

## AMS260 Final Term Coding Project

**Posted on Tue, Mar 2, 2021**  
**Due 11:59 pm, Thu, Mar 18, 2021**

**Submit your homework to your Git repository by 11:59 pm**

---

### 1D Finite Volume Shock Capturing Methods for the Euler Equations

---

In this project we implement finite volume conservative methods to solve the compressible 1D Euler equations,

$$\mathbf{U}_t + \left( \mathbf{F}(\mathbf{U}) \right)_x = 0, \quad (1)$$

where the conservative variables  $\mathbf{U}$  and the associated flux function  $\mathbf{F}(\mathbf{U})$  are given by

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho E \end{pmatrix}, \text{ and } \mathbf{F} = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ u(\rho E + p) \end{pmatrix}. \quad (2)$$

The fluid variables in the Euler equations include the mass density  $\rho$ , the  $x$ -momentum  $m = \rho u$ , and the total energy per unit mass as a sum of the kinetic energy  $\rho u^2/2$  and the internal energy  $\rho e$ ,

$$\rho E = \rho \left( \frac{u^2}{2} + e \right), \quad (3)$$

with the specific internal energy  $e$  given by a simple ideal gas law equation of state (EoS),

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

We set the ratio of specific heats  $\gamma$  as a constant value,  $\gamma = 1.4$ , for the project. You may use different values of  $\gamma$ , for instance,  $\gamma = 5/3 = 1.666667$  to explore what differences you get (e.g., check the shock jump strength of the Sod shock tube problem using different  $\gamma$  values), but all test results you report should use  $\gamma = 1.4$  unless you are told otherwise.

The current template code (provided) works on the following minimum set of configurations:

- (a) the first-order Godunov (FOG) method for reconstruction
- (b) the Minmod slope limiter
- (c) the HLL Riemann solver
- (d) the first-order forward Euler temporal update from  $\mathbf{U}^n$  to  $\mathbf{U}^{n+1}$  – this is all you need for FOG; however, you will need to implement the second-order characteristic tracing method for PLM and PPM

- (e) outflow and reflecting boundary conditions
- (f) the Sod's 1D shock tube problem

The goals in the project are to:

1. study the first-order Godunov (FOG) template code,
2. extend the FOG code to the second-order piecewise linear method (PLM) and the third-order piecewise parabolic method (PPM),
3. implement two other slope limiters of van Leer and MC for both PLM and PPM,
4. implement two other Riemann solvers of HLLC (the reference sections from the book by Toro are attached) and Roe,
5. run four benchmarked problems (see Section 2.) using different reconstruction methods with different Riemann solvers,
6. analyze your results by conducting comparison studies with different combinations of reconstruction methods and Riemann solvers, e.g.,
  - ( $\alpha$ ) FOG+HLL
  - ( $\beta$ ) FOG+HLLC
  - ( $\gamma$ ) FOG+Roe
  - ( $\delta$ ) PLM+HLL
  - ( $\epsilon$ ) PLM+HLLC
  - ( $\zeta$ ) PLM+Roe
  - ( $\eta$ ) PPM+HLL
  - ( $\theta$ ) PPM+HLLC
  - ( $\iota$ ) PPM+Roe

As a reference, you're given one example problem called the Sod's 1D shock tube problem. Use this Sod problem to check and see if all your implementations (the above nine cases) are correctly working on the Sod's shock tube problem. Once you see all your implementations are correct on the Sod problem, you're going to setup and run three new test problems.

## 1. Grid Discretization

We adopt our 1D grid as before, following the cell-centered (rather than cell interface-centered) notation for discrete cells  $x_i$  and the conventional temporal discretization  $t^n$ :

$$x_i = (i - \frac{1}{2})\Delta x, \quad (5)$$

$$t^n = n\Delta t. \quad (6)$$

Then the cell interface-centered grid points are written using the ‘half-integer’ indices:

$$x_{i+\frac{1}{2}} = x_i + \frac{\Delta x}{2}. \quad (7)$$

We take the number of guardcells  $N_{ngc} = 2$  on each side of the domain, resulting the following grid configuration with  $N_x$  interior grid resolutions:

- two guardcells on the left:  $\{x_i : 1 \leq i \leq 2\}$ ,
- $N_x$  interior points:  $\{x_i : N_{ngc} + 1 \leq i \leq N_{ngc} + N_x\}$
- two guardcells on the right:  $\{x_i : N_{ngc} + N_x + 1 \leq i \leq 2N_{ngc} + N_x\}$ .

## 2. Four Test Problems

Take a look at the result section of Lee *et al.* JCP, 341 (2017) (attached) for code verification. Your code results are expected to look very similar to the results therein. Another good resource is the FLASH code website where you can find the user’s guide with examples: FLASH User’s Guide.

### 2.1. A given example: Sod’s Shock Tube Problem

The Sod problem (Sod 1978) is a one-dimensional flow discontinuity problem that provides a good test of a compressible code’s ability to capture shocks and contact discontinuities with a small number of cells and to produce the correct profile in a rarefaction. It also tests a code’s ability to correctly satisfy the Rankine-Hugoniot shock jump conditions.

We construct the initial conditions for the Sod problem on the computational domain  $[0, 1]$  by establishing a single jump discontinuity. The fluid is initially at rest on either side of the interface, and the density and pressure jumps are chosen so that all three types of nonlinear, hydrodynamic waves (shock, contact, and rarefaction) develop. To the “left” and “right” of the interface we have

$$\mathbf{V}(x, 0) = \begin{cases} \begin{pmatrix} \rho \\ u \\ p \end{pmatrix}_L = \begin{pmatrix} 1.0 \\ 0.0 \\ 1.0 \end{pmatrix} & \text{if } x \leq 0.5, \\ \begin{pmatrix} \rho \\ u \\ p \end{pmatrix}_R = \begin{pmatrix} 0.125 \\ 0.0 \\ 0.1 \end{pmatrix} & \text{if } x > 0.5. \end{cases} \quad (8)$$

The ratio of specific heats  $\gamma$  is chosen to be 1.4 on both sides of the interface. The outflow boundary condition is used.

### 2.2. Rarefaction Wave

This problem does not contain any jump discontinuities and is smooth, hence it is a good test problem for convergence test. The initial condition on the computational domain  $[0, 1]$  is given by:

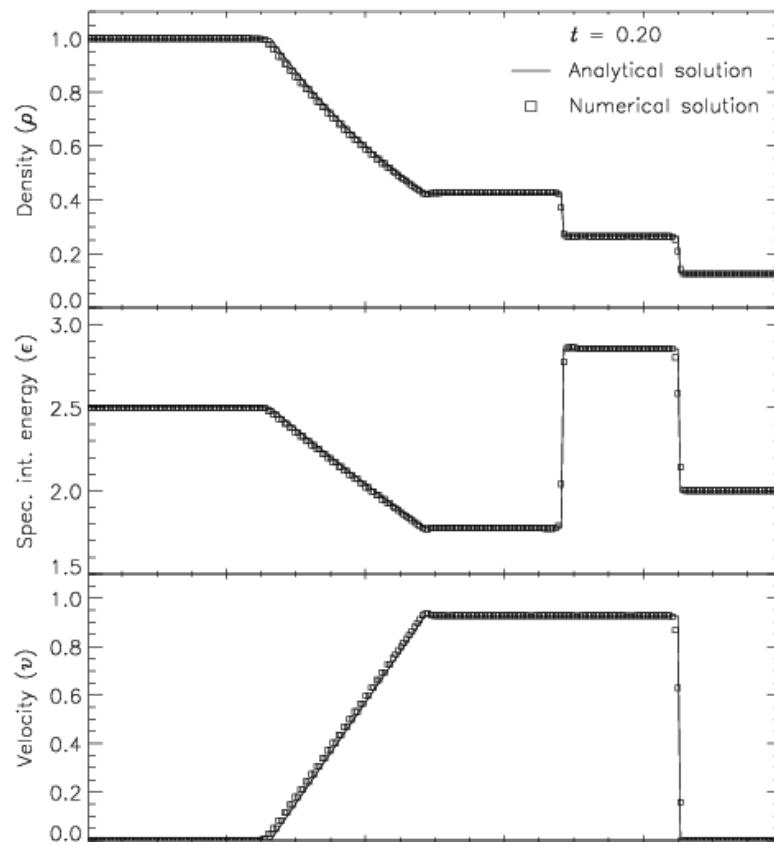


Figure 1. Comparison of numerical and analytical solutions to the Sod problem using the FLASH code. The simulated result is sampled at  $t = 0.2$ .

$$\mathbf{V}(x, 0) = \begin{cases} \begin{pmatrix} \rho \\ u \\ p \end{pmatrix}_L = \begin{pmatrix} 1.0 \\ -2.0 \\ 0.4 \end{pmatrix} & \text{if } x \leq 0.5, \\ \begin{pmatrix} \rho \\ u \\ p \end{pmatrix}_R = \begin{pmatrix} 1.0 \\ 2.0 \\ 0.4 \end{pmatrix} & \text{if } x > 0.5. \end{cases} \quad (9)$$

The ratio of specific heats  $\gamma$  is chosen to be 1.4 on both sides of the interface. Please use  $t_{\max} = 0.15$ . The outflow boundary condition is applied to this problem.

### 2.3. Interacting Blast-Wave: Blast2

This Blast2 problem was originally used by Woodward and Colella (1984) to compare the performance of several different hydrodynamical methods on problems involving strong shocks and narrow features. It has no analytical solution (except at very early times), but since it is one-dimensional, it is easy to produce a converged solution by running the code with a very large number of cells, permitting a reference solution to compare with.

Reflecting boundary conditions are used, where the velocity  $u$  is negated in the guardcell regions in a symmetric way, i.e., assuming  $N_{ngc} = 2$ ,

$$u_i = -u_{k-i} \quad \text{for } i = 1, 2, \quad (10)$$

on the left boundary with  $k = 2N_{ngc} + 1$ . Similarly on the right boundary we have

$$u_i = -u_{k-i} \quad \text{for } i = N_x + N_{ngc} + 1, N_x + 2N_{ngc}, \quad (11)$$

where  $k = N_{ngc} + 2N_x + 1$ .

The other primitive variables, density and pressure, are mirrored in the guardcell regions,

$$\rho_i = \rho_{k-i} \quad \text{and} \quad p_i = p_{k-i} \quad \text{for } i = 1, 2, \quad (12)$$

on the left boundary with  $k = 2N_{ngc} + 1$ . Similarly on the right boundary we have

$$\rho_i = \rho_{k-i} \quad \text{and} \quad p_i = p_{k-i} \quad \text{for } i = N_x + N_{ngc} + 1, N_x + 2N_{ngc}, \quad (13)$$

where  $k = N_{ngc} + 2N_x + 1$ .

The initial conditions consist of two parallel, planar flow discontinuities on the computational domain  $[0, 1]$ . The density is unity and the velocity is initially zero everywhere. The pressure is large at the left and right and small in the center

$$p_L = 1000, \quad p_M = 0.01, \quad p_R = 100. \quad (14)$$

$$\mathbf{V}(x, 0) = \begin{cases} \begin{pmatrix} \rho \\ u \\ p \end{pmatrix}_L = \begin{pmatrix} 1.0 \\ 0.0 \\ 1000.0 \end{pmatrix} & \text{if } x \leq 0.1, \\ \begin{pmatrix} \rho \\ u \\ p \end{pmatrix}_M = \begin{pmatrix} 1.0 \\ 0.0 \\ 0.01 \end{pmatrix} & \text{if } 0.1 < x \leq 0.9, \\ \begin{pmatrix} \rho \\ u \\ p \end{pmatrix}_R = \begin{pmatrix} 1.0 \\ 0.0 \\ 100.0 \end{pmatrix} & \text{if } x > 0.9. \end{cases} \quad (15)$$

Use  $\gamma = 1.4$  and the final time  $t_{\max} = 0.038$ .

#### 2.4. The Shu-Osher Problem

The problem description is given in Chapter 9 of the lecture note. Also see the computed results and their comparisons in Fig. 5 therein.

This problem requires a special boundary condition that keeps the values of the primitive variables in the guardcells unchanged from the initial conditions. That is, using the computational domain of  $[-4.5, 4.5]$ ,

$$\mathbf{V}(x, t) = \begin{cases} \begin{pmatrix} \rho \\ u \\ p \end{pmatrix}_L = \begin{pmatrix} 3.857143 \\ 2.629369 \\ 10.33333 \end{pmatrix} & \text{if } x < -4.0, \\ \begin{pmatrix} \rho \\ u \\ p \end{pmatrix}_R = \begin{pmatrix} 1 + a_\rho \sin(f_\rho x) \\ 0.0 \\ 1.0 \end{pmatrix} & \text{if } x > -4.0, \end{cases} \quad (16)$$

where  $a_\rho$  is the amplitude and  $f_\rho$  is the frequency of the density perturbations, for which we take  $a_\rho = 0.2$  and  $f_\rho = 5.0$ . The ideal equation of state is used with  $\gamma$  set to 1.4. The location of the initial discontinuity is at  $x_s = -4.0$ . Use the final time  $t_{\max} = 1.8$ .

### 3. Project Tasks

Write a scientific report using LaTeX no more than 20 pages with font size 11. Your report should have three parts: introduction, main code results, and conclusion. Submit your report and source code to your Git repository. Report any issues with coding (e.g., incomplete implementations, bugs, etc.) or any interesting findings and/or issues in your code results (e.g., how your PPM results look different from PLM; Roe fails but HLL works, etc.). Try your best to explain what you observe and justify them.

- Coding:
  1. study the template code

2. test the FOG code and implement PLM and PPM solvers for reconstruction, and two Riemann solvers of HLLC and Roe
  3. complete the full set of three different slope limiters: minmod, van Leer's, and MC for both PLM and PPM
  4. setup the above four problems: Sod, Rarefaction, Blast2, and the Shu-Osher problem
- Code Results: run the four test problems to conduct the following comparison tests.
    1. **[Reconstructions]** Run the Sod shock tube problem on  $N_x = 128$  using HLL. Plot primitive variables (density  $\rho$ , velocity  $u$ , and pressure  $p$ ) at  $t = 0.2$  for FOG, PLM+minmod, and PPM+minmod.
    2. **[Riemann Solvers]** Run the rarefaction problem on  $N_x = 128$  using PLM with minmod. Plot primitive variables (density  $\rho$ , velocity  $u$ , and pressure  $p$ ) at  $t = 0.15$  for each HLL, HLLC, and Roe.
    3. **[Slope Limiters]** Run the Blast2 problem on  $N_x = 128$  using PPM with HLLC. Plot density  $\rho$  at  $t = 0.038$  for each minmod, van Leer's, and MC slope limiters. Compare these results with FOG+HLLC.
    4. **[Grid Resolutions]** Run the Shu-Osher problem using PLM+MC+Roe. on  $N_x = 32, 64, 128$ . Plot density  $\rho$  at  $t = 1.8$  for each grid resolution. In addition, compare these results with FOG+Roe.
    5. **[CFL]** Run the Shu-Osher problem using PLM+vanLeer+HLLC on  $N_x = 128$  using CFL = 0.2, 0.4, 0.6, 0.8, 1.0, and 1.4. Plot density  $\rho$  at  $t = 1.8$  for each CFL number.

## The HLL and HLLC Riemann Solvers

The approximate Riemann solver proposed by Harten Lax and van Leer (HLL) in 1983 requires estimates for the fastest signal velocities emerging from the initial discontinuity at the interface, resulting in a *two-wave model* for the structure of the exact solution. A more accurate method is the HLLC, introduced by Toro and collaborators in 1992. This method assumes a *three-wave model*, resulting in better resolution of intermediate waves.

### 10.1 Introduction

For the purpose of computing a Godunov flux, Harten, Lax and van Leer [244] presented a novel approach for solving the Riemann problem approximately. The resulting Riemann solvers have become known as HLL Riemann solvers. In this approach an *approximation for the intercell numerical flux is obtained directly*, unlike the Riemann solvers presented previously in Chaps. 4 and 9. The central idea is to assume, for the solution, a wave configuration that consists of *two waves* separating three constant states. Assuming that the wave speeds are given by some algorithm, application of the integral form of the conservation laws gives a closed-form, approximate expression for the flux. The approach produced practical schemes after the contributions of Davis [150] and Einfeldt [181], who independently proposed various ways of computing the wave speeds required to completely determine the intercell flux. The two-wave HLL approach, along with the wave speed estimates proposed by Einfeldt [181] is known as the HLLE solver. The resulting HLL-type Riemann solvers form the bases of very efficient and robust approximate Godunov-type methods.

One difficulty with these schemes, however, is the assumption of a *two-wave* configuration. This is correct only for hyperbolic systems of two equations, such as the one-dimensional shallow water equations. For larger systems, such as the Euler equations or the split two-dimensional shallow water equations for example, the two-wave assumption is incorrect. As a con-

sequence the resolution of physical features such as contact surfaces, shear waves and material interfaces, can be very inaccurate. For the limiting case in which these features are stationary relative to the mesh, the resulting numerical smearing is unacceptable. In view of this situation Einfeldt proposed [181] a modification to the HLLE scheme, called HLLM, in which the single intermediate state in the HLL approach is modified by means of a linear distribution. The modification involves some parameters that control the amount of excessive dissipation for intermediate waves. Particular choices of these parameters and of the wave speed estimates reduce the HLLM scheme to a modified version of the Roe solver. See [182] for further details on both HLLE and HLLEM.

A different approach to remedy the problem of intermediate waves in the HLL approach was taken by Toro, Spruce and Speares in 1992 [541], [542]. They proposed the HLLC Riemann solver (C standing for Contact), as applied to the time-dependent Euler equations. HLLC is a three-wave model, resulting two-star states for the intermediate region of the Riemann-problem solution fan. A precursor to HLLC was also anticipated in [505]. Early applications of HLLC include the steady supersonic two-dimensional Euler equations [532] and the time-dependent two dimensional shallow water equations [193], [194]. Batten and collaborators [32] analyzed the HLLC scheme and proposed new ways of estimating the wave speeds. See also the work of Batten, Leschziner and Goldberg [33], in which they proposed implicit versions of the HLLC Riemann solver, with application to turbulent flows. In later work by Linde and others [324], [325] modifications to the HLL two-wave approach were also explored, in order to reduce numerical dissipation of contact waves.

In the last decade or so we have seen further developments of the HLLC method as well as ambitious applications. A quick electronic search, by typing for example *HLLC solver*, will give hundreds of useful entries on the subject. Recall that the Euler equations have *three* distinct characteristic fields in one, two and three space dimensions, see Chapter 3, section 3.2. This is why HLLC as proposed in [541], [542], [96] is a *complete* Riemann solver, for the Euler equations; that is the approximate wave structure of HLLC contains all the characteristic fields of the exact problem. However, for systems with eigenstructure containing more than three distinct characteristic fields, the HLLC becomes *incomplete*, tending to behave like HLL for the one-dimensional Euler equations. The incomplete character of a Riemann solver affects the resolution of intermediate waves, particularly when these move slowly relative to the mesh. Therefore, the obvious way of improving the HLLC approach is to admit the correct number of characteristic fields for the system of interest. Works along these lines include [230], [474] and [75]. Other interesting developments and ambitious applications are found in the following works, to name but a few, [24], [553], [54], [54], [74], [360], [580], [318], [397], [351], [334], [572], [6], [53], [285], [199], [255], [237], [382], [361], [86], [425], [602].

In this Chapter we present the HLL and HLLC Riemann solvers as applied to the three-dimensional, time dependent Euler equations. The principles can

easily be extended to solve other hyperbolic systems. Useful background reading is found in Chaps. 3, 4, 6 and 9. The rest of this chapter is organised as follows: Sect. 10.1 recalls the Riemann problem. In Sect. 10.3 we present the original approach of Harten, Lax and van Leer. In Sect. 10.4 we present the HLLC Riemann solver and in Sect. 10.5 we give various algorithms for computing the required wave speeds. A summary of the HLLC schemes is presented in Sect. 10.6. In Sect. 10.7 we analyse the behaviour of the approximate Riemann solvers in the presence of contacts and passive scalars. Numerical results are shown in Sect. 10.8 and in Sect. 10.9 contains some concluding remarks.

## 10.2 The Riemann Problem

We are concerned with solving numerically the general Initial Boundary Value Problem (IBVP)

$$\left. \begin{array}{l} \text{PDEs : } \mathbf{U}_t + \mathbf{F}(\mathbf{U})_x = \mathbf{0}, \\ \text{ICs : } \mathbf{U}(x, 0) = \mathbf{U}^{(0)}(x), \\ \text{BCs : } \mathbf{U}(0, t) = \mathbf{U}_L(t), \quad \mathbf{U}(L, t) = \mathbf{U}_R(t), \end{array} \right\} \quad (10.1)$$

in a domain  $0 \leq x \leq L$ , with appropriate boundary conditions. We use the explicit conservative formula

$$\mathbf{U}_i^{n+1} = \mathbf{U}_i^n - \frac{\Delta t}{\Delta x} [\mathbf{F}_{i+\frac{1}{2}} - \mathbf{F}_{i-\frac{1}{2}}], \quad (10.2)$$

with the numerical flux  $\mathbf{F}_{i+\frac{1}{2}}$  yet to be defined.

### 10.2.1 The Godunov Flux

In Chap. 6 we defined the Godunov intercell numerical flux as

$$\mathbf{F}_{i+\frac{1}{2}} = \mathbf{F}(\mathbf{U}_{i+\frac{1}{2}}(0)), \quad (10.3)$$

in which  $\mathbf{U}_{i+\frac{1}{2}}(0)$  is the exact similarity solution  $\mathbf{U}_{i+\frac{1}{2}}(x/t)$  of the Riemann problem

$$\left. \begin{array}{l} \mathbf{U}_t + \mathbf{F}(\mathbf{U})_x = \mathbf{0}, \\ \mathbf{U}(x, 0) = \begin{cases} \mathbf{U}_L & \text{if } x < 0, \\ \mathbf{U}_R & \text{if } x > 0, \end{cases} \end{array} \right\} \quad (10.4)$$

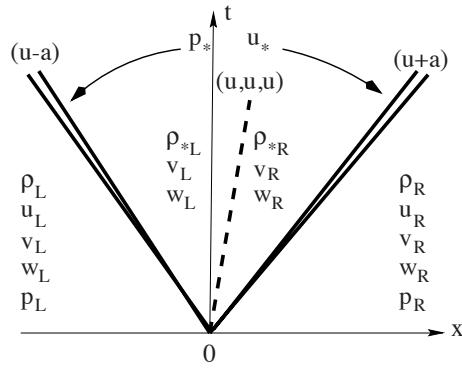
evaluated at  $x/t = 0$ . Fig. 10.1 shows the structure of the exact solution of the Riemann problem for the  $x$ -split, three dimensional Euler equations, for which the vectors of conserved variables and fluxes are

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ \rho u w \\ u(E + p) \end{bmatrix}. \quad (10.5)$$

The value  $x/t = 0$  for the Godunov flux corresponds to the  $t$ -axis. See Chaps. 4 and 6 for details. The piecewise constant initial data, in terms of primitive variables, is

$$\mathbf{W}_L = \begin{bmatrix} \rho_L \\ u_L \\ v_L \\ w_L \\ p_L \end{bmatrix}, \quad \mathbf{W}_R = \begin{bmatrix} \rho_R \\ u_R \\ v_R \\ w_R \\ p_R \end{bmatrix}. \quad (10.6)$$

In Chap. 9 we provided approximations to the state  $\mathbf{U}_{i+\frac{1}{2}}(x/t)$  and obtained



**Fig. 10.1.** Structure of the exact solution of the Riemann problem for the  $x$ -split three dimensional Euler equations. There are five wave families *associated* with the eigenvalues  $u - a$ ,  $u$  (of multiplicity 3) and  $u + a$ .

a corresponding approximate Godunov method by evaluating the physical flux function  $\mathbf{F}$  at this approximate state; see (10.3). The purpose of this chapter is to find *direct approximations to the flux function*  $\mathbf{F}_{i+\frac{1}{2}}$  following the novel approach proposed by Harten, Lax and van Leer [238].

### 10.2.2 Integral Relations

Consider Fig. 10.2, in which the whole of the wave structure arising from the exact solution of the Riemann problem is contained in the control volume  $[x_L, x_R] \times [0, T]$ , that is

$$x_L \leq TS_L, \quad x_R \geq TS_R, \quad (10.7)$$

where  $S_L$  and  $S_R$  are the *fastest signal velocities* perturbing the initial data states  $\mathbf{U}_L$  and  $\mathbf{U}_R$  respectively, and  $T$  is a chosen time. The integral form of the conservation laws in (10.4), in the control volume  $[x_L, x_R] \times [0, T]$  reads

$$\int_{x_L}^{x_R} \mathbf{U}(x, T) dx = \int_{x_L}^{x_R} \mathbf{U}(x, 0) dx + \int_0^T \mathbf{F}(\mathbf{U}(x_L, t)) dt - \int_0^T \mathbf{F}(\mathbf{U}(x_R, t)) dt. \quad (10.8)$$

See Sect. 2.4.1 of Chap. 2 for details on integral forms of conservation laws. Evaluation of the right-hand side of this expression gives

$$\int_{x_L}^{x_R} \mathbf{U}(x, T) dx = x_R \mathbf{U}_R - x_L \mathbf{U}_L + T(\mathbf{F}_L - \mathbf{F}_R), \quad (10.9)$$

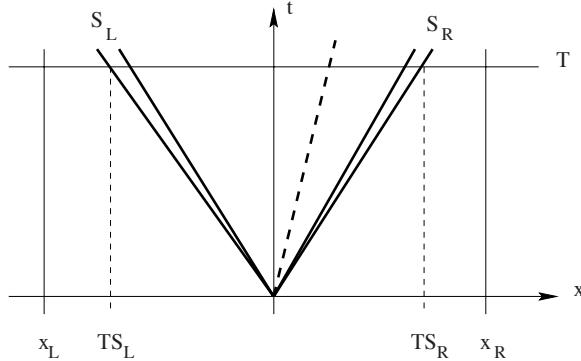
where  $\mathbf{F}_L = \mathbf{F}(\mathbf{U}_L)$  and  $\mathbf{F}_R = \mathbf{F}(\mathbf{U}_R)$ . We call the integral relation (10.9) the *consistency condition*. Now we split the integral on the left-hand side of (10.8) into three integrals, namely

$$\int_{x_L}^{x_R} \mathbf{U}(x, T) dx = \int_{x_L}^{TS_L} \mathbf{U}(x, T) dx + \int_{TS_L}^{TS_R} \mathbf{U}(x, T) dx + \int_{TS_R}^{x_R} \mathbf{U}(x, T) dx$$

and evaluate the first and third terms on the right-hand side. We obtain

$$\int_{x_L}^{x_R} \mathbf{U}(x, T) dx = \int_{TS_L}^{TS_R} \mathbf{U}(x, T) dx + (TS_L - x_L)\mathbf{U}_L + (x_R - TS_R)\mathbf{U}_R. \quad (10.10)$$

Comparing (10.10) with (10.9) gives



**Fig. 10.2.** Control volume  $[x_L, x_R] \times [0, T]$  on  $x$ - $t$  plane.  $S_L$  and  $S_R$  are the fastest signal velocities arising from the solution of the Riemann problem.

$$\int_{TS_L}^{TS_R} \mathbf{U}(x, T) dx = T(S_R \mathbf{U}_R - S_L \mathbf{U}_L + \mathbf{F}_L - \mathbf{F}_R). \quad (10.11)$$

On division through by the length  $T(S_R - S_L)$ , which is the width of the wave system of the solution of the Riemann problem between the slowest and fastest signals at time  $T$ , we have

$$\frac{1}{T(S_R - S_L)} \int_{TS_L}^{TS_R} \mathbf{U}(x, T) dx = \frac{S_R \mathbf{U}_R - S_L \mathbf{U}_L + \mathbf{F}_L - \mathbf{F}_R}{S_R - S_L}. \quad (10.12)$$

Thus, the integral average of the exact solution of the Riemann problem between the slowest and fastest signals at time  $T$  is a known constant, provided

that the signal speeds  $S_L$  and  $S_R$  are known; such constant is the right-hand side of (10.12) and we denote it by

$$\mathbf{U}^{hll} = \frac{S_R \mathbf{U}_R - S_L \mathbf{U}_L + F_L - F_R}{S_R - S_L} . \quad (10.13)$$

We now apply the integral form of the conservation laws to the left portion of Fig. 10.2, that is the control volume  $[x_L, 0] \times [0, T]$ . We obtain

$$\int_{TS_L}^0 \mathbf{U}(x, T) dx = -TS_L \mathbf{U}_L + T(\mathbf{F}_L - \mathbf{F}_{0L}) , \quad (10.14)$$

where  $\mathbf{F}_{0L}$  is the flux  $\mathbf{F}(\mathbf{U})$  along the  $t$ -axis. Solving for  $\mathbf{F}_{0L}$  we find

$$\mathbf{F}_{0L} = \mathbf{F}_L - S_L \mathbf{U}_L - \frac{1}{T} \int_{TS_L}^0 \mathbf{U}(x, T) dx . \quad (10.15)$$

Evaluation of the integral form of the conservation laws on the control volume  $[0, x_R] \times [0, T]$  yields

$$\mathbf{F}_{0R} = \mathbf{F}_R - S_R \mathbf{U}_R + \frac{1}{T} \int_0^{TS_R} \mathbf{U}(x, T) dx . \quad (10.16)$$

The reader can easily verify that the equality

$$\mathbf{F}_{0L} = \mathbf{F}_{0R}$$

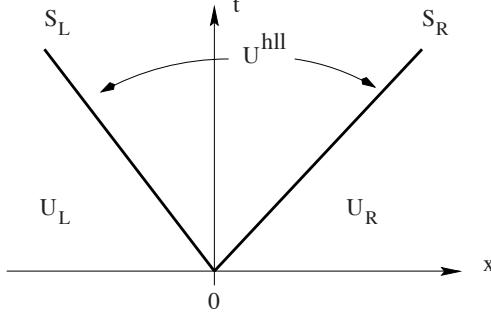
results in the *consistency condition* (10.9). All relations so far are exact, as we are assuming the exact solution of the Riemann problem.

### 10.3 The HLL Approximate Riemann Solver

Harten, Lax and van Leer [244] put forward the following approximate Riemann solver

$$\tilde{\mathbf{U}}(x, t) = \begin{cases} \mathbf{U}_L & \text{if } \frac{x}{t} \leq S_L , \\ \mathbf{U}^{hll} & \text{if } S_L \leq \frac{x}{t} \leq S_R , \\ \mathbf{U}_R & \text{if } \frac{x}{t} \geq S_R , \end{cases} \quad (10.17)$$

where  $\mathbf{U}^{hll}$  is the constant state vector given by (10.13) and the speeds  $S_L$  and  $S_R$  are assumed to be known. Fig. 10.3 shows the structure of this approximate solution of the Riemann problem, called the HLL Riemann solver. Note that this approximation consists of just three constant states separated by two waves. The *Star Region* consists of a *single* constant state; all intermediate states separated by intermediate waves are *lumped* into the single state  $\mathbf{U}^{hll}$ . The corresponding flux  $\mathbf{F}^{hll}$  along the  $t$ -axis is found from the relations (10.15) or (10.16), with the exact integrand replaced by the approximate solution (10.17). Note that we *do not take*  $\mathbf{F}^{hll} = \mathbf{F}(\mathbf{U}^{hll})$ . The non-trivial case of



**Fig. 10.3.** Approximate HLL Riemann solver. Solution in the *Star Region* consists of a single state  $\mathbf{U}^{hll}$  separated from data states by two waves of speeds  $S_L$  and  $S_R$ .

interest is the subsonic case  $S_L \leq 0 \leq S_R$ . Substitution of the integrand in (10.15) or (10.16) by  $\mathbf{U}^{hll}$  in (10.13) gives

$$\mathbf{F}^{hll} = \mathbf{F}_L + S_L(\mathbf{U}^{hll} - \mathbf{U}_L), \quad (10.18)$$

or

$$\mathbf{F}^{hll} = \mathbf{F}_R + S_R(\mathbf{U}^{hll} - \mathbf{U}_R). \quad (10.19)$$

Note that relations (10.18) and (10.19) are also obtained from applying Rankine–Hugoniot conditions across the left and right waves respectively; see Sect. 2.4.2 of Chap. 2 and Sect. 3.1.3 of Chap. 3 for details on the Rankine–Hugoniot conditions. Use of (10.13) in (10.18) or (10.19) gives the HLL flux

$$\mathbf{F}^{hll} = \frac{S_R \mathbf{F}_L - S_L \mathbf{F}_R + S_L S_R (\mathbf{U}_R - \mathbf{U}_L)}{S_R - S_L}. \quad (10.20)$$

The corresponding HLL intercell flux for the approximate Godunov method is then given by

$$\mathbf{F}_{i+\frac{1}{2}}^{hll} = \begin{cases} \mathbf{F}_L & \text{if } 0 \leq S_L, \\ \frac{S_R \mathbf{F}_L - S_L \mathbf{F}_R + S_L S_R (\mathbf{U}_R - \mathbf{U}_L)}{S_R - S_L}, & \text{if } S_L \leq 0 \leq S_R, \\ \mathbf{F}_R & \text{if } 0 \geq S_R. \end{cases} \quad (10.21)$$

Given an algorithm to compute the speeds  $S_L$  and  $S_R$  we have an approximate intercell flux (10.21) to be used in the conservative formula (10.2) to produce an approximate Godunov method. Procedures to estimate the wave speeds  $S_L$  and  $S_R$  are given in Sect. 10.5. Harten, Lax and van Leer [244] showed that the Godunov scheme (10.2), (10.21), if convergent, converges to the weak solution of the conservation laws. In fact they proved that the converged solution is also the physical, entropy satisfying, solution of the conservation laws. Their results actually apply to a larger class of approximate Riemann solvers. One of the requirements is *consistency with the integral form of the conservation laws*.

That is, an approximate solution  $\tilde{\mathbf{U}}(x, t)$  is consistent with the integral form of the conservation laws if, when substituted for the exact solution  $\mathbf{U}(x, t)$  in the consistency condition (10.9), the right-hand side remains unaltered.

A shortcoming of the HLL scheme is exposed by contact discontinuities, shear waves and material interfaces, or any type of *intermediate waves*. For the Euler equations these waves are associated with the multiple eigenvalue  $\lambda_2 = \lambda_3 = \lambda_4 = u$ . See Fig. 10.1. Note that in the integral (10.12), all that matters is the average across the wave structure, without regard for the spatial variations of the solution of the Riemann problem in the *Star Region*. As pointed out by Harten, Lax and van Leer themselves [244], this defect of the HLL scheme may be corrected by restoring the missing waves. Accordingly, Toro, Spruce and Speares [541], [542] proposed the so called *HLLC scheme*, where C stands for *Contact*. In this scheme the missing middle waves are put back into the structure of the approximate Riemann solver.

## 10.4 The HLLC Approximate Riemann Solver

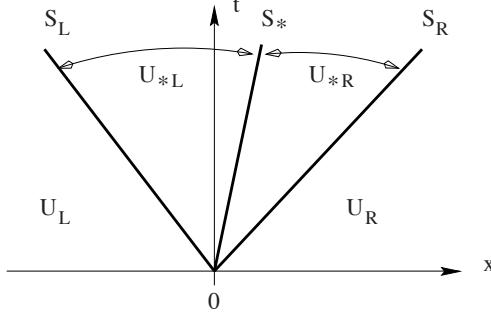
The HLLC scheme is a modification of the HLL scheme described in the previous section, whereby the missing contact and shear waves in the Euler equations are restored. The scheme was first presented in terms of the time-dependent, two dimensional Euler equations [541], [542]. Early applications include the steady supersonic two-dimensional Euler equations [532] and the time-dependent two dimensional shallow water equations [193], [194].

### 10.4.1 Useful Relations

Consider Fig. 10.2, in which the complete structure of the solution of the Riemann problem is contained in a sufficiently large control volume  $[x_L, x_R] \times [0, T]$ . Now, in addition to the slowest and fastest signal speeds  $S_L$  and  $S_R$  we include a middle wave of speed  $S_*$ ; for the Euler equations this corresponds to the multiple eigenvalue  $\lambda_2 = \lambda_3 = \lambda_4 = u$ . See Fig. 10.4. Evaluation of the integral form of the conservation laws in the control volume reproduces the result of equation (10.12), even if variations of the integrand across the wave of speed  $S_*$  are allowed. Note that the consistency condition (10.9) effectively becomes the condition (10.12). By splitting the left-hand side of integral (10.12) into two terms we obtain

$$\left. \begin{aligned} \frac{1}{T(S_R - S_L)} \int_{TS_L}^{TS_R} \mathbf{U}(x, T) dx &= \frac{1}{T(S_R - S_L)} \int_{TS_L}^{TS_*} \mathbf{U}(x, T) dx \\ &\quad + \frac{1}{T(S_R - S_L)} \int_{TS_*}^{TS_R} \mathbf{U}(x, T) dx . \end{aligned} \right\} \quad (10.22)$$

We define the integral averages



**Fig. 10.4.** HLLC approximate Riemann solver. Solution in the *Star Region* consists of two constant states separated from each other by a middle wave of speed  $S_*$ .

$$\left. \begin{aligned} \mathbf{U}_{*L} &= \frac{1}{T(S_* - S_L)} \int_{TS_L}^{TS_*} \mathbf{U}(x, T) dx , \\ \mathbf{U}_{*R} &= \frac{1}{T(S_R - S_*)} \int_{TS_*}^{TS_R} \mathbf{U}(x, T) dx . \end{aligned} \right\} \quad (10.23)$$

By substitution of (10.23) into (10.22) and use of (10.12), the consistency condition (10.9) becomes

$$\left( \frac{S_* - S_L}{S_R - S_L} \right) \mathbf{U}_{*L} + \left( \frac{S_R - S_*}{S_R - S_L} \right) \mathbf{U}_{*R} = \mathbf{U}^{hll} , \quad (10.24)$$

where  $\mathbf{U}^{hll}$  is given by (10.12)–(10.13). The HLLC approximate Riemann solver is given as follows

$$\tilde{\mathbf{U}}(x, t) = \begin{cases} \mathbf{U}_L & , \text{if } \frac{x}{t} \leq S_L , \\ \mathbf{U}_{*L} & , \text{if } S_L \leq \frac{x}{t} \leq S_* , \\ \mathbf{U}_{*R} & , \text{if } S_* \leq \frac{x}{t} \leq S_R , \\ \mathbf{U}_R & , \text{if } \frac{x}{t} \geq S_R . \end{cases} \quad (10.25)$$

We seek a corresponding HLLC numerical flux defined as

$$\mathbf{F}_{i+\frac{1}{2}}^{hllc} = \begin{cases} \mathbf{F}_L & , \text{if } 0 \leq S_L , \\ \mathbf{F}_{*L} & , \text{if } S_L \leq 0 \leq S_* , \\ \mathbf{F}_{*R} & , \text{if } S_* \leq 0 \leq S_R , \\ \mathbf{F}_R & , \text{if } 0 \geq S_R , \end{cases} \quad (10.26)$$

with the intermediate fluxes  $\mathbf{F}_{*L}$  and  $\mathbf{F}_{*R}$  still to be determined. Fig. 10.4 shows the structure of the HLLC approximate Riemann solver.

By integrating over appropriate control volumes, or more directly, by applying Rankine–Hugoniot Conditions across each of the waves of speeds  $S_L$ ,  $S_*$ ,  $S_R$ , we obtain

$$\mathbf{F}_{*L} = \mathbf{F}_L + S_L(\mathbf{U}_{*L} - \mathbf{U}_L) , \quad (10.27)$$

$$\mathbf{F}_{*R} = \mathbf{F}_{*L} + S_*(\mathbf{U}_{*R} - \mathbf{U}_{*L}) , \quad (10.28)$$

$$\mathbf{F}_{*R} = \mathbf{F}_R + S_R(\mathbf{U}_{*R} - \mathbf{U}_R) . \quad (10.29)$$

Compare relations (10.27) and (10.29) for the HLLC scheme with (10.18) and (10.19) for the HLL scheme. Substitution of  $\mathbf{F}_{*L}$  from (10.27) and  $\mathbf{F}_{*R}$  from (10.29) into (10.28) gives identically the consistency condition (10.24). Hence conditions (10.27)–(10.29) are sufficient for ensuring consistency; these are three equations for the four unknowns vectors  $\mathbf{U}_{*L}$ ,  $\mathbf{F}_{*L}$ ,  $\mathbf{U}_{*R}$ ,  $\mathbf{F}_{*R}$ .

#### 10.4.2 The HLLC Flux for the Euler Equations

We seek the solution for the two unknown intermediate fluxes  $\mathbf{F}_{*L}$  and  $\mathbf{F}_{*R}$ . From (10.27)–(10.29) we see that it is sufficient to find solutions for the two intermediate state vectors  $\mathbf{U}_{*L}$  and  $\mathbf{U}_{*R}$ . There are more unknowns than equations and some extra conditions need to be imposed, in order to solve the algebraic problem. Obvious conditions to impose are those satisfied by the exact solution; for pressure and normal component of velocity we have

$$\left. \begin{aligned} p_{*L} &= p_{*R} = p_* , \\ u_{*L} &= u_{*R} = u_* , \end{aligned} \right\} \quad (10.30)$$

and for tangential velocity components we have

$$\left. \begin{aligned} v_{*L} &= v_L , \quad v_{*R} = v_R , \\ w_{*L} &= w_L , \quad w_{*R} = w_R . \end{aligned} \right\} \quad (10.31)$$

See Chap. 4. In addition, it is entirely justified, and convenient, to set

$$S_* = u_* \quad (10.32)$$

and thus if an estimate for  $S_*$  is known, the normal velocity component  $u_*$  in the *Star Region* is known. Now equations (10.27) and (10.29) can be rearranged as

$$S_L \mathbf{U}_{*L} - \mathbf{F}_{*L} = S_L \mathbf{U}_L - \mathbf{F}_L , \quad (10.33)$$

and

$$S_R \mathbf{U}_{*R} - \mathbf{F}_{*R} = S_R \mathbf{U}_R - \mathbf{F}_R , \quad (10.34)$$

where the right-hand sides of (10.33) and (10.34) are known constant vectors. We also note the useful relation between  $\mathbf{U}$  and  $\mathbf{F}$ , namely

$$\mathbf{F}(\mathbf{U}) = u \mathbf{U} + p \mathbf{D} , \quad \mathbf{D} = [0, 1, 0, 0, u]^T . \quad (10.35)$$

Assuming that the wave speeds  $S_L$  and  $S_R$  are known and performing algebraic manipulations of the first and second components of equations (10.33)–(10.34) one obtains the following solutions for pressure in the two *Star Regions*

$$p_{*L} = p_L + \rho_L (S_L - u_L)(S_* - u_L) , \quad p_{*R} = p_R + \rho_R (S_R - u_R)(S_* - u_R) . \quad (10.36)$$

From (10.30)  $p_{*L} = p_{*R}$ , which from (10.36) allows us to obtain an expression for the speed  $S_*$  purely in terms of the assumed speeds  $S_L$  and  $S_R$ , namely

$$S_* = \frac{p_R - p_L + \rho_L u_L (S_L - u_L) - \rho_R u_R (S_R - u_R)}{\rho_L (S_L - u_L) - \rho_R (S_R - u_R)}. \quad (10.37)$$

Thus, we only need to provide estimates for  $S_L$  and  $S_R$ , just as for the simpler HLL solver.

Algebraic manipulation of (10.33) and (10.34) and using the corresponding values  $p_{*L}$  and  $p_{*R}$  from (10.36) gives the intermediate fluxes  $\mathbf{F}_{*L}$  and  $\mathbf{F}_{*R}$  as

$$\mathbf{F}_{*K} = \mathbf{F}_K + S_K (\mathbf{U}_{*K} - \mathbf{U}_K), \quad (10.38)$$

for K=L and K=R, with the intermediate states given as

$$\mathbf{U}_{*K} = \rho_K \left( \frac{S_K - u_K}{S_K - S_*} \right) \begin{bmatrix} 1 \\ S_* \\ v_K \\ w_K \\ \frac{E_K}{\rho_K} + (S_* - u_K) \left[ S_* + \frac{p_K}{\rho_K (S_K - u_K)} \right] \end{bmatrix}. \quad (10.39)$$

The final choice of the HLLC flux is made according to (10.26).

A variation in the formulation of the HLLC solver (10.38)–(10.39) is the following. From equations (10.33) and (10.34) we may write the following solutions for the state vectors  $\mathbf{U}_{*L}$  and  $\mathbf{U}_{*R}$

$$\mathbf{U}_{*K} = \frac{S_K \mathbf{U}_K - \mathbf{F}_K + p_{*K} \mathbf{D}_*}{S_L - S_*}, \quad \mathbf{D}_* = [0, 1, 0, 0, S_*], \quad (10.40)$$

with  $p_{*L}$  and  $p_{*R}$  as given by (10.36). Substitution of  $p_{*K}$  from (10.36) into (10.40) followed by use of (10.27) and (10.29) gives direct expressions for the intermediate fluxes as

$$\mathbf{F}_{*K} = \frac{S_* (S_K \mathbf{U}_K - \mathbf{F}_K) + S_K (p_K + \rho_L (S_K - u_K) (S_* - u_K)) D_*}{S_K - S_*}, \quad (10.41)$$

with the final choice of the HLLC flux made again according to (10.26).

We remark here that the HLLC formulation (10.38)–(10.39) enforces the condition  $p_{*L} = p_{*R}$ , which is satisfied by the exact solution. In the alternative HLLC formulation (10.41) we relax such condition, being more consistent with the pressure approximations (10.36).

A different HLLC flux is obtained by assuming a single mean pressure value in the *Star Region*, and given by the arithmetic average of the pressures in (10.36), namely

$$P_{LR} = \frac{1}{2} [p_L + p_R + \rho_L(S_L - u_L)(S_* - u_L) + \rho_R(S_R - u_R)(S_* - u_R)] . \quad (10.42)$$

Then the intermediate state vectors are given by

$$\mathbf{U}_{*K} = \frac{S_K \mathbf{U}_K - \mathbf{F}_K + P_{LR} \mathbf{D}_*}{S_K - S_*} . \quad (10.43)$$

Substitution of these into (10.27) and (10.29) gives the fluxes  $\mathbf{F}_{*L}$  and  $\mathbf{F}_{*R}$  as

$$\mathbf{F}_{*K} = \frac{S_*(S_K \mathbf{U}_K - \mathbf{F}_K) + S_K P_{LR} \mathbf{D}_*}{S_K - S_*} . \quad (10.44)$$

Again the final choice of HLLC flux is made according to (10.26).

*Remark: general equation of state.* All manipulations so far, assuming that wave speed estimates for  $S_L$  and  $S_R$  are available, are valid for any equation of state. The equation of state only enters in prescribing estimates for  $S_L$  and  $S_R$ .

#### 10.4.3 Multidimensional and Multicomponent Flow

Here we consider extensions of the HLLC solver to two areas of application, namely multidimensional flow and multicomponent flow.

The presentation of the HLLC scheme has been made for the  $x$ -split three-dimensional Euler equations, for which the corresponding eigenvalues are denoted here as  $\lambda_1 = u - a$ ,  $\lambda_2 = u$  (multiplicity 3),  $\lambda_3 = u + a$ , where  $u$  is the *normal* velocity component and  $a$  is the speed of sound. In a general multidimensional situation, see Chapt. 16, one usually requires the flux in the direction *normal* to a volume (or element) interface, which is not necessarily aligned with any of the Cartesian directions. In this case the form of the governing equations remains identical to the  $x$ -split system (10.4), (10.5). There will be a normal and two tangential components of velocity as before, and all the results obtained so far will be applicable.

In the study of multicomponent flow, one considers the advection of *chemical species* by the flow, the carrier fluid. For example, let us consider  $m$  species of concentrations  $q_l$ , for  $l = 1, \dots, m$ , advected with the normal fluid speed  $u$ . This means that for each species we can write the following advection equation

$$\partial_t q_l + u \partial_x q_l = 0 ,$$

for  $l = 1, \dots, m$ . Note that these equations are written in non-conservative form. However, by combining these with the continuity equation we obtain a conservative form of these equations, namely

$$(\rho q_l)_t + (\rho u q_l)_x = 0 , \text{ for } l = 1, \dots, m .$$

The eigenvalues of the enlarged system are as before, with the exception of  $\lambda_2 = u$ , which now, in three space dimensions, has multiplicity  $m + 3$ . These conservation equations can then be added as new components to the conservation equations in (10.1) or (10.4), with the enlarged vectors of conserved variables and fluxes given as

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E \\ \rho q_1 \\ \dots \\ \rho q_l \\ \dots \\ \rho q_m \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ \rho u w \\ u(E + p) \\ \rho u q_1 \\ \dots \\ \rho u q_l \\ \dots \\ \rho u q_m \end{bmatrix}. \quad (10.45)$$

The HLLC flux accommodates these new equations in a very natural way, and nothing special needs to be done. If the HLLC flux (10.38) is used, with  $\mathbf{F}$  as in (10.45), then the intermediate state vectors are given by

$$\mathbf{U}_{*K} = \rho_K \left( \frac{S_K - u_K}{S_K - S_*} \right) \begin{bmatrix} 1 \\ S_* \\ v_K \\ w_K \\ \frac{E_K}{\rho_K} + (S_* - u_K) \left[ S_* + \frac{p_K}{\rho_K(S_K - u_K)} \right] \\ (q_1)_K \\ \dots \\ (q_l)_K \\ \dots \\ (q_m)_K \end{bmatrix}. \quad (10.46)$$

for  $K = L$  and  $K = R$ . In this manner the HLLC flux will resolve the additional intermediate fields as the exact Riemann solver.

Note that the tangential velocity components  $v$  and  $w$  are special cases of passive scalars; compare (10.46) with (10.39) for  $q = v$  and  $q = w$ .

## 10.5 Wave-Speed Estimates

In order to determine completely the numerical fluxes in both the HLL and HLLC Riemann solvers we need to provide an algorithm for computing the wave speeds  $S_L$  and  $S_R$ . For the HLLC scheme one requires in addition an estimate for the speed of the middle wave  $S_*$ , but as seen in (10.37), this can in fact be computed once  $S_L$  and  $S_R$  are known. Thus the pending task is to

determine estimates for  $S_L$  and  $S_R$ . One approach is to estimate the speeds directly; another approach relies on pressure estimates in the *Star Region*, which are then utilised to obtain  $S_L$  and  $S_R$  using exact wave relations.

### 10.5.1 Direct Wave Speed Estimates

The most well known approach for estimating bounds for the minimum and maximum signal velocities present in the solution of the Riemann problem is to provide, directly, wave speeds  $S_L$  and  $S_R$ . Davis [150] suggested the simple estimates

$$S_L = u_L - a_L, \quad S_R = u_R + a_R \quad (10.47)$$

and

$$S_L = \min \{u_L - a_L, u_R - a_R\}, \quad S_R = \max \{u_L + a_L, u_R + a_R\}. \quad (10.48)$$

These estimates make use of data values only, are exceedingly simple but are not recommended for practical computations. Both Davis [150] and Einfeldt [181], proposed to use the Roe [407] average eigenvalues for the left and right non-linear waves, that is

$$S_L = \tilde{u} - \tilde{a}, \quad S_R = \tilde{u} + \tilde{a}, \quad (10.49)$$

where  $\tilde{u}$  and  $\tilde{a}$  are the Roe-average particle and sound speeds respectively, given as follows

$$\tilde{u} = \frac{\sqrt{\rho_L} u_L + \sqrt{\rho_R} u_R}{\sqrt{\rho_L} + \sqrt{\rho_R}}, \quad \tilde{a} = \left[ (\gamma - 1) \left( \tilde{H} - \frac{1}{2} \tilde{u}^2 \right) \right]^{1/2}, \quad (10.50)$$

with the enthalpy  $H = (E + p)/\rho$  approximated as

$$\tilde{H} = \frac{\sqrt{\rho_L} H_L + \sqrt{\rho_R} H_R}{\sqrt{\rho_L} + \sqrt{\rho_R}}. \quad (10.51)$$

Complete details of the Roe Riemann solver are given in Chap. 11.

Motivated by the Roe eigenvalues Einfeldt [181] proposed the estimates

$$S_L = \bar{u} - \bar{d}, \quad S_R = \bar{u} + \bar{d}, \quad (10.52)$$

for his HLLE solver, where

$$\bar{d}^2 = \frac{\sqrt{\rho_L} a_L^2 + \sqrt{\rho_R} a_R^2}{\sqrt{\rho_L} + \sqrt{\rho_R}} + \eta_2 (u_R - u_L)^2 \quad (10.53)$$

and

$$\eta_2 = \frac{1}{2} \frac{\sqrt{\rho_L} \sqrt{\rho_R}}{(\sqrt{\rho_L} + \sqrt{\rho_R})^2}. \quad (10.54)$$

These wave speed estimates are reported to lead to effective and robust Godunov-type schemes. More details on the HLLE solver and its modification HLLEM, are found in [182]. In this paper the authors also analyze the effect of the choice of wave speed estimates on the Riemann solver and introduce the concept of *positively conservative* Riemann solvers, for the Euler equations. These are solvers for which, for physically admissible data, density and internal energy remain positive during the calculations. See Batten et al. [32] for further discussion on direct wave speed estimates.

Davis made some observations regarding the relationship between the chosen wave speeds and some well-known numerical methods. Suppose that for a given Riemann problem we can identify a positive speed  $S^+$ . Then by choosing  $S_L = -S^+$  and  $S_R = S^+$  in the HLL flux (10.20) one obtains a Rusanov flux [418]

$$\mathbf{F}_{i+1/2} = \frac{1}{2}(\mathbf{F}_L + \mathbf{F}_R) - \frac{1}{2}S^+(\mathbf{U}_R - \mathbf{U}_L). \quad (10.55)$$

As to the choice of the speed  $S^+$ , Davis [150] considered

$$S^+ = \max \{|u_L - a_L|, |u_R - a_R|, |u_L + a_L|, |u_R + a_R|\}.$$

Actually, the above speed is bounded by

$$S^+ = \max \{|u_L| + a_L, |u_R| + a_R\}. \quad (10.56)$$

This choice is likely to produce a more robust scheme and is also simpler than Davis's choice.

Another possible choice is  $S^+ = S_{max}^n$ , the maximum wave speed present at the appropriate time found by imposing the Courant stability condition; see Sect. 6.3.2 of Chap. 6. This speed is related to the time step  $\Delta t$  and the grid spacing  $\Delta x$  via

$$S_{max}^n = \frac{C_{cfl}\Delta x}{\Delta t}, \quad (10.57)$$

where  $C_{cfl}$  is the Courant number coefficient, usually chosen (empirically) to be  $C_{cfl} \approx 0.9$ , for a scheme of linear stability limit of unity. For  $C_{cfl} = 1$  one has  $S^+ = \frac{\Delta x}{\Delta t}$ , which results in the Lax–Friedrichs numerical flux

$$F_{i+1/2} = \frac{1}{2}(F_L + F_R) - \frac{1}{2}\frac{\Delta x}{\Delta t}(U_R - U_L). \quad (10.58)$$

See Sect. 5.3.4 of Chap. 5 and Sect. 7.3.1 of Chap. 7.

In the next section we propose a different way of finding wave-speed estimates.

### 10.5.2 Pressure-Based Wave Speed Estimates

A different approach for finding wave speed estimates was proposed by Toro et. al. [542], whereby one first finds an estimate for the pressure  $p_*$  in

the *Star Region*. Then, estimates for  $S_L$  and  $S_R$  are derived. This is a simple task and several reliable choices are available. Suppose we have an estimate  $p_*$  for the pressure in the *Star Region*. Then we choose the following wave speeds

$$S_L = u_L - a_L q_L, \quad S_R = u_R + a_R q_R, \quad (10.59)$$

where

$$q_K = \begin{cases} 1 & \text{if } p_* \leq p_K \\ \left[ 1 + \frac{\gamma+1}{2\gamma} (p_*/p_K - 1) \right]^{1/2} & \text{if } p_* > p_K. \end{cases} \quad (10.60)$$

This choice of wave speeds discriminates between shock and rarefaction waves. If the  $K$  wave ( $K = L$  or  $K = R$ ) is a rarefaction then the speed  $S_K$  corresponds to the characteristic speed of the head of the rarefaction, which carries the fastest signal. If the wave is a shock wave then the speed corresponds to an approximation of the true shock speed; the wave relations used are exact but the pressure ratio across the shock is approximated, because  $p_*$  is an approximation to the pressure behind the shock wave. We propose to use the state approximations of Chap. 9 to find  $p_*$ .

The PVRS approximate Riemann solver [502] presented in Sect. 9.3 of Chap. 9 gives

$$p_{pvrs} = \frac{1}{2}(p_L + p_R) - \frac{1}{2}(u_R - u_L)\bar{\rho}\bar{a}, \quad (10.61)$$

where

$$\bar{\rho} = \frac{1}{2}(\rho_L + \rho_R), \quad \bar{a} = \frac{1}{2}(a_L + a_R). \quad (10.62)$$

This approximation for pressure can be used directly into (10.59)–(10.60) to obtain wave speed estimates for the HLL and HLLC schemes. See also Eq. (9.28) of Chapt. 9 for alternative estimates for  $p_*$ .

Another choice is furnished by the Two–Rarefaction Riemann solver TRRS of Sect. 9.4.1 of Chap. 9, namely

$$p_{tr} = \left[ \frac{a_L + a_R - \frac{\gamma-1}{2}(u_R - u_L)}{a_L/p_L^z + a_R/p_R^z} \right]^{1/z}, \quad (10.63)$$

where

$$P_{LR} = \left( \frac{p_L}{p_R} \right)^z; \quad z = \frac{\gamma-1}{2\gamma}. \quad (10.64)$$

The Two–Shock Riemann solver TSRS of Sect. 9.4.2 of Chap. 9 gives

$$p_{ts} = \frac{g_L(p_0)p_L + g_R(p_0)p_R - \Delta u}{g_L(p_0) + g_R(p_0)}, \quad (10.65)$$

where

$$g_K(p) = \left[ \frac{A_K}{p + B_K} \right]^{1/2}, \quad p_0 = \max(0, p_{pvrs}), \quad (10.66)$$

for  $K = L$  and  $K = R$ .

In computational practice we could use the hybrid scheme of Sect. 9.5.2 of Chap. 9 to determine  $p_*$ . See Chap. 9 for full details. The HLL approximate Riemann solver with the hybrid pressure-based wave speed estimates has been implemented in the NAG routine D03PXF [319] for Godunov-type methods to solve the time-dependent, one dimensional Euler equations for ideal gases. For ideal gases we find that the simplified PVRS scheme, with  $p_* = \max(0, p_{pvrs})$  is very simple and also is found to be sufficiently robust.

## 10.6 Summary of HLLC Fluxes

Here we summarize the HLLC scheme, based on a particular choice of wave speeds. To compute the HLLC flux one performs the following steps:

- *Step I: pressure estimate.* Compute estimate for the pressure  $p_*$  in the *Star Region* as

$$\left. \begin{aligned} p_* &= \max(0, p_{pvrs}), \quad p_{pvrs} = \frac{1}{2}(p_L + p_R) - \frac{1}{2}(u_R - u_L)\bar{\rho}\bar{a}, \\ \bar{\rho} &= \frac{1}{2}(\rho_L + \rho_R), \quad \bar{a} = \frac{1}{2}(a_L + a_R). \end{aligned} \right\} \quad (10.67)$$

There are other possible choices for estimating the pressure  $p_*$ . See (10.63) and (10.65).

- *Step II: wave speed estimates.* Compute the wave speed estimates for  $S_L$  and  $S_R$  as

$$S_L = u_L - a_L q_L, \quad S_R = u_R + a_R q_R, \quad (10.68)$$

with

$$q_K = \begin{cases} 1 & \text{if } p_* \leq p_K \\ \left[ 1 + \frac{\gamma+1}{2\gamma} \left( p_*/p_K - 1 \right) \right]^{1/2} & \text{if } p_* > p_K. \end{cases} \quad (10.69)$$

Then compute the intermediate speed  $S_*$  in terms of  $S_L$  and  $S_R$  as

$$S_* = \frac{p_R - p_L + \rho_L u_L (S_L - u_L) - \rho_R u_R (S_R - u_R)}{\rho_L (S_L - u_L) - \rho_R (S_R - u_R)}. \quad (10.70)$$

Other choices of  $S_L$  and  $S_R$  are possible. See for example (10.49) and (10.52)

- *Step III: HLLC flux.* Compute the HLLC flux, according to

$$\mathbf{F}_{i+\frac{1}{2}}^{hllc} = \begin{cases} \mathbf{F}_L & \text{if } 0 \leq S_L, \\ \mathbf{F}_{*L} & \text{if } S_L \leq 0 \leq S_*, \\ \mathbf{F}_{*R} & \text{if } S_* \leq 0 \leq S_R, \\ \mathbf{F}_R & \text{if } 0 \geq S_R, \end{cases} \quad (10.71)$$

with

$$\mathbf{F}_{*K} = \mathbf{F}_K + S_K(\mathbf{U}_{*K} - \mathbf{U}_K) \quad (10.72)$$

and

$$\mathbf{U}_{*K} = \rho_K \left( \frac{S_K - u_K}{S_K - S_*} \right) \begin{bmatrix} 1 \\ S_* \\ v_K \\ w_K \\ \frac{E_K}{\rho_K} + (S_* - u_K) \left[ S_* + \frac{p_K}{\rho_K(S_K - u_K)} \right] \end{bmatrix}. \quad (10.73)$$

There are two variants of the HLLC flux in the third step, as seen below.

- *Step III: HLLC flux, Variant 1.* Compute the numerical fluxes as

$$\left. \begin{aligned} \mathbf{F}_{*K} &= \frac{S_*(S_K \mathbf{U}_K - \mathbf{F}_K) + S_K(p_K + \rho_L(S_K - u_K)(S_* - u_K))\mathbf{D}_*}{S_K - S_*}, \\ \mathbf{D}_* &= [0, 1, 0, 0, S_*]^T, \end{aligned} \right\} \quad (10.74)$$

and the final HLLC flux chosen according to (10.71).

- *Step III: HLLC flux, Variant 2.* Compute the numerical fluxes as

$$\mathbf{F}_{*K} = \frac{S_*(S_K \mathbf{U}_K - \mathbf{F}_K) + S_K P_{LR} \mathbf{D}_*}{S_K - S_*}, \quad (10.75)$$

with  $\mathbf{D}_*$  as in (10.74) and

$$P_{LR} = \frac{1}{2}[p_L + p_R + \rho_L(S_L - u_L)(S_* - u_L) + \rho_R(S_R - u_R)(S_* - u_R)]. \quad (10.76)$$

The final HLLC flux is chosen according to (10.71).



# The Piecewise Cubic Method (PCM) for computational fluid dynamics



Dongwook Lee<sup>a,\*</sup>, Hugues Faller<sup>a,c</sup>, Adam Reyes<sup>b</sup>

<sup>a</sup> Applied Mathematics and Statistics, University of California, Santa Cruz, CA, USA

<sup>b</sup> Department of Physics, University of California, Santa Cruz, CA, USA

<sup>c</sup> Département de Physique, École Normale Supérieure, Paris, France

## ARTICLE INFO

### Article history:

Received 19 October 2016

Received in revised form 21 March 2017

Accepted 1 April 2017

Available online 5 April 2017

### Keywords:

High-order methods

Piecewise cubic method

Finite volume method

Gas dynamics

Magnetohydrodynamics

Godunov's method

## ABSTRACT

We present a new high-order finite volume reconstruction method for hyperbolic conservation laws. The method is based on a piecewise cubic polynomial which provides its solutions a fifth-order accuracy in space. The spatially reconstructed solutions are evolved in time with a fourth-order accuracy by tracing the characteristics of the cubic polynomials. As a result, our temporal update scheme provides a significantly simpler and computationally more efficient approach in achieving fourth order accuracy in time, relative to the comparable fourth-order Runge–Kutta method. We demonstrate that the solutions of PCM converges at fifth-order in solving 1D smooth flows described by hyperbolic conservation laws. We test the new scheme on a range of numerical experiments, including both gas dynamics and magnetohydrodynamics applications in multiple spatial dimensions.

© 2017 Elsevier Inc. All rights reserved.

## 1. Introduction

In this paper we are interested in solving multidimensional conservation laws of the Euler equations and the ideal MHD equations, written as

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathcal{F}(\mathbf{U}) = 0, \quad (1)$$

where  $\mathbf{U}$  is the vector of the conservative variables, and

$$\mathcal{F}(\mathbf{U}) = [F(\mathbf{U}), G(\mathbf{U}), H(\mathbf{U})]^T = [\mathbf{F}, \mathbf{G}, \mathbf{H}]^T \quad (2)$$

is the flux vector.

We present a new high-order piecewise cubic method (PCM) algorithm that is extracted from the classical PPM [1] and WENO schemes [2,3]. These two algorithms, by far, have been extremely successful in various scientific fields where there are challenging computational needs for *both* high-order accuracy in smooth flows and well-resolved solutions in shock/discontinuous flows. With the advent of high-performance computing (HPC) in recent years, such needs have been more and more desired, and have become a necessary requirement in conducting large scale, cutting edge simulations of gas dynamics and magnetohydrodynamics (MHD) [4–7].

\* Corresponding author.

E-mail address: dlee79@ucsc.edu (D. Lee).

As observed in the success stories of the PPM and WENO methods, discrete algorithms of data interpolation and reconstruction play a key role in numerical methods for PDE approximations [8–10] within the broad framework of finite difference and finite volume discretization methods. In view of this, computational improvements of such interpolation and reconstruction schemes, particularly focused on the high-order property with great shock-capturing capability, take their positions at the center of HPC in modern computational fluid dynamics.

The properties of enhanced solution accuracy with lower numerical errors on a given grid resolution and faster convergence-to-solution rates are the key advantages in high-order schemes. The advantage of using high-order methods in HPC is therefore clear: one can obtain reproducible, admissible, and highly accurate numerical solutions in a faster computational time at the expense of increased rate of floating point operations, while at the same time, with the use of smaller size of grid resolutions. This is by no means exceedingly efficient in high-performance computing (HPC), in view of the fact that the increase of grid resolutions has a direct impact to an increase of memory footprints which are bounded in all modern computing architectures.

In this regard, our goal in this paper is to lay down a mathematical foundation in designing a new high-order method using piecewise cubic polynomials. For the scope of the current paper we focus on describing the detailed PCM algorithm in 1D finite volume framework in which a fourth order PCM reconstruction is formulated in each normal dimension without cross derivative terms. For multidimensional problems we adopt the classical<sup>1</sup> *dimension-by-dimension* [12] approach for simplicity, where a fourth order PCM reconstruction in each normal direction is used to compute second order time averaged normal fluxes. The multidimensional conservation laws are updated using a directionally unsplit fashion. Although this approach has an advantage in its simplicity, it unfortunately fails to retain the high-order accurate property of the 1D baseline algorithm in multidimensional problems. Instead, it provides only a second-order accuracy in multidimensional nonlinear advection in finite volume method due to the lack of accuracy in approximating a face-averaged flux function as a result of mis-using an averaged quantity in place of a pointwise quantity, or vice versa [11–14]. We understand that the baseline 1D PCM scheme can be extended to multiple spatial dimensions preserving its high-order accuracy by following more sophisticated treatments in two different ways. A first approach can be a traditional approach of combining high-order spatial reconstructions and high-order Runge–Kutta (RK) temporal updates [11–14]. Alternatively, a second approach can be available with more computationally efficient formulations based on one-step ADER formulations [15–22] to extend the baseline 1D PCM scheme to multiple spatial dimensions with high-order accuracy. The ADER methods have shown great promises in achieving high-order accurate solutions in multiple spatial dimensions using efficient temporal updating strategies for structured and unstructured grid geometries [19,23], as well as for adaptive mesh refinement geometry [20]. As for the computational performance, the ADER schemes can yield about an order of magnitude improved time-to-solution performance relative to the RK integration methods of the same solution accuracy [21]. Such a work will be considered in our future research to deliver high-order accuracy in designing the PCM scheme multidimensionally.

For a finite volume scheme in 1D we take the spatial average of Eq. (1) over the cell  $I_i = [x_{i-1/2}, x_{i+1/2}]$ , yielding a semi-discrete form,

$$\frac{\partial \bar{\mathbf{U}}_i}{\partial t} = -\frac{1}{\Delta x} (\mathbf{F}_{i+1/2} - \mathbf{F}_{i-1/2}) \quad (3)$$

to get an equation for the evolution of the volume averaged variables,  $\bar{\mathbf{U}}_i = \frac{1}{\Delta x} \int_{I_i} \mathbf{U}(x, t) dx$ . Typically to achieve high-order accuracy in time the temporal update is done using a TVD Runge–Kutta scheme in method-of-lines form [14,24]. In this approach the high-order accuracy comes from taking the *multiple Euler stages* of the RK time discretizations, which require repeated reconstructions in a single time step, increasing the computational cost.

Instead, as will be fully described in Section 2, one of the novel ideas in PCM is to employ the simple *single stage* predictor–corrector type temporal update formulation in which we take the time-average of Eq. (3)

$$\bar{\mathbf{U}}_i^{n+1} = \bar{\mathbf{U}}_i^n - \frac{\Delta t}{\Delta x} (\mathbf{F}_{i+1/2}^{n+1/2} - \mathbf{F}_{i-1/2}^{n+1/2}). \quad (4)$$

Here  $\bar{\mathbf{U}}_i^n = \bar{\mathbf{U}}_i(t^n)$  is the volume averaged quantity at  $t^n$ , and  $\mathbf{F}_{i\pm 1/2}^{n+1/2} = \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} \mathbf{F}_{i\pm 1/2}(t) dt$  is the time average of the interface flux from  $t^n$  to  $t^{n+1}$ . In this way high-order in space and time is accomplished with a single reconstruction in contrast to the multiple Euler stages of the RK time discretizations, providing significant benefits in computational efficiency per solution accuracy.

The organization of the paper is as follows: Section 2 describes the fifth-order accurate spatial reconstruction algorithm of PCM in 1D. We highlight several desirable properties of the PCM scheme in terms of computational efficiency and solution accuracy. Section 3 introduces the fourth-order accurate temporal updating scheme of PCM using a predictor–corrector type characteristic tracing, which is much simpler than the typical high-order Runge–Kutta ODE updates. In Section 5 we discuss how we extend the 1D scheme in Section 2 to multiple spatial dimensions following the *dimension-by-dimension* [12,13] approach to reconstruction, meaning that the PCM reconstruction uses one-dimensional reconstructions in each normal

<sup>1</sup> This approach is the same as the Class A approach in [13], and should not be confused with the so-called dimensionally split approach. Our spatial integration scheme in this paper is directionally unsplit.

direction, followed by evaluating 1D-based fluxes using a single-point quadrature rule at each cell face, followed by updating the system of multidimensional equations in Eq. (1) in a directionally unsplit fashion.

In Section 6 we test the PCM scheme on a wide spectrum of benchmark problems in 1D, 2D and 3D, both for hydrodynamics and magnetohydrodynamics (MHD) applications. We also compare the PCM solutions with PPM and WENO solutions in order to examine numerical accuracy, capability and efficiency in both smooth and shock flow regimes. We conclude our paper in Section 7 with a brief summary.

## 2. The one-dimensional Piecewise Cubic Method (PCM) spatial reconstruction

In this section we describe a new PCM scheme in a one-dimensional finite volume formulation for solving hyperbolic conservation laws of hydrodynamics and magnetohydrodynamics. The new PCM scheme is a higher-order extension of Godunov's method [25], bearing its key components in the reconstruction algorithm on the relevant ideas of its high-order predecessors, the PPM scheme [1], the WENO schemes [2,26–28], and Hermite-WENO schemes [29–32].

For the purpose of this section, we take the  $3 \times 3$  hyperbolic system of conservation laws of the 1D Euler equations

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial F(\mathbf{U})}{\partial x} = 0. \quad (5)$$

The notations used are the vector of the conservative variables  $\mathbf{U}$  and fluxes  $F(\mathbf{U})$ , respectively, defined as

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ E \end{bmatrix}, \quad F(\mathbf{U}) = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ u(E + p) \end{bmatrix}. \quad (6)$$

Here  $\rho$  is the fluid density,  $u$  is the fluid velocity in  $x$ -direction, and  $E$  is the total energy as the sum of the internal energy  $\epsilon = p/(\gamma - 1)$  and the kinetic energy obeying the ideal gas law,

$$E = \frac{p}{\gamma - 1} + \frac{\rho u^2}{2}, \quad (7)$$

where  $p$  is the gas pressure, with the ratio of specific heats denoted as  $\gamma$ . We denote the cells in  $x$ -direction by  $I_i = [x_{i-1/2}, x_{i+1/2}]$ . We assume our grid is configured on an equidistant uniform grid for simplicity.

In addition to the system of the Euler equations in the conserved variables  $\mathbf{U}$  as given in Eq. (5), we often use the two other equivalent system of equations each of which can be written either in the primitive variables  $\mathbf{V} = [\rho, u, p]^T$  or in the characteristic variables  $\mathbf{W}$ . The characteristic variable  $\mathbf{W}$  is readily obtained from  $\mathbf{U}$  or  $\mathbf{V}$  by multiplying the left eigenvectors corresponding to either  $\mathbf{U}$  or  $\mathbf{V}$ , for instance,  $\mathbf{W} = \mathbf{L}\mathbf{U}$ . In the latter  $\mathbf{L}$  ( $\equiv \mathbf{R}^{-1}$ ) represents the  $3 \times 3$  matrix obtained from diagonalizing the coefficient matrix  $\mathbf{A} = \partial F/\partial \mathbf{U} = \mathbf{R}\mathbf{A}\mathbf{R}^{-1}$ , whose rows are the  $k$ -th left eigenvectors  $\ell^{(k)}$ ,  $k = 1, 2, 3$ . The representation of the system in  $\mathbf{W}$  furnishes a completely linearly decoupled 1D system of equations,

$$\frac{\partial \mathbf{W}}{\partial t} + \Lambda \frac{\partial \mathbf{W}}{\partial x} = 0. \quad (8)$$

The above system in the characteristic variables  $\mathbf{W}$  is therefore very handy for analyses, and also is a preferred choice of variable in order to furnish numerical solutions more accurate than third-order especially with better non-oscillatory controls, in particular when considering wave-by-wave propagations in a system of equations [11]. For this reason the characteristic variable  $\mathbf{W}$  is taken as our default variable choice in the 1D PCM reconstruction steps via characteristic decompositions, albeit with an increased computational cost, among the other two choices of the primitive  $\mathbf{V}$  or the conservative variables  $\mathbf{U}$ .

The methodology presented below can be similarly applied to the 1D ideal MHD equations (see for instance, [33]).

### 2.1. Piecewise cubic profile

To begin with we first define a cubic polynomial  $p_i(x)$  to approximate a  $k$ -th characteristic variable  $q \in \mathbf{W}$  on each interval  $I_i$  by

$$p_i(x) = c_0 + c_1(x - x_i) + c_2(x - x_i)^2 + c_3(x - x_i)^3. \quad (9)$$

The goal is now to determine the four coefficients  $c_i$ ,  $i \in \mathbb{Z}$ ,  $0 \leq i \leq 3$ , which can be achieved by imposing the following four conditions:

$$\frac{1}{\Delta x} \int_{I_i} p_i(x) dx = \bar{q}_i, \quad (10)$$

$$p_i(x_{i-1/2}) = q_{L,i}, \quad (11)$$

$$p_i(x_{i+1/2}) = q_{R,i}, \quad (12)$$

$$p'_i(x_i) = q'_{C,i}, \quad (13)$$

where

$$\bar{q}_i = \frac{1}{\Delta x} \int_{I_i} q(x, t^n) dx \quad (14)$$

is the cell-averaged quantity at  $t^n$  on  $I_i$  which is given as an initial condition;

$$q_{L,i} = q(x_{i-1/2}, t^n) + \mathcal{O}(\Delta x^p), \quad q_{R,i} = q(x_{i+1/2}, t^n) + \mathcal{O}(\Delta x^p) \quad (15)$$

are respectively the  $p$ -th order accurate pointwise left and the right Riemann states at  $t^n$  on the cell  $I_i$  that are unknown yet but are to be determined as described below; and lastly

$$q'_{C,i} = q'(x_i, t^n) + \mathcal{O}(\Delta x^r) \quad (16)$$

is the  $r$ -th order accurate approximation to the slope of  $q$  at  $t^n$  evaluated at  $x_i$ , which is again unknown at this point but is to be determined as below.

For the moment let us assume that all four quantities  $\bar{q}_i$ ,  $q_{L,i}$ ,  $q_{R,i}$  and  $q'_{C,i}$  are known. It can be shown that the system of relations in Eqs. (10)–(13) is equivalent to a system given as

$$c_0 + c_2 \frac{\Delta x^2}{12} = \bar{q}_i, \quad (17)$$

$$c_0 - c_1 \frac{\Delta x}{2} + c_2 \frac{\Delta x^2}{4} - c_3 \frac{\Delta x^3}{8} = q_{L,i}, \quad (18)$$

$$c_0 + c_1 \frac{\Delta x}{2} + c_2 \frac{\Delta x^2}{4} + c_3 \frac{\Delta x^3}{8} = q_{R,i}, \quad (19)$$

$$c_1 = q'_{C,i}, \quad (20)$$

which, in turn, can be solved for all four  $c_i$ ,  $i = 1, \dots, 4$ . The final expressions of the coefficients in terms of  $\bar{q}_i$ ,  $q_{L,i}$ ,  $q_{R,i}$ , and  $q'_{C,i}$  are given as:

$$c_0 = \frac{1}{4}(-q_{R,i} - q_{L,i} + 6\bar{q}_i), \quad (21)$$

$$c_1 = q'_{C,i}, \quad (22)$$

$$c_2 = \frac{3}{\Delta x^2}(q_{R,i} + q_{L,i} - 2\bar{q}_i), \quad (23)$$

$$c_3 = \frac{4}{\Delta x^3}(q_{R,i} - q_{L,i} - \Delta x q'_{C,i}). \quad (24)$$

Therefore once we figure out the three unknowns,  $q_{L,i}$ ,  $q_{R,i}$ , and  $q'_{C,i}$ , the cubic profile  $p_i(x)$  in Eq. (9) can be completely determined and is ready to approximate  $q$  on each  $I_i$ .

We now devote the following sections to describe how to determine  $q_{L,i}$ ,  $q_{R,i}$ , and  $q'_{C,i}$  so that the resulting PCM approximation to the variable  $q$  lend its accuracy a fifth-order in space (Sections 2.2 and 2.3) and a fourth-order in time (Section 3).

## 2.2. Reconstruction of the Riemann states $q_{L,i}$ and $q_{R,i}$

We follow the fifth-order finite volume WENO approach, either of the classical WENO-JS [2], WENO-M [34], or WENO-Z [27,28], in order to reconstruct the left and right Riemann states,  $q_{L,i}$  and  $q_{R,i}$ , on each cell  $I_i$ ,

$$q_{L,R,i} = \sum_{\ell=1}^3 \omega_\ell^\pm p_\ell(x_{i\pm 1/2}), \quad (25)$$

where the explicit forms of  $p_\ell$  are given in Appendix A. The nonlinear weights  $\omega_\ell^\pm$  are determined by using one of the following schemes:

- For WENO-JS:

$$\omega_\ell^{\pm, JS} = \frac{\tilde{\omega}_\ell^\pm}{\sum_s \tilde{\omega}_s^\pm}, \text{ where } \tilde{\omega}_\ell^\pm = \frac{\gamma_\ell^\pm}{(\epsilon + \beta_\ell)^m} \quad (26)$$

- For WENO-M:

$$\omega_{\ell}^{\pm,M} = \frac{\tilde{\omega}_{\ell}^{\pm}}{\sum_s \tilde{\omega}_s^{\pm}}, \text{ where } \tilde{\omega}_{\ell}^{\pm} = g_{\ell}(\omega_{\ell}^{\pm,JS}) = \frac{\omega_{\ell}^{\pm,JS} (\gamma_{\ell}^{\pm} + (\gamma_{\ell}^{\pm})^2 - 3(\gamma_{\ell}^{\pm})\omega_{\ell}^{\pm,JS} + (\omega_{\ell}^{\pm,JS})^2)}{(\gamma_{\ell}^{\pm})^2 + \omega_{\ell}^{\pm,JS}(1 - 2\gamma_{\ell}^{\pm})} \quad (27)$$

- For WENO-Z:

$$\omega_{\ell}^{\pm,Z} = \frac{\tilde{\omega}_{\ell}^{\pm}}{\sum_s \tilde{\omega}_s^{\pm}}, \text{ where } \tilde{\omega}_{\ell}^{\pm} = \gamma_{\ell}^{\pm} \left( 1 + \left( \frac{|\beta_0 - \beta_2|}{\epsilon + \beta_{\ell}} \right)^m \right) \quad (28)$$

In the above, the smoothness indicators  $\beta_{\ell}$  are given by following the standard fifth order WENO reconstruction approach [11]. We also display their explicit forms in [Appendix A](#).

In Eqs. (26), (27), and (28),  $\epsilon$  is any arbitrarily small positive number that prevents division by zero, for which we choose  $\epsilon = 10^{-36}$ . One of the classical choice of  $\epsilon$  in many WENO literatures is found to be  $\epsilon = 10^{-6}$  [2,26]; however, it was suggested in [27,34] that  $\epsilon$  should be chosen to be much smaller in order to force this parameter to play only its original role of avoiding division by zero in the definitions of the weights, Eqs. (26) and (28).

Another closely related point of discussion is with the value of  $m$ , the power in the denominators in Eqs. (26) and (28). The parameter  $m$  determines the rate of changes in  $\beta_{\ell}$ , and most of the WENO literatures use  $m = 2$ . However, we observe that using  $m = 1$  resolves discontinuities sharper in most of our numerical simulations without exhibiting any numerical instability, so became the default value in our implementation. For more detailed discussions on the choices of  $\epsilon$  and  $m$ , see [27,28,34]. Using these nonlinear weights, we complete the WENO reconstruction procedure of producing the fifth-order spatially accurate, non-oscillatory reconstructed values at each cell interface at each time step  $t^n$  as defined in Eq. (25).

### 2.3. Reconstruction of the derivative $q'_{C,i}$

The spatial reconstruction part of PCM proceeds to the next final step to obtain the derivative  $q'_{C,i}$  in Eq. (13). The approach resembles with the  $r = 4$  WENO interpolation (i.e., pointwise) algorithm described in [3] in terms of using a set of third degree polynomials. The  $r = 4$  WENO algorithm evaluates seventh order accurate cell interface values  $q_{L,R,i}$  over a seven-point stencil using four third degree polynomials. In contrast, our approach is to reconstruct an approximation to a first derivative of a function  $q'(x_i, t^n)$  [11] over a five-point stencil, using two third degree polynomials.

For this, we might consider using the same strategy in Section 2.2 in which the three second degree ENO polynomials in Eq. (25) are constructed over the five-point stencil  $S = \cup_{\ell=1}^3 S_{\ell}$ . However, this setup will provide only a third-order accurate approximation  $q'_{C,i}$  to the exact derivative  $q'(x_i)$ . To see this, we first observe that the smoothness indicators  $\beta_{\ell}$  with this setup will be including only a single term,

$$\beta_{\ell} = \Delta x^3 \int_{I_i} \left[ p_{\ell}''(x) \right]^2 dx, \quad \ell = 1, 2, 3. \quad (29)$$

Through a Taylor expansion analysis on Eq. (29) we see

$$\beta_{\ell} = D(1 + \mathcal{O}(\Delta x)), \quad (30)$$

where  $D = (q'' \Delta x^2)^2$  is a nonzero quantity independent of  $\ell$  but may depend on  $\Delta x$ , assuming  $q'' \neq 0$  on  $S$ . This results in a set of three nonlinear weights  $\omega_{\ell}$ ,  $\ell = 1, 2, 3$ , obtained either by Eq. (26) or Eq. (28), satisfying

$$\omega_{\ell} = \gamma_{\ell} + \mathcal{O}(\Delta x), \quad (31)$$

where the linear constant weights  $\gamma_{\ell}$  are assumed to exist, when  $q'(x, t^n)$  is smooth in  $S$ , such that

$$q'_{C,i} = \sum_{\ell=1}^3 \gamma_{\ell} p'_{\ell}(x_i) = q'(x_i, t^n) + \mathcal{O}(\Delta x^3). \quad (32)$$

This finally implies the accuracy of  $q'_{C,i}$  is found out to be third-order,

$$q'_{C,i} = \sum_{\ell=1}^3 \omega_{\ell} p'_{\ell}(x_i) = q'(x_i, t^n) + \mathcal{O}(\Delta x^3), \quad (33)$$

because

$$\sum_{\ell=1}^3 \omega_{\ell} p'_{\ell}(x_i) - \sum_{\ell=1}^3 \gamma_{\ell} p'_{\ell}(x_i) = \sum_{\ell=1}^3 (\omega_{\ell} - \gamma_{\ell}) (p'_{\ell}(x_i) - q'(x_i, t^n)) = \sum_{\ell=1}^3 \mathcal{O}(\Delta x) \mathcal{O}(\Delta x^2) = \mathcal{O}(\Delta x^3). \quad (34)$$

In the last equality, we used the fact that, for each  $\ell$ ,  $p'_\ell(x)$  is only a first degree polynomial which is accurate up to second-order when approximating  $q'(x, t^n)$ .

For this reason, we want a better strategy to obtain an approximation  $q'_{C,i}$  at least fourth-order accurate in order that the overall nominal accuracy of the 1D PCM scheme achieves *at least* fourth-order accurate in both space and time.

*Step 1: PPM-build.* An alternate strategy for this goal therefore would be to use a set of two third degree polynomials  $\phi_\pm(x)$  over the same five-point stencil  $S = \cup_{\ell=1}^2 S_\ell$ , where each polynomial is defined on  $S_1 = \{I_{i-2}, \dots, I_{i+1}\}$  and  $S_2 = \{I_{i-1}, \dots, I_{i+2}\}$ . This can be designed using the two third degree polynomials,  $\phi_\pm(x)$ , from the PPM algorithm [1],

$$\phi_\pm(x) = \sum_{k=0}^3 a_k^\pm (x - x_{i \pm 1/2})^k, \quad (35)$$

where the coefficients are determined in such a way that  $\phi_\pm$  preserves the volume average on each cell  $I_i$ . The explicit forms of  $a_k^\pm$  are given in [Appendix B](#).

*Step 2: Linear constant weights.* Now that the polynomials are determined over the stencil  $S = \cup_{\ell=1}^2 S_\ell$ , we use their first derivatives  $\phi'_\pm$  to obtain a convex combination with two linear weights  $\gamma_-$  and  $\gamma_+$ ,

$$q'_{C,i} = \gamma_- \phi'_-(x_i) + \gamma_+ \phi'_+(x_i). \quad (36)$$

The two linear weights are determined to obtain a fourth order accuracy in  $q'_{C,i}$ ,

$$\gamma_- = \gamma_+ = \frac{1}{2}. \quad (37)$$

We display more details in [Appendix B](#).

*Step 3: Nonlinear weights.* The smoothness indicators  $\beta_\pm$  are now constructed using  $\phi_\pm(x)$  as

$$\beta_\pm = \sum_{s=2}^3 \left( \Delta x^{2s-1} \int_{I_i} \left[ \frac{d^s}{dx^s} \phi_\pm(x) \right]^2 dx \right). \quad (38)$$

It is to be noted that the summation begins from  $s = 2$  to measure their smoothness from the second derivative of  $\phi_\pm(x)$  because  $q'_{C,i}$  are our building blocks in this case. They can be written as

$$\beta_- = 4(a_2^-)^2 \Delta x^4 + 12(a_2^-)(a_3^-) \Delta x^5 + 48(a_3^-)^2 \Delta x^6 \quad (39)$$

and

$$\beta_+ = 4(a_2^+)^2 \Delta x^4 - 12(a_2^+)(a_3^+) \Delta x^5 + 48(a_3^+)^2 \Delta x^6. \quad (40)$$

Upon conducting Taylor series expansion analysis on  $\beta_\pm$ , we can see that

$$\beta_\pm = D(1 + \mathcal{O}(\Delta x)), \quad (41)$$

where  $D = (q'' \Delta x^2)^2$  is a nonzero quantity independent of  $\pm$  but might depend on  $\Delta x$ , assuming  $q'' \neq 0$  on  $S$ . It is observed that unlike the smoothness indicators in Eqs. (A.8)–(A.10), the new two smoothness indicators in Eqs. (39) and (40) are no longer written as a sum of perfect squares. Recently, Balsara et al. [35] reformulated the smoothness indicators as a sum of perfect squares for  $r = 3$  to  $r = 9$  finite difference WENO schemes. This compact form helps their new finite difference adaptive WENO-AO schemes become computationally more efficient than the previous approach reported in [3].

The remaining procedure is to obtain the two nonlinear weights  $\omega_\pm$  in the similar way done in the edge reconstructions in Eqs. (26)–(28):

- For WENO-JS:

$$\omega_\pm^{JS} = \frac{\tilde{\omega}_\pm}{\tilde{\omega}_- + \tilde{\omega}_+}, \text{ where } \tilde{\omega}_\pm = \frac{\gamma_\pm}{(\epsilon + \beta_\pm)^m} \quad (42)$$

- For WENO-M:

$$\omega_\pm^M = \frac{\tilde{\omega}_\pm}{\tilde{\omega}_- + \tilde{\omega}_+}, \text{ where } \tilde{\omega}_\pm = g(\omega_\pm^{JS}) = \frac{\omega_\pm^{JS} (\gamma_\pm + (\gamma_\pm)^2 - 3(\gamma_\pm)\omega_\pm^{JS} + (\omega_\pm^{JS})^2)}{(\gamma_\pm)^2 + \omega_\pm^{JS}(1 - 2\gamma_\pm)} \quad (43)$$

- For WENO-Z:

$$\omega_{\pm}^Z = \frac{\tilde{\omega}_{\pm}}{\tilde{\omega}_{-} + \tilde{\omega}_{+}}, \text{ where } \tilde{\omega}_{\pm} = \gamma_{\pm} \left( 1 + \left( \frac{|\beta_{+} - \beta_{-}|}{\epsilon + \beta_{\pm}} \right)^m \right) \quad (44)$$

The final representation of the approximation  $q'_{C,i}$  becomes

$$q'_{C,i} = \omega_{-}\phi'_{-}(x_i) + \omega_{+}\phi'_{+}(x_i). \quad (45)$$

Let us now verify that this approximation is fourth-order accurate after all, that is,

$$q'_{C,i} = \omega_{-}\phi_{-}(x_i) + \omega_{+}\phi_{+}(x_i) = q'(x_i, t^n) + \mathcal{O}(\Delta x^4). \quad (46)$$

Similarly as before, using Eqs. (41), (42), and (44), we can see that, with the help of the binomial series expansion,

$$\omega_{\pm} = \gamma_{\pm} + \mathcal{O}(\Delta x). \quad (47)$$

Therefore, the desired accuracy claimed in Eq. (46) is readily verified by repeating the similar relationship in Eq. (34):

$$\sum_{\ell=-,+} \omega_{\ell} \phi'_{\ell}(x_i) - \sum_{\ell=-,+} \gamma_{\ell} \phi'_{\ell}(x_i) = \sum_{\ell=-,+} (\omega_{\ell} - \gamma_{\ell}) (\phi'_{\ell}(x_i) - q'(x_i, t^n)) = \sum_{\ell=-,+} \mathcal{O}(\Delta x) \mathcal{O}(\Delta x^3) = \mathcal{O}(\Delta x^4). \quad (48)$$

Comparing Eq. (48) with Eq. (34), we now see that it is fourth-order accurate due to the improved third-order accuracy in calculating  $\phi'_{\ell}(x_i) - q'(x_i, t^n)$ . This is a result of using the third degree PPM polynomials  $\phi_{\pm}(x)$ , with which  $q'(x, t^n)$  can be accurately approximated by the second degree polynomials  $\phi'_{\pm}(x)$  up to third-order.

This completes the PCM spatial reconstruction steps that provide the fifth-order accurate Riemann states  $q_{L,R,i}$ , and the fourth-order accurate derivative  $q'_{C,i}$  in space. The remaining task includes conducting a temporal updating step via tracing the characteristic lines using the piecewise cubic polynomials in Eq. (9). This step produces the Riemann states  $(q_L, q_R) = (q_{R,i}^{n+1/2}, q_{L,i+1}^{n+1/2})$  as predictor. We will show in the next section that these predictors are at least fourth-order accurate in time for 1D cases, and they are provided as the initial value problems for the Godunov fluxes at each interface  $x_{i+1/2}$ .

### 3. The PCM characteristic tracing for temporal updates

The PCM proceeds to the last step which advances the pointwise Riemann interface states at  $t^n$

$$q_{L,R,i} = p_i(x_{i \pm 1/2}), \quad (49)$$

where  $p_i(x)$  is the piecewise cubic polynomial in Eq. (9), to the half-time updated predictor states

$$q_{L,R,i}^{n+1/2} \quad (50)$$

by tracing characteristics. The idea is the same as how the PPM characteristic tracing is performed [1], in which we seek a time averaged state. For instance, at the interface  $x_{i+1/2}$ , we consider

$$q_{x_{i+1/2}}^{n+1/2} = \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} q(x_{i+1/2}, t) dt. \quad (51)$$

The initial condition at  $t^n$  of a Generalized Riemann Problem (GRP) is given as

$$q(x_{i+1/2}, t^n) = \begin{cases} p_i(x_{i+1/2}), & x \in I_i \\ p_{i+1}(x_{i+1/2}), & x \in I_{i+1}. \end{cases} \quad (52)$$

Given a linear characteristic equation as in Eq. (8), and for  $t > t^n$  we then have

$$q(x_{i+1/2}, t) = \begin{cases} p_i(x_{i+1/2} - \lambda_i(t - t^n)), & x \in I_i, \lambda_i > 0, \\ p_{i+1}(x_{i+1/2} - \lambda_{i+1}(t - t^n)), & x \in I_{i+1}, \lambda_{i+1} < 0. \end{cases} \quad (53)$$

Here  $\xi(t) = x_{i+1/2} - \lambda(t - t^n)$  is a characteristic line for an eigenvalue  $\lambda$ , assuming  $t - t^n < \Delta t$ .

In this approach we do not solve the Generalized Riemann Problem directly. Rather we use a trapezium rule approximation in time to the integral in Eq. (51), where  $q(x_{i+1/2}, t)$  the solution of the GRP along the  $t$ -axis at the interface  $x_{i+1/2}$  in the  $t$ - $x$  plane. Although this approach is very simple and can be considered to be very similar to the second order characteristic tracing algorithms of PLM and PPM [1,36] we see there is a drawback in general. As it stands, the state  $q(x_{i+1/2}, t)$  constructed using the linear characteristic line  $\xi(t)$  in Eq. (53) is not fully justified in the cases of the Generalized Riemann problem where the characteristics are curved lines instead of straight lines in the  $x$ - $t$  plane. On the other hand, it has

been shown by Qian et al. [22] that a framework of high-order GRP methods can be formulated for second and third order accuracy by using a linear and quadratic GRP solvers respectively. Their approaches, relying on new treatments of resolving curved rarefaction and discontinuity waves, illustrate that the use of linear characteristics is not entirely supported in the third and higher order cases, especially for the system of cases.

Regardless, our simple characteristic tracing strategy has shown that a fast rate of convergence can be achieved for  $\Delta t \sim \Delta x$ , which is now described below for the case with  $x \in I_i$  with  $\lambda_i > 0$  first. Using  $\uparrow$  to denote the state from the left of  $x_{i+1/2}$ ,

$$q_{R,i}^{n+1/2} = q_{x+1/2,\uparrow}^{n+1/2} = \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} q(x_{i+1/2}, t) dt = \frac{1}{\lambda_i \Delta t} \int_{x_{i+1/2}-\lambda_i \Delta t}^{x_{i+1/2}} p_i(x) dx. \quad (54)$$

Again, as seen in Eq. (54), the half-time advancement of the spatially reconstructed state is given by the average of the reconstructed variable  $p_i(x)$  over the domain of dependence  $[x_{i+1/2}-\lambda_i \Delta t, x_{i+1/2}]$  of the interface  $x_{i+1/2}$ . Therefore the accuracy of  $q_{R,i}^{n+1/2}$  is inherited from that of the reconstruction algorithm of  $p_i(x)$ .

The outcome of the integration yields

$$\begin{aligned} q_{R,i}^{n+1/2} = q_{x+1/2,\uparrow}^{n+1/2} &= c_0 + \frac{c_1}{2} \left( 1 - \frac{\lambda_i \Delta t}{\Delta x} \right) \Delta x + \frac{c_2}{4} \left( 1 - 2 \frac{\lambda_i \Delta t}{\Delta x} + \frac{4}{3} \left( \frac{\lambda_i \Delta t}{\Delta x} \right)^2 \right) \Delta x^2 \\ &+ \frac{c_3}{8} \left( 1 - 3 \frac{\lambda_i \Delta t}{\Delta x} + 4 \left( \frac{\lambda_i \Delta t}{\Delta x} \right)^2 - 2 \left( \frac{\lambda_i \Delta t}{\Delta x} \right)^3 \right) \Delta x^3. \end{aligned} \quad (55)$$

The case for  $x \in I_{i+1}$  with  $\lambda_{i+1} < 0$  can be obtained similarly,

$$\begin{aligned} q_{L,i+1}^{n+1/2} = q_{x+1/2,\downarrow}^{n+1/2} &= c_0 + \frac{c_1}{2} \left( -1 - \frac{\lambda_{i+1} \Delta t}{\Delta x} \right) \Delta x + \frac{c_2}{4} \left( 1 + 2 \frac{\lambda_{i+1} \Delta t}{\Delta x} + \frac{4}{3} \left( \frac{\lambda_{i+1} \Delta t}{\Delta x} \right)^2 \right) \Delta x^2 \\ &+ \frac{c_3}{8} \left( -1 - 3 \frac{\lambda_{i+1} \Delta t}{\Delta x} - 4 \left( \frac{\lambda_{i+1} \Delta t}{\Delta x} \right)^2 - 2 \left( \frac{\lambda_{i+1} \Delta t}{\Delta x} \right)^3 \right) \Delta x^3. \end{aligned} \quad (56)$$

In the general case of a system of Euler equations, the above treatment is to be extended to include multiple characteristic waves correspondingly depending on the sign of each  $k$ -th eigenvalue  $\lambda_i^{(k)}$ . This gives us, for the two predictor states  $\mathbf{v}_{L,R,i}^{n+1/2}$  on each cell  $I_i$  in primitive form,

$$\begin{aligned} \mathbf{v}_{R,i}^{n+1/2} = \mathbf{C}_0 &+ \frac{1}{2} \sum_{k; \lambda_i^{(k)} > 0} \left( 1 - \frac{\lambda_i^{(k)} \Delta t}{\Delta x} \right) r^{(k)} \Delta \mathbf{C}_1^{(k)} + \frac{1}{4} \sum_{k; \lambda_i^{(k)} > 0} \left( 1 - 2 \frac{\lambda_i^{(k)} \Delta t}{\Delta x} + \frac{4}{3} \left( \frac{\lambda_i^{(k)} \Delta t}{\Delta x} \right)^2 \right) r^{(k)} \Delta \mathbf{C}_2^{(k)}, \\ &+ \frac{1}{8} \sum_{k; \lambda_i^{(k)} > 0} \left( 1 - 3 \frac{\lambda_i^{(k)} \Delta t}{\Delta x} + 4 \left( \frac{\lambda_i^{(k)} \Delta t}{\Delta x} \right)^2 - 2 \left( \frac{\lambda_i^{(k)} \Delta t}{\Delta x} \right)^3 \right) r^{(k)} \Delta \mathbf{C}_3^{(k)}, \end{aligned} \quad (57)$$

and

$$\begin{aligned} \mathbf{v}_{L,i}^{n+1/2} = \mathbf{C}_0 &+ \frac{1}{2} \sum_{k; \lambda_i^{(k)} < 0} \left( -1 - \frac{\lambda_i^{(k)} \Delta t}{\Delta x} \right) r^{(k)} \Delta \mathbf{C}_1^{(k)} + \frac{1}{4} \sum_{k; \lambda_i^{(k)} < 0} \left( 1 + 2 \frac{\lambda_i^{(k)} \Delta t}{\Delta x} + \frac{4}{3} \left( \frac{\lambda_i^{(k)} \Delta t}{\Delta x} \right)^2 \right) r^{(k)} \Delta \mathbf{C}_2^{(k)}, \\ &+ \frac{1}{8} \sum_{k; \lambda_i^{(k)} < 0} \left( -1 - 3 \frac{\lambda_i^{(k)} \Delta t}{\Delta x} - 4 \left( \frac{\lambda_i^{(k)} \Delta t}{\Delta x} \right)^2 - 2 \left( \frac{\lambda_i^{(k)} \Delta t}{\Delta x} \right)^3 \right) r^{(k)} \Delta \mathbf{C}_3^{(k)}. \end{aligned} \quad (58)$$

Those new notations introduced in Eqs. (57) and (58) represent the  $k$ -th right eigenvector  $r^{(k)}$ ,  $k = 1, 2, 3$ , which is the  $k$ -th column vector of the  $3 \times 3$  matrix  $\mathbf{R}$  evaluated at  $I_i$ ,

$$\mathbf{R} = \begin{bmatrix} r^{(1)} | r^{(2)} | r^{(3)} \end{bmatrix}, \quad (59)$$

and the  $k$ -th characteristic variable vector  $\Delta \mathbf{C}_m^{(k)}$  given as

$$\Delta \mathbf{C}_m^{(k)} = \Delta x^m \ell^{(k)} \cdot \mathbf{C}_m, \quad (60)$$

where, for  $m = 0, \dots, 3$ ,

$$\mathbf{C}_m = \begin{bmatrix} c_m^{(1)} | c_m^{(2)} | c_m^{(3)} \end{bmatrix}^T, \quad (61)$$

in which  $c_m^{(k)}$  is the  $m$ -th coefficient in Eqs. (21)–(24) of the piecewise cubic polynomial in Eq. (9) applied to each of the  $k$ -th characteristic variable  $\bar{q}_i$ .

It is worth mentioning that, unlike the characteristic tracing of PPM (see [1]), the PCM scheme does not necessarily require any extra monotonicity enforcements on  $\mathbf{V}_{L;R,i}^{n+1/2}$ . This is mainly because the PCM reconstruction makes full use of the WENO reconstruction algorithms, providing non-oscillatory approximations to  $q_{L;R,i}$  and  $q'_{C,i}$  by design.

#### 4. Final update step in 1D

We now update  $\bar{\mathbf{U}}_i^n$  to  $\bar{\mathbf{U}}_i^{n+1}$  using the high-order Godunov fluxes  $\mathbf{F}_{i+1/2}^{n+1/2} = \mathcal{RP}(\mathbf{U}_L, \mathbf{U}_R) = \mathcal{RP}(\mathbf{U}_{R,i}^{n+1/2}, \mathbf{U}_{L,i+1}^{n+1/2})$  as corrector, where  $\mathcal{RP}$  implies a solution of the Riemann problem. Note that the Riemann states in conservative variables  $\mathbf{U}_{R,i}^{n+1/2}, \mathbf{U}_{L,i+1}^{n+1/2}$  are obtained either by conversions from  $\mathbf{V}_{R,i}^{n+1/2}, \mathbf{V}_{L,i+1}^{n+1/2}$  in Eq. (57) and Eq. (58), or projecting the characteristic variables directly to the conservative variables in Eq. (57) and Eq. (58). We note that the first needs to be processed using high-order approximation [14], in particular for multidimensional problems, while such a high-order conversion is not required in 1D. In this regard the latter could be a better choice in multi-spatial dimensions, because there is no need for any high-order conversion from the primitive Riemann states to the conservative Riemann states, knowing the fact that the conservative states variables are the type of inputs for the Riemann problems.

#### 5. Multidimensional extension of the 1D PCM scheme

Our primary purpose in the current paper is to focus on laying down the key algorithmic components of PCM in 1D. As described, the 1D PCM algorithm is formally fifth-order in space and fourth-order in time. Our test problems of one-dimensional smooth flows in Section 6.1.1 show that the algorithm delivers nominally a fifth-order accurate convergence rate, particularly with smaller  $L_1$  errors than WENO-JS with RK4.

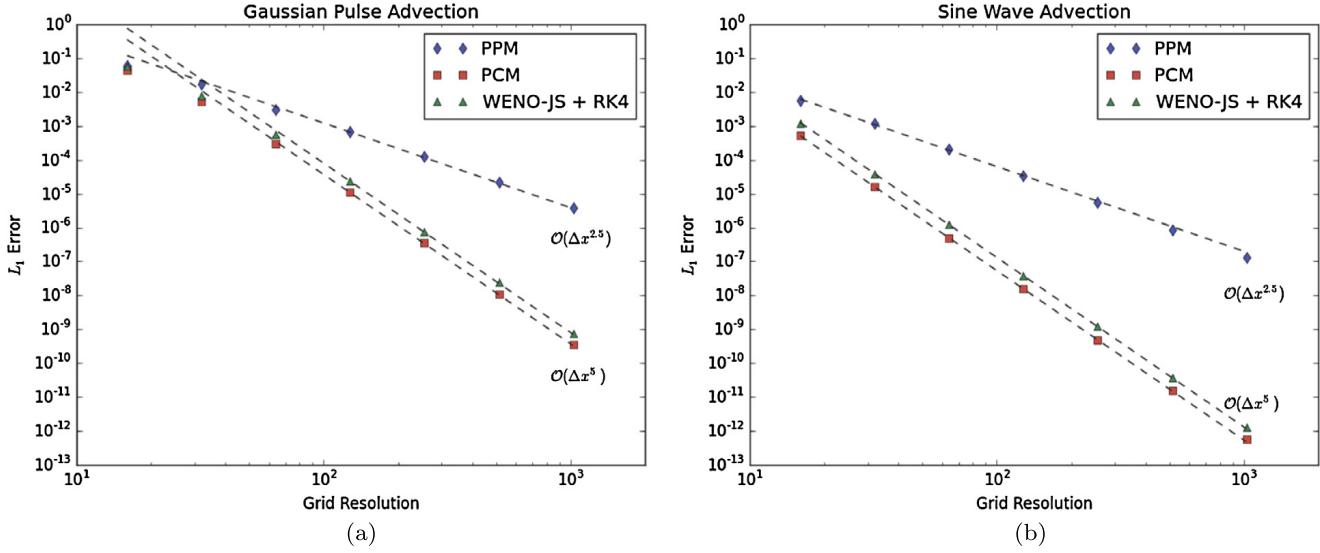
Although possible, extending such a high-order 1D algorithm to multiple spatial dimensions in a way to preserve the same order of convergence in 1D is an attentive task that requires some extra care and attentions [11–14] in the finite volume formulation. On the other hand, one of the simplest and easiest directionally unsplit multidimensional extensions that has been widely adopted in many algorithmic choices [37–47] is to use the so-called *dimension-by-dimension* formalism [12]. In this approach, a finite volume scheme adopts a 1D reconstruction procedure to obtain high order accurate edge values at cell faces, use them to approximate fluxes at a single barycentric point on each cell face, and update the conserved variables using those fluxes in a directionally unsplit way. However, the order of convergence from the resulting multidimensional extension is limited to be at most second-order due to the lack of accuracies that may arise in a couple of places in code implementations: (i) mis-using averaged quantities in place of pointwise quantities for Riemann states, (ii) using low-order approximations in converting between primitive and conservative variables, (iii) and applying low-order quadrature rules in flux function estimations [11–14]. In our case, it takes more coding efforts, practically because the multidimensional PCM results we demonstrate in this paper have been obtained by integrating the PCM algorithm in the FLASH code framework [41,42]; hence carrying out the above-mentioned code changes in a large code such as FLASH requires extra efforts that are not the main points of the current paper. We leave such a high-order, multidimensional extension in our future work, and instead, we adopt the simple dimension-by-dimension formalism for our multidimensional extension of the 1D PCM algorithm.

Additionally, for our choice of multidimensional extension we use the computationally efficient unsplit corner transport upwind (CTU) formulation in FLASH [43,44], which requires smaller number of Riemann solves in both 2D and 3D than the conventional CTU approaches [48–50], while achieving the maximum Courant condition of  $CFL \approx 1$  [43,44].

#### 6. Results

In this section we present numerical results of PCM in 1D, 2D and 3D for hydrodynamics and magnetohydrodynamics. The PCM results are compared with numerical solutions of other popular choices of reconstruction schemes including the second-order PLM [36], the third-order PPM [1] and the fifth-order WENO methods [2,27,28]. As mentioned, the second-order accurate dimension-by-dimension approach has been adopted to extend all of the above baseline 1D reconstruction algorithms to multiple spatial dimensions. As this is the case, for multidimensional problems we have chosen the predictor-corrector type of characteristic tracing methods (charTr) for PLM, PPM, and WENO<sup>2</sup>, not to mention PCM by design. In 1D problems, however, we treat WENO differently and integrate its spatial reconstruction with RK4 in consideration of fully demonstrating its orders of accuracy due from both space (i.e.,  $\mathcal{O}(\Delta x^5)$ ) and time (i.e.,  $\mathcal{O}(\Delta t^4)$ ). It should also be noted that the orders of WENO + RK4 in 1D are to be well comparable to those of PCM. Hence the choice provides a set of good informative comparisons between PCM and WENO + RK4 in particular, in which we will illuminate the advantages of PCM.

<sup>2</sup> To implement a characteristic tracing for WENO we first reconstruct the fifth-order Riemann states,  $q_{S,i}^{weno,n}$ ,  $S = L, R$ , using WENO. They are then temporally evolved by  $\Delta t/2$  to get  $q_{S,i}^{weno,n+1/2}$  by integrating over each corresponding domain of dependence the piecewise parabolic polynomials defined by  $q_{L,i}^{weno,n}$ ,  $q_{R,i}^{weno,n}$  and  $\bar{q}_i$ .



**Fig. 1.** Convergence test of (a) 1D Gaussian pulse advection, and (b) 1D sinusoidal wave advection.

In what follows, unless otherwise mentioned, we set the WENO-JS approach in Eq. (42) as the default choice for  $q'_{C,i}$  in our PCM results. This default setting will be referred to as *PCM-JS* (or simply *PCM*), while the choice with the WENO-Z approach in Eq. (44) will be referred to as *PCM-Z*.

### 6.1. 1D tests

#### 6.1.1. 1D convergence and performance tests

**Gaussian and sinusoidal wave advections.** In our first test we consider two configurations of 1D passive advection of smooth flows, involving initial density profiles of Gaussian and sinusoidal waves. We initialize the both problems on a computational box on  $[0, 1]$  with periodic boundary conditions. The initial density profile of the Gaussian advection is defined by  $\rho(x) = 1 + e^{-100(x-x_0)^2}$ , with  $x_0 = 0.5$ , whereas for the sinusoidal advection the density is initialized by  $\rho(x) = 1.5 - 0.5 \sin(2\pi x)$ . In both cases, we set constant velocity,  $u = 1$ , and pressure,  $P = 1/\gamma$ , and the specific heat ratio,  $\gamma = 5/3$ .

The resulting profiles are propagated for one period through the boundaries, reaching  $t = 1$ . At this point, both profiles return to its initial positions at which we conduct the  $L_1$  error convergence tests compared with the initial conditions on the grid resolutions of  $N_x = 16, 32, 64, 128, 256, 512$  and  $1024$ . Since the nature of the problem is a pure advection in both, any deformation of the initial profile is due to either phase errors or numerical diffusion. For stability we use a fixed Courant number,  $C_{\text{cfl}} = 0.8$  for both tests. We choose the HLLC Riemann solver [51] in all cases.

The results of this study are shown in Fig. 1. From these numerical experiments, the PCM reconstruction shows the fifth-order convergence rates in both tests. Although both PCM and WENO-JS + RK4 demonstrate the same fifth-order of convergence rate, the  $L_1$  errors of PCM are more than twice smaller than those of WENO-JS + RK4. The solutions of PPM converge with the rate of 2.5 which is the slowest among the three. Parameter choices for the PPM runs include the use of the MC slope limiter applied to characteristic variables, no flattening, no contact discontinuity steepening, and no artificial viscosity (this setting for PPM remains the same in what follows).

In Table 1 we compare the relative performance speedups of PCM, PPM and WENO-JS + RK4, all testing the Gaussian and sinusoidal advection problems. We can clearly see that there is a big performance advantage in PCM over WENO-JS + RK4 in delivering the target fifth-order accuracy. The major gain in PCM lies in its predictor–corrector type of characteristic tracing which affords not only the accuracy but also the computational efficiency. Such a relative computational efficiency of PCM in 1D is expected to grow much larger in multidimensional problems, considering that there have to be added algorithmic complexities in achieving high-order accurate solutions in multidimensional finite volume reconstruction [11–14] from the perspectives of balancing optimal numerical stability and accuracy. We will report our strategies of multidimensional extension of PCM in our future work.

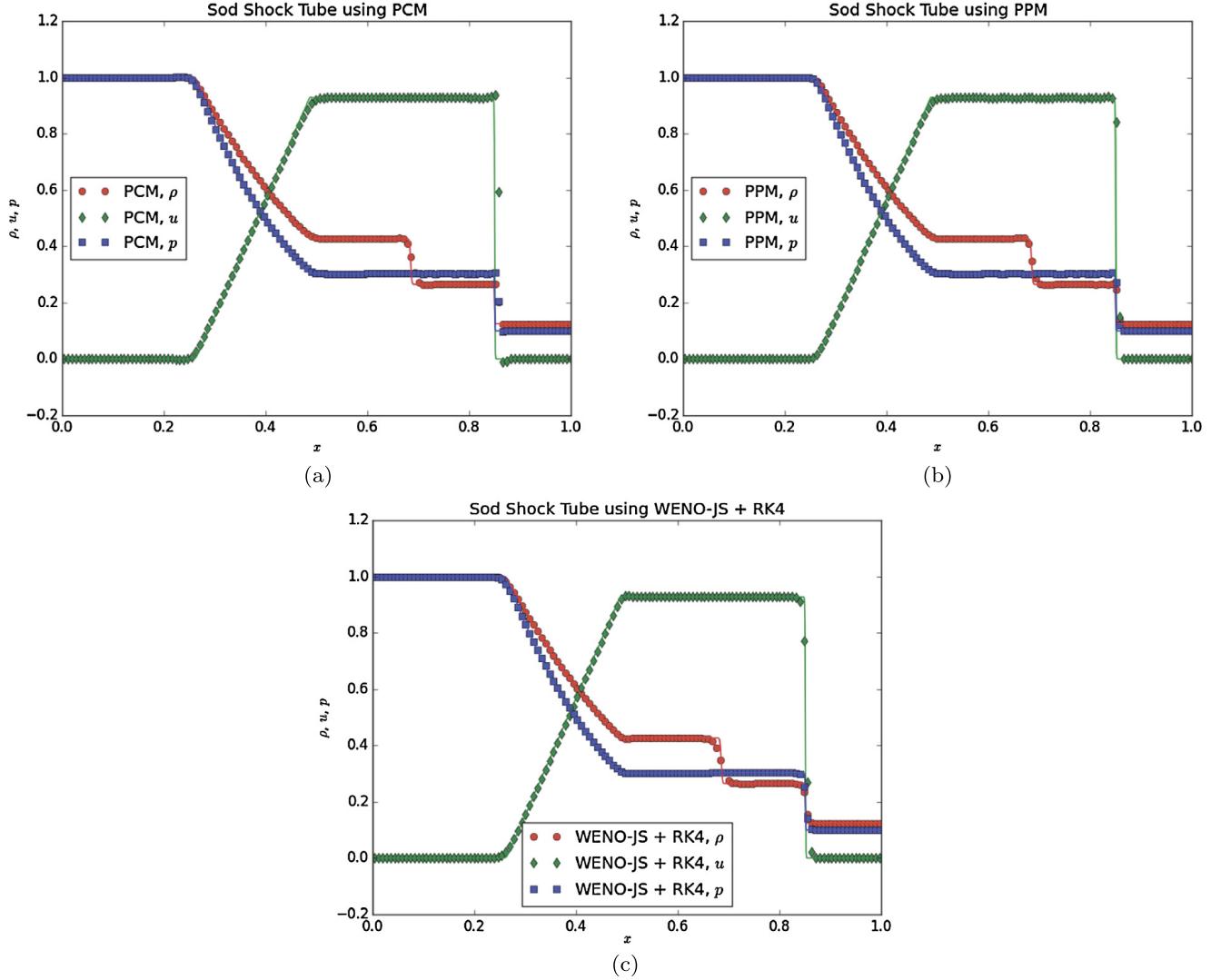
#### 6.1.2. 1D Discontinuous tests

In this section we test PCM on a series of well-benchmarked shock-tube problems of one dimensional hydrodynamics and MHD that involve discontinuities and shocks. As all the tests here have already been well discussed in various literatures, we will describe their setups only briefly and put our emphasis more on discussing the code performance of PCM. Readers are encouraged to refer to the cited references in the texts for more detailed descriptions on each setup.

**Table 1**

Relative speedup of the PPM and WENO schemes compared to the PCM scheme for the 1D Gaussian and sine advection problems. The comparisons have been obtained from a serial calculation on a single CPU.

Scheme	Speedup
PPM	0.65
PCM	1.00
WENO-JS + RK4	1.71

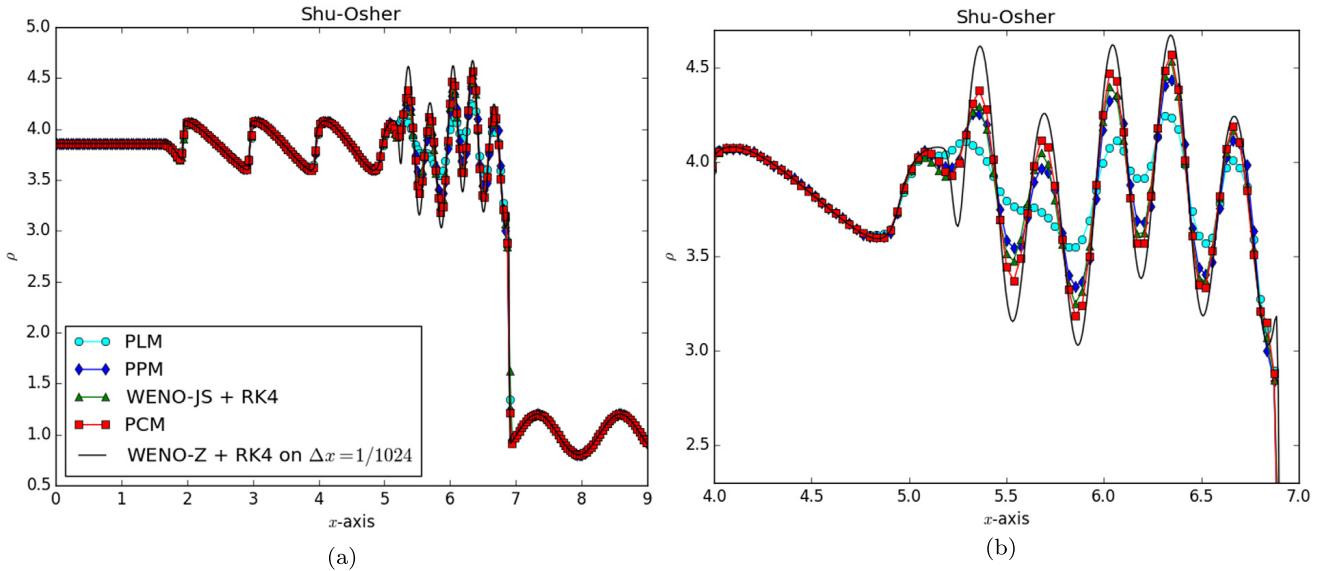


**Fig. 2.** The Sod shock tube problem at  $t = 0.2$ . (a) PCM, (b) PPM with MC slope limiter, and (c) WENO-JS + RK4.

(a) *Sod shock tube*. The Sod's problem [52] has been one of the most widely chosen popular tests in 1D to assess a code's capability to handle shocks and contact discontinuities. The initial condition is consist of the left and the right states given as

$$(\rho, u, p) = \begin{cases} (1, 0, 1) & x < 0.5, \\ (0.125, 0, 0.1) & x > 0.5, \end{cases} \quad (62)$$

with the ratio of specific heats  $\gamma = 1.4$  on the entire domain  $[0, 1]$ . The outflow boundary conditions are imposed at  $x = 0$  and  $x = 1$ . Shown in Fig. 2 include numerical solutions of PCM, PPM and WENO-JS + RK4. The Roe Riemann solver [53] was used in all cases. The test cases (denoted in symbols) are resolved on the grid size of  $N_x = 128$ , and are compared with the reference solutions (denoted in solid curves) computed using WENO-JS + RK4 on the grid resolution of  $N_x = 1024$ . A fixed value of  $C_{\text{cfl}} = 0.8$  was used for all tests.



**Fig. 3.** The Shu-Osher Riemann problem at  $t = 1.8$ . (a) All four reconstruction schemes on  $N_x = 256$  are compared with WENO-JS + RK4 on  $N_x = 1024$ . For PPM, the MC slope limiter is used. (b) A close-up view to demonstrate the schemes' numerical diffusivity.

The result in Fig. 2(a) shows that the solutions of PCM well predict all nonlinear flow characteristics of the rarefaction wave, the contact discontinuity, and the shock. A notable thing in PCM is the number of points at the shock. We see in Fig. 2(a) that at the shock there is only one single point in all flow variables, whereas in all other cases, there are two points spread over the shock width.

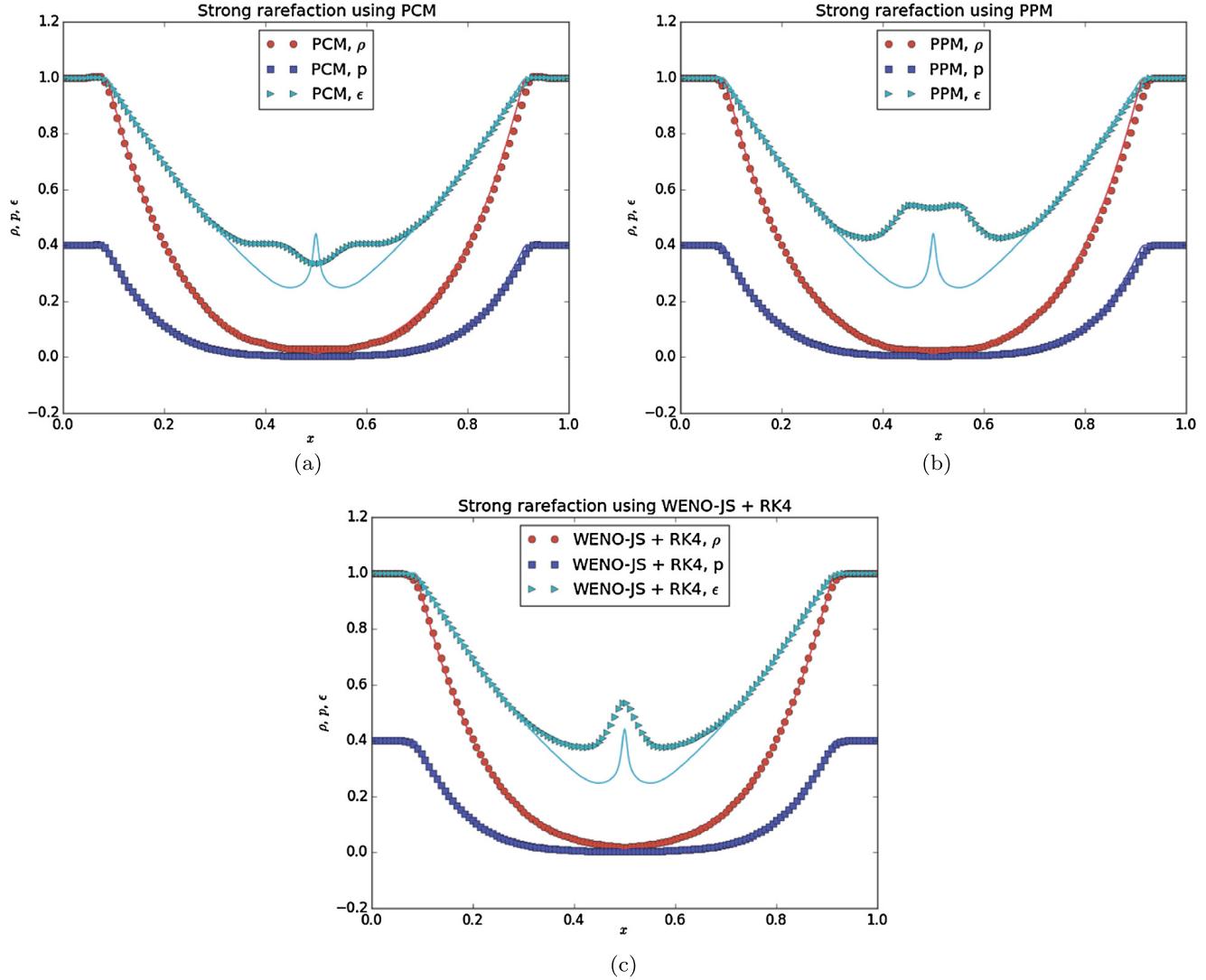
(b) *The Shu-Osher test.* The second test is the Shu-Osher problem [54]. In this problem we test PCM's ability to resolve both small-scale smooth flow features and the shock. On  $[-4.5, 4.5]$ , the initial condition launches a nominally Mach 3 shock wave at  $x = -4.0$  propagating into a region ( $x > -4.0$ ) of a constant density field with sinusoidal perturbations. As the shock advances, two sets of density features appear behind the shock. The first set has the same spatial frequency as the un-shocked perturbations, whereas the second set behind the shock involves the frequency that is doubled. The important point of the test is to see how well a code can accurately resolve strengths of the oscillations behind the shock, as well as the shock itself.

The results of this test are shown for PLM, PPM, WENO-JS + RK4, and PCM in Fig. 3. The solutions are calculated at  $t = 1.8$  using a resolution of  $N_x = 256$  and are compared to a reference solution resolved on  $N_x = 1024$ . All methods were solved using the Roe Riemann solver, with  $C_{\text{cfl}} = 0.8$ . It is evident in Fig. 3(b) that the PCM solution exhibits the least diffusive solution among the tested methods, producing a very-high order accurate solution that is more quickly approaching to the high resolution reference solution.

(c) *The Einfeldt strong rarefaction test.* First described by Einfeldt et al. [55] the main test point in this problem is to see how satisfactorily a code can compute physical variables,  $p, u, \rho, \epsilon$ , etc. in the low density region. Among the variables the internal energy  $\epsilon = p/(\rho(\gamma - 1))$ , where  $\gamma = 1.4$ , is the hardest to get it right due to the ratio of the pressure and density that are both close to zero. The ratio of the two small quantities will amplify any small errors in each, hence making the error in  $\epsilon$  appear to be the largest in general [10].

The large errors in  $\epsilon$  are indeed observed in Fig. 4(b)–Fig. 4(c) in that the error is the largest at or around  $x = 0.5$  in the presence of sudden increase of its peak values. On the contrary, the internal energy computed using PCM shown in Fig. 4(a) behaves in a uniquely different way such that the value continues to drop when approaching  $x = 0.5$ . From this viewpoint, and with the help of the exact solution available in [10], it's fair to say that the PCM solution in Fig. 4(a) appears to predict the internal energy most accurately.

(d) *Two-blast.* This problem was introduced by Woodward and Colella [56] and was designed to test a code performance particularly on interactions of strong shocks and discontinuities. We follow the original setup to test PCM, and compare its solution with those of PPM and WENO-JS + RK4, using 128 grid points to resolve the domain  $[0, 1]$ . In Fig. 5 the three density profiles at  $t = 0.038$  are plotted against the high-resolution solution of WENO-JS + RK4 on 1024 grid points. Overall, all methods we tested here produce an acceptable quality of solutions as illustrated in Fig. 5(a). Note however that, among the three methods, the PCM solution in Fig. 5(b) demonstrates the highest peak heights, following more closely the high-resolution solutions. As reported in [40] we see that all methods also smear out the contact discontinuity at  $x \approx 0.6$  pretty much the same amount.

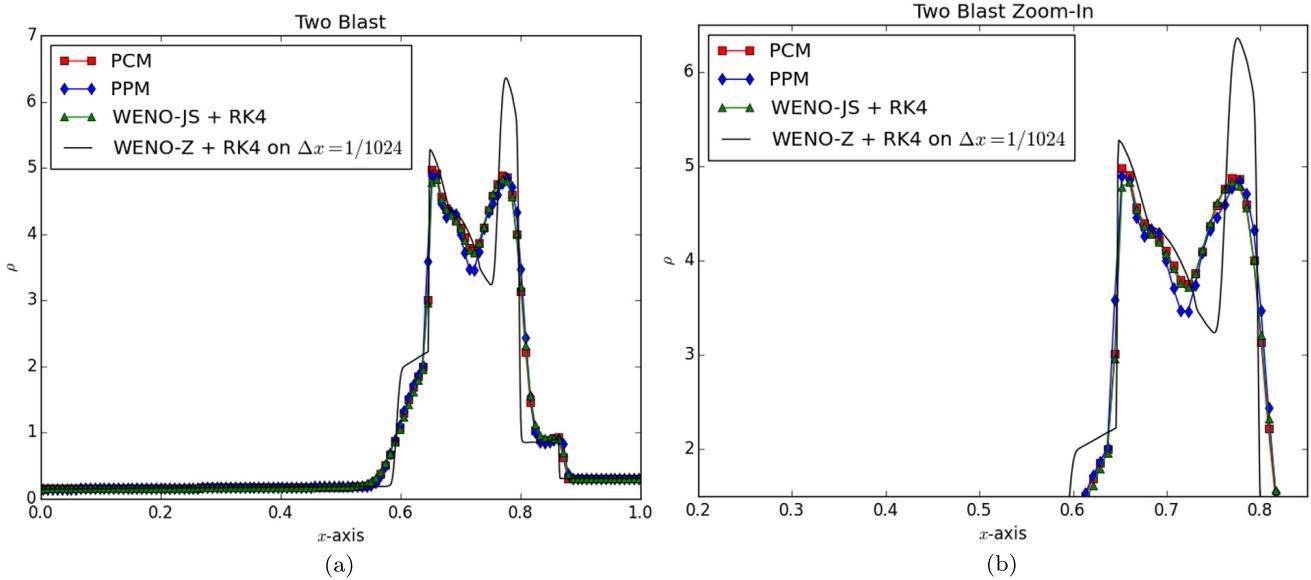


**Fig. 4.** The Einfeldt strong rarefaction test at  $t = 0.15$ . All tests used the HLLC Riemann solver on  $N_x = 128$  with  $C_{\text{cfl}} = 0.8$ . The solid curves represent the reference solution computed using WENO-JS + RK4 on  $N_x = 1024$ .

(e) *Brio–Wu MHD shock tube*. An MHD version of the Sod’s shock tube problem was first studied by Brio and Wu [33], and it has become a must-to-do test for MHD codes. Since then, the problem has revealed a couple of interesting findings including not only the discovery of the compound wave [33], but also the existence of non-unique solutions [57,58]. More recently, Lee [59] realized that there are unphysical numerical oscillations in using PPM and studied an approach to suppress the level of oscillations based on the upwind slope limiter. The presence of such oscillations in PPM has been also briefly reported in [40]. The study reported in [59] shows that the origin of the oscillations arise from the numerical nature of a slowly moving shock as a function of the magnetic strength of tangential component. The slowly moving shock was first identified in [56], and the oscillatory behaviors have been studied by many researchers for more than 30 years, yet there is no ultimate resolution [60–65].

In this test, as just mentioned, there are observable numerical oscillations found in all methods, PPM, PCM and WENO-JS + charTr. The results in Fig. 6 show that the oscillations are the largest in PPM, consistent with the findings in [59], and there are less amount in PCM and WENO-JS + charTr. The PPM solutions are suffering from significant amount of spurious oscillations in all four variables,  $\rho$ ,  $u$ ,  $p$  and  $B_y$ , as shown in Fig. 6(b). Such behaviors are less significant in PCM and WENO-JS + charTr, respectively illustrated in Fig. 6(a) and Fig. 6(c), in that, the oscillations in  $\rho$ ,  $p$ ,  $B_y$  are much more controlled now, while the most outstanding oscillations are found in  $u$  near  $x \approx 0.8$ . It is worth mentioning that the oscillatory behaviors remain to be consistent regardless of the choice of Riemann solvers such as HLL [66], HLLC [67], HLLD [68], or Roe [53] (tested here).

(f) *RJ2a MHD shock tube*. Ryu et al. [69] studied a class of one dimensional MHD shock tube problems that are informative to run as a code verification test. We have chosen one of their setups, introduced in their Fig. 2a. In what follows the problem is referred to be as the RJ2a test. The viewpoint of this test is to monitor if all three dimensional MHD waves



**Fig. 5.** Two blast problem at  $t = 0.038$ . (a) All three reconstruction schemes on  $N_x = 128$  are compared with WENO-JS + RK4 on  $N_x = 1024$ . For PPM, the MC slope limiter is used. (b) A close-up view to demonstrate the schemes' numerical accuracy. In all tests we used the HLLC Riemann solver with  $C_{\text{cfl}} = 0.8$ .

are successfully captured. We see that in Fig. 7 all structures of left- and right-going fast shocks, left- and right-going slow shocks, and a contact discontinuity are well captured in all methods tested, including PCM.

## 6.2. 2D tests

We present two dimensional tests of hydrodynamics and MHD in this section. All test cases are computed using the second-order dimension-by-dimension extension of the baseline 1D algorithms, including PCM.

### 6.2.1. 2D convergence test of the isentropic vortex advection – hydro case

The first 2D test problem, considered in [3,70], consists of the advection of an isentropic vortex along the diagonal of a Cartesian computational box. The dynamics of the problem allows to quantify a code's dissipative properties and the correct discretization balance of multidimensional terms through monitoring the preservation of the initial circular shape of the vortex. At  $t = 10$  the vortex finishes one periodic advection over the domain and returns to the initial position, where we can measure the solution accuracy against the initial condition. As such we have chosen this problem particularly to access the PCM's order of convergence rate in 2D. We omit the details of the initial problem setup which can be found in [3,70].

As expected, the results presented in Fig. 8 clearly confirm that, when PCM is extended to 2D using the simple dimension-by-dimension formulation, the overall numerical solution accuracy converges in second-order, regardless of its inherent fifth-order property in 1D.

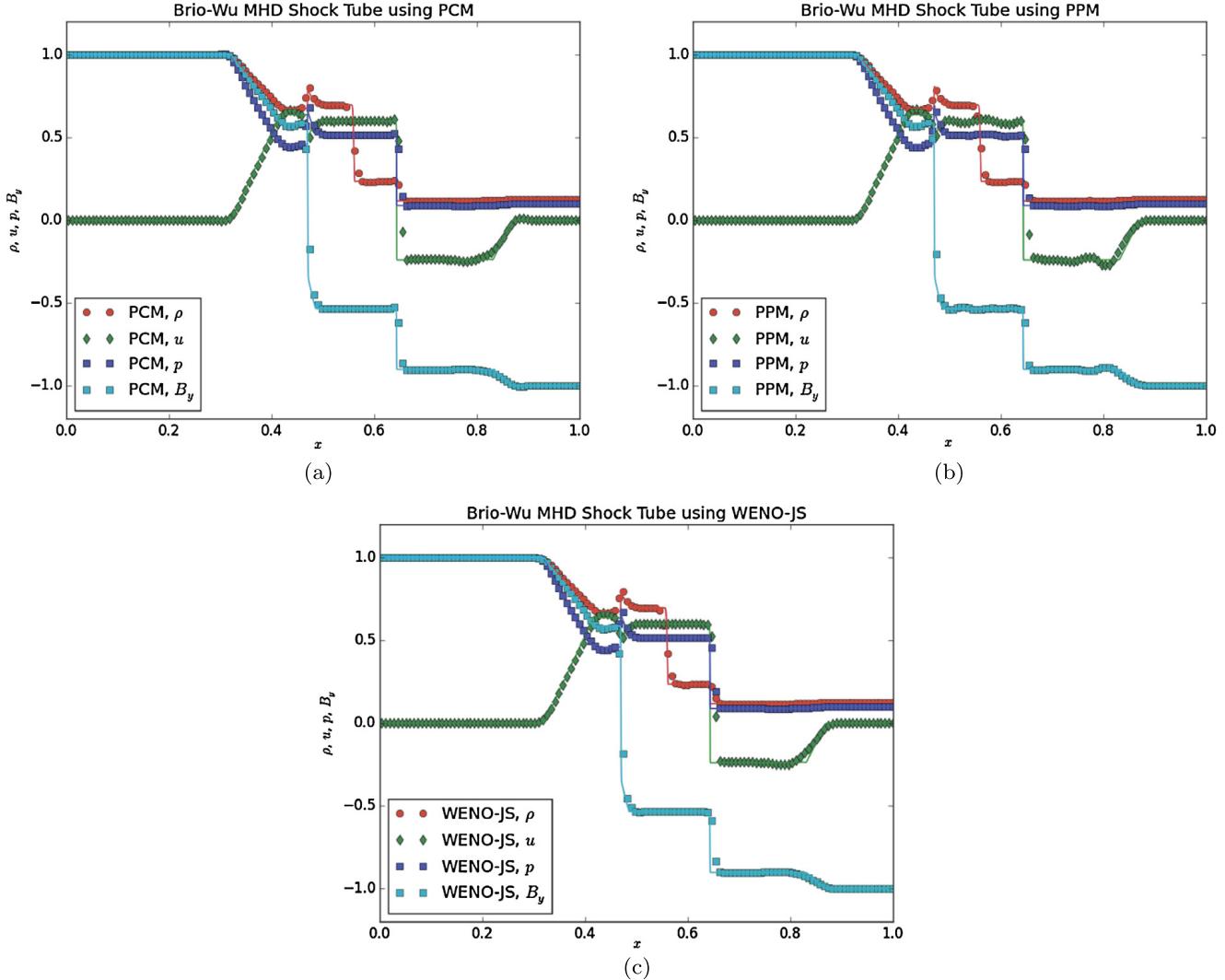
### 6.2.2. 2D convergence test of the isentropic vortex advection – MHD case

We also ran an MHD 2D isentropic vortex test which is an analogous MHD version of the previous hydro vortex test, in order to demonstrate PCM's convergence rate on the magnetic fields as well. The test was presented by Balsara [71] and we omit the details of the setup for brevity. As expected, the results presented in Fig. 9 clearly confirm that, PCM demonstrate second order accuracy in two dimensional magnetic field evolutions.

### 6.2.3. 2D Discontinuous tests

(a) Sedov. We consider the Sedov blast test [72] to check PCM's ability to handle a spherical symmetry of the strong hydro-dynamical shock explosion. The problem studies a self-similar evolution of a spherical shock wave propagation due to an initial point-source of a highly pressurized perturbation. The test has been used widely in various literatures, and we follow the same setup found in [41]. Panels in Fig. 10 show the density field in linear scale at  $t = 0.05$  resolved on a grid size of  $256 \times 256$  for the domain  $[0, 1] \times [0, 1]$ . The HLLC Riemann solver was used in all runs with  $C_{\text{cfl}} = 0.8$ . The range of the plotted densities in colors in all four panels is  $0.01 \leq \rho \leq 4.9$ , the same is also used for the 30 levels of density contour lines that are plotted in logarithmic scale.

As can be seen, the PCM solution in Fig. 10(a) is superior not only in preserving a great deal of the spherical symmetry at the outermost shock front, but also in revealing more flow structures in the central low density region, again in the most spherical manner. This great ability of preserving the spherical symmetry in PCM is also found in Fig. 11(a) where the two curves are the two section cuts of density fields along  $y = 0.5$  (black) and  $y = x$  (cyan), respectively. We see that the two

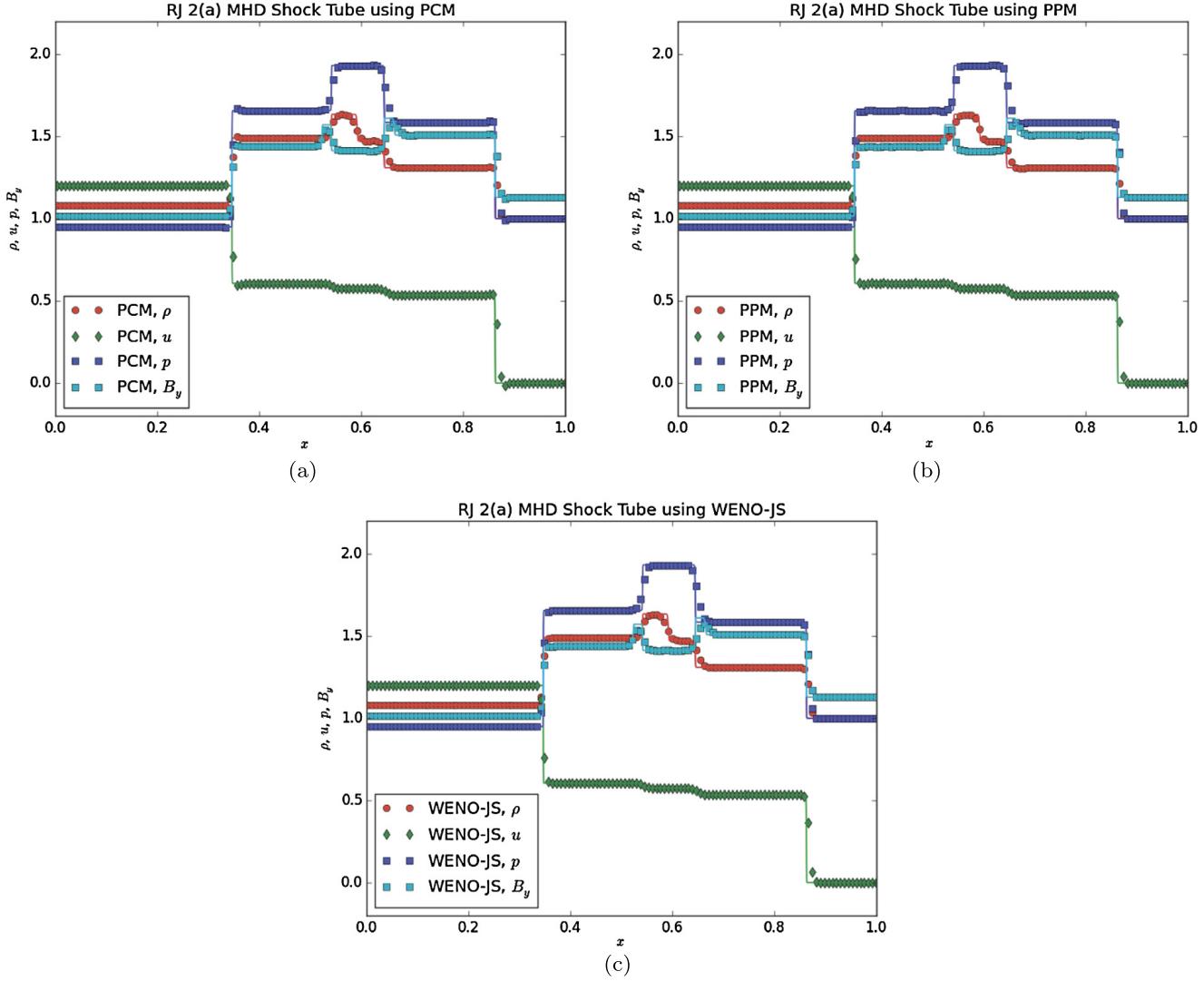


**Fig. 6.** The Brio–Wu MHD shock tube problem. The Roe Riemann solver is consistently used for all methods. The test problems are computed on  $N_x = 128$ , while the high-resolution reference solution is obtained using PLM on  $N_x = 2048$  with the MC slope limiter. The same slope limiter was used in PPM too. The Courant number is fixed as 0.8 in all runs.

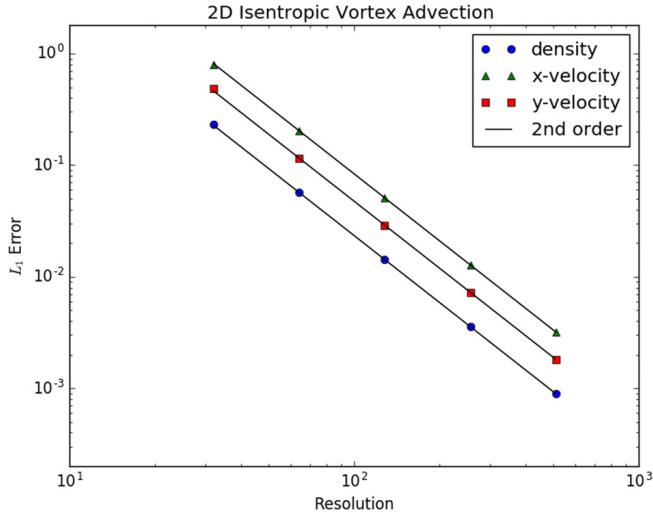
peak values of each section cut are matching each other very closely in terms of both their locations and their magnitudes. In the other schemes there are clearly much larger disagreements in the magnitude of the peak density values. Balsara and his collaborators [73,74] also have shown that the use of multidimensional Riemann solvers provide an excellent capability of preserving isentropic propagation of the blast wave as well.

(b) 2D Riemann problems. Next we test PCM for a family of well-known benchmarked Riemann problems whose mathematical classification was originally put forward by Zhang and Zheng [75], in which the original 16 of admissible configurations were conjectured on polytropic gas. This claim was corrected by Schultz-Rinne [76] that one of them was impossible, and the numerical testings for such 15 configurations were studied in [77]. Later, Lax and Liu showed that there are total of 19 genuinely different configurations available, providing numerical solutions of all 19 cases too [78]. See also [79]. Until today, this family of 2D Riemann problems has been chosen by many people to demonstrate that their numerical algorithms can predict these 19 configurations successfully in pursuance of code verification purposes [12,80–82]. We follow the setup as described in [81] in the following two verification tests, Configuration 3 and Configuration 5. In both cases the calculations show numerical solutions on  $[0, 1] \times [0, 1]$  using outflow boundary conditions.

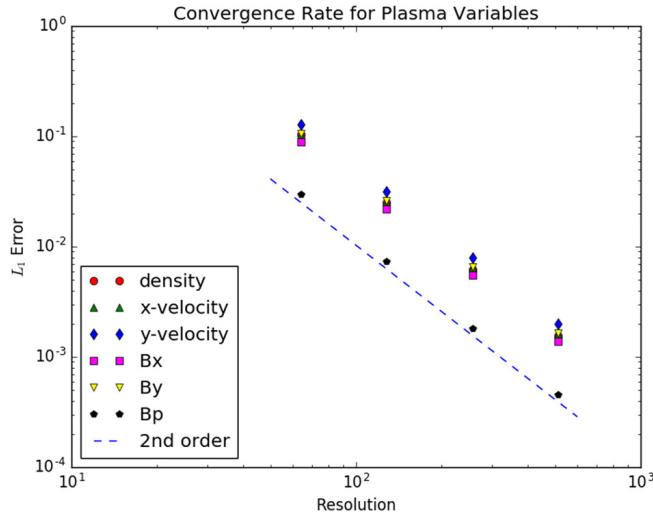
(b) – Configuration 3. Panels in Fig. 12 show numerical solutions of density at  $t = 0.8$  resolved on  $400 \times 400$  using the HLLC Riemann solvers with  $C_{\text{cfl}} = 0.8$ . Also shown are the 40 contour lines of  $\rho$ . The range of  $\rho$  is fixed as  $0.1 \leq \rho \leq 1.8$  in both the pseudo-color figures and the contour lines. For the PCM method we employed both approaches of WENO-JS (see Eq. (42)) and WENO-Z (see Eq. (44)) for the calculations of the smoothness stencils. The figures can be directly compared with Fig. 6 in [81] where they used the same grid resolution for their hybrid compact-WENO scheme. First of all, including the two PCM solutions, we see that all calculations have produced their solutions successfully, in particular, without



**Fig. 7.** The RJ2a MHD shock tube problem. All runs used the HLLD Riemann solver, computed on  $N_x = 128$  with  $C_{\text{cfl}} = 0.8$ . The reference solution was obtained using PLM on  $N_x = 2048$ . The PLM and PPM methods used the MC slope limiter.



**Fig. 8.** Convergence test of the 2D isentropic hydrodynamical vortex advection problem. The errors in  $\rho$ ,  $u$ , and  $v$  are calculated in  $L_1$  sense against the initial conditions. The tested PCM solutions are solved on  $N_x \times N_y$ , where  $N_x = N_y = 32, 64, 128, 256$  and  $512$ . All runs reached to  $t = 10$  using the HLLC Riemann solver with  $C_{\text{cfl}} = 0.8$ .



**Fig. 9.** Convergence test of the 2D isentropic MHD vortex advection problem. The errors in  $\rho$ ,  $u$ ,  $v$ ,  $B_x$ ,  $B_y$  and the magnetic pressure  $B_p$  are calculated in  $L_1$  sense against the initial conditions. The tested PCM solutions are solved on  $N_x \times N_y$ , where  $N_x = N_y = 32, 64, 128, 256$  and  $512$ . All runs reached to  $t = 10$  using the HLLC Riemann solver with  $C_{\text{cfl}} = 0.8$ .

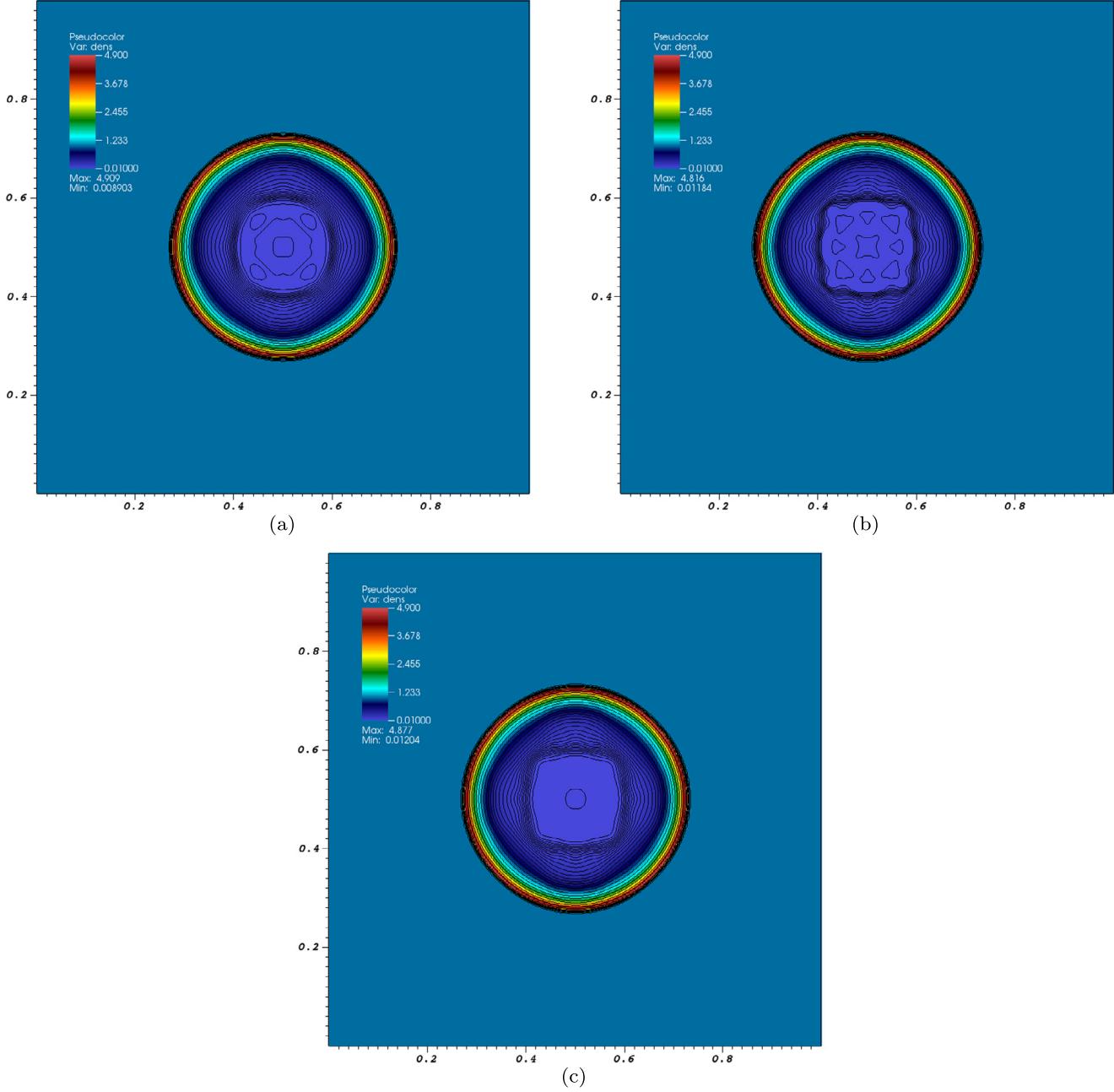
suffering unphysical oscillations near shocks and contact discontinuities. This test confirms that the PCM results are well comparable to the other solutions, except for some expected minor discrepancies.

One thing to notice is that the PPM solution in Fig. 12(b) has interestingly much more formations of Kelvin-Helmholtz instabilities, identified as vortical rollups along the slip lines (shown as the interface boundaries between the green triangular regions and the sky blue areas surrounding the mushroom-shaped jet). This feature is also often found in a test known as “Double Mach reflection” [56], where the similar pattern of rollups are detected along the slip line. As found in various studies [13,30,37] it is conventional to say that the amount of such vortical rollups at slip lines is one of the key factors to measure inherent numerical dissipations in a code. If we follow this approach, it then leads us to say that the PPM method is the least dissipative method among the six methods we tested. However, we find that this conclusion is somewhat arguable considering the nominal order of accuracy of PPM is lower than those of WENO-JS + CharTr, WENO-Z + CharTr and PCM. We think that there is to be more accurate assessment regarding this type of conclusion. Readers can find a very similar numerical comparison between PPM and WENO-JS in [37].

(b) – Configuration 5. As a second 2D Riemann problem we consider Configuration 5 to test PCM and compare its solution with three other solutions of PLM, PPM and WENO-JS + CharTr. The WENO-JS approaches in Eqs. (26) and (42) are adopted for PCM for the calculations of the smoothness stencils. The choice of grid resolution is  $1024 \times 1024$  in this test in order to directly compare our results with the results reported in Fig. 4 and Fig. 5 of [12]. As obtained in Fig. 13, the PCM solution satisfactorily compares well with the solutions of PPM and WENO-JS + CharTr, as well as with the high-resolution results in [12]. As also reported in [12,13], when considering discontinuous flows in multiple space dimensions, the dimension-by-dimension approach works just as fine in terms of producing comparably accurate solutions. Likewise, the results in Fig. 13 show that we observe the same qualitative performances in all the methods we tested here, including PCM.

Although all the solutions are comparably admissible, there are few distinctive features in the PCM solution, displayed in Fig. 13(d). We note that the minimum and maximum values of the computed density are respectively the smallest and the largest among the four results. This trend is consistent with the increasing order of accuracy in the four panels here. The same observation can be found also in Fig. 12 too. We consider this as an indication that the small scale features are better resolved in PCM with less amount of numerical diffusion.

(c) *MHD rotor*. Next, we consider the MHD rotor problem [83,84]. As the problem has been discussed by various people we rather focus on discussing the solution of PCM here. We use the same setup conditions as described in [43]. Exhibited in Fig. 14(e) and Fig. 14(f) are respectively the density and the 30 contour lines of the Mach number on  $400 \times 400$  cells, both at  $t = 0.15$ . With minor discrepancies, we see that the PCM solution successfully demonstrates its ability to solve MHD flows in multiple space dimensions. To test PCM for multidimensional MHD flows, we integrated the PCM algorithm in the MHD scheme [43,44] of the FLASH code [41,42]. Of noteworthy point is that the contour lines of the Mach number in Fig. 14(f) remain concentric in the central region without any distortion from the near-perfect symmetry. In all runs the HLLD Riemann solver [68] was used with a fixed value  $C_{\text{cfl}} = 0.8$ . The PPM run used the MC slope limiter for monotonicity.



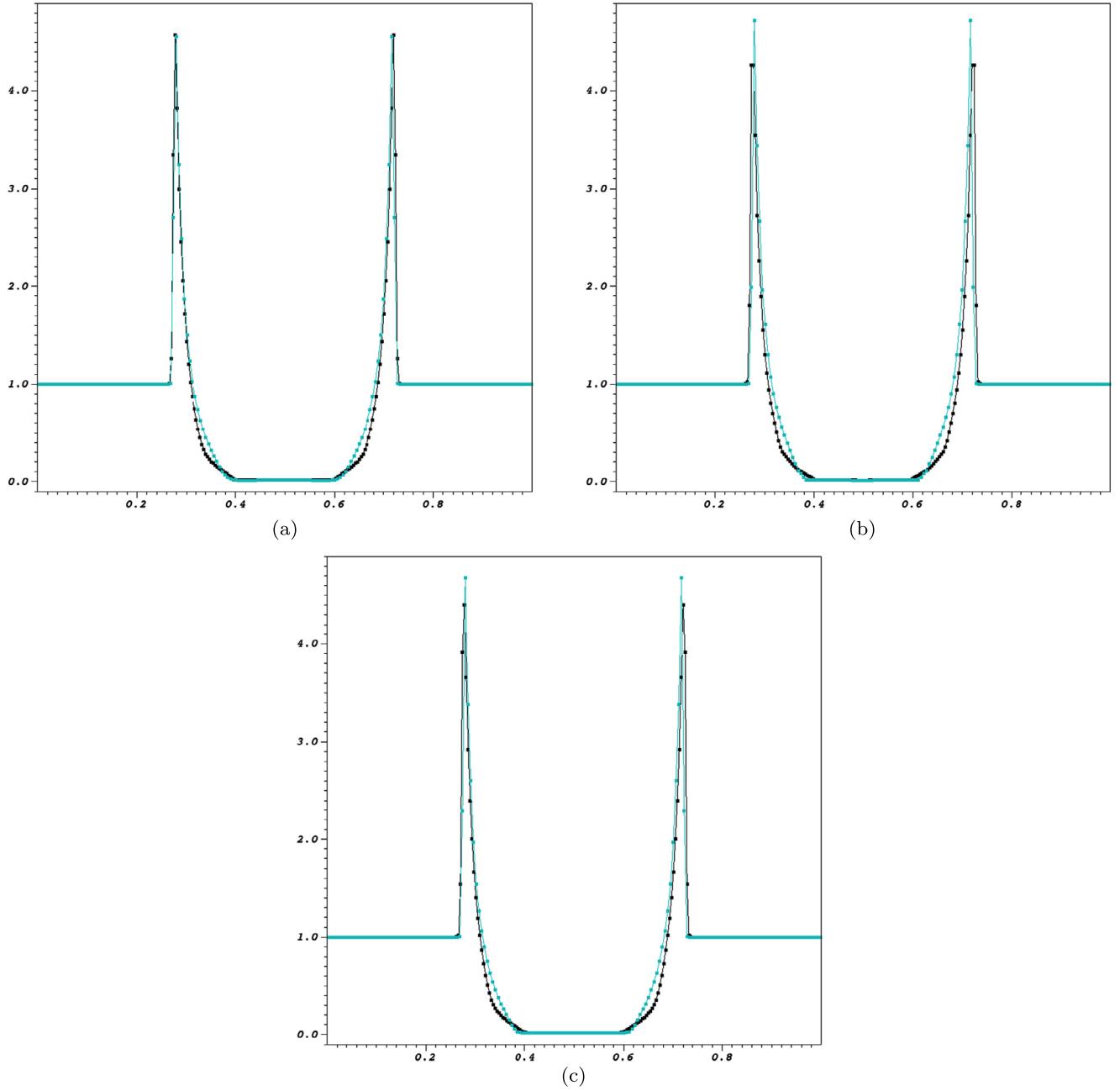
**Fig. 10.** The Sedov explosion test. (a) PCM, (b) PPM, and (c) WENO-JS + charTr. All calculations are done on  $256 \times 256$  cells with the HLLC Riemann solver and with  $C_{\text{cfl}} = 0.8$ . PPM is computed using the MC slope limiter. 30 equally spaced levels of density contour lines are shown in logarithmic scale. (For interpretation of the colors in this figure, the reader is referred to the web version of this article.)

### 6.3. 3D tests

Lastly, for 3D cases, we have selected three test problems in MHD in such a way that we can fully quantify the performance of PCM both for assessing its convergence rate in 3D and for verifying its code capability in discontinuous flows.

#### 6.3.1. 3D convergence test

**Alfvén Wave convergence test – UG.** We solve the circularly polarized Alfvén Wave propagation problem [44,49] as our first 3D test problem to quantify the PCM's order of accuracy in full 3D. The computational domain is resolved on  $2N_x \times N_y \times N_z$  grid cells, where we choose  $N_x = N_y = N_z = 8, 16, 32$  and 64 for the grid convergence study. As in [44] we ran the same two configurations of the wave mode that are the standing wave mode and the traveling wave mode until  $t = 1$ . In both we choose the Roe Riemann solver with  $C_{\text{cfl}} = 0.95$ .



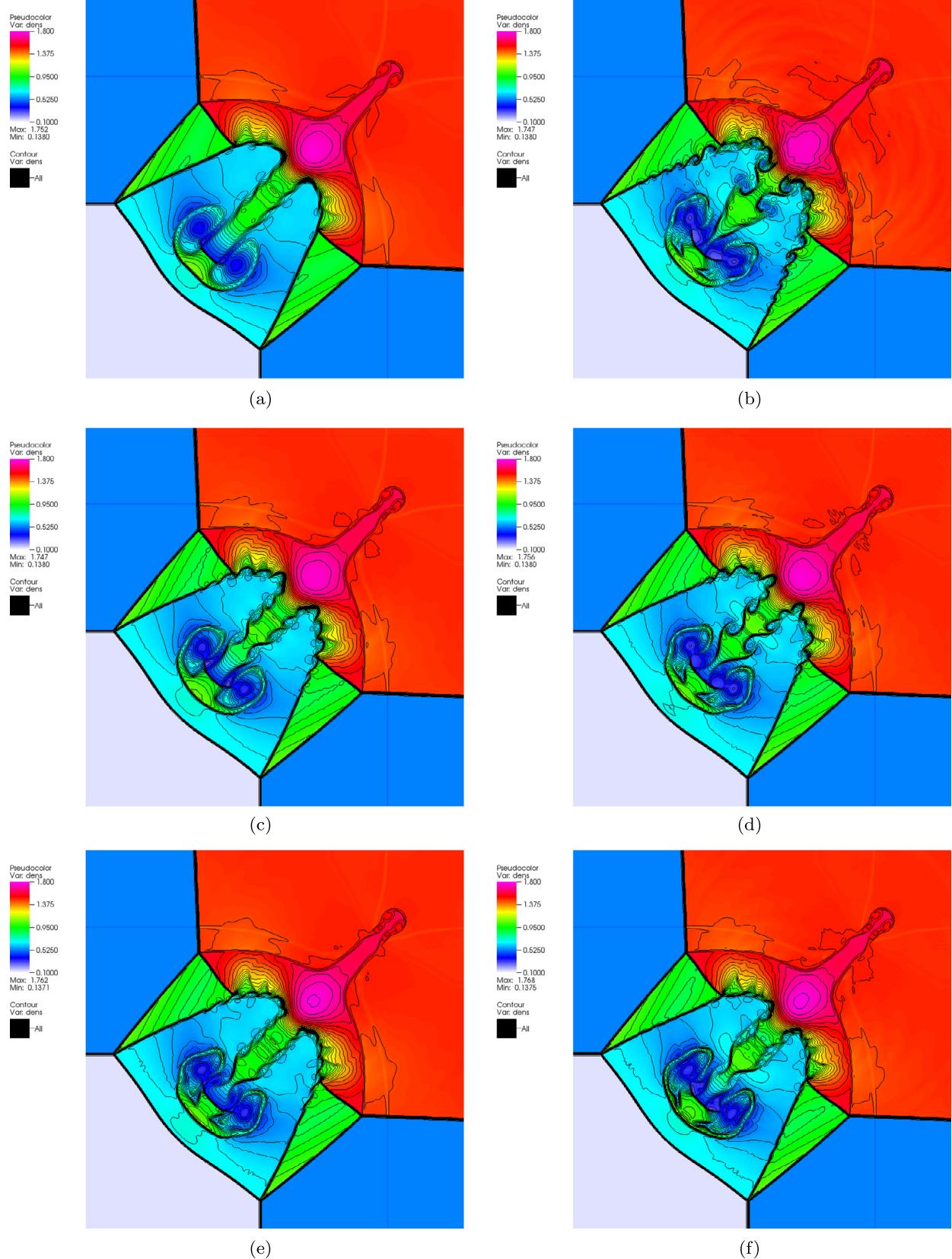
**Fig. 11.** Two section cuts of  $\rho$  from the Sedov test in Fig. 10. The two lines represent section cuts of densities along  $y = 0.5$  (black curves) and  $y = x$  (cyan curves). (a) PCM, (b) PPM, and (c) WENO-JS + charTr. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Respectively, Fig. 15(a) and Fig. 15(b) are the  $L_1$  numerical errors on a logarithmic scale for the standing wave case and the traveling wave case. We observe that the rate of PCM convergence in 3D is second-order as expected, which agrees with the results reported in [44]. One difference is noted in the standing wave case in Fig. 15(a) that PCM's  $L_1$  error in each grid resolution is much lower than those obtained with PPM + HLLD + F-CTU with  $C_{\text{cfl}} = 0.95$  in [44]. However, the magnitudes of the PCM error in the traveling wave case in Fig. 15(b) look pretty much similar to the equivalent run in [44].

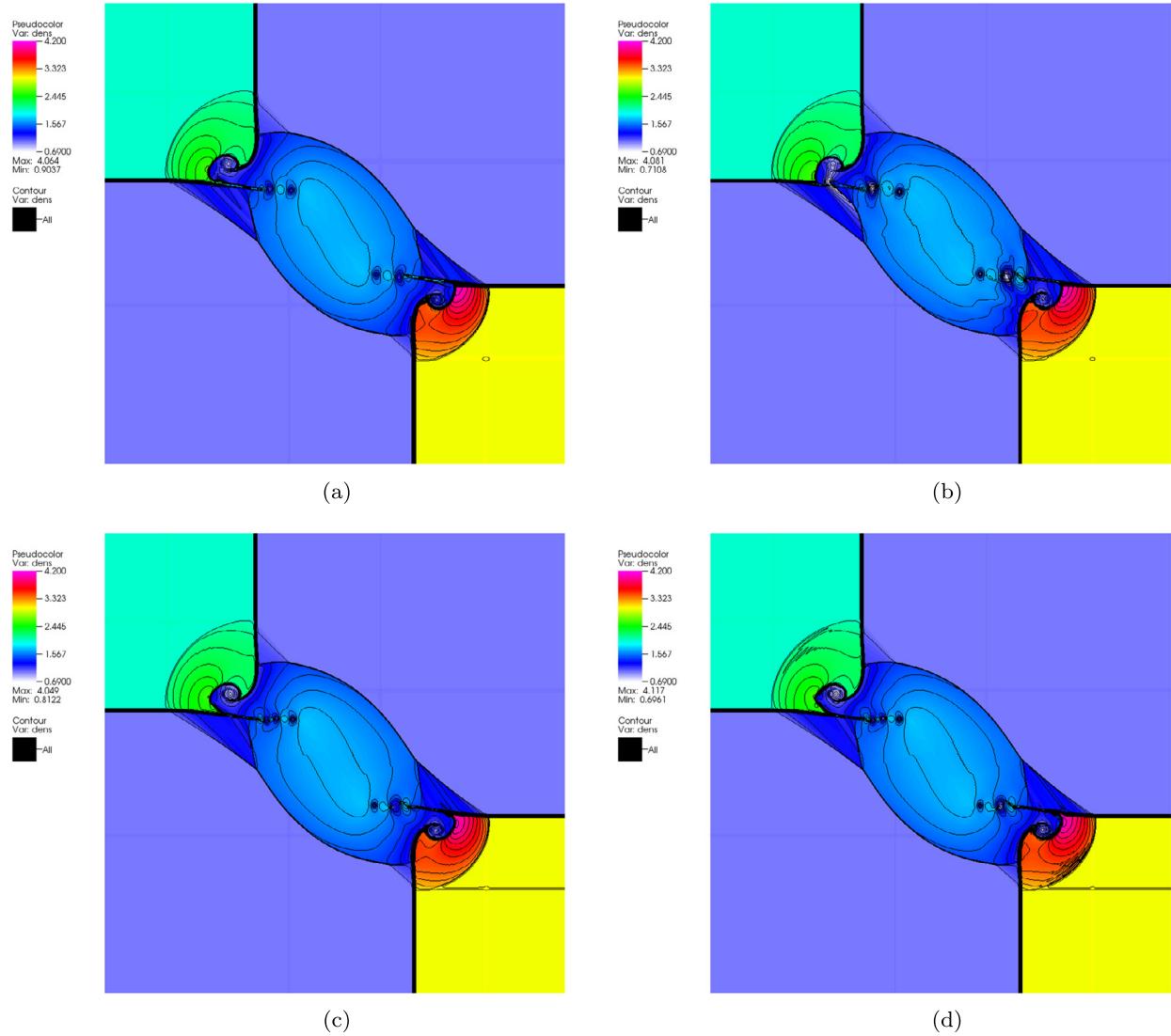
### 6.3.2. 3D discontinuous tests

(a) **3D MHD blast – UG.** We consider the 3D variant of the MHD blast problem by adopting the setup conditions in [44] to demonstrate the three-dimensional propagation of strong MHD shocks using the PCM algorithm. The original 2D version of the spherical blast wave problem was studied in [85], and later various people adopted the similar setup conditions [40,43, 44,83,86–92] for their code verifications in strongly magnetized shock flows.

We display four different fluid variables in each panel in Fig. 16. From the top right quadrant to the bottom right quadrant in counter clockwise direction, we show the gas pressure  $p$ , the density  $\rho$ , the total velocity  $U = \sqrt{u^2 + v^2 + w^2}$ ,



**Fig. 12.** 2D Riemann problem – Configuration 3. (a) PLM, (b) PPM, (c) WENO-JS + CharTr, (d) WENO-Z + CharTr, (e) PCM-JS, and (f) PCM-Z. Each panel shows the density values at  $t = 0.8$  between  $[0.1, 1.8]$  in linear scale, calculated using  $400 \times 400$  grid cells. The total of 40 contour lines are over-plotted. The MC slope limiter is used in (a) and (b). (For interpretation of the colors in this figure, the reader is referred to the web version of this article.)

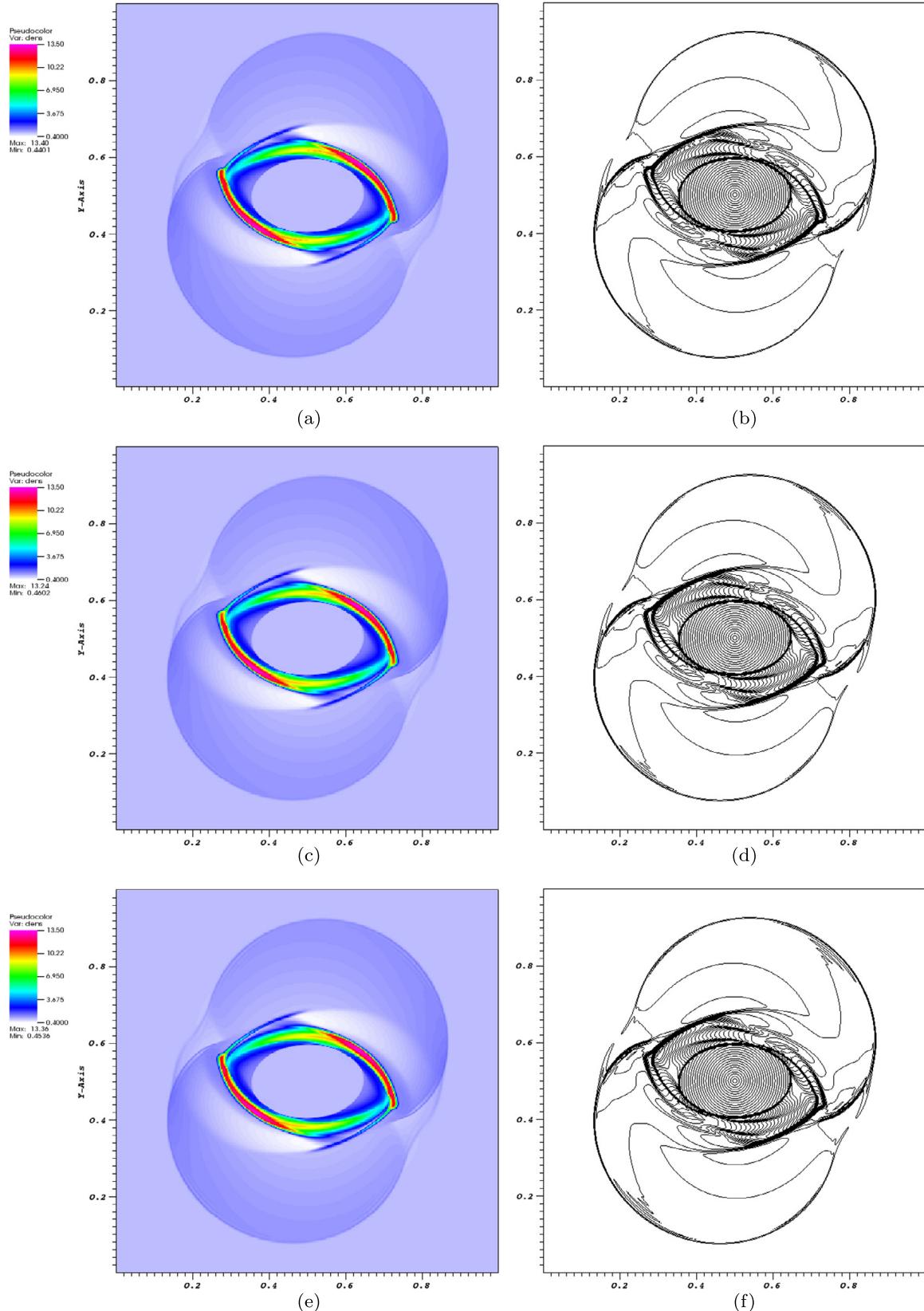


**Fig. 13.** 2D Riemann problem – Configuration 5. (a) PLM, (b) PPM, (c) WENO-JS + CharTr, and (d) PCM-JS. Each panel shows the density values at  $t = 0.23$  between  $[0.69, 4.2]$  in linear scale, resolved on  $1024 \times 1024$  grid cells. The total of 40 contour lines are over-plotted. The MC slope limiter is used in (a) and (b). All runs used the Roe Riemann solver with  $C_{\text{cfl}} = 0.8$ .

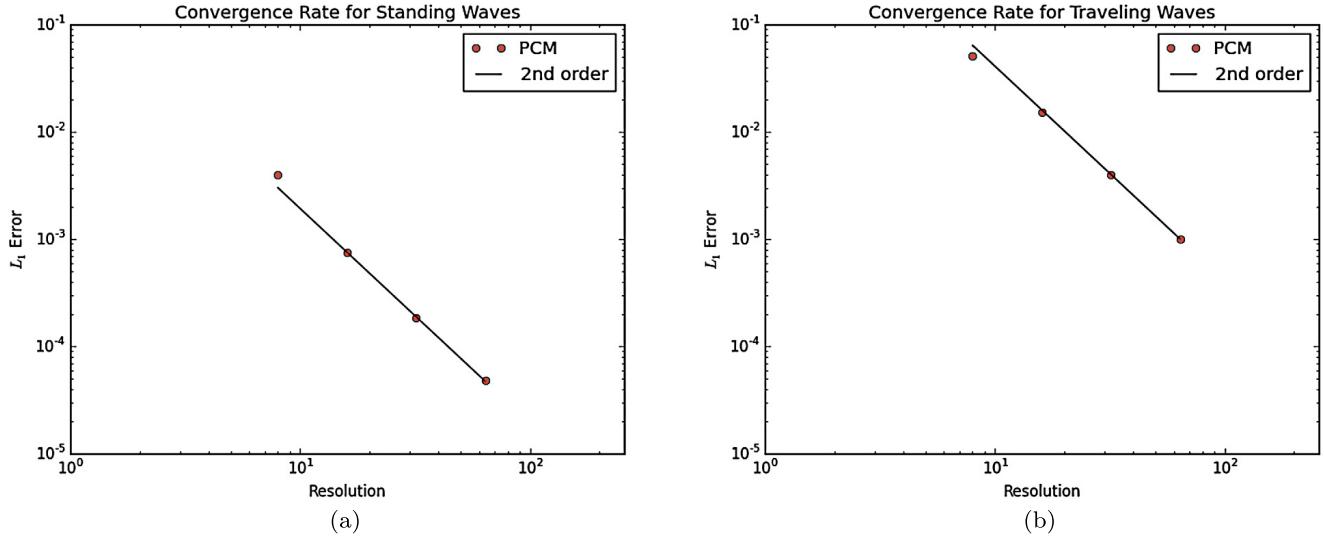
and the magnetic pressure  $B_p$ , all plotted at  $t = 0.01$ . The grid resolution  $128 \times 128 \times 128$  as well as all other parameters are chosen as same as in [44] in order to provide a direct comparison.

We tested PCM in three different plasma conditions defined by the three different strengths of  $B_x = 0$ ,  $\frac{50}{\sqrt{4\pi}}$ , and  $\frac{100}{\sqrt{4\pi}}$ , as displayed in Fig. 16(a)–Fig. 16(c). Of particular interest to note is with the initial low plasma  $\beta$  conditions in the last two cases,  $\beta = 1 \times 10^{-3}$  and  $2.513 \times 10^{-4}$ , respectively. On the other hand, the first setup in Fig. 16(a) produces the non-magnetized plasma flow, hence it allows us to test the PCM algorithm in the pure hydrodynamical limit in 3D. As clearly seen, all results have produced confidently accurate solutions. We also note that PCM has produced larger values of extrema in each variable than those reported in [44], without exhibiting any unphysical oscillations. This test demonstrate that the PCM algorithm is well-suited for simulating low- $\beta$  flows in full 3D.

**(b) Magnetic field loop advection – UG.** Since this problem was originally studied and reported in [93], the problem has become a popular benchmark case among various code developers to demonstrate their MHD algorithms' capabilities in advecting the initial field loop which is weakly magnetized with a very high plasma  $\beta = 2 \times 10^6$ . The problem is known to be challenging [49,93], however, many have demonstrated that their codes can successfully produced comparable results [21,43,44, 82,90,94,95]. In addition to the two original setups [49,93] where the initial field loops advect with the angle diagonal to the domain, Lee [44] recently reported that a small-angle advection is much more challenging. As an example, Lee adopted the advection angle  $\theta \approx 0.573^\circ$  relative to the  $x$ -axis for the small-angle advection case in both 2D and 3D. The study found that a proper amount of multidimensional numerical dissipation plays a key role in maintaining the clean small-angle advection, and designed the algorithm called *upwind-MEC*. Here we repeat all three configurations (two large-angle advection



**Fig. 14.** 2D MHD Rotor problem. The panels on the left column show the density at  $t = 0.15$ , and the panels on the right show the 30 contour lines of the Mach number at the same time. (a)  $\rho$  using PPM, (b) Mach number using PPM, (c)  $\rho$  using WENO-JS + CharTr, (d) Mach number using WENO-JS + CharTr, (e)  $\rho$  using PCM, and (f) Mach number using PCM.



**Fig. 15.** The circularly polarized Alfvén wave convergence rate for both the standing and traveling wave problems using PCM combined with the Roe Riemann solver.

cases and one small-angle advection case) by following the same setups in [44]. All the results in Fig. 17 were obtained using PCM and the Roe Riemann solver with  $C_{\text{cfl}} = 0.8$  on  $64 \times 64 \times 128$  cells.

First, Fig. 17(a) shows the small-angle advection with  $\theta \approx 0.573^\circ$  relative to the  $x$ -axis with the velocity fields given by  $\mathbf{U} = (\cos \theta, \sin \theta, 2)^T$ . Compared to this, in Fig. 17(b), we use  $\mathbf{U} = (1, 1, 2)^T$  which yields the large-angle advection. In both cases the tilt angle  $\omega$  (see [44] for details) is set to be same as  $\theta$ . As manifested, both runs cleanly preserve the initial geometry of the field loop, convincing us that the PCM algorithm is robust and accurate in this challenging problem. As a final test we also perform the standard field loop advection setup of Gardiner and Stone [49]. The result is shown in Fig. 17(c). We see clearly that the PCM algorithm has produced well-behaving, accurate and confident solutions in this test. The results in Fig. 17 can be directly compared to the results reported in [44].

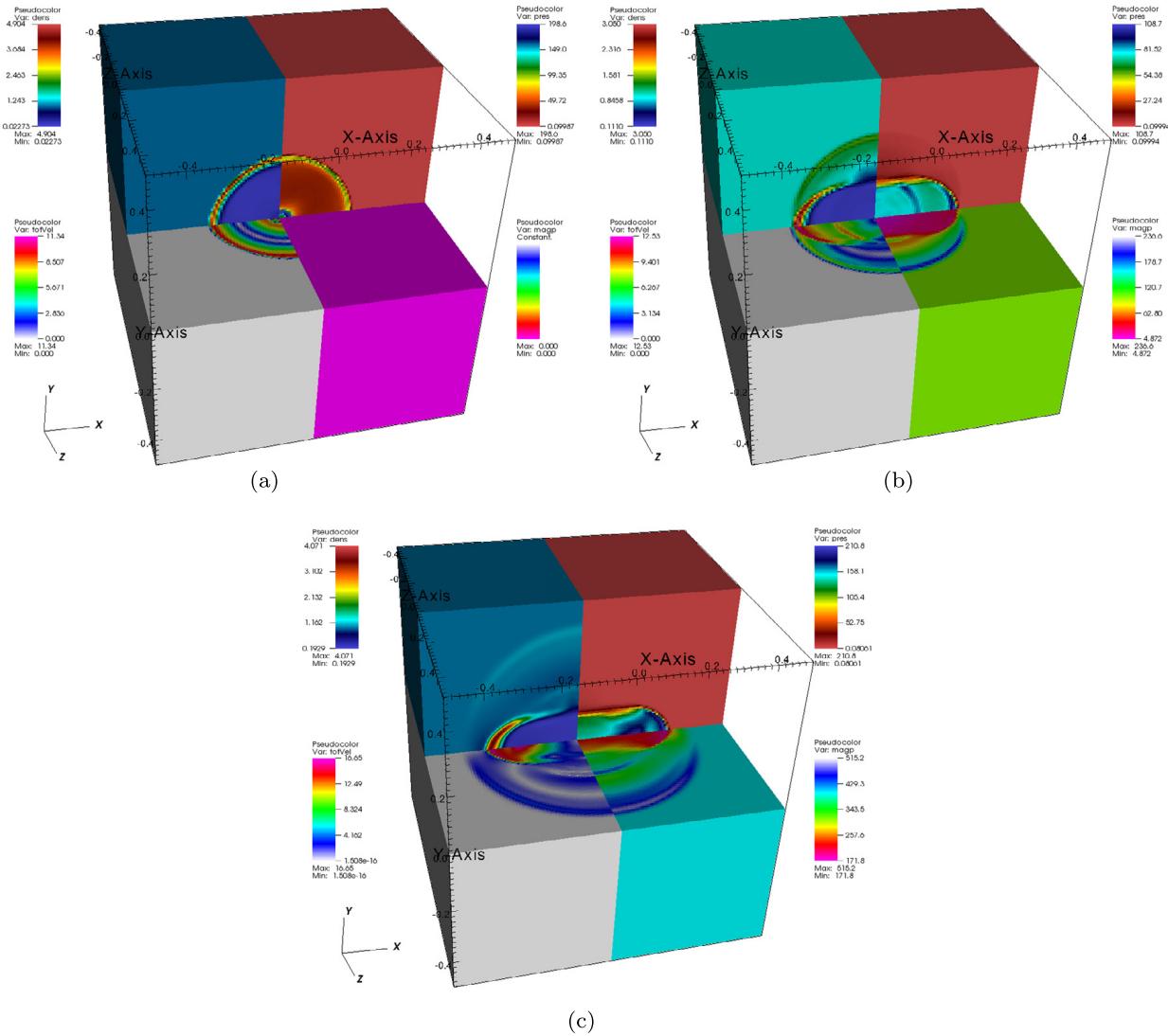
## 7. Conclusions

We summarize key features of the PCM algorithm studied in this paper.

- We have presented a new high-order finite volume scheme for the solutions of the compressible gas dynamics and ideal MHD equations in 1D. This baseline 1D algorithm uses piecewise cubic polynomials for spatial reconstruction by adopting the non-oscillatory approximations of the fifth order WENO schemes to determine the unique piecewise cubic polynomial on each cell. To provide the nominal fifth-order accuracy in space, we have developed a new non-oscillatory WENO-type reconstruction for  $q'_{C,i}$  approximation. The new approach makes use of the two parabolic polynomials, termed as *PPM-Build*, to achieve fourth-order accuracy in establishing  $q'_{C,i}$  approximation.
- We have formulated a new fourth-order temporal updating scheme, all integrated in PCM by design, based on the simple predictor-corrector type characteristic tracing approach. The overall solution accuracy of the baseline 1D PCM scheme, combining both spatially and temporally, seemingly converges with fifth-order. We show the PCM scheme compares greatly with the spatially fifth-order WENO integrated with RK4, demonstrating even smaller  $L_1$  errors in PCM.
- A comparison of the computational expenses of PCM, PPM and WENO-JS + RK4 in 1D reveals that PCM has a superior advantage over the fifth-order counterpart WENO-JS + RK4 by a factor of 1.71.
- We have integrated the baseline 1D PCM algorithm for multidimensional cases by adopting the simple dimension-by-dimension approach. As anticipated, this approach yields at most second-order accurate solutions in multidimensional simulations of smooth flows. In the presence of flow discontinuities and shocks, however, the results obtained with the present simple multidimensional PCM extension shows a great level of confidence in predicting numerical solutions of hydrodynamics and MHD. An approach to extend the fifth-order property of the baseline 1D PCM to multiple spatial dimensions will be further investigated in our future work.

## Acknowledgements

The software used in this work was in part developed by the DOE NNSA-ASC OASCR Flash Center at the University of Chicago. D. Lee also gratefully acknowledges the FLASH group for supporting the current work. The authors also thank D.S. Balsara and an anonymous referee for very helpful suggestions and comments on the manuscript.



**Fig. 16.** The 3D MHD blast problem. All results used the PCM scheme with the hybrid Riemann solver using  $C_{\text{eff}} = 0.8$ . (a)  $B_x = 0$ , (b)  $B_x = \frac{50}{\sqrt{4\pi}}$ , and (c)  $B_x = \frac{100}{\sqrt{4\pi}}$ . In each panel, we show four different plasma quantities, gas pressure in the top right quadrant, density in top left, total velocity in bottom left, and magnetic pressure in bottom right. Each corresponding color bar is shown immediately next to the corresponding quantity. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

#### Appendix A. Reconstruction of the Riemann states $q_{L,i}$ and $q_{R,i}$ at fifth order

We use the standard fifth order WENO reconstruction procedure to obtain the Riemann state values at cell interfaces. On each cell  $I_i = [x_{i-1/2}, x_{i+1/2}]$ , they can be explicitly written as

$$q_{L,R,i} = \sum_{\ell=1}^3 \omega_{\ell}^{\pm} p_{\ell}(x_{i\pm 1/2}), \quad (\text{A.1})$$

where

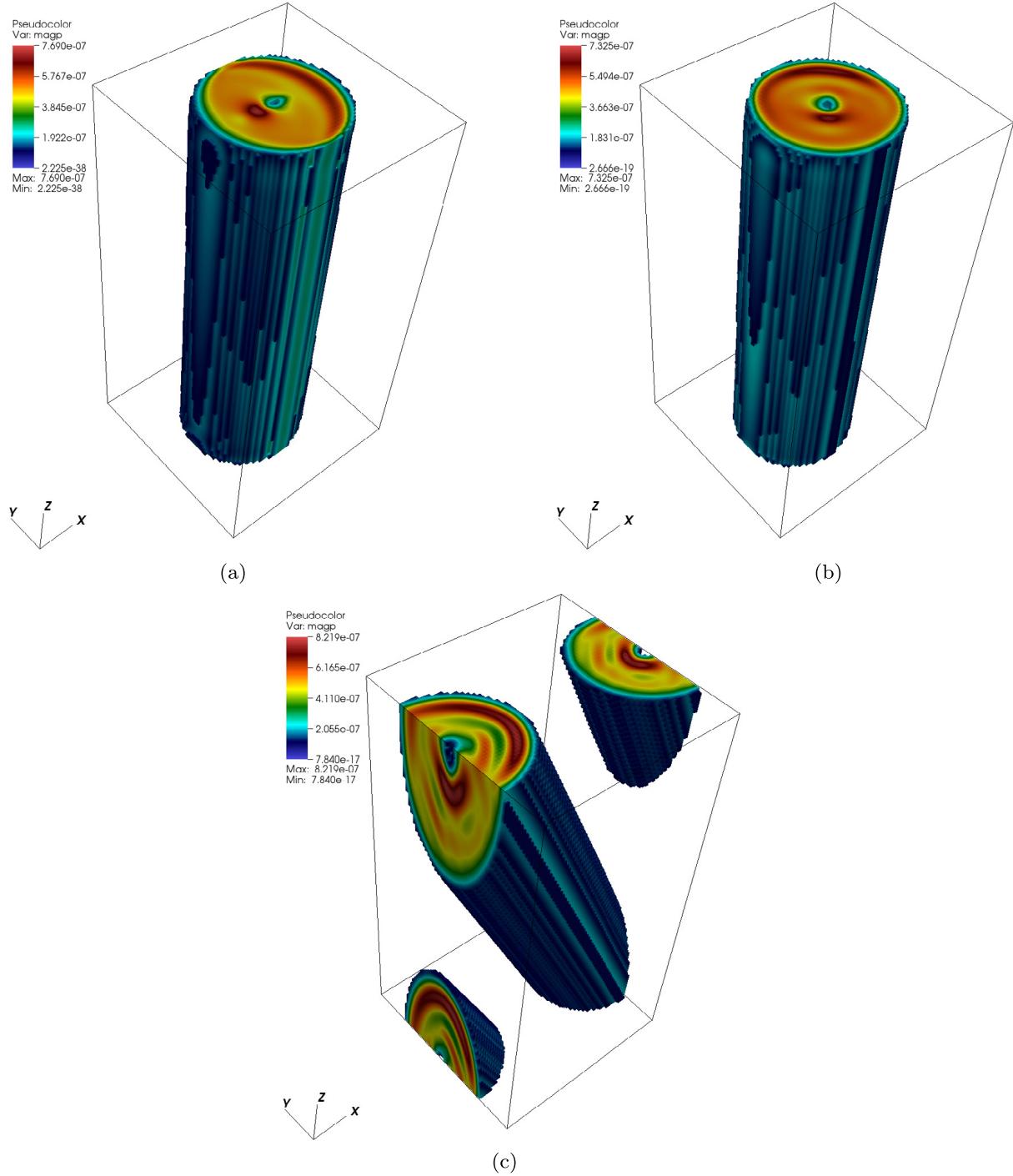
$$p_1(x_{i-1/2}) = -\frac{1}{6}\bar{q}_{i-2} + \frac{5}{6}\bar{q}_{i-1} + \frac{1}{3}\bar{q}_i, \quad (\text{A.2})$$

$$p_2(x_{i-1/2}) = \frac{1}{3}\bar{q}_{i-1} + \frac{5}{6}\bar{q}_i - \frac{1}{6}\bar{q}_{i+1}, \quad (\text{A.3})$$

$$p_3(x_{i-1/2}) = \frac{11}{6}\bar{q}_i - \frac{7}{6}\bar{q}_{i+1} + \frac{1}{3}\bar{q}_{i+2}, \quad (\text{A.4})$$

and

$$p_1(x_{i+1/2}) = \frac{1}{3}\bar{q}_{i-2} - \frac{7}{6}\bar{q}_{i-1} + \frac{11}{6}\bar{q}_i, \quad (\text{A.5})$$



**Fig. 17.** The 3D field loop advection using PCM. (a) The small angle advection with  $\theta \approx 0.537^\circ$  at  $t = 2$ , (b) the large angle advection at  $t = 2$  using  $\mathbf{U} = (1, 1, 2)^T$ , and (c) the standard Gardiner–Stone advection at  $t = 1$ . All runs were calculated on  $64 \times 64 \times 128$  cells using the upwind-MEC.

$$p_2(x_{i+1/2}) = -\frac{1}{6}\bar{q}_{i-1} + \frac{5}{6}\bar{q}_i + \frac{1}{3}\bar{q}_{i+1}, \quad (\text{A.6})$$

$$p_3(x_{i+1/2}) = \frac{1}{3}\bar{q}_i + \frac{5}{6}\bar{q}_{i+1} - \frac{1}{6}\bar{q}_{i+2}. \quad (\text{A.7})$$

The nonlinear weights  $\omega_\ell^\pm$  are constructed based on one of the WENO-JS, WENO-M, or WENO-Z schemes in Eqs. (26)–(28). The smoothness indicators in Eqs. (26)–(28) are given by following the standard fifth order WENO reconstruction [11]:

$$\beta_1 = \frac{13}{12}(\bar{q}_{i-2} - 2\bar{q}_{i-1} + \bar{q}_i)^2 + \frac{1}{4}(\bar{q}_{i-2} - 4\bar{q}_{i-1} + 3\bar{q}_i)^2, \quad (\text{A.8})$$

$$\beta_2 = \frac{13}{12} (\bar{q}_{i-1} - 2\bar{q}_i + \bar{q}_{i+1})^2 + \frac{1}{4} (\bar{q}_{i-1} - \bar{q}_{i+1})^2, \quad (\text{A.9})$$

$$\beta_3 = \frac{13}{12} (\bar{q}_i - 2\bar{q}_{i+1} + \bar{q}_{i+2})^2 + \frac{1}{4} (3\bar{q}_i - 4\bar{q}_{i+1} + \bar{q}_{i+2})^2. \quad (\text{A.10})$$

## Appendix B. Reconstruction of the derivative $q'_{C,i}$ at fourth order

We provide explicit forms to determine  $q'_{C,i}$  at fourth order on the five-point stencil  $S = \{I_{i-2}, \dots, I_{i+2}\}$ . The stencil  $S$  is subdivided into two four-point sub-stencil  $S_1 = \{I_{i-2}, \dots, I_{i+1}\}$  and  $S_2 = \{I_{i-1}, \dots, I_{i+2}\}$  so that  $S = \cup_{\ell=1}^2 S_\ell$ . This strategy is based on the  $r = 4$  WENO description of Balsara and Shu [3], and is slightly modified to obtain the slope  $q'_{C,i}$  in our case. Here we provide detailed expressions and derivations in Step 1 and Step 2 in Section 2.3.

*Step 1.* The coefficients of  $\phi_\pm$  in Eq. (35) are given as

$$a_0^\pm = \frac{1}{12} \left( -\bar{q}_{i-2+s} + 7\bar{q}_{i-1+s} + 7\bar{q}_{i+s} - \bar{q}_{i+1+s} \right), \quad (\text{B.1})$$

$$a_1^\pm = \frac{1}{12\Delta x} \left( \bar{q}_{i-2+s} - 15\bar{q}_{i-1+s} + 15\bar{q}_{i+s} - \bar{q}_{i+1+s} \right), \quad (\text{B.2})$$

$$a_2^\pm = \frac{1}{4\Delta x^2} \left( \bar{q}_{i-2+s} - \bar{q}_{i-1+s} - \bar{q}_{i+s} + \bar{q}_{i+1+s} \right), \quad (\text{B.3})$$

$$a_3^\pm = \frac{1}{6\Delta x^3} \left( -\bar{q}_{i-2+s} + 3\bar{q}_{i-1+s} - 3\bar{q}_{i+s} + \bar{q}_{i+1+s} \right), \quad (\text{B.4})$$

where  $s = 1$  for  $a_k^+$ , while  $s = 0$  for  $a_k^-$ .

*Step 2.* To determine the two linear weights  $\gamma_\pm$  in Eq. (36), we first define the fourth degree polynomial  $\phi(x)$  over the five-point stencil  $S$ ,

$$\phi(x) = \sum_{k=0}^4 b_k (x - x_i)^k. \quad (\text{B.5})$$

Now  $\gamma_\pm$  are determined by solving

$$\gamma_- \phi'_-(x_i) + \gamma_+ \phi'_+(x_i) = \phi'(x_i). \quad (\text{B.6})$$

This is equivalent to solving

$$\gamma_- \left( a_1^- + a_2^- \Delta x + 3a_3^- \frac{\Delta x^2}{4} \right) + \gamma_+ \left( a_1^+ - a_2^+ \Delta x + 3a_3^+ \frac{\Delta x^2}{4} \right) = b_1 \quad (\text{B.7})$$

where  $b_1$  is given explicitly as

$$b_1 = \frac{5}{48} \bar{q}_{i-2} - \frac{17}{24} \bar{q}_{i-1} + \frac{17}{24} \bar{q}_{i+1} - \frac{5}{48} \bar{q}_{i+2}. \quad (\text{B.8})$$

This results in  $\gamma_- = \gamma_+ = \frac{1}{2}$ .

## References

- [1] P. Colella, P.R. Woodward, The piecewise parabolic method (PPM) for gas-dynamical simulations, *J. Comput. Phys.* 54 (1) (1984) 174–201.
- [2] G.-S. Jiang, C.-W. Shu, Efficient implementation of weighted ENO schemes, *J. Comput. Phys.* 126 (1) (1996) 202–228.
- [3] D.S. Balsara, C.-W. Shu, Monotonicity preserving weighted essentially non-oscillatory schemes with increasingly high order of accuracy, *J. Comput. Phys.* 160 (2) (2000) 405–452.
- [4] J.J. Dongarra, H.W. Meuer, H.D. Simon, E. Strohmaier, Recent trends in high performance computing, in: *The Birth of Numerical Analysis*, 2010, p. 93.
- [5] J. Dongarra, *On the Future of High Performance Computing: How to Think for Peta and Exascale Computing*, Hong Kong University of Science and Technology, 2012.
- [6] A. Subcommittee, Top Ten Exascale Research Challenges, US Department of Energy Report, 2014.
- [7] D.E. Keyes, L.C. McInnes, C. Woodward, W. Gropp, E. Myra, M. Pernice, J. Bell, J. Brown, A. Clo, J. Connors, et al., Multiphysics simulations challenges and opportunities, *Int. J. High Perform. Comput. Appl.* 27 (1) (2013) 4–83.
- [8] R.J. LeVeque, *Finite Volume Methods for Hyperbolic Problems*, vol. 31, Cambridge University Press, 2002.
- [9] R.J. LeVeque, *Finite Difference Methods for Ordinary and Partial Differential Equations: Steady-State and Time-Dependent Problems*, vol. 98, SIAM, 2007.
- [10] E. Toro, *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*, Springer, 2009, <http://books.google.com/books?id=SqEjX0um8oOC>.
- [11] C.-W. Shu, High order weighted essentially nonoscillatory schemes for convection dominated problems, *SIAM Rev.* 51 (1) (2009) 82–126.
- [12] P. Buchmüller, C. Helzel, Improved accuracy of high-order WENO finite volume methods on Cartesian grids, *J. Sci. Comput.* 61 (2) (2014) 343–368.

- [13] R. Zhang, M. Zhang, C.-W. Shu, On the order of accuracy and numerical performance of two classes of finite volume WENO schemes, *Commun. Comput. Phys.* 9 (3) (2011) 807–827.
- [14] P. McCrorquodale, P. Colella, A high-order finite-volume method for conservation laws on locally refined grids, *Commun. Appl. Math. Comput. Sci.* 6 (1) (2011) 1–25.
- [15] V.A. Titarev, E.F. Toro, ADER: arbitrary high order Godunov approach, *J. Sci. Comput.* 17 (1) (2002) 609–618.
- [16] E.F. Toro, V.A. Titarev, Derivative Riemann solvers for systems of conservation laws and ADER methods, *J. Comput. Phys.* 212 (1) (2006) 150–165.
- [17] V. Titarev, E. Toro, ADER schemes for three-dimensional non-linear hyperbolic systems, *J. Comput. Phys.* 204 (2) (2005) 715–736, <http://dx.doi.org/10.1016/j.jcp.2004.10.028>, <http://www.sciencedirect.com/science/article/pii/S0021999104004358>.
- [18] M. Dumbser, D.S. Balsara, E.F. Toro, C.-D. Munz, A unified framework for the construction of one-step finite volume and discontinuous Galerkin schemes on unstructured meshes, *J. Comput. Phys.* 227 (2008) 8209–8253, <http://dx.doi.org/10.1016/j.jcp.2008.05.025>.
- [19] D.S. Balsara, T. Rumpf, M. Dumbser, C.-D. Munz, Efficient, high accuracy ADER-WENO schemes for hydrodynamics and divergence-free magnetohydrodynamics, *J. Comput. Phys.* 228 (7) (2009) 2480–2516.
- [20] M. Dumbser, O. Zanotti, A. Hidalgo, D.S. Balsara, ADER-WENO finite volume schemes with space-time adaptive mesh refinement, *J. Comput. Phys.* 248 (2013) 257–286.
- [21] D.S. Balsara, C. Meyer, M. Dumbser, H. Du, Z. Xu, Efficient implementation of ADER schemes for Euler and magnetohydrodynamical flows on structured meshes—speed comparisons with Runge–Kutta methods, *J. Comput. Phys.* 235 (2013) 934–969.
- [22] J. Qian, J. Li, S. Wang, The generalized Riemann problems for compressible fluid flows: towards high order, *J. Comput. Phys.* 259 (2014) 358–389.
- [23] M. Dumbser, M. Käser, V.A. Titarev, E.F. Toro, Quadrature-free non-oscillatory finite volume schemes on unstructured meshes for nonlinear hyperbolic systems, *J. Comput. Phys.* 226 (1) (2007) 204–243.
- [24] C.-W. Shu, Total-variation-diminishing time discretizations, *SIAM J. Sci. Stat. Comput.* 9 (6) (1988) 1073–1084.
- [25] S.K. Godunov, A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics, *Mat. Sb.* 89 (3) (1959) 271–306.
- [26] J. Shi, C. Hu, C.-W. Shu, A technique of treating negative weights in WENO schemes, *J. Comput. Phys.* 175 (1) (2002) 108–127.
- [27] R. Borges, M. Carmona, B. Costa, W.S. Don, An improved weighted essentially non-oscillatory scheme for hyperbolic conservation laws, *J. Comput. Phys.* 227 (6) (2008) 3191–3211.
- [28] M. Castro, B. Costa, W.S. Don, High order weighted essentially non-oscillatory WENO-Z schemes for hyperbolic conservation laws, *J. Comput. Phys.* 230 (5) (2011) 1766–1792.
- [29] J. Qiu, C.-W. Shu, Hermite WENO schemes and their application as limiters for Runge–Kutta discontinuous Galerkin method: one-dimensional case, *J. Comput. Phys.* 193 (1) (2004) 115–135.
- [30] J. Qiu, C.-W. Shu, Hermite WENO schemes and their application as limiters for Runge–Kutta discontinuous Galerkin method II: two dimensional case, *Comput. Fluids* 34 (6) (2005) 642–663.
- [31] J. Zhu, J. Qiu, Hermite WENO schemes and their application as limiters for Runge–Kutta discontinuous Galerkin method, III: unstructured meshes, *J. Sci. Comput.* 39 (2) (2009) 293–321.
- [32] D.S. Balsara, C. Altmann, C.-D. Munz, M. Dumbser, A sub-cell based indicator for troubled zones in RKDG schemes and a novel class of hybrid RKDG+HWENO schemes, *J. Comput. Phys.* 226 (1) (2007) 586–620.
- [33] M. Brio, C.C. Wu, An upwind differencing scheme for the equations of ideal magnetohydrodynamics, *J. Comput. Phys.* 75 (2) (1988) 400–422.
- [34] A.K. Henrick, T.D. Aslam, J.M. Powers, Mapped weighted essentially non-oscillatory schemes: achieving optimal order near critical points, *J. Comput. Phys.* 207 (2) (2005) 542–567.
- [35] D.S. Balsara, S. Garain, C.-W. Shu, An efficient class of WENO schemes with adaptive order, *J. Comput. Phys.* 326 (2016) 780–804.
- [36] P. Colella, A direct Eulerian MUSCL scheme for gas dynamics, *SIAM J. Sci. Stat. Comput.* 6 (1) (1985) 104–117.
- [37] A. Mignone, G. Bodo, S. Massaglia, T. Matsakos, O. Tesileanu, C. Zanni, A. Ferrari, PLUTO: a numerical code for computational astrophysics, *Astrophys. J. Suppl. Ser.* 170 (1) (2007) 228.
- [38] A. Mignone, C. Zanni, P. Tzeferacos, B. van Straalen, P. Colella, G. Bodo, The PLUTO code for adaptive mesh computations in astrophysical fluid dynamics, *Astrophys. J. Suppl. Ser.* 198 (1) (2011) 7.
- [39] A. Mignone, P. Tzeferacos, A second-order unsplit Godunov scheme for cell-centered MHD: the CTU-GLM scheme, *J. Comput. Phys.* 229 (6) (2010) 2117–2138.
- [40] J.M. Stone, T.A. Gardiner, P. Teuben, J.F. Hawley, J.B. Simon, Athena: a new code for astrophysical MHD, *Astrophys. J. Suppl. Ser.* 178 (1) (2008) 137.
- [41] B. Fryxell, K. Olson, P. Ricker, F. Timmes, M. Zingale, D. Lamb, P. MacNeice, R. Rosner, J. Truran, H. Tufo, FLASH: an adaptive mesh hydrodynamics code for modeling astrophysical thermonuclear flashes, *Astrophys. J. Suppl. Ser.* 131 (1) (2000) 273.
- [42] A. Dubey, K. Antypas, M.K. Ganapathy, L.B. Reid, K. Riley, D. Sheeler, A. Siegel, K. Weide, Extensible component-based architecture for FLASH, a massively parallel, multiphysics simulation code, *Parallel Comput.* 35 (10) (2009) 512–522.
- [43] D. Lee, A.E. Deane, An unsplit staggered mesh scheme for multidimensional magnetohydrodynamics, *J. Comput. Phys.* 228 (4) (2009) 952–975.
- [44] D. Lee, A solution accurate, efficient and stable unsplit staggered mesh scheme for three dimensional magnetohydrodynamics, *J. Comput. Phys.* 243 (2013) 269–292.
- [45] G.L. Bryan, M.L. Norman, J.M. Stone, R. Cen, J.P. Ostriker, A piecewise parabolic method for cosmological hydrodynamics, *Comput. Phys. Commun.* 89 (1) (1995) 149–168.
- [46] G.L. Bryan, M.L. Norman, B.W. O’Shea, T. Abel, J.H. Wise, M.J. Turk, D.R. Reynolds, D.C. Collins, P. Wang, S.W. Skillman, et al., Enzo: an adaptive mesh refinement code for astrophysics, *Astrophys. J. Suppl. Ser.* 211 (2) (2014) 19.
- [47] R. Teyssier, Cosmological hydrodynamics with adaptive mesh refinement – a new high resolution code called RAMSES, *Astron. Astrophys.* 385 (1) (2002) 337–364.
- [48] P. Colella, Multidimensional upwind methods for hyperbolic conservation laws, *J. Comput. Phys.* 87 (1) (1990) 171–200.
- [49] T.A. Gardiner, J.M. Stone, An unsplit Godunov method for ideal MHD via constrained transport in three dimensions, *J. Comput. Phys.* 227 (8) (2008) 4123–4141.
- [50] J. Saltzman, An unsplit 3D upwind method for hyperbolic conservation laws, *J. Comput. Phys.* 115 (1) (1994) 153–168.
- [51] E.F. Toro, M. Spruce, W. Speares, Restoration of the contact surface in the HLL-Riemann solver, *Shock Waves* 4 (1) (1994) 25–34.
- [52] G.A. Sod, A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws, *J. Comput. Phys.* 27 (1) (1978) 1–31.
- [53] P.L. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, *J. Comput. Phys.* 43 (2) (1981) 357–372.
- [54] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, II, *J. Comput. Phys.* 83 (1) (1989) 32–78.
- [55] B. Einfeldt, C.-D. Munz, P.L. Roe, B. Sjögren, On Godunov-type methods near low densities, *J. Comput. Phys.* 92 (2) (1991) 273–295.
- [56] P. Woodward, P. Colella, The numerical simulation of two-dimensional fluid flow with strong shocks, *J. Comput. Phys.* 54 (1) (1984) 115–173.
- [57] M. Torrilhon, Uniqueness conditions for Riemann problems of ideal magnetohydrodynamics, *J. Plasma Phys.* 69 (03) (2003) 253–276.
- [58] M. Torrilhon, Non-uniform convergence of finite volume schemes for Riemann problems of ideal magnetohydrodynamics, *J. Comput. Phys.* 192 (1) (2003) 73–94.
- [59] D. Lee, An upwind slope limiter for PPM that preserves monotonicity in magnetohydrodynamics, in: 5th International Conference of Numerical Modeling of Space Plasma Flows (ASTRONUM 2010), vol. 444, 2011, p. 236.

- [60] S. Karni, S. Čanić, Computations of slowly moving shocks, *J. Comput. Phys.* 136 (1) (1997) 132–139.
- [61] Y. Stiriba, R. Donat, A numerical study of postshock oscillations in slowly moving shock waves, *Comput. Math. Appl.* 46 (5) (2003) 719–739.
- [62] M. Arora, P.L. Roe, On postshock oscillations due to shock capturing schemes in unsteady flows, *J. Comput. Phys.* 130 (1) (1997) 25–40.
- [63] S. Jin, J.-G. Liu, The effects of numerical viscosities: I. Slowly moving shocks, *J. Comput. Phys.* 126 (2) (1996) 373–389.
- [64] T.W. Roberts, The behavior of flux difference splitting schemes near slowly moving shock waves, *J. Comput. Phys.* 90 (1) (1990) 141–160.
- [65] E. Johnsen, S. Lele, Numerical Errors Generated in Simulations of Slowly Moving Shocks, Center for Turbulence Research, Annual Research Briefs, 2008, pp. 1–12.
- [66] A. Harten, P.D. Lax, B. Van Leer, On upstream differencing and Godunov-type schemes for hyperbolic conservation laws, in: Upwind and High-Resolution Schemes, Springer, 1997, pp. 53–79.
- [67] S. Li, An HLLC Riemann solver for magneto-hydrodynamics, *J. Comput. Phys.* 203 (1) (2005) 344–357.
- [68] T. Miyoshi, K. Kusano, A multi-state HLL approximate Riemann solver for ideal magnetohydrodynamics, *J. Comput. Phys.* 208 (1) (2005) 315–344.
- [69] D. Ryu, T. Jones, Numerical magnetohydrodynamics in astrophysics: algorithm and tests for one-dimensional flow, arXiv preprint, arXiv:astro-ph/9404074.
- [70] H.C. Yee, M. Vinokur, M.J. Djomehri, Entropy splitting and numerical dissipation, *J. Comput. Phys.* 162 (1) (2000) 33–81.
- [71] D.S. Balsara, Second-order-accurate schemes for magnetohydrodynamics with divergence-free reconstruction, *Astrophys. J. Suppl. Ser.* 151 (1) (2004) 149, <http://stacks.iop.org/0067-0049/151/i=1/a=149>.
- [72] L.I. Sedov, Similarity and Dimensional Methods in Mechanics, CRC Press, 1993.
- [73] D.S. Balsara, A two-dimensional HLLC Riemann solver for conservation laws: application to Euler and magnetohydrodynamic flows, *J. Comput. Phys.* 231 (22) (2012) 7476–7503.
- [74] D.S. Balsara, M. Dumbser, R. Abgrall, Multidimensional HLLC Riemann solver for unstructured meshes – with application to Euler and MHD flows, *J. Comput. Phys.* 261 (2014) 172–208.
- [75] T. Zhang, Y.X. Zheng, Conjecture on the structure of solutions of the Riemann problem for two-dimensional gas dynamics systems, *SIAM J. Math. Anal.* 21 (3) (1990) 593–630.
- [76] C.W. Schulz-Rinne, Classification of the Riemann problem for two-dimensional gas dynamics, *SIAM J. Math. Anal.* 24 (1) (1993) 76–88.
- [77] C.W. Schulz-Rinne, J.P. Collins, H.M. Glaz, Numerical solution of the Riemann problem for two-dimensional gas dynamics, *SIAM J. Sci. Comput.* 14 (6) (1993) 1394–1414.
- [78] P.D. Lax, X.-D. Liu, Solution of two-dimensional Riemann problems of gas dynamics by positive schemes, *SIAM J. Sci. Comput.* 19 (2) (1998) 319–340.
- [79] T. Chang, G.-Q. Chen, S. Yang, On the 2-D Riemann problem for the compressible Euler equations. I. Interaction of shocks and rarefaction waves, *Discrete Contin. Dyn. Syst.* 1 (1995) 555–584.
- [80] A. Kurganov, E. Tadmor, Solution of two-dimensional Riemann problems for gas dynamics without Riemann problem solvers, *Numer. Methods Partial Differ. Equ.* 18 (5) (2002) 584–608.
- [81] W.-S. Don, Z. Gao, P. Li, X. Wen, Hybrid compact-WENO finite difference scheme with conjugate Fourier shock detection algorithm for hyperbolic conservation laws, *SIAM J. Sci. Comput.* 38 (2) (2016) A691–A711.
- [82] D.S. Balsara, Multidimensional HLLE Riemann solver: application to Euler and magnetohydrodynamic flows, *J. Comput. Phys.* 229 (6) (2010) 1970–1993.
- [83] D. Balsara, D. Spicer, A staggered mesh algorithm using high order Godunov fluxes to ensure solenoidal magnetic fields in magnetohydrodynamic simulations, *J. Comput. Phys.* 149 (2) (1999) 270–292.
- [84] G. Tóth, The  $\nabla \cdot B = 0$  constraint in shock-capturing magnetohydrodynamics codes, *J. Comput. Phys.* 161 (2) (2000) 605–652.
- [85] D. Koessl, E. Mueller, W. Hillebrandt, Numerical simulations of axially symmetric magnetized jets. I-The influence of equipartition magnetic fields. II-Apparent field structure and theoretical radio maps. III-Collimation of underexpanded jets by magnetic fields, *Astron. Astrophys.* 229 (1990) 378–415.
- [86] A.L. Zachary, A. Malagoli, P. Colella, A higher-order Godunov method for multidimensional ideal magnetohydrodynamics, *SIAM J. Sci. Comput.* 15 (2) (1994) 263–284.
- [87] D.S. Balsara, M. Dumbser, Divergence-free MHD on unstructured meshes using high order finite volume schemes based on multidimensional Riemann solvers, *J. Comput. Phys.* 299 (2015) 687–715.
- [88] U. Ziegler, A semi-discrete central scheme for magnetohydrodynamics on orthogonal–curvilinear grids, *J. Comput. Phys.* 230 (4) (2011) 1035–1063.
- [89] A. Mignone, P. Tzeferacos, G. Bodo, High-order conservative finite difference GLM–MHD schemes for cell-centered MHD, *J. Comput. Phys.* 229 (17) (2010) 5896–5920.
- [90] S. Li, High order central scheme on overlapping cells for magneto-hydrodynamic flows with and without constrained transport method, *J. Comput. Phys.* 227 (15) (2008) 7368–7393.
- [91] S. Kawai, Divergence-free-preserving high-order schemes for magnetohydrodynamics: an artificial magnetic resistivity method, *J. Comput. Phys.* 251 (2013) 292–318.
- [92] P. Londrillo, L. Del Zanna, High-order upwind schemes for multidimensional magnetohydrodynamics, *Astrophys. J.* 530 (1) (2000) 508.
- [93] T.A. Gardiner, J.M. Stone, An unsplit Godunov method for ideal MHD via constrained transport, *J. Comput. Phys.* 205 (2) (2005) 509–539.
- [94] R. Käppeli, S. Whitehouse, S. Scheidegger, U.-L. Pen, M. Liebendorfer, FISH: a three-dimensional parallel magnetohydrodynamics code for astrophysical applications, *Astrophys. J. Suppl. Ser.* 195 (2) (2011) 20.
- [95] J.M. Stone, T. Gardiner, A simple unsplit Godunov method for multidimensional MHD, *New Astron.* 14 (2) (2009) 139–148.