},j,k} + F_{i\pm\frac{1}{2},j+1,k})$$
$$+ \frac{1}{24}(F_{i\pm\frac{1}{2},j,k-1} - 2F_{i\pm\frac{1}{2},j,k} + F_{i\pm\frac{1}{2},j,k+1}). \tag{33}$$

Higher-order accurate averagings can also be done (see Buchmüller & Helzel 2014; Buchmüller et al. 2016, for instance, for a fifth-order accurate averaging).

Hereafter, this modification of transverse fluxes is termed as the *transverse-flux averaging* for short. We expect that this improves the performance of multidimensional problems, as shown in Section 3.2.5.

2.6. Suppression of Carbuncle Instability

The carbuncle instability often appears at shocks in simulations based on upwind schemes including WENO, as mentioned in the introduction. If a shock propagates almost aligned with the computational grid in multidimensional problems, the disturbances traveling along the shock front at low Mach numbers tend to become unstable owing to insufficient dissipation. It is related to the well-known low Mach number problem of upwind schemes. Fleischmann et al. (2020) suggested that the instability could be suppressed through a modification of eigenvalues for two acoustic modes as follows:

$$c'_s = \min(\phi|v_x|, c_s), \quad \lambda_{1,5} = v_x \pm c'_s, \tag{34}$$

where ϕ is a positive number of order $\mathcal{O}(1)$. Then, the Mach numbers of the acoustic modes increase to $M \geqslant 1/\phi$ in low Mach number regions, and hence the modification in effect decreases the imbalance of advective and acoustic dissipations, easing the low Mach number problem. Following the suggestion

of Fleischmann et al. (2020), we modify the eigenvalues of two acoustic modes in RHDs by limiting the local sound speed:

$$c'_s = \min(\phi|v_x|, c_s),$$
$$\lambda_{1,5} = \frac{(1 - c'^2_s)v_x \pm c'_s/\Gamma\sqrt{\mathcal{Q}}}{1 - c'^2_s v^2}, \tag{35}$$

where $\mathcal{Q} = 1 - v_x^2 - c'^2_s(v_y^2 + v_z^2)$ (see Equation (43) of Ryu et al. 2006).

As presented in Section 3.3.2, the carbuncle instability appears and deforms the bow shock surface in some relativistic jet simulations with our code. It is demonstrated that the above modification effectively suppresses the instability, while the flow structures away from the bow shock are almost intact.

3. Code Verification Tests

We have carried out a number of numerical tests to evaluate the performance of our RHD code described in the previous section. The default setup of the code consists of WENO-Z, SSPRK, and RC along with CFL = 0.8, and includes the fourth-order accurate transverse-flux averaging for multidimensional problems, unless stated otherwise.

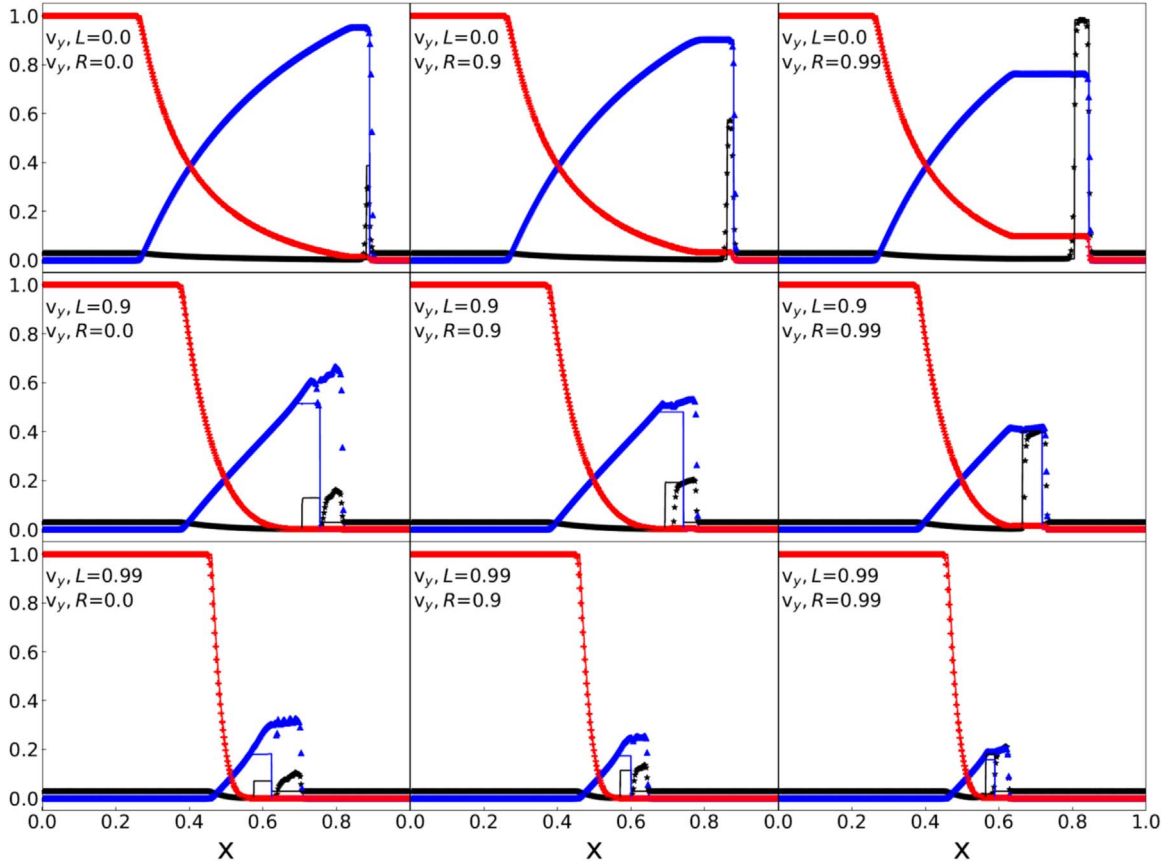### 3.1. 1D Tests: Relativistic Shock Tubes

Relativistic shock-tube tests are set up with different RHD states in the left and right regions of the computational domain of $x = [0, 1]$ with the initial discontinuity at $x = 0.5$. In Newtonian hydrodynamics, the transverse velocities, that is, y- and z-velocities, do not affect the solutions of shock-tube problems. In relativistic regimes, on the other hand, the Lorentz factor depends on the transverse velocities and so do the solutions. Hence, in papers for RHD codes, tests have normally included nonzero transverse velocities (see the references in the *Introduction*). Below, we present shock-tube tests, setting the transverse velocity along the y-direction without loss of generality; then, the primitive variables of a RHD state is given as $u = (\rho, v_x, v_y, p)$, and the left and right states are denoted by u_L and u_R.

We point out that, with nonzero v_y, the tests implicitly assume v_y in $-\infty < y < \infty$ and the simultaneous occurrence of events along the y-axis at $t = 0$, potentially violating the causality in the fluid-rest frame. While such a setup is unphysical, that is the limitation of 1D tests. In addition, shocks with nonzero transverse velocity commonly appear in relativistic jets, and a clear example is the recollimation shock that forms in the jet spine flow (see Paper II). So we have also performed shock-tube tests with nonzero v_y and show them below.

The first test comprises shock-tube problems with different combinations of transverse velocities in the left and right states, where the initial condition is given as

$$u_L = (1, 0, v_{y,L}, 10^3), \; u_R = (1, 0, v_{y,R}, 10^{-2}), \tag{36}$$

with $v_{y,L} = 0, 0.9, 0.99$ and $v_{y,R} = 0, 0.9, 0.99$. Here, ρ, v_x, and p are the same as in Equation (38) and hence the case with $v_{y,L} = 0.99$ and $v_{y,R} = 0.99$ is the same as in Figure 3. Pons et al. (2000) presented the analytic solutions for these shock-tube problems in the case of the ID EOS with $\gamma = 5/3$. Mignone et al. (2005) employed these as a test for their PPM-based RHD code with the ID EOS.

Figure 1. 1D relativistic shock-tube tests with different combinations of transverse velocities. The initial condition is given in Equation (36). The transverse velocities in the left ($x < 0.5$) and right ($x > 0.5$) sides are $v_{y,R} = (0, 0.9, 0.99)$ from left to right and $v_{y,L} = (0, 0.9, 0.99)$ from top to bottom, respectively. The results, $\rho/35$ (black asterisks), v_x (blue triangles), and $p/10^3$ (red plus signs), of simulations with the default setup (WENO-Z, SSPRK, RC, and CFL = 0.8) using 400 grid zones are plotted at $t = 0.4$. Simulations of very high-resolution (2^{17} grid zones) are shown with the solid lines for comparison.

Following Mignone et al. (2005), Figure 1 shows the results with RC using 400 grid zones for nine combinations of $v_{y,L}$ and $v_{y,R}$. Since the exact analytic solution is not available for the RC EOS, we compare the results with those of high-resolution simulations. Apart from the overall differences due to the different EOS employed, the trend in the numerical solutions of our WENO code is similar to those of the PPM code (see Figure 5 in Mignone et al. (2005)); the solutions degrade when the transverse velocity is included. It turns out to be a very "severe" test, especially when the transverse velocity is large with $v_{y,L} \geqslant v_{y,R}$; the contact discontinuity shows further smearing and the shock location deviates from the correct position. While Mignone et al. (2005) attributed these features possibly to relativistic effects, we find that such degrading occurs with relativistic transverse velocities, independent of the EOS and numerical schemes. The overall capability of our WENO code to capture shocks and contact discontinuities in this test looks comparable to that of the PPM code.

The second test is intended to compare the RK4 and SSPRK time-integration methods. The initial condition is given as

$$\boldsymbol{u}_L = (1, 0, 0.9, 10^3), \quad \boldsymbol{u}_R = (1, 0, 0, 10^{-2}). \qquad (37)$$

This is the same problem as that in the middle-left panel of Figure 1 and is one of the cases where the reproduction of acceptable numerical solutions requires a large number of grid zones. Figure 2 shows the results using 20,000 grid zones. This test illustrates that SSPRK performs better than RK4, in terms

of reproducing the correct solutions with a limited number of grid zones. Hence, we adopt SSPRK as the default time-integration method. Note that with SSPRK, although the computation time is about 25% longer per time step, a larger CFL number is allowed; hence, the overall computational efficiency could be comparable in SSPRK and RK4.

We next examine the dependence of the results of a shock-tube test on different EOSs. The initial condition is given as

$$\boldsymbol{u}_L = (1, 0, 0.99, 10^3), \quad \boldsymbol{u}_R = (1, 0, 0.99, 10^{-2}). \qquad (38)$$

Figure 3 shows the results of very high-resolution simulations with 2^{17} grid zones, demonstrating the differences due to different EOSs. This is the same as that in the bottom-right panel of Figure 1. This test was also presented in Ryu et al. (2006), and the results agree with those in their Figure 6 (which were produced with a different code). The numerical solutions with ID, both of $\gamma = 5/3$ and $4/3$, are quite different from those with RC and TM. Considering that RC and TM are designed to fit the EOS for perfect relativistic gas, RP, it is clear that the simulations with ID fail to reproduce the correct solutions of RHDs with the RP EOS. The differences in the RC and TM solutions are relatively small. Yet, as mention in Section 2.2, RC approximates RP better than TM, so we expect that the simulation using RC reproduces the solution with RP more closely. Hence, we here adopt RC as the fiducial EOS for our code.
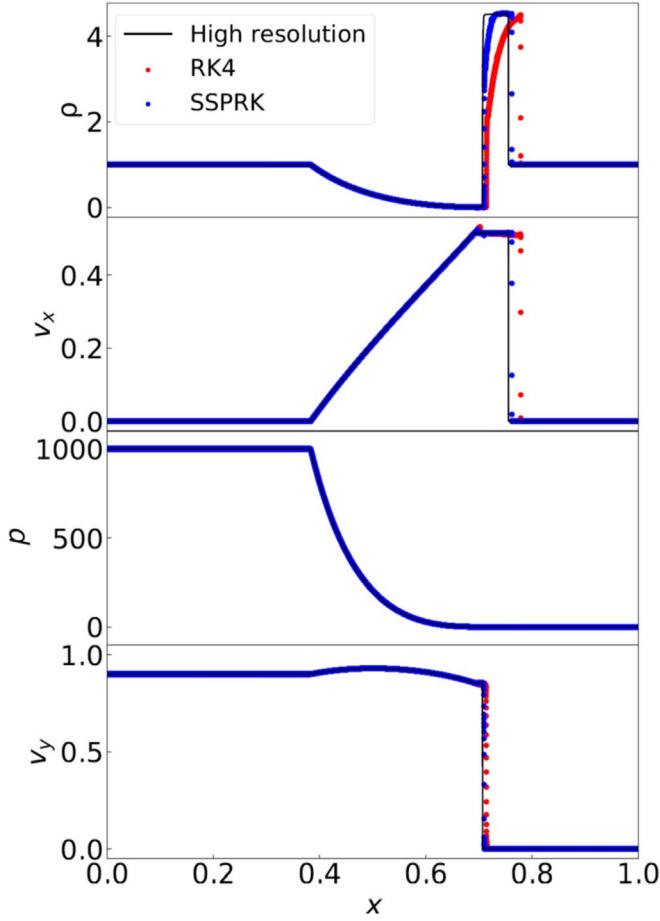
Figure 2. 1D relativistic shock-tube test to compare the RK4 (red) and SSPRK (blue) time-integration methods. The initial condition is given in Equation (37). The results of simulations with WENO-Z, RC, and CFL = 0.8 using 20,000 grid zones are plotted at $t = 0.45$. Simulations of very high-resolution (2^{17} grid zones) are shown with the solid lines for comparison.

3.2. 2D Tests

3.2.1. 2D Relativistic Shock Tube

The first 2D test is a shock-tube problem where the shock, contact discontinuity, and rarefaction wave propagate along the off-axis in a rectangular domain. It is designed to evaluate the ability of our code to capture discontinuous structures and reproduce waves in 2D. The initial condition is given as

$$\mathbf{u}(x, y) = \begin{cases} \mathbf{u}_L & \text{for } y < 2(1-x), \\ \mathbf{u}_R & \text{for } y > 2(1-x), \end{cases} \quad (39)$$

where

$$\mathbf{u}_L = (10, 0, 0, 13.3), \ \mathbf{u}_R = (1, 0, 0, 10^{-6}). \quad (40)$$

Here and below, $\mathbf{u} = (\rho, v_x, v_y, p)$ in 2D tests. The computational domain consists of $[0,1] \times [0, 2]$. Here, the surfaces of the shock and contact discontinuity are formed with the inclination angle of $\arctan(1/2) = 26°.565$ with respect to the x-axis, and the continuous boundary inclined with the same angle is applied (see the right panels of Figure 4). This is the 2D version of a popular 1D shock-tube test that has been widely performed in previous works (e.g., Schneider et al. 1993, and the papers that cite it).
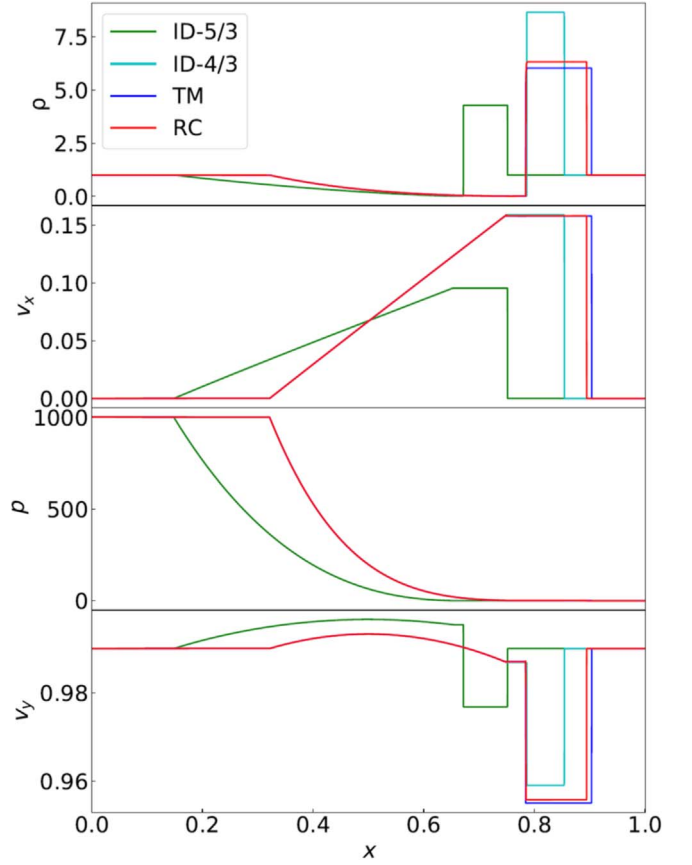
Figure 3. 1D relativistic shock-tube test to demonstrate the effects of different EOSs. The initial condition is given in Equation (38). Simulations of very high-resolution (2^{17} grid zones) adopting WENO-Z, SSPRK, and CFL = 0.8 are shown at $t = 1.8$.
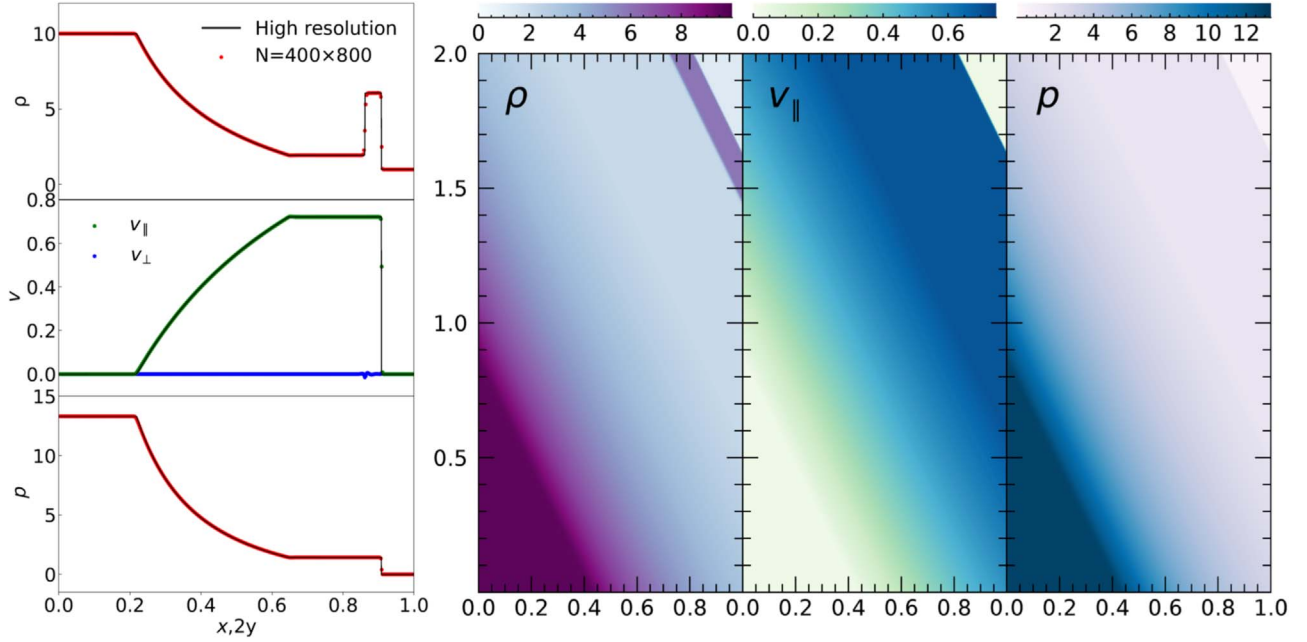
Figure 4 shows the results using 400×800 grid zones. Since the exact analytic solution is not available for the RC EOS, we again compare the results with those of a 1D high-resolution simulation. The smooth structure of the rarefaction wave is well reproduced, while the shock and contact discontinuity are resolved with two to four grid zones. In the left-middle panel, along with the fluid velocity parallel to the propagation direction of the structures, $v_\parallel$ (green dots), the transverse velocity, $v_\perp$ (blues dots), are shown. While $v_\perp = 0$ expected, small, nonzero $v_\perp$ appears and it may be used to estimate the numerical error involved in capturing the shock and contact discontinuity in this test. Compared to $v_\parallel$ behind the shock, which is 0.721, we find that $|v_\perp|$ at the shock and contact discontinuity is at most $\sim 1\%-2\%$.

3.2.2. Convergence of Code

We next consider a 2D advection test to quantify the accuracy of our code in the default setup. The initial condition is given as

$$\rho(x, y) = 1 + 0.2 \sin[2\pi(x \cos\theta + y \sin\theta)],$$
$$v_x(x, y) = 0.2, v_y(x, y) = -0.1,$$
$$p(x, y) = 1, \quad (41)$$

with $\theta = 30°$ in the periodic x–y domain of $[0, 2/\sqrt{3}] \times [0, 2]$, represented with $N \times 2N$ grid zones. Here, the density fluctuation

Figure 4. 2D relativistic shock-tube test. The initial condition is given in Equations (39) and (40). The results with the default setup (WENO-Z, SSPRK, RC, and CFL = 0.8) in the computational domain of [0,1]×[0,2] using 400 × 800 grid zones are shown at $t = 0.4 \times \sqrt{5}$. Left panels: The flow quantities along $x = 2/y$ (dots) are compared to those from a 1D high-resolution simulation with 20,000 grid zones (black solid lines). In the middle panel, $v_\parallel$ (green dots) and $v_\perp$ (blue dots) are the fluid velocity components parallel and perpendicular to the propagation direction of structures. Right panels: The 2D images of the rest-mass density, $v_\parallel$, and pressure are shown.

propagates with the angle of $\arctan(-1/2) = -26°.565$ with respect to the x-axis. We note that this is a test similar to those performed before, for instance, in He & Tang (2012) and Núñez-de la Rosa & Munz (2016). While those previous studies set $v_y = 0$, our test includes nonzero v_y to make it a true 2D test.

The L_2 error and the order of convergence are estimated with the rest-mass density as

$$\Delta_N = \frac{\sqrt{\sum_{i=1}^{N}\sum_{j=1}^{2N}(\rho_{i,j}^{num} - \rho_{i,j}^{exact})^2}}{\sqrt{\sum_{i=1}^{N}\sum_{j=1}^{2N}(\rho_{i,j}^{exact})^2}}, \quad (42)$$

$$R_N = \log_2\left(\frac{\Delta_{N/2}}{\Delta_N}\right). \quad (43)$$

Here, $\rho_{i,j}^{num}$ is the numerical solution, while $\rho_{i,j}^{exact}$ is the exact solution. Table 1 gives Δ_N and R_N at $t = 1$ for different numerical resolutions. $R_N \approx 4.0$ implies that our code achieves the fourth-order convergence, possibly due to the fourth-order accuracy of the SSPRK time-integration method.

3.2.3. 2D Riemann Problem

One of conventional 2D tests is the Riemann problem, which consists of four square domains with different initial states (e.g., Del Zanna & Bucciantini 2002; Mignone et al. 2005; He & Tang 2012; Núñez-de la Rosa & Munz 2016). This test is designed to evaluate how a multidimensional code handles the evolution of shocks, contact discontinuities, and a jet-like structure. We consider, for instance, the specific problem performed by He & Tang (2012) with the following initial

Table 1
2D Advection Test

$N \times 2N^a$	Δ_N^a	R_N^b
20 × 40	2.69×10^{-4}	⋯
40 × 80	1.66×10^{-5}	4.02
80 × 160	1.03×10^{-6}	4.01
160 × 320	6.44×10^{-8}	4.00

Notes.
[a] Resolution.
[b] L_2 error and the order of convergence.

condition:

$$\begin{cases} \boldsymbol{u}_1 = (0.035145216124503, 0, 0, 0.162931056509027), & x > 0, y > 0, \\ \boldsymbol{u}_2 = (0.1, 0.7, 0, 1), & x < 0, y > 0, \\ \boldsymbol{u}_3 = (0.5, 0, 0, 1), & x < 0, y < 0, \\ \boldsymbol{u}_4 = (0.1, 0, 0.7, 1), & x > 0, y < 0, \end{cases}$$
(44)

in the computational domain of $[-1, 1] \times [-1, 1]$, covered with 600×600 grid zones. The continuous boundary is imposed on the four faces.

Simulations have been carried out for nine combinations of different EOSs and different WENO versions, and are compared in Figure 5. The case of the ID EOS with $\gamma = 5/3$ reproduces the results of previous studies (see, e.g., Figure 5 in He & Tang 2012). However, the ID results are clearly different from those with the RC and TM EOSs. Hence, this should be a multidimensional example, which illustrates the differences induced by the use of different EOSs. Looking closely at the center region of the domain, the simulations with WENO-ZA (bottom panels) display wiggles in the jet-like structure in blue,
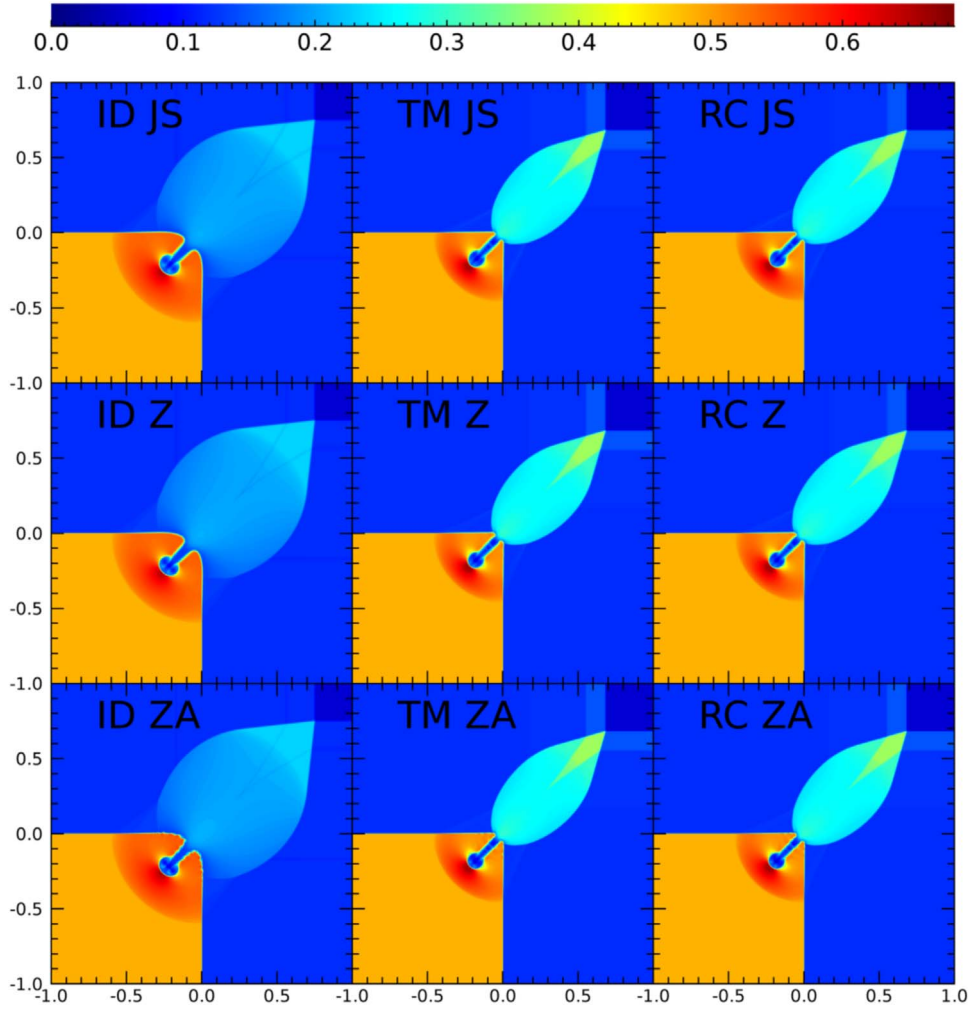
Figure 5. 2D Riemann problem with the initial condition given in Equation (44). The rest-mass density, ρ, from simulations with different EOSs, ID with $\gamma = 5/3$ (left), TM (center), and RC (right), using different WENO versions, WENO-JS (top), WENO-Z (middle), and WENO-ZA (bottom), is shown at $t = 0.8$. In all the simulations, the transverse-flux averaging is implemented, and the SSPRK time-integration method with CFL = 0.8 is used. The computational domain of $[-1, 1] \times [-1, 1]$ is represented with 600×600 grid zones.

which expands in the lower-left direction. By contrast, those features are less noticeable in the simulations with WENO-JS and WENO-Z.

3.2.4. Double-Mach Reflection Problem

The double-Mach reflection test was first introduced by Woodward & Colella (1984) for a Newtonian hydrodynamic code, and later employed for RHD codes in several studies (e.g., Zhang & MacFadyen 2006; He & Tang 2012; Radice & Rezzolla 2012; Núñez-de la Rosa & Munz 2016). In this test, a shock of Mach number $M_s = 10$ travels with the angle of 60° with respect to the x-axis and is reflected at the wall along the x-axis, mimicking a shock colliding with an inclined wedge.

While previously the test was performed with the ID EOS ($\gamma = 1.4$), we here consider the RC and TM EOSs as well. The initial condition for ID with $\gamma = 4/3$ is given as

$$\boldsymbol{u}(x, y) = \begin{cases} \boldsymbol{u}_L & \text{for } y > \sqrt{3}\,(x - 1/6), \\ \boldsymbol{u}_R & \text{for } y < \sqrt{3}\,(x - 1/6), \end{cases} \tag{45}$$

where

$$\boldsymbol{u}_L = (8.564, 0.4247 \sin 60^{\circ}, -0.4247 \cos 60^{\circ}, 0.3808),$$
$$\boldsymbol{u}_R = (1.4, 0, 0, 0.0025), \tag{46}$$

in the computational domain of $[0, 4] \times [0, 1]$ (e.g., He & Tang 2012). The boundary condition is reflecting at the wall along the x-axis in $x > 1/6$, and is continuous elsewhere; at the top boundary of $y = 1$, the condition is set to either $\boldsymbol{u}_L$ or $\boldsymbol{u}_R$ depending on the exact position of the shock.

The characteristics of the induced shock depend rather sensitively on the EOS. To mitigate the differences, we prescribe different post-shock states $\boldsymbol{u}_L$ for different EOSs, while keeping the same pre-shock state $\boldsymbol{u}_R$ and $M_s = 10$. Then, the initial condition is given as

$$\boldsymbol{u}_L = (6.0638, 0.4342 \sin 60^{\circ},$$
$$-0.4342 \cos 60^{\circ}, 0.4275) \tag{47}$$

for the TM EOS and

$$\boldsymbol{u}_L = (6.2506, 0.4340 \sin 60^{\circ},$$
$$-0.4340 \cos 60^{\circ}, 0.4322) \tag{48}$$
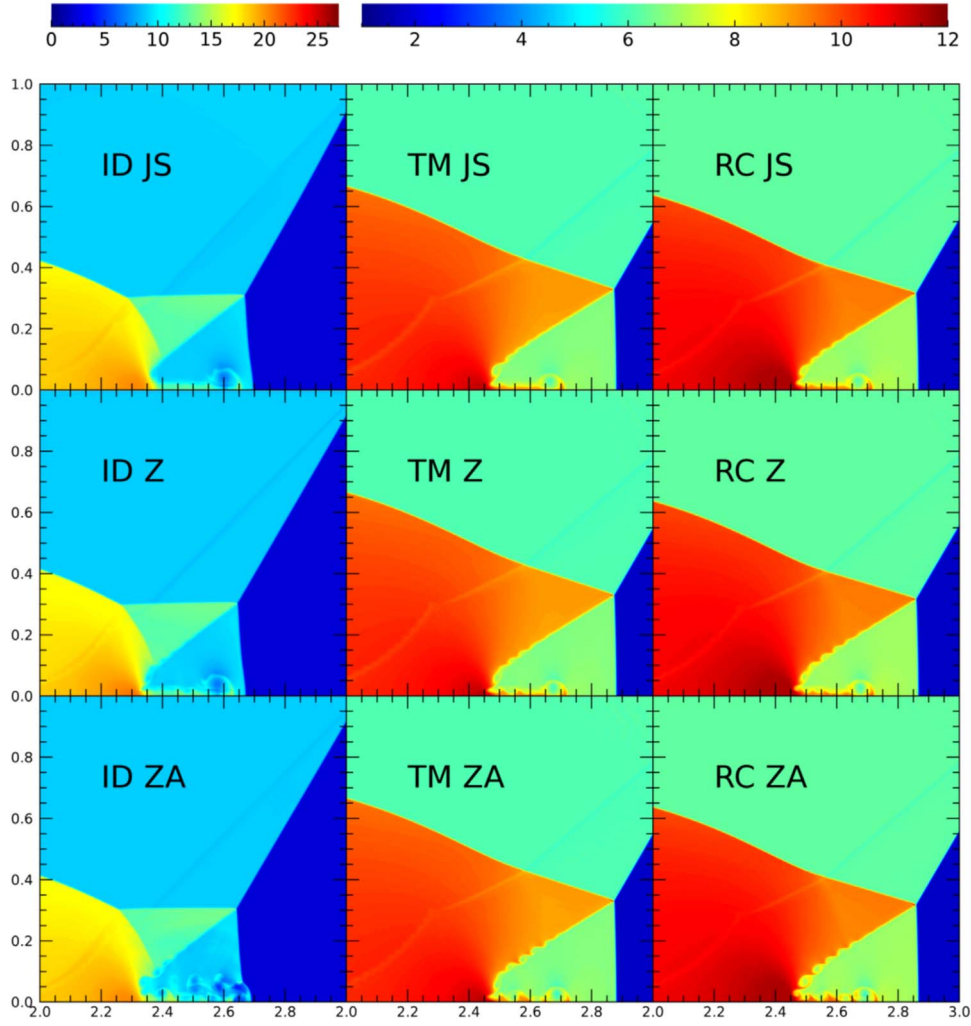
Figure 6. Relativistic double-Mach reflection problem with an inclined shock of $M_s = 10$. The rest-mass density, ρ, from simulations with different EOSs, ID with $\gamma = 1.4$ (left), TM (center), and RC (right), using different WENO versions, WENO-JS (top), WENO-Z (middle), and WENO-ZA (bottom), is shown at $t = 4$. The initial conditions are given in Equation (46) for ID, in Equation (47) for TM, and in Equation (48) for RC. While the computational domain includes $[0, 4] \times [0, 1]$ with 1600×400 grid zones, only the region of $[2, 3] \times [0, 1]$ with 400×400 grid zones is plotted. The transverse-flux averaging is implemented, and the SSPRK time-integration method with CFL = 0.8 is used, except in the case of ID and WENO-ZA (bottom left panel) where CFL = 0.5 is used. Note that the color bar for the ID case is different from that for the TM and RC cases.

for the RC EOS. Note that even with the same ρ and p in the pre-shock region, the sound speed, c_s, differs if the EOS is different (see, e.g., Ryu et al. (2006) for the formulae of c_s as a function of p/ρ for different EOSs). So the shock speed, v_s, is different, even if the Mach number is fixed; in this test with $M_s = 10$, $v_s = 0.4984, 0.5440$, and 0.5438 for ID, TM, and RC, respectively. Again, this test demonstrates the differences owing to different EOSs.

Simulations have been performed with different WENO versions as well as with different EOSs, using 1600×400 grid zones, and compared in Figure 6. In this test, a contact discontinuity extends from a triple shock point to the lower-left direction, and is pushed by the high density and pressure blob and forms a jet-like structure along the x-axis. Figure 6 focuses on the region of $[2, 3] \times [0, 1]$ that includes those structures. The contact discontinuity and jet-like structure are Kelvin–Helmholtz (KH) unstable, as was shown for the Newtonian test (see, e.g., Reyes et al. 2019). In simulations with WENO-ZA, the KH vortices are well visible along the contact discontinuity, while they are less obvious with WENO-Z and almost absent

with WENO-JS. In general, WENO-Z and WENO-ZA, equipped with higher-order global smoothness indicators, tend to produce richer structures than WENO-JS. On the other hand, in the case of ID and WENO-ZA (bottom left panel), the jet-like structure near the bottom reflecting wall looks distorted; as a matter of fact, in this case, the simulation crashes with CFL = 0.8, and hence CFL = 0.5 is used. This indicates that WENO-ZA may be more prone to numerical artifacts than WENO-JS and WENO-Z.

3.2.5. Kelvin–Helmholtz Instability

The KH instability occurs at the interface of two slipping fluids, such as the jet-backflow interface in relativistic jets (see Paper II). It has been widely employed to evaluate the ability to handle the linear growth and the development of nonlinear structures in hydrodynamic codes (e.g., Zhang & MacFadyen 2006; Radice & Rezzolla 2012, for the test of RHD codes). We here present a test of KH instability to illustrate the effects of the transverse-flux averaging, described in

Section 2.5, as well as the ability to reproduce the linear growth. The initial condition is given as

$$
\boldsymbol{u}(x, y) = \begin{cases} (-(\rho_1 - \rho_2)/2 \tanh[(y - 0.25)/\delta] + (\rho_1 + \rho_2)/2, \ -U \tanh[(y - 0.25)/\delta], \ 0.001 \sin(kx), \ p) & \text{for } y > 0, \\ (\ (\rho_1 - \rho_2)/2 \tanh[(y + 0.25)/\delta] + (\rho_1 + \rho_2)/2, \ \ U \tanh[(y + 0.25)/\delta], \ 0.001 \sin(kx), \ p) & \text{for } y < 0, \end{cases} \tag{49}
$$

in the periodic domain of $[-0.5, 0.5] \times [-0.5, 0.5]$. Then, $\rho \approx \rho_1$ and $v_x \approx U$ at $y = 0$, and $\rho \approx \rho_2$ and $v_x \approx -U$ at $y = \pm 0.5$, with shear interfaces of thickness δ at $y = \pm 0.25$. We set $\rho_1 = 2$, $\rho_2 = 1$, $U = 0.2$, $p = 2.5$, $\delta = 0.01$, and $k = 2\pi$. A perturbation of the box size with a small amplitude is introduced in v_y to initiate the instability. The computational domain is covered with either 600×600 or 1200×1200 grid zones.

The linear growth of KH instability can be analyzed with the linearized version of the RHD Equations in (1)–(3). The growth rate was obtained, for instance, in Bodo et al. (2004), for the case of continuous density across the interface, i.e., $\rho_1 = \rho_2$. Following the same procedure, it can be derived for the case with a density jump across the interface, such as the one above. Denoting the perturbed quantities as $\delta \tilde{q} \propto \exp[i(kx - \omega t)]$, the dispersion relation is

$$
\left(\frac{\omega}{k}\right)^6 \left(\mathcal{R}_- + \frac{U^2}{c^4}\mathcal{C}_-\right)
$$
$$
+ 2\left(\frac{\omega}{k}\right)^5 U\left[\mathcal{R}_+ + \left(\frac{1}{c^2} - 2\frac{U^2}{c^4}\right)\mathcal{C}_+\right]
$$
$$
- \left(\frac{\omega}{k}\right)^4 U^2\left[\mathcal{R}_- - \left(\frac{1}{U^2} - \frac{8}{c^2} + 6\frac{U^2}{c^4}\right)\mathcal{C}_-\right]
$$
$$
- 4\left(\frac{\omega}{k}\right)^3 U^3\left[\mathcal{R}_+ + \left(\frac{1}{U^2} - \frac{3}{c^2} + \frac{U^2}{c^4}\right)\mathcal{C}_+\right]
$$
$$
- \left(\frac{\omega}{k}\right)^2 U^4\left[\mathcal{R}_- - \left(\frac{6}{U^2} - \frac{8}{c^2} + \frac{U^2}{c^4}\right)\mathcal{C}_-\right]
$$
$$
+ 2\left(\frac{\omega}{k}\right) U^5\left[\mathcal{R}_+ - \left(\frac{2}{U^2} - \frac{1}{c^2}\right)\mathcal{C}_+\right]
$$
$$
+ U^6\left(\mathcal{R}_- + \frac{1}{U^2}\mathcal{C}_-\right) = 0, \tag{50}
$$

where $\mathcal{R} = (\rho_1 h_1 c_{s,1})/(\rho_2 h_2 c_{s,2})$, $\mathcal{R}_\pm = 1 \pm \mathcal{R}^2$, and $\mathcal{C}_\pm = c_{s,2}^2 \mathcal{R}^2 \pm c_{s,1}^2$. Here, the subscripts 1 and 2 mark the quantities with the density ρ_1 and ρ_2, and the pressure is continuous across the interface.

The left panel of Figure 7 compares the growth of perturbation in the linear phase as measured by the root-mean-square (rms) y-velocity, $\langle v_y^2 \rangle^{1/2}$, with the predicted growth of the $k = 2\pi$ mode from Equation (50). After the initial adjustment, the simulations show an exponential growth before reaching saturation. Our test demonstrates that, with the resolutions employed, the simulations closely reproduce the predicted growth of the linear analysis, regardless of whether the transverse-flux averaging is included or not.

The right panel of Figure 7 presents the rest-mass density from the simulations without and with the transverse-flux averaging, as well as those using two different grid resolutions; the results are presented at $t = 5$ (the end of the linear stage) for

the simulations with 600×600 grid zones, and at $t = 6.5$ (the nonlinear stage) for the simulations with 1200×1200 grid zones, in order to depict the differences caused by the transverse-flux averaging at different evolution phases. Overall, the eddy structures of KH vortices, especially on small scales, better develop by including the transverse-flux averaging. Such structures commonly emerge in astrophysical flows, for instance, in relativistic jets. Hence, we incorporate the transverse-flux averaging as a default component of our code.

3.3. Relativistic Jets

As the main purpose of this paper is to describe an RHD code developed for studies of ultra-relativistic astrophysical jets, we have extensively performed test simulations of jets with different EOSs and different WENO versions as well as different setups for the components of the code. Only some of them are presented here. We point out that jet is one of the most popular test problems for RHD codes, and hence many papers for RHD codes presented jet tests (see the *Introduction* for references).

3.3.1. 2D Plane-parallel Jet

We first show a test of a 2D plan-parallel (slab) jet. The structures of jet and cocoon materials can be visualized more easily with 2D jets than with 3D jets. Simulations have been performed with a light relativistic jet, injected horizontally into the uniform background medium, in a pressure equilibrium:

$$
\begin{cases} \boldsymbol{u}_{\text{jet}} = (10^{-2}, 0.99, 0, 10^{-3}), \\ \boldsymbol{u}_{\text{bkg}} = (1, 0, 0, 10^{-3}), \end{cases} \tag{51}
$$

where $\boldsymbol{u}_{\text{jet}}$ and $\boldsymbol{u}_{\text{bkg}}$ denote the jet and background states, respectively. The jet speed of $v_{\text{jet}} = 0.99$ corresponds to the Lorentz factor of $\Gamma_{\text{jet}} = 7$. The computational domain covers $[0, 3.5] \times [0, 1]$ with 1050×300 grid zones. A jet nozzle with the width (diameter) of 0.1 is placed at $(x, y) = (0.0, 0.5)$. The boundary condition is inflow at the jet nozzle, and outflow elsewhere.

Figure 8 compares the results of simulations with different WENO versions. In all three cases, the convergence-divergence structure inside the jet spine and the turbulent structure inside the cocoon are well reproduced. With WENO-ZA, clearly more structures develop; however, feather-like patterns appear at the interface of the jet-backflow and also in the backflow, and it is not clear whether they are all physical. Between WENO-JS and WENO-Z, the latter produces richer structures. The results of our simulations are favorably compared with those of previous studies, for example, Figure 8 in Núñez-de la Rosa & Munz (2016), although the jet parameters as well as the EOS in other studies may not be necessarily identical to ours.

3.3.2. Carbuncle Instability in 3D Jet Simulations

As described in Section 2.6, in order to suppress the carbuncle instability at shocks, our code includes the modification of
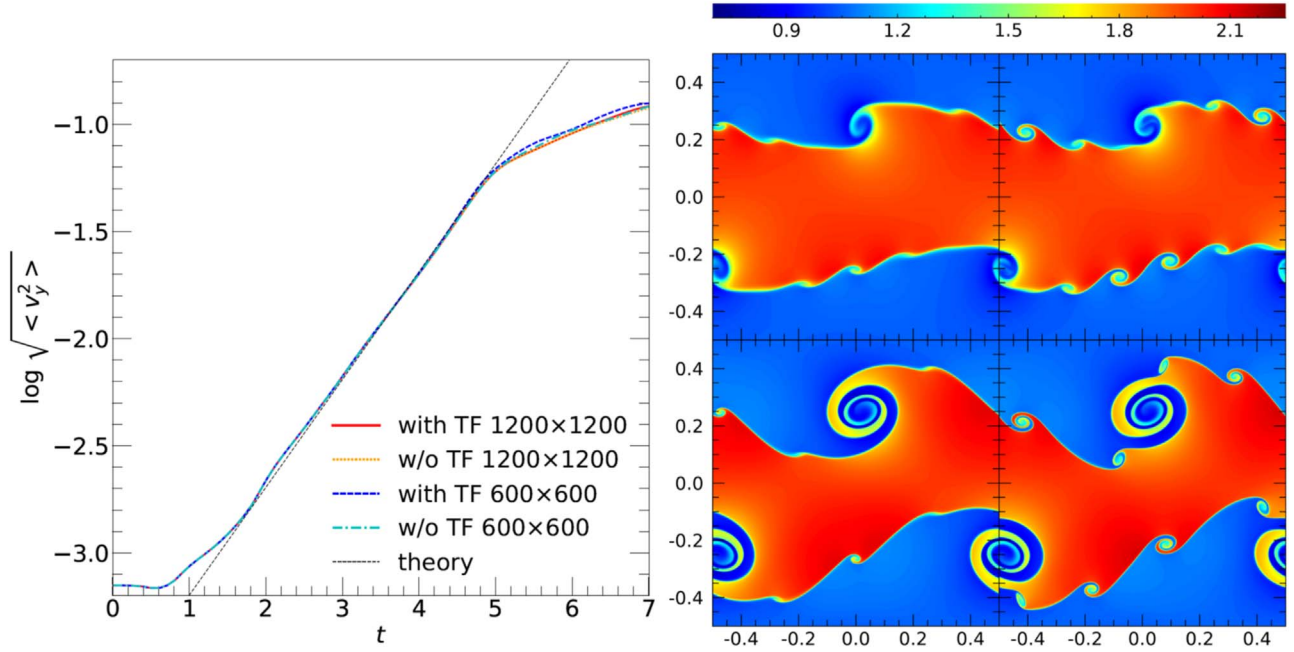
Figure 7. Relativistic KH instability test with the initial condition in Equation (49). Left panel: The time evolution of the rms y-velocity, $\langle v_y^2 \rangle^{1/2}$, is plotted for simulations with/without transverse-flux averaging and with 600×600 and 1200×1200 grid zones. The black doted line draws the predicted linear growth of the instability from Equation (50). Right panel: The rest-mass density, ρ, is shown for simulations without (left) and with (right) transverse-flux averaging and with 600×600 grid zones at $t = 5$ (top) and 1200×1200 grid zones at $t = 6.5$ (bottom). The computational domain covers $[-0.5, 0.5] \times [-0.5, 0.5]$. The code equipped with the default setup (WENO-Z, SSPRK, RC, and CFL = 0.8) is used.

eigenvalues for the acoustic modes, suggested by Fleischmann et al. (2020). We here present 3D jet simulations to demonstrate that the carbuncle instability in the jet is effectively suppressed with this modification.

The morphology of radio galaxy jets, including the FR-I and FR-II dichotomy, is known to depend mainly on the energy injection rate, called the jet power,

$$Q_{\rm jet} = \pi r_{\rm jet}^2 v_{\rm jet}(\Gamma_{\rm jet}^2 \rho_{\rm jet} h_{\rm jet} - \Gamma_{\rm jet} \rho_{\rm jet}), \qquad (52)$$

defined with the jet quantities including the jet radius, $r_{\rm jet}$ (see, e.g., Kaiser & Alexander 1997; Godfrey & Shabala 2013, Paper II). We have found that, in simulations of high-power jets, the head advances fast and the bow shock surface is relatively stable to the carbuncle instability. By contrast, in low-power jets, the head strolls rather slowly, and the bow shock surface is more prone to the instability.

To illustrate this, we show simulations for two cases representing high-power and low-power jets. The 3D jets are set up with the RHD state, $\boldsymbol{u} = (\rho, v_x, v_y, v_z, p)$, as follows:

$$\begin{cases} \boldsymbol{u}_{\rm jet} = (\eta,\, v_{\rm jet}\sin(\theta_{\rm pre})\cos(2\pi t/t_{\rm pre}),\, v_{\rm jet}\sin(\theta_{\rm pre})\sin(2\pi t/t_{\rm pre}),\, v_{\rm jet}\cos(\theta_{\rm pre}),\, p), \\ \boldsymbol{u}_{\rm bkg} = (1, 0, 0, 0, p). \end{cases}$$
$$(53)$$

For the high-power jet, the density contract is set to be $\eta = 10^{-5}$ and the jet speed is $v_{\rm jet} = 0.9999$ (or $\Gamma_{\rm jet} = 71$); for the low-power jet, $\eta = 10^{-4}$ and $v_{\rm jet} = 0.9729$ (or $\Gamma_{\rm jet} = 4.3$). The computational domain consists of $[0, 1] \times [0, 1] \times [0, 2]$ for the high-power jet and $[0, 1] \times [0, 1] \times [0, 1]$ for the low-power jet, respectively. The radius of the jet nozzle is $r_{\rm jet} = 0.03$ for both cases.

The pressure is set to be $p = 7.64 \times 10^{-6}$ in both cases, which corresponds to that of the typical ICM with hydrogen number density, $n_{\rm H,ICM} = 10^{-3}$ cm^{-3}, and temperature, $T_{\rm ICM} = 5 \times 10^7$ K.

Assuming that the jet radius is $r_{\rm jet} = 1$ kpc, the jet power is $Q_{\rm jet} = 3.34 \times 10^{47}$ erg s^{-1} and 3.34×10^{45} erg s^{-1} for the high-power and low-power jets, respectively. Note that $Q_{\rm jet}$ for the high-power jet is relevant for FR-II jets, while $Q_{\rm jet}$ for the low-power jet lies at the border dividing the FR-I and FR-II dichotomy (e.g., Godfrey & Shabala 2013).

In the case of both high-power and low-power jets, to break the rotational symmetry, a slow, small-angle precession is added to the jet velocity, $v_{\rm jet}$, with precession angle and period, $\theta_{\rm pre} = 0°\!.5$ and $t_{\rm pre} = 10\, t_{\rm cross}$, respectively. To ensure a smooth start-up of the jets, preventing possible developments of unphysical structures at the tip of the jets in the early stage, a windowing,

$$v_{\rm jet}^* = v_{\rm jet}[1 - (r/r_{\rm jet})^n] \text{ with } n = 20, \qquad (54)$$

is applied up to $t = 1\, t_{\rm cross}$, where r is the radial distance from the jet center. Here, $t_{\rm cross}$ is the jet crossing time defined as $t_{\rm cross} \equiv r_{\rm jet}/v_{\rm head}^*$, where $v_{\rm head}^*$ is the advance speed of the jet head, approximately estimated assuming the balance between the jet ram pressure and the background pressure (Martí et al. 1997, Paper II). The dynamical evolution of jet-induced flows can be characterized with $t_{\rm cross}$ (see paper II).

Figure 9 shows the rest-mass density, ρ, at $t_{\rm end} = 50\, t_{\rm cross}$ for the high-power jet and at $t_{\rm end} = 13\, t_{\rm cross}$ for the low-power jet, from simulations using $200 \times 200 \times 400$ and $200 \times 200 \times 200$ grid zones, respectively. From the left to right panels, the cases with increasingly greater degrees of the modification of eigenvalues described in Equation (35), $\phi = \infty$, 10, and 5, are shown . Without the modification ($\phi = \infty$), the carbuncle instability appears at the bow shock in both cases, and will eventually disrupt the jets, if the simulations run longer. The figure illustrates that the instability is suppressed, if the modifications with $\phi = 10$ and 5 are applied to the high-power and low-power jets, respectively. The structures in the jet spine,
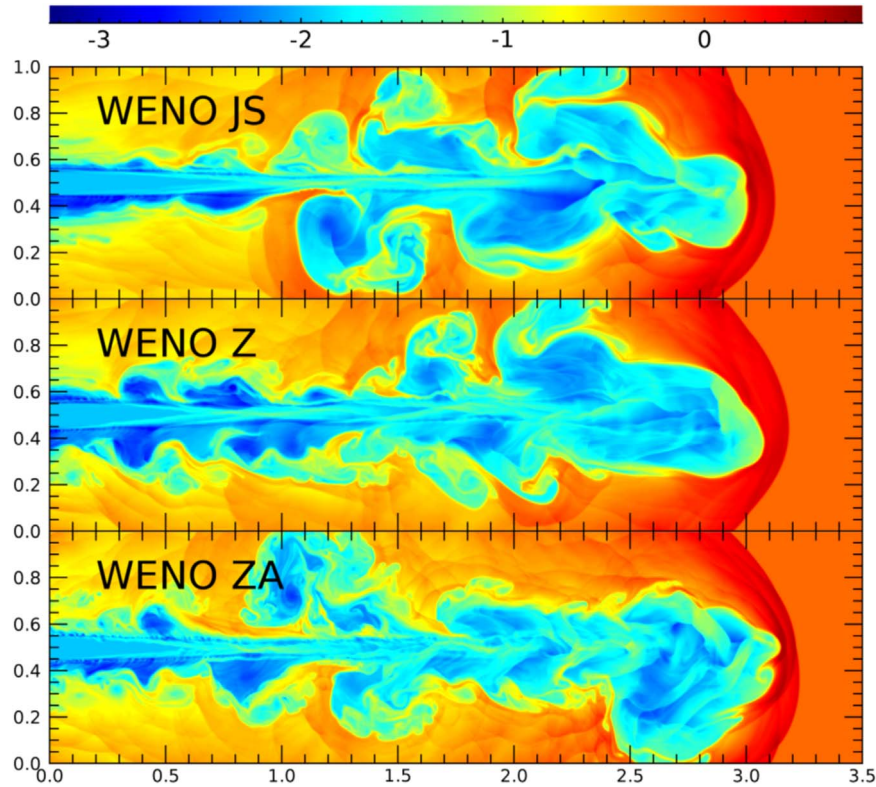
Figure 8. 2D plane-parallel (slab) jet with the Lorentz factor $\Gamma_{\rm jet} = 7$, injected from the left boundary into a uniform background. The jet and background states are given in Equation (51). The computational domain consists of $[0, 3.5] \times [0, 1]$ with 1050×300 grid zones, and the width of the jet nozzle is 0.1. The rest-mass density, $\log \rho$, from simulations using different WENO versions, WENO-JS (top), WENO-Z (middle), and WENO-ZA (bottom), is shown. Otherwise, the default code setup with SSPRK, RC, CFL = 0.8, and the transverse-flux averaging is adopted.

backflow, and shocked ICM, on the other hand, are not much affected by the modification. In general, with our code, the instability is effectively suppressed in simulations with $Q_{\rm jet} \gtrsim 10^{46}$ erg s^{-1} if $\phi \simeq 10$ is set, while $\phi \simeq 5$ is required in simulations with $Q_{\rm jet} \lesssim 10^{46}$ erg s^{-1}.

3.3.3. 3D Relativistic Jet

As the final test, we present 3D simulations of a high-power FR-II jet with different EOSs and different WENO versions. The jet parameters are the same as those of the high-power jet in the upper panels of Figure 9. The computational domain is [0, 1] × [0, 1] × [0, 1] with 200 × 200 × 200 grid zones, smaller than that of Figure 9. Figure 10 shows the rest-mass density, ρ, at $t_{\rm end} = 25 \, t_{\rm cross}$ from simulations with 12 combinations of different EOSs and different WENO versions. The modification of eigenvalues with $\phi = 10$ is included. When the upward-moving jet penetrates into the ICM, the jet flow is halted at the jet head, and reflected backward, producing a cocoon of downward-moving backflow with the low-density jet material. In addition, a bow shock forms in the ICM, encompassing the shocked ICM and the jet cocoon. The KH instability commonly develops at the interface between the jet spine and backflow and also at the interface between the backflow and shocked ICM.

Figure 10 demonstrates that the advance of the jet and also the overall morphology depend on the EOS. The structures developed in the simulations with RC and TM are comparable. However, the results using ID, with both $\gamma = 5/3$ and $4/3$, look sufficiently different. In fact, in the simulations with RC and TM, while the injected jet material has the adiabatic index $\gamma \simeq 1.45$, it varies from $\gamma \simeq 4/3$ at the shocked jet flow to

$\gamma \simeq 5/3$ at the shocked ICM. Inspecting the figure closely reveals that WENO-JS is the most diffusive among the three WENO versions, while WENO-ZA is the least diffusive and produces the most complex structures. However, as noted above (e.g., Figures 5–6), WENO-ZA sometimes produces suspicious features. Considering both the accuracy and robustness, we hence employ WENO-Z as the default WENO reconstruction scheme for our RHD code.

4. Summary and Discussion

To study the properties of ultra-relativistic astrophysical jets with numerical simulations, we have developed an RHD code based on the fifth-order accurate FD WENO scheme (Jiang & Shu 1996). As an effort to find an optimal code to emulate accurately and robustly the nonlinear structures and flow dynamics as well as the thermodynamics in the jets, we have incorporated several recent developments available in the literature: (1) ID, TM, and RC EOSs; (2) WENO-JS, WENO-Z, and WENO-ZA versions; (3) RK4 and SSPRK time-integration methods; (4) a high-order accurate averaging of fluxes along the transverse directions; and (5) a modification of eigenvalues for the acoustic modes to suppress the carbuncle instability.

We have performed an extensive set of test simulations for 1D, 2D, and 3D problems, to evaluate the effectiveness of these implementations and hence to identify the best combination of different options for the studies of relativistic jets. In conclusion, we have chosen the RC EOS, WENO-Z scheme, and SSPRK integration method as the fiducial setup of our code. In addition, we include the fourth-order accurate averaging of fluxes along the transverse directions and the modification of eigenvalues for the
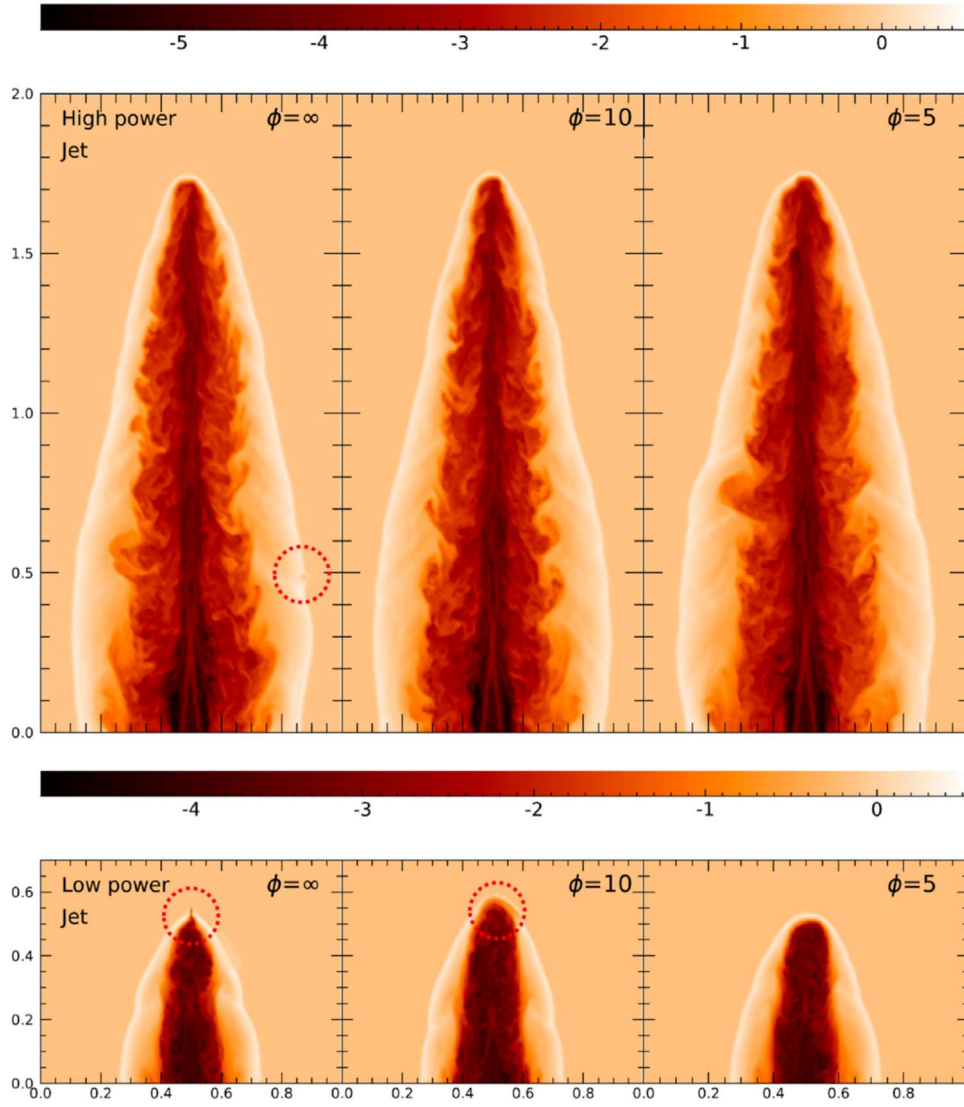
Figure 9. 3D jet simulations to demonstrate the suppression of the carbuncle instability with the modification of eigenvalues described in Equation (35). The red circles enclose the regions where the instability starts to appear. The left panels show the simulations *without* the modification of eigenvalues, while the center and right panels are specified *with* the modification parameter of $\phi = 10$ and 5, respectively. The top panels display a high-power jet of the Lorentz factor of $\Gamma_{\rm jet} = 71$, while the bottom panels show a low-power jet of the Lorentz factor of $\Gamma_{\rm jet} = 4.3$. The jet is injected from the center of the bottom x–y plane into a uniform background. The computational domain of $[0, 1] \times [0, 1] \times [0, 2]$ and $[0, 1] \times [0, 1] \times [0, 1]$ is covered with $200 \times 200 \times 400$ and $200 \times 200 \times 200$ grid zones for the high-power and low-power jets, and the radius of the jet nozzle is 0.03 in both the jets. See the main text for the details of the jet setups. The slice images of $\log_{10} \rho$ in the x–z plane through $y = 0.5$ are shown at $t_{\rm end} = 50 t_{\rm cross}$ (top) and 13 $t_{\rm cross}$ (bottom). The bottom panels include a cropped region of $0 \leqslant z \leqslant 0.7$. The default code setup with WENO-Z, SSPRK, RC, CFL = 0.8, and the transverse-flux averaging is adopted.

acoustic modes suggested by Fleischmann et al. (2020) in the code.

The main points of this paper can be summarized as follows:

1. In simulations of ultra-relativistic jets, in which the jet of $\gamma \simeq 4/3$ with relativistic temperature penetrates into the ICM of $\gamma \simeq 5/3$ with subrelativistic temperature, the two fluids are mixed by turbulence in the cocoon, producing the transition from $\gamma \simeq 4/3$ to $\gamma \simeq 5/3$. In order to follow such process, either the RC or TM EOS should be used, instead of the ID EOS with a fixed γ. We find RC and TM produce comparable results in various tests presented here. Yet, considering that RC approximates RP more closely than TM (Ryu et al. 2006), we adopt RC as the default EOS.

2. Among the three versions of WENO schemes considered here, WENO-JS is the most diffusive, while WENO-ZA is the least diffusive, as expected. WENO-ZA tends to produce richer structures, but at the same time seems to be less robust under harsh conditions such as those of relativistic jets. Considering that WENO-Z is sufficiently accurate and also robust, it is selected as the default option to be used for our simulations of relativistic jets.

3. Between the two fourth-order accurate time-integration methods, SSPRK yields better results than RK4 for simulations of shocks with large transverse velocities. We hence adopt SSPRK as the default time-integration method.

4. Borrowing the idea of the modified dimension-by-dimension method suggested for FV WENO schemes
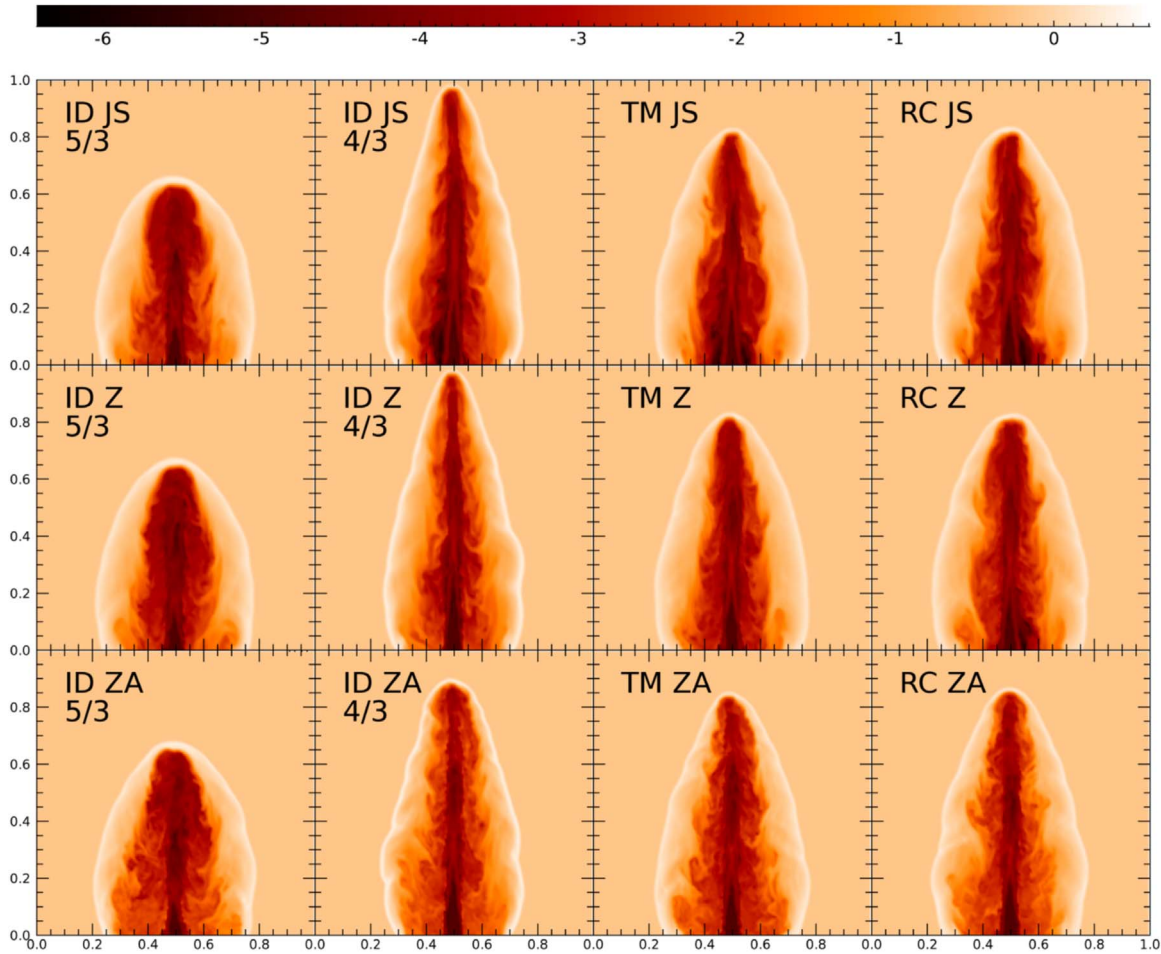
Figure 10. 3D jet of the Lorentz factor, $\Gamma_{jet} = 71$, injected from the center of the bottom x–y plane into a uniform background. The computational domain of $[0, 1] \times [0, 1] \times [0, 1]$ is covered with $200 \times 200 \times 200$ grid zones, and the radius of the jet nozzle is 0.03. See the main text for the details of the jet setup. The slice images of $\log_{10} \rho$ in the x–z plane through $y = 0.5$ from simulations with different EOSs, ID with $\gamma = 5/3$ (first column), ID with $\gamma = 4/3$ (second column), TM (third column), and RC (fourth column), using different WENO versions, WENO-JS (top), WENO-Z (middle), and WENO-ZA (bottom), are shown at at $t_{end} = 25 \, t_{cross}$. In all the simulations, the transverse-flux averaging and the fix for the carbuncle instability with $\phi = 10$ are implemented, and the SSPRK time-integration method with CFL = 0.8 is used.

(Buchmüller & Helzel 2014; Buchmüller et al. 2016), we implement a fourth-order accurate averaging of fluxes along the transverse directions in smooth-flow regions. It improves the performance in some multidimensional problems, especially in those involving complex flows.

5. In our simulations of relativistic jets, the bow shock surface is frequently subject to the carbuncle instability. We find that the modification of eigenvalues for the acoustic modes described in Equation (35) effectively suppresses the instability with the parameter, $\phi \sim 5$–10.

In an accompanying paper (Paper II), using the RHD code presented here, we report a simulation study of ultra-relativistic jets ejected from AGNs into the uniform ICM. In particular, we describe in detail the flow dynamics, such as shocks, velocity shear, and turbulence, induced by the jets.

ORCID iDs

Jeongbhin Seo (ID) https://orcid.org/0000-0002-5550-8667
Hyesung Kang (ID) https://orcid.org/0000-0002-4674-5687
Dongsu Ryu (ID) https://orcid.org/0000-0002-5455-2957
Indranil Chattopadhyay (ID) https://orcid.org/0000-0002-2133-9324

References

Aloy, M. A., Ibáñez, J. M., Martí, J. M., & Müller, E. 1999, ApJS, 122, 151
Bodo, G., Mignone, A., & Rosner, R. 2004, PhRvE, 70, 036304
Borges, R., Carmona, M., Costa, B., & Don, W. S. 2008, JCoPh, 227, 3191
Buchmüller, P., Dreher, J., & Helzel, C. 2016, ApMaC, 272, 460
Buchmüller, P., & Helzel, C. 2014, JSCom, 61, 343
Chattopadhyay, I., & Ryu, D. 2009, ApJ, 694, 492
Del Zanna, L., & Bucciantini, N. 2002, A&A, 390, 1177
Dolezal, A., & Wong, S. 1995, JCoPh, 120, 266
Duffell, P. C., & MacFadyen, A. I. 2011, ApJS, 197, 15
Dumbser, M., Moschetta, J.-M., & Gressier, J. 2004, JCoPh, 197, 647
Falle, S. A. E. G., & Komissarov, S. S. 1996, MNRAS, 278, 586
Fleischmann, N., Adami, S., Hu, X. Y., & Adams, N. A. 2020, JCoPh, 401 109004

Gaensler, B. M., & Slane, P. O. 2006, ARA&A, 44, 17
Godfrey, L. E. H., & Shabala, S. S. 2013, ApJ, 767, 12
Gottlieb, S. 2005, JSCom, 25, 105
Ha, Y., Kim, C. H., Lee, Y. J., & Yoon, J. 2013, JCoPh, 232, 68
Hanawa, T., Mikami, H., & Matsumoto, T. 2008, in ASP Conf. Ser., 385,
    Numerical Modeling of Space Plasma Flows, ed. N. V. Pogorelov,
    E. Audit, & G. P. Zank (San Francisco, CA: ASP), 259
Hardcastle, M. J., & Croston, J. H. 2020, NewAR, 88, 101539
He, P., & Tang, H. 2012, CCoPh, 11, 114
Henrick, A. K., Aslam, T. D., & Powers, J. M. 2005, JCoPh, 207, 542
Hu, X., Wang, Q., & Adams, N. A. 2010, JCoPh, 229, 8952
Jiang, G.-S., & Shu, C.-W. 1996, JCoPh, 126, 202
Jiang, G.-S., & Wu, C.-C. 1999, JCoPh, 150, 561
Joshi, R. K., Chattopadhyay, I., Ryu, D., & Yadav, L. 2021, MNRAS,
    502, 5227
Kaiser, C. R., & Alexander, P. 1997, MNRAS, 286, 215
Landau, L. D., & Lifshitz, E. M. 1959, Fluid Mechanics (Oxford: Pergamon)
Lister, M. L., Aller, M. F., Aller, H. D., et al. 2013, AJ, 146, 120
Liu, S., Shen, Y., Zeng, F., & Yu, M. 2018, IJNMF, 87, 271
Liu, X.-D., Osher, S., & Chan, T. 1994, JCoPh, 115, 200
Martı, J. M., & Müller, E. 1996, JCoPh, 123, 1
Martí, J. M., & Müller, E. 2003, LRR, 6, 7
Martí, J. M., & Müller, E. 2015, LRCA, 1, 3
Martí, J.-M., Müller, E., Font, J. A., Ibáñez, J.-M., & Marquina, A. 1997, ApJ,
    479, 151
Mathews, W. G. 1971, ApJ, 165, 147
Mignone, A., & Bodo, G. 2005, MNRAS, 364, 126

Mignone, A., Plewa, T., & Bodo, G. 2005, ApJS, 160, 199
Nava, L., Desiante, R., Longo, F., et al. 2017, MNRAS, 465, 811
Núñez-de la Rosa, J., & Munz, C.-D. 2016, MNRAS, 460, 535
Pandolfi, M., & D'Ambrosio, D. 2001, JCoPh, 166, 271
Peery, K., & Imlay, S. 1988, in 24th Joint Propulsion Conf. (Reston, VA:
    AIAA Journal), 2904
Piran, T. 2005, Rev. Mod. Phys., 76, 1143
Pons, J. A., Ma Martí, J., & Müller, E. 2000, JFM, 422, 125
Qin, T., Shu, C.-W., & Yang, Y. 2016, JCoPh, 315, 323
Radice, D., & Rezzolla, L. 2012, A&A, 547, A26
Reyes, A., Lee, D., Graziani, C., & Tzeferacos, P. 2019, JCoPh, 381, 189
Ryu, D., Chattopadhyay, I., & Choi, E. 2006, ApJS, 166, 410
Savolainen, T., Homan, D. C., Hovatta, T., et al. 2010, A&A, 512, A24
Schneider, V., Katscher, U., Rischke, D., et al. 1993, JCoPh, 105, 92
Seo, J., Kang, H., & Ryu, D. 2021, ApJ, 920, 144
Shu, C.-W. 2009, SIAMR, 51, 82
Shu, C.-W., & Osher, S. 1988, JCoPh, 77, 439
Shu, C.-W., & Osher, S. 1989, JCoPh, 83, 32
Spiteri, R. J., & Ruuth, S. J. 2002, SJNA, 40, 469
Spiteri, R. J., & Ruuth, S. J. 2003, Mathematics and Computers in Simulation,
    62, 125
Synge, J. L. 1957, The Relativistic Gas, Series in Physics (Amsterdam: North-
    Holland)
Taub, A. H. 1948, PhRv, 74, 328
Woodward, P., & Colella, P. 1984, JCoPh, 54, 115
Zhang, R., Zhang, M., & Shu, C.-W. 2011, CCoPh, 9, 807
Zhang, W., & MacFadyen, A. I. 2006, ApJS, 164, 255