

REVIEW

Open Access



Two-stage fourth order: temporal-spatial coupling in computational fluid dynamics (CFD)

Jiequan Li^{1,2}

Correspondence:

li_jiequan@iapcm.ac.cn

¹Laboratory of Computational Physics, Institute of Applied Physics and Computational Mathematics, Beijing, People's Republic of China

²Center for Applied Physics and Technology, Peking University, Beijing, People's Republic of China

Abstract

With increasing engineering demands, there need high order accurate schemes embedded with precise physical information in order to capture delicate small scale structures and strong waves with correct “physics”. There are two families of high order methods: One is the method of line, relying on the Runge-Kutta (R-K) time-stepping. The building block is the Riemann solution labeled as the solution element “1”. Each step in R-K just has first order accuracy. In order to derive a fourth order accuracy scheme in time, one needs four stages labeled as “ $1 \odot 1 \odot 1 \odot 1 = 4$ ”. The other is the one-stage Lax-Wendroff (LW) type method, which is more compact but is complicated to design numerical fluxes and hard to use when applied to highly nonlinear problems. In recent years, the pair of solution element and dynamics element, labeled as “2”, are taken as the building block. The direct adoption of the dynamics implies the inherent temporal-spatial coupling. With this type of building blocks, a family of two-stage fourth order accurate schemes, labeled as “ $2 \odot 2 = 4$ ”, are designed for the computation of compressible fluid flows. The resulting schemes are compact, robust and efficient. This paper contributes to elucidate how and why high order accurate schemes should be so designed. To some extent, the “ $2 \odot 2 = 4$ ” algorithm extracts the advantages of the method of line and one-stage LW method. As a core part, the pair “2” is expounded and LW solver is revisited. The generalized Riemann problem (GRP) solver, as the discontinuous and nonlinear version of LW flow solver, and the gas kinetic scheme (GKS) solver, the microscopic LW solver, are all reviewed. The compact Hermite-type data reconstruction and high order approximation of boundary conditions are proposed. Besides, the computational performance and prospective discussions are presented.

Keywords: Compressible fluid dynamics, Hyperbolic balance laws, High order methods, Temporal-spatial coupling, Multi-stage two-derivative methods, Lax-Wendroff type flow solvers, GRP solver

1 Introduction

In the simulation of compressible fluid flows or related problems, there are two families of commonly-used high order accurate numerical schemes: One is the family of methods of line, for which the fluid dynamical system is written in semi-discrete form and the Runge-Kutta (RK) temporal iteration is employed for the temporal discretization, such as RK-WENO [1], RK-DG [2] and their variants. The building blocks comprises of the solution element, the associated Riemann solution, which is labeled as “1” in order to be in contrast with the Lax-Wendroff (LW) type flow solvers. The

fourth order RK temporal iteration is labeled as “ $1 \odot 1 \odot 1 \odot 1 = 4$ ”. This family of schemes have very favorable properties such as simplicity in time-stepping for complex engineering problems. The limitation is also obvious such as compactness, efficiency and fidelity. The other is the family of one-stage LW type methods, the numerical realization of Cauchy-Kowalevski (CK) approach [3] for the corresponding partial differential equations. This family of methods have the strong temporal-spatial coupling property, leading to very compact numerical schemes. However, when applied to high nonlinear problems, the complex construction of numerical fluxes hampers to develop high (more than two) order accurate schemes. Particularly, as strong waves (discontinuities) are present in flows (solutions), the CK procedure loses its physical and mathematical meanings, exhibiting the instability of the resulting schemes near discontinuities.

Careful inspection of these two families of methods motivates to combine the merits of both methods: The simplicity of multi-stage RK methods and the temporal-spatial coupling of LW type methods. This straightforward combination immediately yields a two-stage fourth order accurate temporal discretization for the LW type flow solvers [4], which is labeled as “ $2 \odot 2 = 4$ ”. Here “4” just represents “fourth” order accurate temporal discretization, but “2” has much deeper implications, some of which are enumerated below.

- (i) **“2” represents a pair.** Unlike the methods of line, this method adopts the pair, the conservative variables and their dynamics, e.g., the velocity and the acceleration, as the building block to design numerical schemes. In [4], we call this pair as the Riemann solver and the LW type solver.
- (ii) **“2” implies the temporal-spatial coupling.** The LW flow solver implies the temporal-spatial coupling property of resulting schemes. This is necessary to simulate the temporal-spatial coherent structures of fluid flows.
- (iii) **“2” stands for second order accuracy in time.** Of course, “2” also symbolizes the temporal accuracy of resulting schemes and requires that at each of the two stages the above pair should be the building block.
- (iv) **“2” indicates the exchange of kinematics and thermodynamics.** The Gibbs relation plays a fundamental role in compressible fluid flows. In the dynamical process, there is always the interaction of kinematics and thermodynamics. The stronger nonlinear waves, e.g. shocks, exist in the fluid flows, the more fundamental role the thermodynamics plays.
- (v) **“2” guarantees the compactness and efficiency.** Since only two stages are taken to achieve fourth order temporal accuracy, half amount of spatial discretization treatments are saved and much smaller computational stencils are needed. Hence the resulting schemes are more compact and efficient.
- (vi) **“2” reflects the consistency of mathematical and physical expressions of fluid dynamics.** The fluid dynamical systems essentially consist of balance laws, which say the relation between the change rate of physical quantities and the associated fluxes. The form of balance laws always makes sense no matter whether there are discontinuities in the solution. The Lax-Wendroff type flow solvers inherently reflects the consistency between the physical implication of fluid dynamical systems and their mathematical formulation.

In this paper we will elucidate the idea of this new family of schemes by interpreting the philosophy from ordinary differential equations (ODEs) to fluid dynamical systems, reviewing the well-used GRP and GKS solvers as the representatives of the Lax-Wendroff type solvers, building high order temporal-spatially coupled high order accurate schemes with favorable computational performance.

We organize this paper in the following sections. In Section 2, we propose this new family of methods and the corresponding “2 ⊙ 2” algorithm. In Section 3, we review the generalized Riemann problem (GRP) solver and in Section 4 continue to review a kinetic solver, the gas kinetic scheme (GKS) solver. In Section 5, we introduce the compact Hermite-type interpolation for the data reconstruction. In Section 6, we discuss the approximation of boundary conditions to suit for the 2 ⊙ 2 algorithm. In Section 7, we remark the computational performance of this approach in terms of computational efficiency, robustness and fidelity.

2 What is “2 ⊙ 2 = 4”?

This section serves to elucidate the meaning of “2 ⊙ 2 = 4” for hyperbolic problems and particularly compressible fluid flows, and review the two-stage fourth order accurate schemes proposed in [4]. We remind that this strategy may not be suitable for incompressible flows or it needs some modifications but certainly awaits for further improvement.

2.1 Start with ODEs and philosophic thinking

Let’s recall the Runge-Kutta (RK) method for an ordinary differential equation

$$\frac{dy}{dt} = f(t, y). \tag{1}$$

The Runge-Kutta method takes the iteration procedure

$$y_{n+1} = y_n + h \sum_{i=1}^s b_i k_i$$

$$k_i = f \left(t_n + c_i h, y_n + h \sum_{j=1}^{i-1} a_{ij} k_j \right), \quad i = 1, \dots, s, \tag{2}$$

where h is the time increment, a_{ij} , b_i and c_i satisfy the Butcher tableau [5]. The building block of RK is the solution element y . In order to devise a s -th order accurate scheme, one needs s -stage iteration, which is parameter-dependent. In this paper, we focus on fourth order accurate schemes and therefore label the fourth order RK schemes as $1 \odot 1 \odot 1 \odot 1 = 4$. The notation “⊙” is an operation satisfying certain requirement such as stability.

The RK method lays the foundation of numerical approximations to ODEs. Note that this method only uses the solution element “ y ”, but ignores the dynamics element dy/dt . This sounds confusing, however, one may pay his attention to the role of the dynamics element if he is familiar with the symplectic algorithm for Hamiltonian system [6] for which the pair of the position and momentum are together used for the computation in order to preserve the symplectic structure. The momentum can be regarded as the dynamics element of the position (solution element). The word “*symplectic*” itself has the meaning of “*pair*”.

With this philosophical thinking, it is reasonable to construct multi-stage multi-derivative algorithms for dynamical systems (ODEs). Indeed, this was achieved in [7]

with many subsequent works [8–11]. The building block is the pair, the solution element and the dynamics element (the derivatives). Specifically, a multi-stage two-derivative algorithm is written as

$$\begin{aligned}
 Y_i &= y_n + h \sum_{j=1}^{i-1} a_{ij} f(Y_j) + h^2 \sum_{j=1}^{i-1} \hat{a}_{ij} g(Y_j), \quad i = 1, \dots, s, \\
 y_{n+1} &= y_n + h \sum_{i=1}^s b_i f(Y_i) + h^2 \sum_{i=1}^s \hat{b}_i g(Y_i),
 \end{aligned}
 \tag{3}$$

where we suppress the dependence of f on t for simplicity so that $f = f(y)$, the coefficients a_{ij} , \hat{a}_{ij} , b_i , and \hat{b}_i can be displayed in an extended Butcher tableau [7]. Here the function $g(y) = f'(y)f(y)$ is given using the chain rule

$$g(y) = \frac{d}{dt} f(y) = f'(y) \frac{dy}{dt} = f'(y) f(y).
 \tag{4}$$

The dynamical element is implicitly used in the construction of algorithm (3). This is why this method is of multi-stage two-derivative type with the pair $(y, dy/dt)$ as the building block. In particular, as $s = 2$, we have the two-stage fourth order accurate time-stepping algorithm in the form

$$\begin{aligned}
 y_* &= y_n + \frac{h}{2} f(y_n) + \frac{h^2}{8} f'(y_n) f(y_n), \\
 y_{n+1} &= y_n + h f(y_n) + \frac{h^2}{6} (f'(y_n) f(y_n) + 2f'(y_*) f(y_*)),
 \end{aligned}
 \tag{5}$$

labelled as the “ $2 \odot 2 = 4$ ” algorithm, which was independently derived in [4] for hyperbolic conservation laws. See the discussion in the subsequent sections. For (5), the first “2” represents the two-stage approach, the second “2” means the pair of the solution element and the dynamics element, and “4” stands for the fourth order accurate approximation to (1). Certainly, the first “2” has more implications when applied to the fluid dynamical systems for compressible flows. Besides, the notation “ \odot ” is used here to symbolize the mathematical operation currently. Probably in the future, this notation could be replaced by a better one.

2.2 Lax-Wendroff flow solvers

The Lax-Wendroff method [12] plays a fundamental role in the development of high order accurate schemes for hyperbolic equations. The corresponding scheme is unique in the class of three-point schemes of second order both in space and time. The feature of uniqueness implies that it is the reference of high order accurate schemes, and the three-point stencil hints at the compactness. Here we are going to show more fundamentals of this method, which is taken as the building block or higher order accurate schemes. Moreover, we would like to present as many details as possible because it is unusual that it has not been received “enough” attention since its birth. Part of the reason may be the presence of oscillations near discontinuities when used to simulated compressible fluid flows although it was modified, e.g., the flux limiter methods in 1980s ([13] and its successors), to be suited for the capture of discontinuities; part of the reason is, more possibly, the seeming complexity compared to methods of line. Even more seriously, the misuse in various contexts, such as diffusion equations and (dispersive) KdV type equations, leads to many controversial issues.

2.3 The revisit of Lax-Wendroff method

Let us first recall the Lax-Wendroff method [12]. Consider the advection equation

$$u_t + au_x = 0, \quad t > 0, \quad x \in [0, 1], \tag{6}$$

where a is a constant. The boundary condition remains to be discussed in Section 6. We approximate (6) by assuming that the solution is sufficiently regular, and take the Taylor series expansion at any point (x, t) to obtain,

$$u(x, t + \Delta t) = u(x, t) + \Delta t \frac{\partial u}{\partial t}(x, t) + \frac{\Delta t^2}{2} \frac{\partial^2 u}{\partial t^2}(x, t) + \mathcal{O}(\Delta t^3). \tag{7}$$

A key step is the *temporal-spatial coupling* technique by taking use of (6) to quantify the differentiation relation between the change rate of u and the spatial variation,

$$\frac{\partial u}{\partial t} = -a \frac{\partial u}{\partial x}, \quad \frac{\partial^2 u}{\partial t^2} = a^2 \frac{\partial^2 u}{\partial x^2}. \tag{8}$$

Ignoring truncation errors of order more than three, the Lax-Wendroff scheme is derived as (cf. [12]),

$$u_j^{n+1} = u_j^n - \frac{\lambda}{2} (u_{j+1}^n - u_{j-1}^n) + \frac{\lambda^2}{2} (u_{j+1}^n - 2u_j^n + u_{j-1}^n), \quad \lambda = a \frac{\Delta t}{\Delta x}, \tag{9}$$

where central difference approximations are made to guarantee the spatial accuracy, u_j^n represents the point value $u(x_j, t_n)$ at the grid point (x_j, t_n) , $x_j = j\Delta x$, $t_n = n\Delta t$, with the spatial and temporal increments Δx and Δt . The Taylor expansion process is the same as that in the Cauchy-Kowalevski approach (see [3]), and therefore (9) is regarded as the numerical realization of the CK approach. Note that this process determines the feature of this scheme, implying its application only in the range of hyperbolic problems (local behavior or finite propagation property). Any other extension needs serious and cautious treatments.

The Taylor expansion also relies on the smoothness of the solution. The successive differentiation (6) gives rise to the risk in the following sense.

- (i) Once Eq. 6 admits discontinuities in the solution, the manipulation for (8) does not make any sense. This is the main reason that (9) produces oscillations near discontinuities [12].
- (ii) As this method is applied to highly nonlinear dynamical systems, this manipulation becomes horrible and hampers to develop higher order accurate schemes, due to the successive differentiations.

We will comment on this manipulation appropriately at later sections. Rather now, we reinspect (6) and (7) from another point of view (after ignoring high order truncation errors), actually in the finite volume framework,

$$\begin{aligned} u(x, t + \Delta t) &= u(x, t) + \Delta t \frac{\partial}{\partial t} \left[u + \frac{\Delta t}{2} \frac{\partial u}{\partial t} \right] \\ &= u(x, t) - a\Delta t \frac{\partial}{\partial x} \left[u + \frac{\Delta t}{2} \frac{\partial u}{\partial t} \right], \end{aligned} \tag{10}$$

where the differentiation relation $\frac{\partial}{\partial t} = -a \frac{\partial}{\partial x}$ is applied. We immediately realize that for any (x, t)

$$u(x, t) + \frac{\Delta t}{2} \frac{\partial u}{\partial t} = u \left(x, t + \frac{\Delta t}{2} \right) + \mathcal{O}(\Delta t^2), \tag{11}$$

as long as the solution is smooth in t (temporal direction or flow direction). Viewing (10) in the finite volume framework, we obtain over the control volume $[x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}] \times [t_n, t_{n+1})$, $x_{j+\frac{1}{2}} = \frac{1}{2}(x_j + x_{j+1})$,

$$\begin{aligned} u_{j+\frac{1}{2}}^* &:= u\left(x_{j+\frac{1}{2}}, t_n\right) + \frac{\Delta t}{2} \frac{\partial u}{\partial t}\left(x_{j+\frac{1}{2}}, t_n\right), \\ u_j^{n+1} &= u_j^n - \lambda \left(u_{j+\frac{1}{2}}^* - u_{j-\frac{1}{2}}^*\right). \end{aligned} \tag{12}$$

The prediction of the value $u_{j+\frac{1}{2}}^*$ depends on the approximations to $u\left(x_{j+\frac{1}{2}}, t_n\right)$ and $\frac{\partial u}{\partial t}\left(x_{j+\frac{1}{2}}, t_n\right)$. This is achieved by the *Lax-Wendroff solver*.

Lax-Wendroff solver. A *Lax-Wendroff solver* for (6) is the numerical algorithm approximating the values

$$u_{j+\frac{1}{2}}^n := \lim_{t \rightarrow t_n+0} u\left(x_{j+\frac{1}{2}}, t\right), \quad \left(\frac{\partial u}{\partial t}\right)_{j+\frac{1}{2}}^n := \lim_{t \rightarrow t_n+0} \frac{\partial u}{\partial t}\left(x_{j+\frac{1}{2}}, t\right) \tag{13}$$

for the given initial data at $t = t_n$ for (6).

This pair of values actually provide all quite detailed information along the interface $x = x_{j+\frac{1}{2}}$ of control volume and also the flux

$$\begin{aligned} \frac{1}{\Delta t} \int_{t_n}^{t_{n+1}} au\left(x_{j+\frac{1}{2}}, t\right) dt &= au\left(x_{j+\frac{1}{2}}, t_n + \frac{\Delta t}{2}\right) + \mathcal{O}(\Delta t^2) \\ &= a \left[u_{j+\frac{1}{2}}^n + \frac{\Delta t}{2} \left(\frac{\partial u}{\partial t}\right)_{j+\frac{1}{2}}^n \right] + \mathcal{O}(\Delta t^2). \end{aligned} \tag{14}$$

The two formulae (9) and (12) are equivalent for smooth flows. However, the new formulation (12) is fundamentally different from (9) in the following sense.

- (i) The formulae (12) is actually the finite volume formulation for (6). The formulation is more straightforward for fluid dynamical systems than other formulations because it is just the numerical version of balance laws and allows discontinuities as its solution.
- (ii) The manipulation (11) is legal because the flow should be smooth in time (but not in space), unlike the difference approximation for LW approach.
- (iii) The temporal-spatial coupling feature again plays an important role, e.g.,

$$\frac{\partial u}{\partial t}\left(x_{j+\frac{1}{2}}, t_n\right) = -a \frac{\partial u}{\partial x}\left(x_{j+\frac{1}{2}}, t_n\right). \tag{15}$$

This feature is crucial for a numerical scheme to preserve the fluid dynamical properties such as the Galilean invariance.

- (iv) The successive differentiation (8) can be avoided, which is extremely important for nonlinear problems when discontinuities are involved because the manipulation (8) makes no sense both mathematically and physically.

For (6), we label “2” for the pair $(u, \frac{\partial u}{\partial t})$ in the Lax-Wendroff solver (13), which is the building block, as we see, in the Lax-Wendroff scheme. It is interesting to observe that (13) can be approximated in an upwind or central way. The upwind approximation can avoid superfluous information in the scheme.

2.4 Lax-Wendroff flow solvers for nonlinear hyperbolic balance laws

We consider hyperbolic conservation laws

$$\mathbf{u}_t + \mathbf{f}(\mathbf{u})_x = 0, \tag{16}$$

where the vector \mathbf{u} is the conservative variable. The natural formulation of (16) is in the finite volume framework, the balance law over any interval $I_j = (x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}})$,

$$\begin{aligned} \frac{d}{dt} \bar{\mathbf{u}}_j(t) &= -\frac{1}{\Delta x} \left(\mathbf{f}(\mathbf{u}(x_{j+\frac{1}{2}}, t)) - \mathbf{f}(\mathbf{u}(x_{j-\frac{1}{2}}, t)) \right), \\ \bar{\mathbf{u}}_j(t) &= \frac{1}{\Delta x} \int_{I_j} \mathbf{u}(x, t) dx, \end{aligned} \tag{17}$$

or the control volume $(x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}) \times (t_n, t_{n+1})$,

$$\bar{\mathbf{u}}_j^{n+1} = \bar{\mathbf{u}}_j^n - \frac{\Delta t}{\Delta x} \left(\mathbf{f}_{j+\frac{1}{2}}(t_n; t_{n+1}) - \mathbf{f}_{j-\frac{1}{2}}(t_n; t_{n+1}) \right), \tag{18}$$

with

$$\bar{\mathbf{u}}_j^n = \frac{1}{\Delta x} \int_{I_j} \mathbf{u}(x, t_n) dx, \quad \mathbf{f}_{j+\frac{1}{2}}(t_n; t_{n+1}) = \frac{1}{\Delta t} \int_{t_n}^{t_{n+1}} \mathbf{f}(\mathbf{u}(x_{j+\frac{1}{2}}, t)) dt. \tag{19}$$

If one would prefer to other formulations, such as the discontinuous Galerkin method [2], the following statements still hold.

We shift (x_j, t_n) to $(0, 0)$ due to the invariance of (16) with respect to the translation of coordinates. In order to proceed in one of those frameworks, we have to solve (16) approximately subject to the initial data

$$\mathbf{u}(x, 0) = \mathbf{P}_\pm(x), \quad \text{for } \pm x > 0, \tag{20}$$

where $\mathbf{P}_\pm(x)$ are smooth functions, typically polynomials, with a jump at $x = 0$. The same as in the linear case (13), a *Lax-Wendroff flow solver* for such a problem is an algorithm approximating

$$\mathbf{u}_0 := \lim_{t \rightarrow 0^+} \mathbf{u}(0, t), \quad \left(\frac{\partial \mathbf{u}}{\partial t} \right)_0 = \lim_{t \rightarrow 0^+} \frac{\partial \mathbf{u}}{\partial t}(0, t). \tag{21}$$

In general, we consider hyperbolic balance laws in multi-dimensions,

$$\mathbf{u}_t + \nabla \cdot \mathbf{F}(\mathbf{u}) = \mathbf{h}, \quad \mathbf{F} = (\mathbf{f}_x, \mathbf{f}_y, \mathbf{f}_z). \tag{22}$$

where \mathbf{h} is the source term resulting from physics or geometry, $\mathbf{x} = (x, y, z)$ is the spatial coordinate. The initial data for (22) is set to be

$$\mathbf{u}(x, y, z, 0) = \mathbf{P}_\pm(x, y, z), \quad \text{for } \pm \boldsymbol{\mu} \cdot \mathbf{x} > 0, \tag{23}$$

where $\boldsymbol{\mu}$ is the unit normal of a line or plane $L : \boldsymbol{\mu} \cdot \mathbf{x} = 0$ pointing from the negative side to the positive side, corresponding to the outer normal of interfaces of computational volume. The Lax-Wendroff solver for (22) is to find the pair of values with the same form as in (21),

$$\mathbf{u}_{L,0} := \lim_{t \rightarrow 0^+} \mathbf{u}(L, t), \quad \left(\frac{\partial \mathbf{u}}{\partial t} \right)_{L,0} := \lim_{t \rightarrow 0^+} \frac{\partial \mathbf{u}}{\partial t}(L, t), \tag{24}$$

where the limit is taken along the spatial-temporal interface $L \times (0, \Delta t)$.

We want to remark here that the pair $(\mathbf{u}_{L,0}, (\partial \mathbf{u} / \partial t)_{L,0})$ can be modulated to any direction in order to suit for an arbitrary Lagrangian-Eulerian (ALE) method.

2.5 Rough comments on the correlation between LW solver and temporal-spatial coupling

The instantaneous temporal derivatives in (21) and (24) can be roughly using the Lax-Wendroff approach

$$\frac{\partial \mathbf{u}}{\partial t} = -\nabla \cdot \mathbf{F}(\mathbf{u}) + \mathbf{h}, \tag{25}$$

and then $-\nabla \cdot \mathbf{F}(\mathbf{u})$ and \mathbf{h} are approximated using certain technologies such as WENO etc. The same as in (15), the coherent relation of spatial and temporal variations is rooted in this formula.

The intuitive outcome of this coupling is the following.

- (i) *The multidimensional effect, in particular the transversal effect, is input into the flux directly.* Thinking of a single advection problem

$$u_t + au_x + bu_y = 0, \tag{26}$$

where a, b are constants. For an interface with the normal in the x -direction, the transversal effect, expressed in the y -direction, is ignored for the standard Riemann solver. This is further verified for the wave system

$$p_t + c_0u_x + c_0v_y = 0, \quad u_t + c_0p_x = 0, \quad v_t + c_0p_y = 0, \tag{27}$$

where c_0 is a constant. In Table 1, we use three methods to simulate the periodic wave problem. It is observed that even with the same convergence rate, the RK method produces also ten times of errors than what the second order GRP does for which the transversal effect is included. The solution cannot even converge with the refinement of meshes if only normal flux is used but the transversal effect is not included. See [14].

- (ii) *The source effect \mathbf{h} is also reflected through such a process.* It is simple to see that

$$\frac{\partial u}{\partial t} = -\frac{\partial f(u)}{\partial x} + h(u, x), \tag{28}$$

for hyperbolic balance law

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = h(u, x). \tag{29}$$

This input is essential and indispensable for the well-balancedness, as verified for the shallow water equations [15].

There are more fundamentals, such as the thermodynamical effect [16], resulting from the temporal-spatial coupling.

Table 1 L_1 error and convergence order of u for the periodic wave problem at final time $t = 2$ with the methods *GRP2D*, *RK2* and *GRP1D*

N	<i>GRP2D</i>		<i>RK2</i>		<i>GRP1D</i>	
	L_1 error	order	L_1 error	order	L_1 error	order
40	4.54E-2		1.38E-1		4.16E-1	
80	7.32E-3	2.63	3.55E-2	1.96	2.25E-1	0.89
160	1.33E-3	2.46	8.96E-3	1.99	1.17E-1	0.94
320	2.81E-4	2.25	2.25E-3	2.00	6.66E-2	0.81
640	6.53E-5	2.10	5.62E-4	2.00	1.88E-1	-1.50

2.6 2 ⊙ 2 = 4: Two-stage fourth order accurate schemes

In [4], the fourth order accurate method is developed for hyperbolic conservation laws. We start with the review of the dynamical system

$$\frac{d}{dt} \mathbf{w} = \mathcal{L}(\mathbf{w}), \tag{30}$$

where \mathcal{L} is a linear or nonlinear operator of \mathbf{w} . Then we have the following two-stage algorithm for (30).

Stage 1. Define intermediate values

$$\begin{aligned} \mathbf{w}^* &= \mathbf{w}^n + \frac{1}{2} \Delta t \mathcal{L}(\mathbf{w}^n) + \frac{1}{8} \Delta t^2 \frac{\partial}{\partial t} \mathcal{L}(\mathbf{w}^n), \\ \frac{\partial}{\partial t} \mathcal{L}(\mathbf{w}^*) &= \frac{\partial}{\partial \mathbf{u}} \mathcal{L}(\mathbf{w}^*) \mathcal{L}(\mathbf{w}^*), \end{aligned} \tag{31}$$

where the second equation follows from (30), using the chain rule.

Stage 2. Advance the solution using the formula

$$\mathbf{w}^{n+1} = \mathbf{w}^n + \Delta t \mathcal{L}(\mathbf{w}^n) + \frac{1}{6} \Delta t^2 \left(\frac{\partial}{\partial t} \mathcal{L}(\mathbf{w}^n) + 2 \frac{\partial}{\partial t} \mathcal{L}(\mathbf{w}^*) \right). \tag{32}$$

This algorithm provides a fourth order accurate approximation to \mathbf{w} . Originally, this algorithm was proposed in [7, 17], and independently in [4] based on Lax-Wendroff flow solvers. Along this direction, one can derive as high order accurate approximations as what he likes [7, 18].

When applied to hyperbolic problems (16) and (22), one can formulate them in any appropriate framework such as finite volume framework [4] or discontinuous Galerkin (DG) framework [19]. Hence we assume that the computational domain Ω is meshed as $\Omega = \cup_{j \in \mathcal{J}} \Omega_j$ and formulate the problem in the form

$$\frac{d}{dt} \mathbf{w}_j(t) = \mathcal{L}_j(\mathbf{w}), \quad \mathbf{w}_j = \frac{1}{|\Omega_j|} \int_{\Omega_j} \mathbf{u}(\mathbf{x}, t) d\mathbf{x}, \quad \mathbf{w} = \{\mathbf{w}_j; j \in \mathcal{J}\}. \tag{33}$$

Thus, this problem boils down to the dynamical system in the form (30). Then we have a two-stage fourth order time-stepping method, now symbolized as the “2⊙2 = 4” method. The intuitive meaning is that we adopt the second order flow solvers as building blocks and use a two-stage time-stepping to achieve fourth order accurate numerical methods for hyperbolic problems or convection-dominated problems. We make a diagram in the following.

$$\boxed{\text{“4” : A fourth order scheme}} = \boxed{\text{“2” : Second order Lax-Wendroff type flow solvers}} + \boxed{\text{“2” : A two-stage time stepping}}$$

Careful readers may observe the validity of (33) when the above two-stage algorithm applies to the current case, which is why we have to develop the Lax-Wendroff type flow solvers based on hyperbolic balance laws rather than the formal partial differential equations (Ben-Artzi, M, Li, J: On the consistency and convergence of finite volume approximations to hyperbolic balance laws, submitted). There are at least two points that we should concern: (i) *System (33) is index-dependent and therefore each equation for fixed j is related to the neighboring equations;* (ii) *the continuity of \mathcal{L}_j is crucial when applying the above two-stage algorithm.* It is at these points that (30) is substantially different from (1). Therefore, the regularity of flux is a key factor to guarantee the validity of this

algorithm. Physically speaking, this regularity is natural by recalling the Lagrangian form of fluid dynamical systems [20]. Hence in the computation of instantaneous values (21) or (24), we must be aware of the regularity of the flux that will be further emphasized in the next section about the GRP solver.

3 The GRP solver: a discontinuous and nonlinear LW flow solver

As is well-known, and also pointed out in the last section, the standard Lax-Wendroff solver results in an algorithm that producing oscillatory solutions if discontinuities are present. The GRP solver, the abbreviation of the generalized Riemann problem (GRP) solver, can be regarded as the discontinuous and nonlinear version of the Lax-Wendroff solver. This solver was originally proposed in [21] for compressible fluid flows and related problems. See [22] for the comprehensive summary of works before 2003. Later on a direct Eulerian version of GRP solver was derived in [23] and further extended to general hyperbolic conservation laws [24, 25]. The presentation below will follow the direct Eulerian GRP. The application to non-conservative systems is referred to [26].

3.1 1-D GRP solver

We first review one-dimensional GRP solver for hyperbolic balance laws

$$\mathbf{u}_t + \mathbf{f}(\mathbf{u})_x = \mathbf{h}(x, \mathbf{u}), \tag{34}$$

subject to the initial data of form (20). An important prototype is the compressible Euler equations with cross section,

$$\begin{aligned} \frac{\partial(A(x)\rho)}{\partial t} + \frac{\partial(A(x)\rho u)}{\partial x} &= 0, \\ \frac{\partial(A(x)\rho u)}{\partial t} + \frac{\partial(A(x)\rho u^2)}{\partial x} + A(x)\frac{\partial p}{\partial x} &= 0, \\ \frac{\partial(A(x)\rho E)}{\partial t} + \frac{\partial(A(x)u(\rho E+p))}{\partial x} &= 0, \end{aligned} \tag{35}$$

where the variables ρ, u, p and E are the density, velocity, pressure and the total specific energy. The total specific energy consists of two parts $E = \frac{u^2}{2} + e$, e is the internal specific energy. The function $A(x)$ is the area of the duct. When $A(x) \equiv 1$, the system (35) represents the planar compressible Euler equations. Let T be the temperature. Then the entropy S can be defined, as usual, by Gibbs relation of thermodynamics,

$$TdS = de - \frac{p}{\rho^2}d\rho. \tag{36}$$

The local sound speed c is defined as

$$c^2 = \frac{\partial p(\rho, S)}{\partial \rho}. \tag{37}$$

We will distinguish the linear (acoustic) and nonlinear GRP solvers. Both are related. However, as strong waves are involved, the nonlinear GRP solver becomes crucial. Details can be found first in [27] and later in [24, 25] for general versions.

We denote

$$\begin{aligned} \mathbf{u}_\ell &= \mathbf{P}_-(0-0), & \mathbf{u}_r &= \mathbf{P}_+(0+0), \\ \mathbf{u}'_\ell &= \mathbf{P}'_-(0-0), & \mathbf{u}'_r &= \mathbf{P}'_+(0+0). \end{aligned} \tag{38}$$

(I) Linear GRP solver. Linear hyperbolic equations describe the propagation of waves linearly. For the single linear case (6),

$$u_t + au_x = \alpha u, \tag{39}$$

with a damping constant α , we have obviously

$$u_0 = \frac{a + |a|}{2}u_\ell + \frac{a - |a|}{2}u_r, \quad \left(\frac{\partial u}{\partial t}\right)_0 = -\frac{a + |a|}{2}u'_\ell - \frac{a - |a|}{2}u'_r + \alpha u_0. \tag{40}$$

For the linear or semi-linear system case

$$\mathbf{u}_t + \mathbf{A}\mathbf{u}_x = \mathbf{h}(\mathbf{u}, x), \tag{41}$$

we need to diagonalize the system and pursue the characteristic decomposition. Denote by $\lambda_1, \dots, \lambda_m$ the eigenvalues of \mathbf{A} , $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_m)$, and $|\Lambda| = \text{diag}(|\lambda_1|, \dots, |\lambda_m|)$. Then we decompose \mathbf{A} as $\mathbf{A} = \mathbf{R}\Lambda\mathbf{R}^{-1}$, and denote again $|\mathbf{A}| = \mathbf{R}|\Lambda|\mathbf{R}^{-1}$. It turns out that the instantaneous values take

$$\begin{aligned} \mathbf{u}_0 &= \frac{\mathbf{A} + |\mathbf{A}|}{2}\mathbf{u}_\ell + \frac{\mathbf{A} - |\mathbf{A}|}{2}\mathbf{u}_r, \\ \left(\frac{\partial \mathbf{u}}{\partial t}\right)_0 &= -\frac{\mathbf{A} + |\mathbf{A}|}{2}\mathbf{u}'_\ell - \frac{\mathbf{A} - |\mathbf{A}|}{2}\mathbf{u}'_r + \mathbf{h}(\mathbf{u}_0, 0). \end{aligned} \tag{42}$$

Therefore, the GRP solver for the linear system case is substantially the same as that for the single equation case.

(II) Acoustic Approximation

For nonlinear cases, if $\mathbf{u}_\ell = \mathbf{u}_r$, but $\|\mathbf{u}'_\ell - \mathbf{u}'_r\| \neq 0$, only linear waves emanate from the singularity point $(0, 0)$. Then we can linearize the system (34), at the value $\mathbf{u}_0 = \mathbf{u}_\ell = \mathbf{u}_r$, as

$$\theta_t + \mathbf{A}(\mathbf{u}_0)\theta_x = \mathbf{h}(x, \mathbf{u}_0), \quad \theta = \mathbf{u} - \mathbf{u}_0, \tag{43}$$

where $\mathbf{A}(\mathbf{u}_0)$ is the Jacobian $\mathbf{f}'(\mathbf{u}_0)$. Note that

$$\left(\frac{\partial \mathbf{u}}{\partial t}\right)_0 = \left(\frac{\partial \theta}{\partial t}\right)_0, \quad \text{and} \quad \left(\frac{\gamma\gamma\mathbf{u}}{yx}\right)_0 = \left(\frac{\partial \theta}{\partial x}\right)_0. \tag{44}$$

Immediately we have

$$\left(\frac{\partial \mathbf{u}}{\partial t}\right)_0 = -\frac{\mathbf{A}(\mathbf{u}_0) + |\mathbf{A}(\mathbf{u}_0)|}{2}\mathbf{u}'_\ell - \frac{\mathbf{A}(\mathbf{u}_0) - |\mathbf{A}(\mathbf{u}_0)|}{2}\mathbf{u}'_r + \mathbf{h}(\mathbf{u}_0, 0). \tag{45}$$

We can proceed to obtain any higher temporal derivatives $(\partial^m \mathbf{u} / \partial t^m)_0$, abbreviated as ADER in [28]. However, in the framework of $2 \odot 2 = 4$, we are satisfied with the first order temporal variation in (45).

As $\|\mathbf{u}_\ell - \mathbf{u}_r\| \ll 1$, weakly nonlinear waves emanate from $(0, 0)$. Then we can carry out the so-called *acoustic approximation*. To be precise, we can use either exact or approximate Riemann solvers to obtain the intermediate state \mathbf{u}_0 and linearize the system (34) to be in the form (43) so that the temporal derivative $(\partial \mathbf{u} / \partial t)_0$ is calculated as in (45).

For the Euler Eq. 35, the acoustic approximation is the following.

- (i) As $u_0 - c_0 > 0$ or $u_0 + c_0 < 0$, the acoustic waves moves to one side of the t -axis. Then $(\partial \mathbf{u} / \partial t)_0$ is taken upwind.

(ii) As $u_0 - c_0 < 0 < u_0 + c_0$, the t -axis is located between two acoustic waves. Then we have

$$\begin{aligned} \left(\frac{\partial u}{\partial t}\right)_0 &= -\frac{1}{2} \left[(u_0 + c_0) \left(u'_\ell + \frac{p'_\ell}{\rho_0 c_0} \right) + (u_0 - c_0) \left(u'_r - \frac{p'_r}{\rho_0 c_0} \right) \right], \\ \left(\frac{\partial p}{\partial t}\right)_0 &= -\frac{\rho_0 c_0}{2} \left[(u_0 + c_0) \left(u'_\ell + \frac{p'_\ell}{\rho_0 c_0} \right) - (u_0 - c_0) \left(u'_r - \frac{p'_r}{\rho_0 c_0} \right) \right] \\ &\quad - \frac{A'(0)}{A(0)} \rho_0 c_0^2 u_0. \end{aligned} \tag{46}$$

The quantity $(\partial \rho / \partial t)_0$ is solved according to the direction of the contact discontinuity,

$$\left(\frac{\partial \rho}{\partial t}\right)_0 = \frac{1}{c_0^2} \left[\left(\frac{\partial p}{\partial t}\right)_0 + u_0 (p'_\ell - c_0^2 \rho'_\ell) \right] \tag{47}$$

if $u_0 = u_\ell = u_r > 0$; and

$$\left(\frac{\partial \rho}{\partial t}\right)_0 = \frac{1}{c_0^2} \left[\left(\frac{\partial p}{\partial t}\right)_0 + u_0 (p'_r - c_0^2 \rho'_r) \right] \tag{48}$$

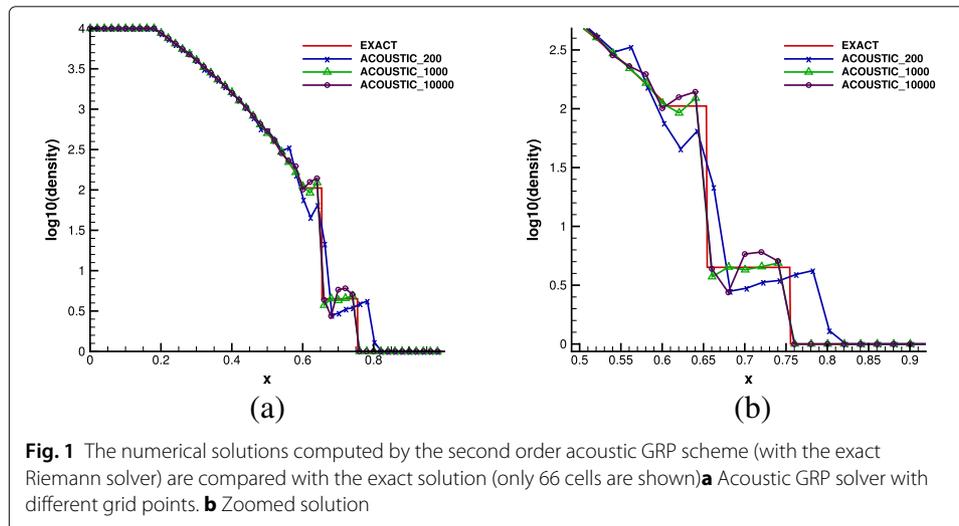
if $u_0 = u_\ell = u_r < 0$.

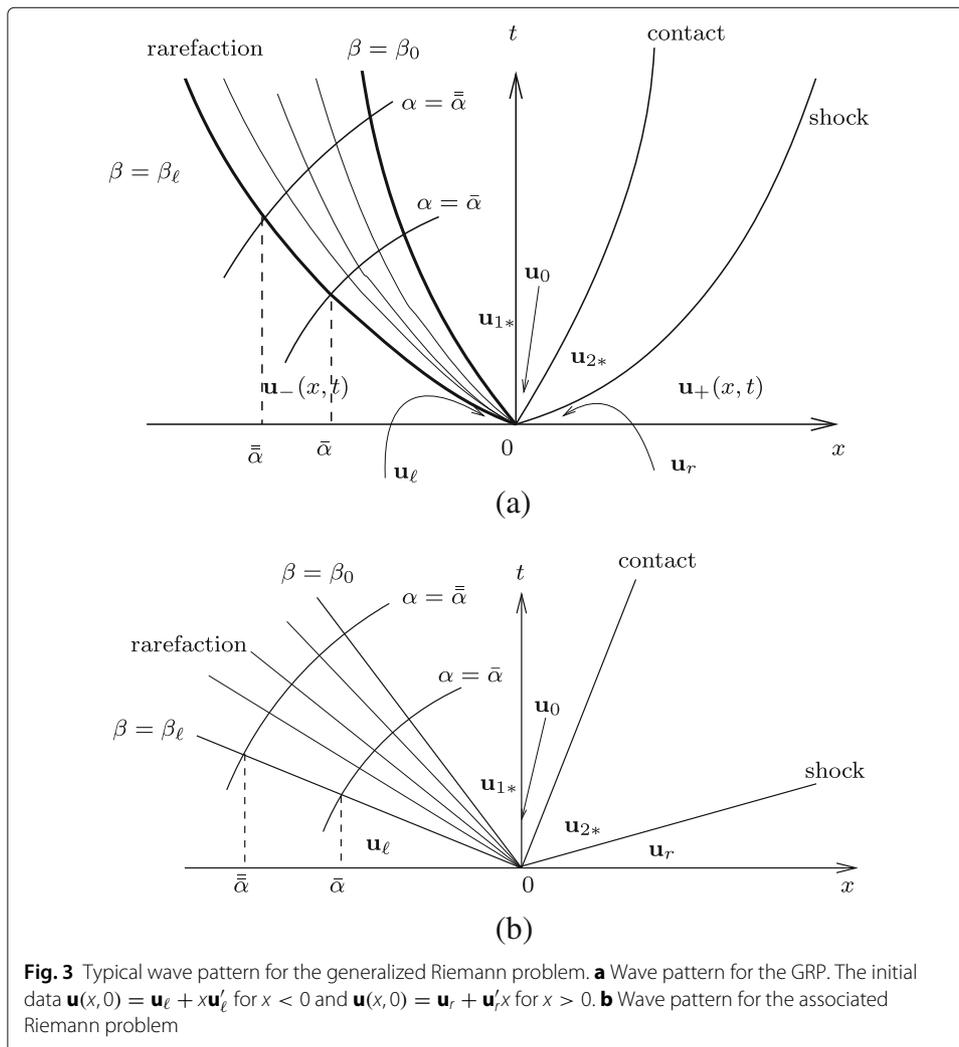
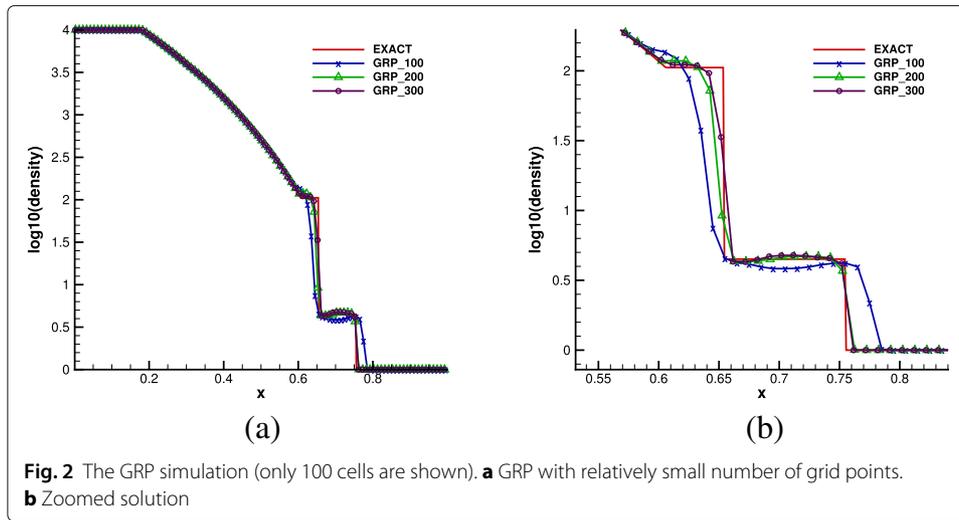
For “cheap” engineering applications, one can use the local Riemann solution \mathbf{u}_0 to linearize the nonlinear system and obtain a “linear” system so that the above acoustic approximation strategy can be adopted.

(III) Nonlinear GRP solver

As $\|\mathbf{u}_\ell - \mathbf{u}_r\| \gg 1$, nonlinear waves emanate from the singularity point $(0, 0)$. The larger the difference between \mathbf{u}_ℓ and \mathbf{u}_r is, the stronger the strength of the waves is. In Fig. 1 we use the acoustic GRP solver to simulate the *big density ratio problem* and observe that the numerical solution has large disparity from the exact solution. In Fig. 2, we use the nonlinear GRP solver that will be described below, and see that the numerical solution is improved prominently [16]. Hence it is essential to develop the nonlinear GRP solver as long as strong waves need resolving. We just illustrate the nonlinear GRP solver for Euler equations with cross section (35). For general hyperbolic balance laws, readers are referred to [24, 25].

We just need to assume a typical case, as shown in Fig. 3, that a rarefaction wave moves to the left, a shock moves to the right and a contact discontinuity lies in the middle. When





the waves move to one side of t -axis, i.e., $u_\ell - c_\ell > 0$ or $u_r + c_r < 0$, it is treated as the acoustic case that $(\partial \mathbf{u} / \partial t)_0$ is obtained upwind.

We rewrite (35) in terms of (ρ, u, S) ,

$$\begin{aligned} \frac{\partial \rho}{\partial t} + u \frac{\partial \rho}{\partial x} + \rho \frac{\partial u}{\partial x} &= -\frac{A'(x)}{A(x)} \rho u, \\ \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + \frac{1}{\rho} \frac{\partial p}{\partial x} &= 0, \\ \frac{\partial S}{\partial t} + u \frac{\partial S}{\partial x} &= 0, \end{aligned} \tag{49}$$

where p is regarded as a function of ρ and S . In terms of ρ, u and p , the third equation of (49) can be replaced by,

$$\frac{\partial p}{\partial t} + u \frac{\partial p}{\partial x} + \rho c^2 \frac{\partial u}{\partial x} = -\frac{A'(x)}{A(x)} \rho c^2 u. \tag{50}$$

In order to resolve strong rarefaction waves, it is particularly essential to introduce the so-called *generalized Riemann invariants*, as in [24],

$$\phi = u - \int^\rho \frac{c(\omega, S)}{\omega} d\omega, \quad \psi = u + \int^\rho \frac{c(\omega, S)}{\omega} d\omega. \tag{51}$$

Together with the entropy variable S , system (35) becomes

$$\begin{cases} \frac{\partial \phi}{\partial t} + (u - c) \frac{\partial \phi}{\partial x} = B_1, \\ \frac{\partial \psi}{\partial t} + (u + c) \frac{\partial \psi}{\partial x} = B_2, \\ \frac{\partial S}{\partial t} + u \frac{\partial S}{\partial x} = 0. \end{cases} \tag{52}$$

where $B_1 = T \frac{\partial S}{\partial x} + \frac{A'(x)}{A(x)} cu$, $B_2 = T \frac{\partial S}{\partial x} - \frac{A'(x)}{A(x)} cu$. Here it is easily seen that the variable section $A(x)$ acts on the dynamical behavior of ϕ and ψ , and thus on that of (ρ, u, p) . The severe change of entropy inevitably leads to the variation of other physical variables. The GRP solver that we will derive tells precisely how the entropy and the cross section affect the dynamics.

The most important ingredient is the application of nonlinear geometric optics for the local expanding of rarefaction waves using local characteristic coordinates (α, β) , as shown in Fig. 3. With that, we can obtain the instantaneous temporal derivatives $\partial S / \partial t$ and $\partial \psi / \partial t$ as (restricted to polytropic gases),

$$\begin{aligned} T \frac{\partial S}{\partial t}(0, \beta) &= -(\beta + c_\ell \theta) \theta^{\frac{2\gamma}{\gamma-1}} T_\ell S'_\ell, \quad \theta = c(0, \beta) / c_\ell, \\ \frac{\partial \psi}{\partial t}(0, \beta) &= G_1 + \frac{A'(0)}{2A(0)} G_2, \end{aligned} \tag{53}$$

where $T_\ell S'_\ell$ is defined through the Gibbs relation,

$$T_\ell S'_\ell = e'_\ell - \frac{p_\ell}{\rho_\ell} \rho'_\ell, \tag{54}$$

and G_1, G_2 are given by,

$$\begin{aligned}
 G_1 &= -\frac{\beta+c_\ell\theta}{c_\ell}\theta^{\frac{\gamma+1}{\gamma-1}}T_\ell S'_\ell + \frac{\beta+2c_\ell\theta}{c_\ell}\theta^{\frac{3-\gamma}{2(\gamma-1)}}\left[\frac{2\gamma}{3\gamma-1}T_\ell S'_\ell - c_\ell\psi'_\ell\right], \\
 G_2 &= \beta(\beta+c_\ell\theta) - (\beta+2c_\ell\theta)\left[u_\ell\theta^{\frac{3-\gamma}{2(\gamma-1)}} + \overline{G}_2\right], \\
 \overline{G}_2 &= \begin{cases} \frac{-2(\gamma+1)c_\ell\theta}{(\gamma-1)(3\gamma-5)}\left(1-\theta^{\frac{5-3\gamma}{2(\gamma-1)}}\right) - \frac{(\gamma+1)\psi_\ell}{\gamma-3}\left[1-\theta^{\frac{3-\gamma}{2(\gamma-1)}}\right], & \text{if } \gamma \neq 3, 5/3, \\ 2c_\ell(\theta-1) - \psi_\ell \ln \theta, & \text{if } \gamma = 3, \\ 2[c_\ell\theta \ln \theta + \psi_\ell(1-\theta)], & \text{if } \gamma = 5/3. \end{cases}
 \end{aligned}
 \tag{55}$$

For general cases, please refer to [29].

Several remarks are in order about the role of entropy variation and the cross section on the dynamics.

- (i) The source term reflecting geometric variation always plays an important role in the dynamics of flows. Inherently, the critical gas indices are clearly exhibited in (55), which cannot be illustrated in other flow solvers. This is just an evidence that only GRP solver can distinguish different gases.
- (ii) The entropy change rate is essential and acts on other flow variables as its initial variation is severe. This tells why the GRP solver is indispensable when strong waves are simulated. As the involved waves are weak or $T\frac{\partial S}{\partial x}$ is small, $\frac{\partial S}{\partial t}$ is negligible and many approximations such as linearization are acceptable.

The shock is resolved by tracking its trajectory described by the Rankine-Hugoniot relation,

$$\sigma = \frac{\rho u - \overline{\rho u}}{\rho - \overline{\rho}}, \quad u = \overline{u} \pm \Phi(p; \overline{p}, \overline{\rho}), \quad \rho = \Psi(p; \overline{p}, \overline{\rho}), \tag{56}$$

where (ρ, u, p) and $(\overline{\rho}, \overline{u}, \overline{p})$ are the states ahead and behind the shock, respectively, and

$$\Phi(p; \overline{p}, \overline{\rho}) = (p - \overline{p})\sqrt{\frac{1 - \mu^2}{\overline{\rho}(p + \mu^2\overline{p})}}, \quad \Psi(p; \overline{p}, \overline{\rho}) = \overline{\rho}\frac{p + \mu^2\overline{p}}{\overline{p} + \mu^2\overline{p}}, \quad \mu^2 = \frac{\gamma - 1}{\gamma + 1}, \tag{57}$$

for polytropic gases.

For the contact discontinuity $x = x(t)$, we make use of the continuity property of pressure and velocity on both sides of the trajectory,

$$u(x(t) - 0, t) = u(x(t) + 0, t), \quad p(x(t) - 0, t) = p(x(t) + 0, t). \tag{58}$$

Then the differentiation along the contact discontinuity gives

$$\frac{Du}{Dt}(x(t) - 0, t) = \frac{Du}{Dt}(x(t) + 0, t), \quad \frac{Dp}{Dt}(x(t) - 0, t) = \frac{Dp}{Dt}(x(t) + 0, t), \tag{59}$$

where $D/Dt = \partial/\partial t + u\partial/\partial x$ is the material derivative. This relation bridges the rarefaction wave and the shock, just like that for the Riemann solver. We just remind that the density and entropy undergo jump across this contact discontinuity.

Thus we come to the nonlinear GRP solver that are distinguished as nonsonic and sonic cases.

Proposition 1 (*Non-sonic case.*) *Assume a typical wave configuration for the generalized Riemann problem for (35) as shown in Fig. 3 that the t -axis is located in the intermediate region. Then $(\partial u/\partial t)_0$ and $(\partial p/\partial t)_0$ satisfies the following pair of linear equations,*

$$\begin{aligned} a_\ell \left(\frac{\partial u}{\partial t} \right)_0 + b_\ell \left(\frac{\partial p}{\partial t} \right)_0 &= d_\ell, \\ a_r \left(\frac{\partial u}{\partial t} \right)_0 + b_r \left(\frac{\partial p}{\partial t} \right)_0 &= d_r, \end{aligned} \tag{60}$$

where a_ℓ, b_ℓ, d_ℓ and a_r, b_r, d_r are specified below. Also the computation of $(\partial\rho/\partial t)_0$ are computed by the following two cases.

(i) If $u_0 > 0$, the contact discontinuity moves to the right and separates two states $(\rho_{0\ell}, u_0, p_0), (\rho_{0r}, u_0, p_0)$. The coefficients a_ℓ, b_ℓ and d_ℓ are given as,

$$(a_\ell, b_\ell, d_\ell) = (\tilde{a}_\ell, \tilde{b}_\ell, \tilde{d}_\ell). \tag{61}$$

The coefficients a_r, b_r and d_r are given by

$$\begin{aligned} a_r &= \frac{c_{0\ell}^2}{c_{0\ell}^2 - u_0^2} \left[\tilde{a}_r \left(1 - \frac{\rho_{0\ell} u_0^2}{\rho_{0r} c_{0r}^2} \right) + \tilde{b}_r (\rho_{0r} - \rho_{0\ell}) u_0 \right], \\ b_r &= \frac{1}{c_{0\ell}^2 - u_0^2} \left[\tilde{a}_r \left(-\frac{1}{\rho_{0\ell}} + \frac{c_{0\ell}^2}{\rho_{0r} c_{0r}^2} \right) u_0 - \tilde{b}_r \left(-\frac{\rho_{0r}}{\rho_{0\ell}} u_0^2 + c_{0\ell}^2 \right) \right], \\ d_r &= \tilde{d}_r + \frac{A'(0)}{A(0)} \frac{u_0^3}{c_{0\ell}^2 - u_0^2} \left[\tilde{a}_r \left(1 - \frac{\rho_{0\ell} c_{0\ell}^2}{\rho_{0r} c_{0r}^2} \right) + \tilde{b}_r (\rho_{0r} - \rho_{0\ell}) c_{0\ell}^2 \right]. \end{aligned} \tag{62}$$

The value $(\partial\rho/\partial t)_0$ is computed from the rarefaction side,

$$\left(\frac{\partial \rho}{\partial t} \right)_0 = \frac{1}{c_{0\ell}^2} \left[\left(\frac{\partial p}{\partial t} \right)_0 + (\gamma - 1) \rho_{0\ell} u_0 \left(\frac{c_{0\ell}}{c_\ell} \right)^{\frac{1+\mu^2}{\mu^2}} T_\ell S'_\ell \right], \tag{63}$$

by using the state equation $p = p(\rho, S)$.

(ii) If $u_0 < 0$, the contact discontinuity moves to the left. The coefficients a_r, b_r and d_r are given as,

$$(a_r, b_r, d_r) = (\tilde{a}_r, \tilde{b}_r, \tilde{d}_r). \tag{64}$$

While the coefficients a_ℓ, b_ℓ and d_ℓ are computed by

$$\begin{aligned} a_\ell &= \frac{c_{0r}^2}{c_{0r}^2 - u_0^2} \left[\tilde{a}_\ell \left(1 - \frac{\rho_{0r} u_0^2}{\rho_{0\ell} c_{0\ell}^2} \right) + \tilde{b}_\ell (\rho_{0\ell} - \rho_{0r}) u_0 \right], \\ b_\ell &= \frac{1}{c_{0r}^2 - u_0^2} \left[\tilde{a}_\ell \left(-\frac{1}{\rho_{0r}} + \frac{c_{0r}^2}{\rho_{0\ell} c_{0\ell}^2} \right) u_0 - \tilde{b}_\ell \left(-\frac{\rho_{0\ell}}{\rho_{0r}} u_0^2 + c_{0r}^2 \right) \right], \\ d_\ell &= \tilde{d}_\ell + \frac{A'(0)}{A(0)} \frac{u_0^3}{c_{0r}^2 - u_0^2} \left[\tilde{a}_\ell \left(1 - \frac{\rho_{0r} c_{0r}^2}{\rho_{0\ell} c_{0\ell}^2} \right) + \tilde{b}_\ell (\rho_{0\ell} - \rho_{0r}) c_{0r}^2 \right]. \end{aligned} \tag{65}$$

The value $(\partial\rho/\partial t)_0$ is computed from the shock side,

$$g_\rho^R \left(\frac{\partial \rho}{\partial t} \right)_0 + g_p^R \left(\frac{Dp}{Dt} \right)_0 + g_u^R \left(\frac{Du}{Dt} \right)_0 = u_0 \cdot f_r, \tag{66}$$

where g_ρ^R, g_p^R, g_u^R and f_r are constant, explicitly given in the following,

$$\begin{aligned}
 g_\rho^R &= u_0 - \sigma_0, \quad g_p^R = \frac{\sigma_0}{c_0^2} - u_0 H_1, \quad g_u^R = \rho_{2*}(\sigma_0 - u_0) \cdot u_0 \cdot H_1, \\
 f_r &= (\sigma_0 - u_r) \cdot H_2 \cdot p'_r + (\sigma_0 - u_r) \cdot H_3 \cdot \rho'_r - \rho_r \cdot (H_2 c_r^2 + H_3) \cdot u'_r \\
 &\quad - \frac{A'(0)}{A(0)} \cdot (H_2 c_r^2 + H_3) \rho_r u_r,
 \end{aligned} \tag{67}$$

and H_1, H_2 and H_3 are expressed by

$$H_1 = \frac{\rho_r(1-\mu^4)p_r}{(p_r+\mu^2p_0)^2}, \quad H_2 = \frac{\rho_r(\mu^4-1)p_0}{(p_r+\mu^2p_0)^2}, \quad H_3 = \frac{p_0+\mu^2p_r}{p_r+\mu^2p_0}. \tag{68}$$

The other coefficients $(\tilde{a}_\ell, \tilde{b}_\ell, \tilde{d}_\ell)$ and $(\tilde{a}_r, \tilde{b}_r, \tilde{d}_r)$ are

$$\begin{aligned}
 \tilde{a}_\ell(0, \beta) &= 1, \\
 \tilde{b}_\ell(0, \beta) &= \frac{1}{\rho(0, \beta)c(0, \beta)}, \\
 \tilde{d}_\ell(\beta) &= \frac{\beta+2\theta c_\ell}{c_\ell} \cdot \theta^{\frac{3-\gamma}{2(\gamma-1)}} \left(\frac{2\gamma}{3\gamma-1} T_\ell S'_\ell - c_\ell \psi'_\ell \right) + \frac{A'(0)}{2A(0)} G_2,
 \end{aligned} \tag{69}$$

and

$$\begin{aligned}
 \tilde{a}_r &= 1 - \frac{\sigma_0 u_0}{u_0^2 - c_{0r}^2} - \frac{\sigma_0 \rho_{0r} c_{0r}^2}{u_0^2 - c_{0r}^2} \cdot \Phi_1, \\
 \tilde{b}_r &= \frac{1}{\rho_{0r}} \frac{\sigma_0}{u_0^2 - c_{0r}^2} - \left(1 - \frac{\sigma_0 u_0}{u_0^2 - c_{0r}^2} \right) \Phi_1, \\
 \tilde{d}_r &= L_p^R \cdot p'_r + L_u^R \cdot u'_r + L_\rho^R \cdot \rho'_r + \frac{A'(0)}{A(0)} j_r,
 \end{aligned} \tag{70}$$

where we use notations

$$\begin{aligned}
 L_p^R &= -\frac{1}{\rho_r} + (\sigma_0 - u_r) \cdot \Phi_2, \\
 L_u^R &= \sigma_0 - u_r - \rho_r \cdot c_r^2 \cdot \Phi_2 - \rho_r \cdot \Phi_3, \\
 L_\rho^R &= (\sigma_0 - u_r) \cdot \Phi_3, \\
 j_r &= -(\Phi_2 c_{0r}^2 + \Phi_3) \rho_r u_r + (1 + \Phi_1 \rho_{0r} u_0) \frac{\sigma_0 c_{0r}^2 u_0}{u_0^2 - c_{0r}^2}; \\
 \sigma_0 &= \frac{\rho_{0r} u_0 - \rho_r u_r}{\rho_{0r} - \rho_r}, \\
 \Phi_1 &= \frac{1}{2} \sqrt{\frac{1-\mu^2}{\rho_r(p_0+\mu^2p_r)}} \cdot \frac{p_0+(1+2\mu^2)p_r}{p_0+\mu^2p_r}, \\
 \Phi_2 &= -\frac{1}{2} \sqrt{\frac{1-\mu^2}{\rho_r(p_0+\mu^2p_r)}} \cdot \frac{(2+\mu^2)p_0+\mu^2p_r}{p_0+\mu^2p_r}, \\
 \Phi_3 &= -\frac{p_0-p_r}{2\rho_r} \sqrt{\frac{1-\mu^2}{\rho_r(p_0+\mu^2p_r)}}.
 \end{aligned} \tag{71}$$

Proposition 2 (Sonic case). Assume that the t -axis is located inside the rarefaction wave associated with $u - c$. Then we have

$$\begin{aligned}
 \left(\frac{\partial u}{\partial t} \right)_0 &= \frac{1}{2} \left[\tilde{d}_\ell + \theta^{\frac{2\gamma}{\gamma-1}} T_\ell S'_\ell + \frac{A'(0)}{A(0)} u_0^2 \right], \\
 \left(\frac{\partial p}{\partial t} \right)_0 &= \frac{\rho_0 c_0}{2} \left[\tilde{d}_\ell - \theta^{\frac{2\gamma}{\gamma-1}} T_\ell S'_\ell - \frac{A'(0)}{A(0)} u_0^2 \right],
 \end{aligned} \tag{72}$$

where \tilde{d}_ℓ is given in (69), with $\theta = c_0/c_\ell$, and (u_0, ρ_0, c_0) is the limiting value of (u, ρ, c) along the t -axis so that $u_0 - c_0 = 0$. Then density change rate is given as in (63).

The above formulae look complicated, but seem irreplaceable. We can go to [24] for technical derivation of them.

3.2 Temporal-spatial coupling and thermodynamical effect

Thermodynamics distinguishes compressible fluid flows from incompressible ones, and the Mach number can be regarded as a parameter of the compressibility. The entropy dissipation is a necessary condition guaranteeing the stability of numerical schemes. Let's consider the compressible Euler equations with uniform cross section ($A(x) \equiv \text{constant}$ in (35)). The entropy inequality says

$$(\rho S)_t + (\rho u S)_x \geq 0, \text{ in } \mathcal{D}'. \tag{73}$$

However, it is a well-known open problem whether this inequality is satisfied at discrete level, particularly for high order accurate schemes. There are two origins of discrete errors: the data projection and the flux approximation. In a general setting of finite volume framework, given the initial data $\mathbf{u}_n(x) \in \mathcal{P}_k$ at $t = t_n$, we have to find the solution $\mathbf{u}_{n+1}(x)$ at next time level $t = t_{n+1}$, satisfying

$$\int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \rho S(\mathbf{u}_{n+1}(x)) dx \geq \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \rho S(\mathbf{u}_n(x)) dx - \Delta t \left[(\rho u S)_{j+\frac{1}{2}}^{approx} - (\rho u S)_{j-\frac{1}{2}}^{approx} \right] + Tol(\Delta x, \Delta t), \tag{74}$$

where $(\rho u S)_{j+\frac{1}{2}}^{approx}$ is the numerical entropy flux, and $Tol(\Delta x, \Delta t)$ is the entropy production that has the maximum tolerance of order three, $Tol(\Delta x, \Delta t) = \mathcal{O}(\Delta t^3 + \Delta x^3)$. We comment on how to achieve this inequality at the discrete level in the following.

- (i) The persistence space \mathcal{P}_k often consists of piecewise polynomials of degree k . Given the initial data $\mathbf{u}_n(x) \in \mathcal{P}_k$, we solve (35) and obtain the (analytic) entropy solution $\mathbf{u}(x, t)$ for $t_n < t < t_{n+1}$, satisfying

$$\int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \rho S(\mathbf{u}(x, t_{n+1})) dx \geq \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \rho S(\mathbf{u}_n(x)) dx - \left[\int_{t_n}^{t_{n+1}} \rho u S(x_{j+\frac{1}{2}}, t) dt - \int_{t_n}^{t_{n+1}} \rho u S(x_{j-\frac{1}{2}}, t) dt \right]. \tag{75}$$

- (ii) The projection of $\mathbf{u}(x, t)$ onto \mathcal{P}_k (reconstruction procedure) is required to satisfy

$$\int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \rho S(\mathbf{u}_{n+1}(x)) dx \geq \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \rho S(\mathbf{u}(x, t_{n+1})) dx + \mathcal{O}(\Delta x^3). \tag{76}$$

This is an extremely difficult step. For scalar conservation laws, there was a nice discussion on MUSCL-type linear reconstruction [30]. In general, the Jensen inequality tells that

$$\begin{aligned} \rho S(\bar{\mathbf{u}}_j^{n+1}) &\geq \frac{1}{\Delta x} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \rho S(x, t_{n+1}) dx, \\ \bar{\mathbf{u}}_j^{n+1} &= \frac{1}{\Delta x} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \mathbf{u}(x, t_{n+1}) dx. \end{aligned} \tag{77}$$

Hence, there is still some room to make (76) hold, which remains an open problem. *Open problem on data projection: Find an optimal reconstruction strategy so that (76) holds.*

- (iii) Assume that (76) holds for certain data projection. As shown above, we approximate the interface value in the following way

$$\mathbf{u}(x_{j+\frac{1}{2}}, t) = \mathbf{u}_{j+\frac{1}{2}}^n + \left(\frac{\partial \mathbf{u}}{\partial t}\right)_{j+\frac{1}{2}}^n (t - t_n) + \mathcal{O}((t - t_n)^2), \quad t_n < t < t_{n+1}. \quad (78)$$

In particular, we make use of the entropy information. It turns out that

$$(\rho u S)_{j+\frac{1}{2}}^{approx} := (\rho u S)(x_{j+\frac{1}{2}}, t_{n+\frac{1}{2}}) = \frac{1}{\Delta t} \int_{t_n}^{t_{n+1}} \rho u S(x_{j+\frac{1}{2}}, t) dt + \mathcal{O}(\Delta t^2). \quad (79)$$

Summarizing all together yields (74).

It is observed that the precise calculation of entropy in (53) is a direct way to achieve (79). Other ways may at most lead to

$$(\rho u S)_{j+\frac{1}{2}}^{approx} := (\rho u S)(x_{j+\frac{1}{2}}, t_{n+\frac{1}{2}}) = \frac{1}{\Delta t} \int_{t_n}^{t_{n+1}} \rho u S(x_{j+\frac{1}{2}}, t) dt + \mathcal{O}(\|\mathbf{u}\|^2). \quad (80)$$

The error of $\mathcal{O}(\|\mathbf{u}\|^2)$ is not tolerated unless for scalar cases or smooth flows, since this type of errors violate the entropy inequality in the limit.

It is no doubt that the achievement of (79) is the outcome of the direct use of the entropy equation in (49), which is actually the Lax-Wendroff procedure, a temporal-spatial coupling procedure.

3.3 M-D GRP solver and transversal effects

When computing multidimensional (M-D) problems, M-D GRP solver is necessary to reflect the transversal effect, which is impossible using the exact or approximate normal Riemann solvers. We restrict to two-dimensional hyperbolic balance laws

$$\mathbf{u}_t + \mathbf{f}(\mathbf{u})_x + \mathbf{g}(\mathbf{u})_y = \mathbf{h}(x, y, \mathbf{u}), \quad (81)$$

where \mathbf{f} and \mathbf{g} are flux functions. 3-D GRP solver is straightforward. The initial data takes the form

$$\mathbf{u}_0(x, y) = \begin{cases} \mathbf{u}_-(x, y), & x < 0, \\ \mathbf{u}_+(x, y), & x > 0, \end{cases} \quad (82)$$

where $\mathbf{u}_\pm(x, y)$ are two polynomials of degree k . The x -direction is the normal and the y -direction is the transversal. A particular case is

$$\mathbf{u}_0(x, y) = \begin{cases} \mathbf{u}_- + \nabla \mathbf{u}_{0,-} \cdot \mathbf{x}, & x < 0, \\ \mathbf{u}_+ + \nabla \mathbf{u}_{0,+} \cdot \mathbf{x}, & x > 0. \end{cases} \quad (83)$$

The M-D GRP solver can be classified as *M-D linear GRP solver*, *acoustic GRP solver*, *nonlinear normal GRP solver with transversal perturbation*, and *genuinely nonlinear M-D GRP solver*.

(I) M-D linear GRP solver. We consider the linear case

$$\mathbf{u}_t + \mathbf{A}\mathbf{u}_x + \mathbf{B}\mathbf{u}_y = 0, \quad (84)$$

where \mathbf{A} and \mathbf{B} are constant matrices, and both of them have their respective real eigenvalues and the complete sets of eigenvectors. We first assume that (84) is subject to the initial data (83). Note that $\nabla \mathbf{u}$ satisfies the same form of (84),

$$(\nabla \mathbf{u})_t + \mathbf{A}(\nabla \mathbf{u})_x + \mathbf{B}(\nabla \mathbf{u})_y = 0, \quad (85)$$

but subject to the initial data

$$\nabla \mathbf{u}_0(x, y) = \begin{cases} \nabla \mathbf{u}_{0,-}, & x < 0, \\ \nabla \mathbf{u}_{0,+}, & x > 0. \end{cases} \tag{86}$$

This boils down to the standard Riemann problem for $\nabla \mathbf{u}$. Therefore, the gradient $\nabla \mathbf{u}$ has an explicit expression,

$$\nabla \mathbf{u}(x, y, t) = \sum_{\ell=1}^{L(x,t)} \nabla v_{0,+}^{\ell} \mathbf{r}^{\ell} + \sum_{\ell=L(x,t)+1}^m \nabla v_{0,-}^{\ell} \mathbf{r}^{\ell}, \tag{87}$$

where the notations $L(x, t)$ is the maximum value of ℓ such that $x - \lambda^{\ell} t > 0$, λ^{ℓ} is the eigenvalue of \mathbf{A} and \mathbf{r}^{ℓ} is the associated eigenvector, \mathbf{v} is so defined that

$$\mathbf{u} = \sum_{\ell=1}^m v^{\ell} \mathbf{r}^{\ell}, \quad \mathbf{v} = (v^1, \dots, v^m). \tag{88}$$

See [31]. In particular, we have

$$\nabla \mathbf{u}(0, y, t) = \sum_{\ell: \lambda^{\ell} < 0} \nabla v_{0,+}^{\ell} \mathbf{r}^{\ell} + \sum_{\ell: \lambda^{\ell} > 0} \nabla v_{0,-}^{\ell} \mathbf{r}^{\ell}. \tag{89}$$

Then we immediately obtain

$$\begin{aligned} \left(\frac{\partial \mathbf{u}}{\partial t}\right)_0 &:= \lim_{t \rightarrow 0^+} \frac{\partial \mathbf{u}}{\partial t}(0, y, t) \\ &= -\mathbf{A} \lim_{t \rightarrow 0^+} \frac{\partial \mathbf{u}}{\partial x}(0, y, t) - \mathbf{B} \lim_{t \rightarrow 0^+} \frac{\partial \mathbf{u}}{\partial y}(0, y, t) \\ &= -(\mathbf{A}, \mathbf{B}) \cdot \nabla \mathbf{u}(0, y, 0^+). \end{aligned} \tag{90}$$

As far as the more general initial data (82) is concerned, the solution \mathbf{u} consists of piecewise polynomials of the same degree as the initial data since (84) is a linear system. Here we are satisfied with the second order GRP solver and solve (84) at any point $(0, y_0, 0)$ to obtain

$$\left(\frac{\partial \mathbf{u}}{\partial t}\right)_0 = -(\mathbf{A}, \mathbf{B}) \cdot \nabla \mathbf{u}(0, y_0, 0^+), \tag{91}$$

where $\nabla \mathbf{u}(0, y_0, 0^+)$ are calculated as the same procedure as above.

(II) M-D acoustic GRP solver.

For nonlinear cases (81), if the initial data (82) is continuous but discontinuous in its derivatives, acoustic waves emanate from the interface $x = 0$, just as one-dimensional case. For this case, we might as well take the initial data (83) and assume that $\mathbf{u}_- = \mathbf{u}_+$ but $\|\nabla \mathbf{u}_{0,-} - \nabla \mathbf{u}_{0,+}\| \neq 0$. Then we linearize (81) around the state $\mathbf{u}_0 = \mathbf{u}_- = \mathbf{u}_+$ to obtain

$$\theta_t + \mathbf{f}'(\mathbf{u}_0)\theta_x + \mathbf{g}'(\mathbf{u}_0)\theta_y = \mathbf{h}(x, y, \mathbf{u}_0), \quad \theta = \mathbf{u} - \mathbf{u}_0. \tag{92}$$

Then we can exactly follow the linear case to calculate $(\partial \mathbf{u} / \partial t)_0$.

The acoustic approximation applies for the case $\|\mathbf{u}_- - \mathbf{u}_+\| \ll 1$. We linearize (81) around the intermediate state \mathbf{u}_0 resulting from the associated Riemann problem. Then the linear GRP solver applies for this case.

(III) M-D genuinely nonlinear GRP solver with transversal description.

As $\|\mathbf{u}_- - \mathbf{u}_+\| \gg 1$, we have to deal with genuinely nonlinear GRP. Thinking of the initial value problem for (81) subject to the initial data (83), the solution is the envelope of Riemann solution along $x = 0$ locally at $t = 0$. Hence along $x = 0$, the associated

Riemann solution is known. For example, at two points $(0, y_1, 0)$ and $(0, y_2, 0)$, we solve the Riemann problem locally for the normal conservation law, respectively,

$$\mathbf{u}_t + \mathbf{f}(\mathbf{u})_x = 0, \tag{93}$$

and obtain the local intermediate values $\mathbf{u}(0, y_1, 0^+)$ and $\mathbf{u}(0, y_2, 0^+)$. Then for any point $(0, y, 0)$, $y_1 < y < y_2$, we can approximate $\frac{\partial \mathbf{u}}{\partial y}(0, y, 0^+)$. Particularly, we approximate

$$\mathbf{u}_y(0, y_0, 0^+) = \frac{\mathbf{u}(0, y_1, 0^+) - \mathbf{u}(0, y_2, 0^+)}{y_1 - y_2} + \mathcal{O}((y_1 - y_2)^2). \tag{94}$$

Then we regard the transversal term $\mathbf{g}(\mathbf{u})_y$ and $\mathbf{h}(x, y, \mathbf{u})$ as a perturbation locally, and solve the following problem,

$$\mathbf{u}_t + \mathbf{f}(\mathbf{u})_x = -\mathbf{g}(\mathbf{u})_y + \mathbf{h}(x, y, \mathbf{u}) =: \mathbf{d}(x, y, \mathbf{u}, \mathbf{u}_y). \tag{95}$$

This boils down to the one-dimensional planar problem locally, for which the GRP solver was proposed in [24]. Detailed and complete M-D GRP solver is proposed in [32].

3.4 Transversal effects for genuinely M-D schemes

For multidimensional (M-D) problems, the balance law can be always written in the form,

$$\frac{d}{dt} \int_{\Omega} \mathbf{u}(\mathbf{x}, t) d\mathbf{x} = - \int_{\partial\Omega} \mathbf{f}(\mathbf{u}) \cdot \mathbf{n} dL, \tag{96}$$

where Ω is the control volume, $\partial\Omega$ is the boundary and \mathbf{n} is the outer unit normal. We ignore the external force just for the clarity of presentation.

Thanks to the Galilean invariance, we always assume that $(1, 0)$ (the direction x -axis) is the normal direction, and $(0, 1)$ (the direction of y -axis) is the transversal direction. The standard Riemann solver just reflects the normal effect. In contrast, the LW procedure can describe the transversal effect precisely. Consider a linear advection problem

$$u_t + au_x + bu_y = 0. \tag{97}$$

Then we use the temporal-spatial coupling property to obtain

$$\left(\frac{\partial u}{\partial t}\right)_{\partial\Omega} = -a \left(\frac{\partial u}{\partial x}\right)_{\partial\Omega} - b \left(\frac{\partial u}{\partial y}\right)_{\partial\Omega}, \tag{98}$$

where $\left(\frac{\partial u}{\partial x}\right)_{\partial\Omega}$ and $\left(\frac{\partial u}{\partial y}\right)_{\partial\Omega}$ can be obtained by solving the associated Riemann problem. Also as remarked in Section 2 for the linear wave system, the transversal effect is substantial even though the convergence rate is formally the same.

4 The kinetic LW flow solver

The fluid dynamics can be described in various viewpoints, such as the kinetic description. The governing equation is the Boltzmann-type equation

$$f_t + \xi \cdot \nabla_{\mathbf{x}} f = \frac{1}{\epsilon} B(f, f), \tag{99}$$

where $f = f(t, \mathbf{x}, \xi)$ is the density distribution, ξ is the velocity of molecules (particle), and $B(f, f)$ is the collision term, ϵ is the Knusner number. Ideally, for a given initial distribution, we solve (99) to obtain the solution $f\left(t, x_{j+\frac{1}{2}}, \xi\right)$ and define the numerical flux

$$\mathbf{f}_{j+\frac{1}{2}}(t_n; t_{n+1}) = \frac{1}{\Delta t} \int_{t_n}^{t_{n+1}} \int_{\mathbb{R}^3} \xi \psi f(t, x_{j+\frac{1}{2}}, \xi) d\xi dt, \tag{100}$$

where $\psi = (1, \xi, \xi^2)^\top$ is the invariant, and the average of macroscopic variables is

$$u_j^{n+1} = \frac{1}{\Delta x} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \int_{\mathbb{R}^3} \psi f(t_{n+1}, x, \xi) d\xi dx. \tag{101}$$

In general, it is difficult to solve the Eq. 99 analytically. To understand the relation of macroscopic equations (Euler and Navier-Stokes equations) and the kinetic equation, we first take the *so-called railroad method* in [33] as an example to illustrate how to devise kinetic schemes.

4.1 Railroad method

Consider the linear advection equation

$$u_t + au_x = 0, \quad u(x, 0) = u_0(x). \tag{102}$$

Introduce a distribution function $f(t, x, \xi)$ and define the macroscopic variable $u(x, t)$ as a moment of f ,

$$u(x, t) = \int_{-\infty}^{\infty} f(t, x, \xi) d\xi. \tag{103}$$

If we choose f to take the form,

$$f(t, x, \xi) = \frac{u(x, t)}{\sqrt{\pi}} \exp[-(\xi - a)^2], \tag{104}$$

then $f(t, x, \xi)$ satisfies

$$f_t + \xi f_x = Q[f] := \frac{(\xi - a)}{\sqrt{\pi}} \exp[-(\xi - a)^2] \frac{\partial u}{\partial x}, \tag{105}$$

subject to initial data

$$f(0, x, \xi) = \frac{u_0(x)}{\sqrt{\pi}} \exp[-(\xi - a)^2]. \tag{106}$$

The initial value problem (102) and the problem (103)-(106) are equivalent: If one solution is known, then the other is defined. We write the solution of (105)-(106) as

$$f(t, x, \xi) = f(0, x - \Delta t\xi, \xi) + \int_0^{\Delta t} Q[f](\tau, x - (\Delta t - \tau)\xi, \tau, \xi) d\tau. \tag{107}$$

Then the solution $u(x, \Delta t)$ is given as

$$u(x, t) = \int_{-\infty}^{\infty} f(0, x - \Delta t\xi, \xi) d\xi + \int_{-\infty}^{\infty} \int_0^{\Delta t} Q[f](\tau, x - (\Delta t - \tau)\xi, \tau, \xi) d\tau d\xi. \tag{108}$$

Note that the LW approach for (107) yields

$$f(t, x, \xi) = f(0, x, \xi) + \Delta t\xi \frac{\partial}{\partial x} f(0, x, \xi) + \Delta tQ[f](0, x, \xi) + \mathcal{O}(\Delta t^2). \tag{109}$$

Therefore we have

$$u(x, t) = \int_{-\infty}^{\infty} (f(0, x, \xi) + \Delta t\xi \frac{\partial}{\partial x} f(0, x, \xi) + \Delta tQ[f](0, x, \xi)) d\xi + \mathcal{O}(\Delta t^2), \tag{110}$$

which yields a second order approximation to the exact solution $u(x, t)$.

The numerical solution is

$$u_j(\Delta t) = u_j(0) - \frac{\Delta t}{\Delta x} [F_{j+\frac{1}{2}} - F_{j-\frac{1}{2}}], \tag{111}$$

where

$$u_j(t) = \frac{1}{\Delta x} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \int_{-\infty}^{\infty} f(t, x, \xi) d\xi dx,$$

$$F_{j+\frac{1}{2}} = \int_0^{\Delta t} \int_{-\infty}^{\infty} \xi f(t, x_{j+\frac{1}{2}}, \xi) d\xi dt.$$

We assume the initial data for (102) is piecewise smooth with possible discontinuity at $x = x_{j+\frac{1}{2}}$. Correspondingly, the initial data (106) for (105) consists of two parts. It turns out that the numerical flux $F_{j+\frac{1}{2}}$ in (111) becomes

$$F_{j+\frac{1}{2}} = F_{j+\frac{1}{2}}^+ + F_{j+\frac{1}{2}}^-, \tag{112}$$

where $F_{j+\frac{1}{2}}^{\pm}$ consist of three parts, respectively,

$$\begin{aligned} F_{j+\frac{1}{2}}^{\pm} &= \int_0^{\Delta t} \int_{\pm\xi>0} \xi f(\tau, x_{j+\frac{1}{2}}, \xi) d\xi d\tau = \Delta t G_{j+\frac{1}{2}}^{\pm} - \frac{\Delta t^2}{2} \left(H_{j+\frac{1}{2}}^{\pm} - K_{j+\frac{1}{2}}^{\pm} \right), \\ G_{j+\frac{1}{2}}^{\pm} &= \int_{\pm\xi>0} \xi f(0, x_{j+\frac{1}{2}} \mp 0, \xi) d\xi, \\ H_{j+\frac{1}{2}}^{\pm} &= \int_{\pm\xi>0} \xi^2 \frac{\partial}{\partial x} f(0, x_{j+\frac{1}{2}} \mp 0, \xi) d\xi, \\ K_{j+\frac{1}{2}}^{\pm} &= \int_{\pm\xi>0} \xi Q[f](0, x_{j+\frac{1}{2}} \mp 0, \xi) d\xi. \end{aligned} \tag{113}$$

As the solution is smooth, the scheme (111) becomes the LW approach immediately. See [33] for details.

4.2 The LW type solver for gas kinetic schemes

Let's now work on a simplified model, the Bhatnagar-Gross-Krook (BGK) model [34],

$$f_t + \xi f_x = \frac{g - f}{\epsilon}, \tag{114}$$

where ϵ is the collision time, and g is the equilibrium state, approached by f as ϵ goes to zero,

$$g = \frac{\rho}{m} \left(\frac{m}{2\pi kT} \right)^{\frac{3}{2}} e^{-(m/2kT)\xi^2}, \tag{115}$$

where m and k are constant, ρ and T are density and temperature, respectively. Indeed, all macroscopic variables ρ , \mathbf{u} and E are defined as

$$(\rho, \mathbf{u}, E)(\mathbf{x}, t) = \int_{\mathbb{R}^3} (1, \xi, \xi^2) f(t, \mathbf{x}, \xi) d\xi. \tag{116}$$

The validity of BGK-model is clearly explained in [34].

Starting with (114), Xu and his collaborators successfully developed gas kinetic scheme (GKS) solver [35–39]. A key ingredient is that the explicit solution formula for (114) is used for the numerical flux approximation,

$$f(t, x_{j+\frac{1}{2}}, \xi) = \frac{1}{\epsilon} \int_0^{\Delta t} g(t', x', \xi) e^{-(t-t')/\epsilon} dt' + e^{-t/\epsilon} f_0(x_{j+\frac{1}{2}} - \xi t, \xi), \tag{117}$$

subject to the initial data $f_0(x, \xi)$, where $x' = x_{j+\frac{1}{2}} - \xi(t - t')$. Here just the case of one-dimension is described. The full information contained in (117) provides “exact” expression of flux across the interface $x = x_{j+\frac{1}{2}}$, which is of course consistent with the LW flow solver. We can go to [38] for comprehensive description of the GKS solver. For the gas-kinetic scheme, the gas evolution is a relaxation process from kinetic to hydrodynamic scale through the exponential function, and the corresponding flux is a complicated function of time.

In order to obtain the time derivatives of the flux function at t_n and $t_* = t_n + \Delta t/2$ with the correct physics, the flux function should be approximated as a linear function of time within a time interval. Let's first introduce the following notation,

$$\mathbb{F}_{i+1/2}(W^n, \delta) = \int_{t_n}^{t_n+\delta} F_{i+1/2}(W^n, t) dt = \int_{t_n}^{t_n+\delta} \int \xi f, tmx_{i+1/2}, \xi) d\xi dt.$$

In the time interval $[t_n, t_n + \Delta t]$, the flux is expanded as the following linear form

$$F_{i+1/2}(W^n, t) = F_{i+1/2}^n + \partial_t F_{i+1/2}^n (t - t_n). \tag{118}$$

The coefficients $F_{i+1/2}^n$ and $\partial_t F_{i+1/2}^n$ can be determined as follows,

$$\begin{aligned} F_{i+1/2}(W^n, t_n) \Delta t + \frac{1}{2} \partial_t F_{i+1/2}(W^n, t_n) \Delta t^2 &= \mathbb{F}_{i+1/2}(W^n, \Delta t), \\ \frac{1}{2} F_{i+1/2}(W^n, t_n) \Delta t + \frac{1}{8} \partial_t F_{i+1/2}(W^n, t_n) \Delta t^2 &= \mathbb{F}_{i+1/2}(W^n, \Delta t/2). \end{aligned}$$

By solving the linear system, we have

$$\begin{aligned} F_{i+1/2}(W^n, t_n) &= (4\mathbb{F}_{i+1/2}(W^n, \Delta t/2) - \mathbb{F}_{i+1/2}(W^n, \Delta t)) / \Delta t, \\ \partial_t F_{i+1/2}(W^n, t_n) &= 4(\mathbb{F}_{i+1/2}(W^n, \Delta t) - 2\mathbb{F}_{i+1/2}(W^n, \Delta t/2)) / \Delta t^2. \end{aligned} \tag{119}$$

Similarly, $F_{i+1/2}(W^*, t_*)$, $\partial_t F_{i+1/2}(W^*, t_*)$ for the intermediate state can be constructed. For the two-dimensional computation, the corresponding fluxes in the y -direction can be obtained as well. Readers are referred to [18].

There are huge numbers of references about kinetic solvers, which are beyond the scope of the current paper. We stop to discuss further.

5 Compact reconstruction using the Hermite interpolation

The compactness is a key factor in the design of high order schemes, determining the dissipation of the schemes near discontinuities and the numerical treatment of boundary conditions. With the increase of time-stepping, the width of computational stencils is inevitably expanded for multi-stage methods. Hence it is very important to construct the data in a compact way.

Unlike WENO using the Lagrangian interpolation, we adopt the Hermite-type interpolation using both the average values of physical (conservative or primitive) variables, and the approximate gradient of the solution. Going back to the original GRP, we construct the data over the computation cell $(x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}})$ as

$$u_n(x) = \bar{u}_j^n + \sigma_j^n (x - x_j), \quad x \in (x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}), \tag{120}$$

where the gradient is chosen through the procedure,

$$\sigma_j^n = \frac{1}{\Delta x} \text{minmod} \left(\alpha (u_j^n - u_{j-1}^n), u_{j+\frac{1}{2}}^{n,-} - u_{j-\frac{1}{2}}^{n,-}, \alpha (u_{j+1}^n - u_j^n) \right), \quad \alpha \in [0, 2). \tag{121}$$

Usually, α is chosen to be as large as possible. As $\alpha \in (1, 2)$, $u_n(x)$ behaves as sawtooth and implies that σ_j^n take mostly

$$\sigma_j^n = \frac{1}{\Delta x} (u_{j+\frac{1}{2}}^{n,-} - u_{j-\frac{1}{2}}^{n,-}) = \frac{1}{\Delta x} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \frac{\partial u}{\partial x}(x, t_n - 0) dx \approx \left(\frac{\partial u}{\partial x} \right)_j^n. \tag{122}$$

This is a natural approximation to the gradient. The boundary value $u_{j+\frac{1}{2}}^{n,-}$, as a strong solution along the cell interface $x = x_{j+\frac{1}{2}}$, is calculated from the history,

$$u_{j+\frac{1}{2}}^{n,-} = u_{j+\frac{1}{2}}^{n-1} + \Delta t \left(\frac{\partial u}{\partial t} \right)_{j+\frac{1}{2}}^{n-1}, \tag{123}$$

where $u_{j+\frac{1}{2}}^{n-1}$ and $\left(\frac{\partial u}{\partial t} \right)_{j+\frac{1}{2}}^{n-1}$ are obtained already from the GRP solver, and no extra efforts need making.

Some remarks on (121) are made here, and they can be applied for later high order data interpolations.

- (i) Compared to the classical limiter algorithm, (121) takes (122) in smooth regions, and limit the gradient near discontinuities in order to suppress possible oscillations. This sawtooth-type reconstruction can produce sharper profiles of discontinuities.
- (ii) The piecewise linear data (120) is the embryonic form of Hermite polynomials for high order schemes. Since all values are already given using the GRP solver, no extra effort is made on the calculation of the gradient, unlike in DG methods or other Hermite interpolation [40]. If one might argue the freedom of solution elements, he could regard the current treatment as the Lagrangian interpolation using five points values.
- (iii) For the data (120), five points values are used for the data reconstruction. Essentially, these values are defined in three computational cells, rather than in five cell, so that computational stencils are almost half saved. This is one of key factors achieving compactness.

Now we extend this to the two-stage fourth order method, by reviewing the result in [41]. Given the average \bar{u}_j and the derivative Δu_j of the function $u(x)$ over the cell I_j ,

$$\bar{u}_j = \frac{1}{h} \int_{I_j} u(x, t) dx, \quad \Delta u_j = \frac{1}{h} \int_{I_j} \frac{\partial u}{\partial x}(x, t) dx, \tag{124}$$

we want to construct a polynomial $p(x)$ such that $u_{j+\frac{1}{2},-}$ is its left limiting values at $x = x_{j+\frac{1}{2}}$. Choose three stencils

$$S^{(-1)} = I_{j-1} \cup I_j, \quad S^{(0)} = I_{j-1} \cup I_j \cup I_{j+1}, \quad S^{(1)} = I_j \cup I_{j+1}. \tag{125}$$

On stencil $S^{(0)}$, \bar{u}_{j-1} , \bar{u}_j and \bar{u}_{j+1} are used to construct a polynomial $p^{(0)}$ for the interpolation. Hence at $x_{j+\frac{1}{2}}$, we have

$$u_{j+\frac{1}{2},-}^{(0)} := p^{(0)}\left(x_{j+\frac{1}{2}}\right) = -\frac{1}{6}\bar{u}_{j-1} + \frac{5}{6}\bar{u}_j + \frac{1}{3}\bar{u}_{j+1}. \tag{126}$$

Similarly, $p^{(-1)}$ and $p^{(1)}$ are constructed by using $\bar{u}_j, \bar{u}_{j-1}, \Delta u_{j-1}$ on $S^{(-1)}$ and by using $\bar{u}_j, \bar{u}_{j+1}, \Delta u_{j+1}$ on $S^{(1)}$, respectively,

$$\begin{aligned} u_{j+\frac{1}{2},-}^{(-1)} &:= p^{(-1)}\left(x_{j+\frac{1}{2}}\right) = -\frac{7}{6}\bar{u}_{j-1} + \frac{13}{6}\bar{u}_j - \frac{2h}{3}\Delta u_{j-1}, \\ u_{j+\frac{1}{2},-}^{(1)} &:= p^{(1)}\left(x_{j+\frac{1}{2}}\right) = \frac{1}{6}\bar{u}_j + \frac{5}{6}\bar{u}_{j+1} - \frac{h}{3}\Delta u_{j+1}. \end{aligned} \tag{127}$$

If the solution is smooth on the large stencil $I_{-1} \cup I_0 \cup I_1$, we have

$$\tilde{u}_{j+\frac{1}{2},-} = \frac{1}{120} (-23\bar{u}_{j-1} + 76\bar{u}_j + 67\bar{u}_{j+1} - 9h\Delta u_{j-1} - 21h\Delta u_{j+1}). \tag{128}$$

Thus the linear weights of the three stencils are

$$\gamma^{(-1)} = \frac{9}{80}, \quad \gamma^{(0)} = \frac{29}{80}, \quad \gamma^{(1)} = \frac{21}{40}, \tag{129}$$

which ensure

$$\tilde{\mathbf{u}}_{j+\frac{1}{2},-} = \sum_{r=-1}^1 \gamma^{(r)} \mathbf{u}_{j+\frac{1}{2},-}^{(r)}.$$

The smoothness indicators are defined by

$$\beta^{(r)} = \sum_{l=1}^2 \int_{I_j} h^{2l-1} \left(\frac{d^l}{dx^l} p^{(r)}(x) \right)^2 dx, \quad r = -1, 0, 1, \tag{130}$$

in the same way as in the WENO reconstructions where $p^{(r)}(x)$ is the interpolation polynomial on stencil $S^{(r)}$. Their explicit expressions are

$$\begin{aligned} \beta^{(-1)} &= (-2\bar{u}_{j-1} + 2\bar{u}_j - h\Delta u_{j-1})^2 + \frac{13}{3} (-\bar{u}_{j-1} + \bar{u}_j - h\Delta u_{j-1})^2, \\ \beta^{(0)} &= \frac{1}{4} (-\bar{u}_{j-1} + \bar{u}_{j+1})^2 + \frac{13}{12} (-\bar{u}_{j-1} + 2\bar{u}_j - \bar{u}_{j+1})^2, \\ \beta^{(1)} &= (2\bar{u}_{j+1} - 2\bar{u}_j - h\Delta u_{j+1})^2 + \frac{13}{3} (\bar{u}_{j+1} - \bar{u}_j - h\Delta u_{j+1})^2. \end{aligned} \tag{131}$$

Then we compute the nonlinear weights in the same way as the WENO-Z method does

$$\omega_r^z = \frac{\alpha_r^z}{\sum_l \alpha_l^z}, \quad \alpha_r^z = \gamma^{(r)} \left(1 + \frac{\tau^z}{\beta^{(r)} + \varepsilon} \right), \quad r = -1, 0, 1, \tag{132}$$

where $\tau^z = |\beta^{(1)} - \beta^{(-1)}|$ and ε is a small parameter in order to avoid a zero denominator. Finally we have

$$\mathbf{u}_{j+\frac{1}{2},-} = \sum_{r=-1}^1 \omega_r^z \mathbf{u}_{j+\frac{1}{2},-}^{(r)}. \tag{133}$$

The right interface value $\mathbf{u}_{j-\frac{1}{2},+}$ can be reconstructed in a similar way by mirroring the above procedure with respect to $x_j = \frac{1}{2} (x_{j-\frac{1}{2}} + x_{j+\frac{1}{2}})$.

Since the GRP solver has to use the spatial derivative $(\partial u / \partial x)_{j+\frac{1}{2},\pm}$, we approximate them using the interpolation,

$$\left(\frac{\partial u}{\partial x} \right)_{j+\frac{1}{2},\pm} := \frac{1}{12h} (\bar{u}_{j-1} - 15\bar{u}_j + 15\bar{u}_{j+1} - \bar{u}_{j+2}). \tag{134}$$

It is observed in this interpolation does not need the WENO-type stencil selection procedure.

We define this procedure as *HWENO*, terming a Hermite type interpolation using the WENO interpolation strategy. *GRP4-HWENO5* refers to the two-stage fourth order scheme based on this Hermite type interpolation using the GRP solver.

For two-dimensional cases, we can develop the similar approach over rectangular meshes. See [41, 42]. As far as unstructured meshes are concerned, there still remains space to explore.

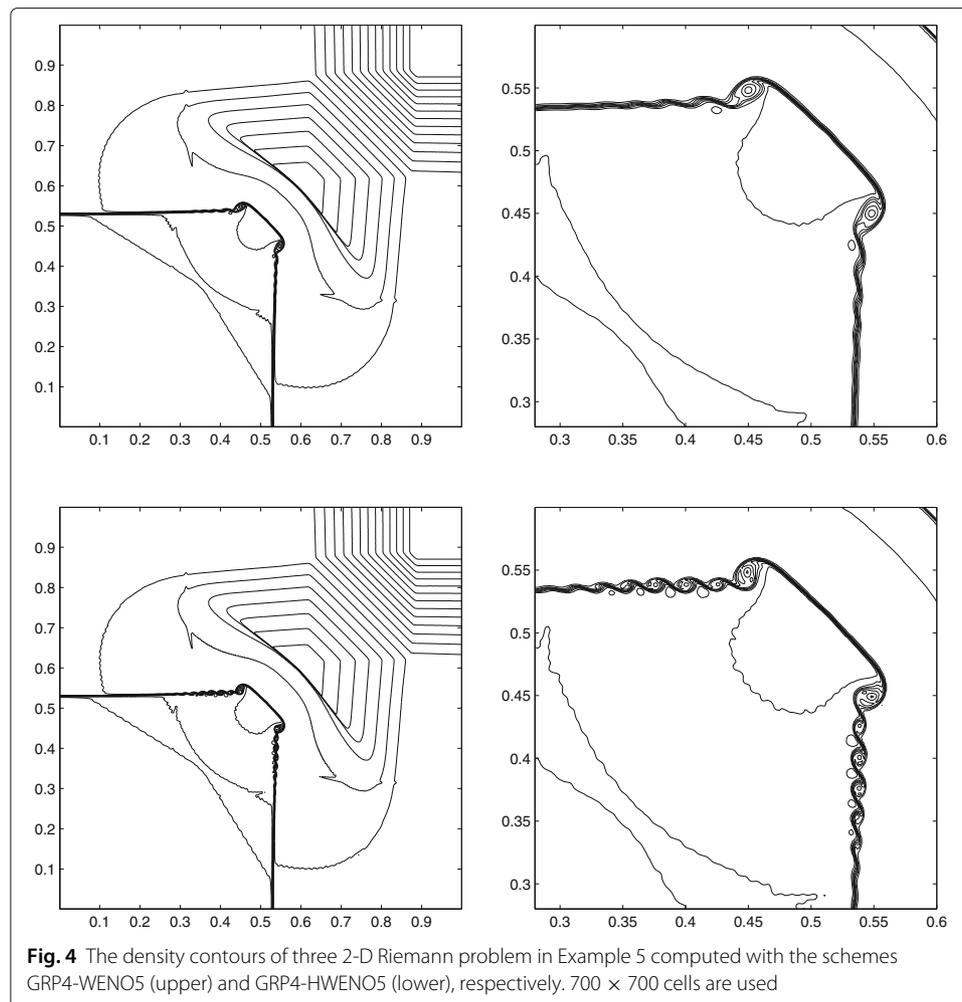
Here we give an example to demonstrate how important the compactness is. We provide an example of two-dimensional Riemann problem taken from [43] involving the interactions of vortex sheets with rarefaction waves. The computation is implemented over the domain $[0, 1] \times [0, 1]$.

$$(\rho, u, v, p)(x, y, 0) = \begin{cases} (1, 0.1, 0.1, 1), & 0.5 < x < 1, 0.5 < y < 1, \\ (0.5197, -0.6259, 0.1, 0.4), & 0 < x < 0.5, 0.5 < y < 1, \\ (0.8, 0.1, 0.1, 0.4), & 0 < x < 0.5, 0 < y < 0.5, \\ (0.5197, 0.1, -0.6259, 0.4), & 0.5 < x < 1, 0 < y < 0.5. \end{cases} \quad (135)$$

The output time is 0.3. The contours of the density and their local enlargements are shown in Fig. 4. We can see that the scheme with the Hermite type reconstruction can resolve more small structures along the vortex sheet.

6 High order boundary conditions

Approximation to boundary conditions may be one of the most challenging issues in CFD. On one hand, mathematical modelings of fluid flows near physical boundaries are diverse. On the other hand, highly nonlinear behaviors and complex boundaries make the approximation notoriously involved.



We briefly illustrate their idea in the finite difference framework by considering the initial boundary value problem (IBVP) for a scalar conservation law

$$\begin{cases} \frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0, & x \in (0, 1), t > 0, \\ u(x, 0) = u_0(x), & x \in (0, 1), \\ u(0, t) = g(t), & t > 0. \end{cases} \tag{136}$$

Assume that $f'(u) > 0$ for all $u \in \mathfrak{R}$ so that $x = 0$ is an inflow boundary and $x = 1$ is an outflow boundary. We equally distribute $M + 1$ points $\{x_j = (j + 1/2)h : j = 0, 1, \dots, M\}$ in the computational domain $(0, 1)$, as shown in Fig. 5. We use u_j to denote the value of u at $x = x_j$ and suppress the index for the time levels. Obviously at the inflow boundary, the solution value at the ghost point x_{-1} is required in order to perform a second-order finite difference at x_0 . For this purpose, a polynomial is constructed in the region around the inflow boundary by using point-wise values u_{-1} , u_0 and u_1 ,

$$L(x) = g_{-1}(x)u_{-1} + g_0(x)u_0 + g_1(x)u_1, \tag{137}$$

from which we want to find the value u_{-1} . The Lagrangian interpolation tells that

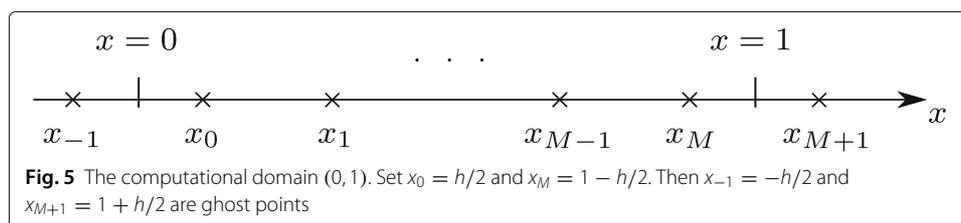
$$\begin{aligned} g_{-1}(x) &= \frac{(x-x_0)(x-x_1)}{2h^2}, \\ g_0(x) &= \frac{(x-x_{-1})(x-x_1)}{-h^2}, \\ g_1(x) &= \frac{(x-x_{-1})(x-x_0)}{2h^2}. \end{aligned} \tag{138}$$

Then u_{-1} can be obtained by solving the linear equation $u(0, t) = L(0)$ where $u(0, t) = g(t)$. At the outflow boundary $x = 1$, we simply use the extrapolation

$$u_{M+1} = 2u_M - u_{M-1} \tag{139}$$

to obtain the value u_{M+1} since the signal goes out of the computational domain at this end.

The extension to high order is highly nontrivial. Let's review the approach developed in [44]. The same as other multi-stage methods (e.g. in [45]), the current two-stage fourth order method needs careful treatment at the intermediate stage for the approximation in order to preserve the accuracy. There are two key points: The construction at ghosts points and the approximation of boundary conditions. The inverse LW approach is applied here [46], but the current treatment is much simpler and easier to be implemented since no higher derivatives need calculating.



6.1 Ghost values

We still consider IBVP (136). Then the values \bar{u}_{-1} , \bar{u}_{-2} , Δu_{-1} and Δu_{-2} , defined over $I_{-1} = (-h, 0)$ and $I_{-2} = (-2h, -h)$, ghost cells, are needed in the reconstruction procedure for the values indexed by $j = 0$ and $j = 1$. To obtain the values mentioned above, a cubic polynomial,

$$p(x) = \alpha_3 x^3 + \alpha_2 x^2 + \alpha_1 x + \alpha_0, \tag{140}$$

is constructed over $I_{-2} \cup I_{-1} \cup I_0 \cup I_1 = (-2h, 2h)$ to interpolate the solution $u(x, t^n)$ such that

$$\frac{1}{h} \int_{I_i} p(x) dx = \bar{u}_i, \quad i = -2, -1, 0, 1. \tag{141}$$

With the constraints (141) into (140), we determine the coefficients $\alpha_0, \alpha_1, \alpha_2$ and α_3 as

$$\begin{aligned} \alpha_3 &= \frac{\bar{u}_1 - 3\bar{u}_0 + 3\bar{u}_{-1} - \bar{u}_{-2}}{6h^3}, & \alpha_2 &= \frac{\bar{u}_1 - \bar{u}_0 - \bar{u}_{-1} + \bar{u}_{-2}}{4h^2}, \\ \alpha_1 &= \frac{-\bar{u}_1 + 15\bar{u}_0 - 15\bar{u}_{-1} + \bar{u}_{-2}}{12h}, & \alpha_0 &= \frac{-\bar{u}_1 + 7\bar{u}_0 + 7\bar{u}_{-1} - \bar{u}_{-2}}{12}, \end{aligned} \tag{142}$$

in which \bar{u}_{-1} and \bar{u}_{-2} are yet to be determined and they are obtained by evaluating $p(0)$ and $p'(0)$ at the boundary $x = 0$,

$$\begin{aligned} p(0) &= \frac{1}{12} (-\bar{u}_1 + 7\bar{u}_0 + 7\bar{u}_{-1} - \bar{u}_{-2}) = g(t) + \mathcal{O}(h^4), \\ p'(0) &= \frac{1}{12h} (-\bar{u}_1 + 15\bar{u}_0 - 15\bar{u}_{-1} + \bar{u}_{-2}) = -f'(g(t))^{-1} g'(t) + \mathcal{O}(h^3). \end{aligned} \tag{143}$$

Solving (143) in terms of \bar{u}_{-1} and \bar{u}_{-2} yields (by ignoring high order terms)

$$\begin{aligned} \bar{u}_{-1} &= \frac{1}{4} (-6g + 6 h f'(g)^{-1} g' + 11\bar{u}_0 - \bar{u}_1), \\ \bar{u}_{-2} &= \frac{1}{4} (-90g + 42 h f'(g)^{-1} g' + 105\bar{u}_0 - 11\bar{u}_1). \end{aligned} \tag{144}$$

Substituting (144) into (142), in turn, gives us the explicit expressions of $\alpha_i, i = 0, \dots, 3$, and then the expression of $p(x)$. Therefore we have (by ignoring high order terms)

$$\begin{aligned} \Delta u_{-1} &= \frac{p(0) - p(-h)}{h} \\ &= \frac{1}{8h} (66g - 34 h f'(g)^{-1} g' - 73\bar{u}_0 + 7\bar{u}_1), \\ \Delta u_{-2} &= \frac{p(-h) - p(-2h)}{h} \\ &= \frac{1}{8h} (294g - 118 h f'(g)^{-1} g' - 331\bar{u}_0 + 37\bar{u}_1). \end{aligned} \tag{145}$$

Thus (144) and (145) together provide the values in the ghost cells I_{-1} and I_{-2} . Not that in (143) the inverse LW approach is used,

As there are discontinuities close to the inflow boundary, a WENO-type stencil selecting procedure can be applied. Assume that there is a discontinuity in either I_0 or I_1 , we shorten the stencil cell by cell. Denote the stencils by

$$S^{(2)} = \{I_{-2}, I_{-1}, I_0, I_1\}, \quad S^{(1)} = \{I_{-2}, I_{-1}, I_0\}, \quad S^{(0)} = \{I_{-2}, I_{-1}\}. \tag{146}$$

Denote by $p^{(r)}(x)$ the interpolation polynomial on $S^{(r)}, r = 0, 1, 2$, just as the polynomial $p(x)$ constructed before. Then define

$$\begin{aligned} \bar{u}_{-1}^{(r)} &= \frac{1}{h} \int_{I_{-1}} p^{(r)}(x) dx, & \bar{u}_{-2}^{(r)} &= \frac{1}{h} \int_{I_{-2}} p^{(r)}(x) dx, \\ \Delta u_{-1}^{(r)} &= \frac{1}{h} (p^{(r)}(0) - p^{(r)}(-h)), & \Delta u_{-2}^{(r)} &= \frac{1}{h} (p^{(r)}(-h) - p^{(r)}(-2h)). \end{aligned} \tag{147}$$

The expressions of $\bar{u}_{-1}^{(r)}, \bar{u}_{-2}^{(r)}, \Delta u_{-1}^{(r)}$ and $\Delta u_{-2}^{(r)}$ for $r = 0, 1, 2$ will be listed in 9.1.

The smoothness indicators are defined in the same way as for the classical WENO interpolation, and the values are given

$$\begin{aligned} \bar{u}_{-1} &= \sum_{r=0}^2 \omega^{(r)} \bar{u}_{-1}^{(r)}, & \bar{u}_{-2} &= \sum_{r=0}^2 \omega^{(r)} \bar{u}_{-2}^{(r)}, \\ \Delta u_{-1} &= \sum_{r=0}^2 \omega^{(r)} \Delta u_{-1}^{(r)}, & \Delta u_{-2} &= \sum_{r=0}^2 \omega^{(r)} \Delta u_{-2}^{(r)}, \end{aligned} \tag{148}$$

where the linear weights of each stencil are

$$\begin{aligned} \alpha^{(r)} &= \frac{d^{(r)}}{(\varepsilon + \beta^{(r)})^2}, & \omega^{(r)} &= \frac{\alpha^{(r)}}{\sum_{l=0}^2 \alpha^{(l)}} \\ d^{(0)} &= h^2, & d^{(1)} &= h, & d^{(2)} &= 1 - d^{(0)} - d^{(1)}. \end{aligned} \tag{149}$$

6.2 Inflow boundary condition treatment at intermediate stages

The same as in other multi-stage temporal discretization [47, 48], the direct use of exact boundary conditions at intermediate stages in the process of multi-stage approaches will cause the lose of the numerical accuracy. In order to offset such a defect, our strategy is made as follows. We first focus on the leftmost control volume I_0 and write out the solution advancing formula,

$$\begin{aligned} \bar{u}_0^{n+1} &= \bar{u}_0^n - \frac{k}{h} \left[f_{\frac{1}{2}}^{4th} - f_{-\frac{1}{2}}^{4th} \right] \\ &= \bar{u}_0^n - \frac{1}{h} \left\{ k \left[f \left(u_{\frac{1}{2}}^n \right) - f \left(u_{-\frac{1}{2}}^n \right) \right] \right. \\ &\quad + \frac{k^2}{6} \left[f' \left(u_{\frac{1}{2}}^n \right) \left(\frac{\partial u}{\partial t} \right)_{\frac{1}{2}}^n - f' \left(u_{-\frac{1}{2}}^n \right) \left(\frac{\partial u}{\partial t} \right)_{-\frac{1}{2}}^n \right] \\ &\quad \left. + \frac{k^2}{3} \left[f' \left(u_{\frac{1}{2}}^{n+\frac{1}{2}} \right) \left(\frac{\partial u}{\partial t} \right)_{\frac{1}{2}}^{n+\frac{1}{2}} - f' \left(u_{-\frac{1}{2}}^{n+\frac{1}{2}} \right) \left(\frac{\partial u}{\partial t} \right)_{-\frac{1}{2}}^{n+\frac{1}{2}} \right] \right\}. \end{aligned} \tag{150}$$

Using the governing Eq. 136 to replace the temporal derivatives by the corresponding spatial ones, we obtain

$$\begin{aligned} \bar{u}_0^{n+1} &= \bar{u}_0^n - \frac{1}{h} \left\{ k \left[f \left(u_{\frac{1}{2}}^n \right) - f \left(u_{-\frac{1}{2}}^n \right) \right] \right. \\ &\quad - \frac{k^2}{6} \left[\left(f' \left(u_{\frac{1}{2}}^n \right) \right)^2 \left(\frac{\partial u}{\partial x} \right)_{\frac{1}{2}}^n - \left(f' \left(u_{-\frac{1}{2}}^n \right) \right)^2 \left(\frac{\partial u}{\partial x} \right)_{-\frac{1}{2}}^n \right] \\ &\quad \left. - \frac{k^2}{3} \left[\left(f' \left(u_{\frac{1}{2}}^{n+\frac{1}{2}} \right) \right)^2 \left(\frac{\partial u}{\partial x} \right)_{\frac{1}{2}}^{n+\frac{1}{2}} - \left(f' \left(u_{-\frac{1}{2}}^{n+\frac{1}{2}} \right) \right)^2 \left(\frac{\partial u}{\partial x} \right)_{-\frac{1}{2}}^{n+\frac{1}{2}} \right] \right\}. \end{aligned} \tag{151}$$

The difficulty results from the presence of $(\partial u / \partial x)_{\frac{1}{2}}^{n+\frac{1}{2}}$ and $(\partial u / \partial x)_{-\frac{1}{2}}^{n+\frac{1}{2}}$ evaluated at the intermediate stage $t = t^{n+\frac{1}{2}}$. In order to restore the fourth-order accuracy of the two-stage fourth-order scheme, we use

$$\begin{aligned} \left(\frac{\partial u}{\partial x} \right)_{-\frac{1}{2}}^{n+\frac{1}{2}} &= - \left(f' \left(g \left(t^{n+\frac{1}{2}} \right) \right) \right)^{-1} (g')^{n+\frac{1}{2}}, \\ \left(\frac{\partial u}{\partial x} \right)_{\frac{1}{2}}^{n+\frac{1}{2}} &= \frac{1}{48h} \left[-49\bar{u}_0^{n+\frac{1}{2}} + 59\bar{u}_1^{n+\frac{1}{2}} - 4\bar{u}_2^{n+\frac{1}{2}} \right. \\ &\quad \left. - 6g^{n+\frac{1}{2}} + 6h \left(f' \left(g \left(t^{n+\frac{1}{2}} \right) \right) \right)^{-1} (g')^{n+\frac{1}{2}} \right], \end{aligned} \tag{152}$$

where the exact boundary values $g \left(t^{n+\frac{1}{2}} \right)$ and $g' \left(t^{n+\frac{1}{2}} \right)$ are replaced by

$$\begin{aligned} g^{n+\frac{1}{2}} &= g \left(t^{n+\frac{1}{2}} \right) - \frac{k^3}{48} g''' \left(t^{n+\frac{1}{2}} \right), \\ (g')^{n+\frac{1}{2}} &= g' \left(t^{n+\frac{1}{2}} \right). \end{aligned} \tag{153}$$

The detailed analysis is given in [44].

6.3 Outflow boundary condition

We set $x_{M+\frac{1}{2}} = 1$ as an outflow boundary, at which no boundary condition is prescribed theoretically. Numerically, we have to set required values \bar{u}_{M+1} , \bar{u}_{M+2} , Δu_{M+1} and Δu_{M+2} in ghost cells. Since the signal propagates out of the computational domain through the boundary $x = 1$, the extrapolation can be used to construct the data in the ghost cells I_{M+1} and I_{M+2} . A cubic polynomial is constructed in order to achieve the fourth-order accuracy,

$$\begin{aligned} q(x) = & \frac{\bar{u}_M - 3\bar{u}_{M-1} + 3\bar{u}_{M-2} - \bar{u}_{M-3}}{6h^3} (x-1)^3 \\ & + \frac{5\bar{u}_M - 13\bar{u}_{M-1} + 11\bar{u}_{M-2} - 3\bar{u}_{M-3}}{4h^2} (x-1)^2 \\ & + \frac{35\bar{u}_M - 69\bar{u}_{M-1} + 45\bar{u}_{M-2} - 11\bar{u}_{M-3}}{12h} (x-1) \\ & + \frac{25\bar{u}_M - 23\bar{u}_{M-1} + 13\bar{u}_{M-2} - 3\bar{u}_{M-3}}{12}. \end{aligned} \quad (154)$$

This gives the values

$$\begin{aligned} \bar{u}_{M+1} &= 4\bar{u}_M - 6\bar{u}_{M-1} + 4\bar{u}_{M-2} - \bar{u}_{M-3}, \\ \bar{u}_{M+2} &= 10\bar{u}_M - 20\bar{u}_{M-1} + 15\bar{u}_{M-2} - 4\bar{u}_{M-3}, \\ \Delta u_{M+1} &= \frac{26\bar{u}_M - 57\bar{u}_{M-1} + 42\bar{u}_{M-2} - 11\bar{u}_{M-3}}{6}, \\ \Delta u_{M+2} &= \frac{47\bar{u}_M - 114\bar{u}_{M-1} + 93\bar{u}_{M-2} - 26\bar{u}_{M-3}}{6}. \end{aligned} \quad (155)$$

If there is a discontinuity in either I_{M-3} , I_{M-2} , I_{M-1} or I_M , a WENO-type stencil selection can be applied.

6.4 Hyperbolic systems

At moment, the boundary treatment for systems of hyperbolic balance laws is basically achieved through the diagonalization process. Then we distinguish various cases such as the solid boundary condition, inflow and outflow boundary conditions for practical applications. Details can be found in [44].

7 Computational performance

In our series of papers, we have demonstrated the performance of current temporal-spatially coupled algorithms through many challenging benchmark problems, particularly in [49] using the GKS solver. Here I would like to give some remarks in terms of computational efficiency, robustness and fidelity.

7.1 Computational efficiency

Computational efficiency is always an important issue for practical engineering problems. We have tested and compared the efficiency with the popular WENO algorithm in [18] and with DG in [19].

Specifically, in [18] we evaluate the computational costs of the WENO-type reconstruction and the flux evaluation quantitatively. The time for each reconstruction is denoted by T_R , the time for second-order gas-kinetic solver is T_{2nd} , and the time for third-order flux solver is T_{3rd} . According to the data provided in ([18], Table 1, Page 203) we can estimate the time used for the computations of flux and reconstruction with the following relations,

$$T_R + 2T_{2nd} = 0.84287s,$$

$$T_R + 2T_{3rd} = 1.38178s,$$

$$2T_R + 12T_{2nd} = 2.20566s,$$

where the estimation is based on the characteristic variable reconstruction and each flux is shared by two cells. Thus, the time for reconstruction is $T_R = 0.71289s$, the time for second-order gas-kinetic flux solver is $T_{2nd} = 0.06499s$ and the time for third-order gas-kinetic flux solver is $T_{3rd} = 0.33445s$. For classical fourth-order Runge-Kutta schemes, the computational time for four spatial reconstruction alone will become much higher than the fourth-order gas-kinetic scheme for the update of each cell averaged values

$$4T_R = 2.85156s > 2T_R + 12T_{2nd} = 2.20566s.$$

Similar estimation can be done for the conservative variables reconstruction. Even without counting on the cost of the flux evaluation in the traditional fourth-order Runge-Kutta method, such as those commonly used with the Lax-Friedrichs flux, the current fourth-order time stepping method is still more efficient than the classical methods.

The efficiency is mainly attributed to the half of reconstruction steps compared to that for the same order of other line methods. This is further verified in the framework of DG methods [19]. In Table 2 and Fig. 6 through simulating shock-vortex interaction problem, demonstrating that nearly 55% CPU time can be saved using the GRP-DG(s2p3) method compared to the same order SSP RKDG(s5p3) method. This result meets the expectation well as compared to the RKDG(s5p3) method which needs five stages of evaluating DoFs and performing reconstruction to achieve fourth order, while the GRP-DG(s2p3) method only takes two stages to provide totally comparable results.

7.2 Robustness

The robustness is always an important indicator for a practical numerical method. In the framework of multistage multi-derivative algorithm, the strong stability preserving (SSP) property was taken over to show the stability [9]. However, SSP seems not work when the Lax-Wendorff type flow solvers are taken as the building block. Therefore new stability framework is worth exploring in the future.

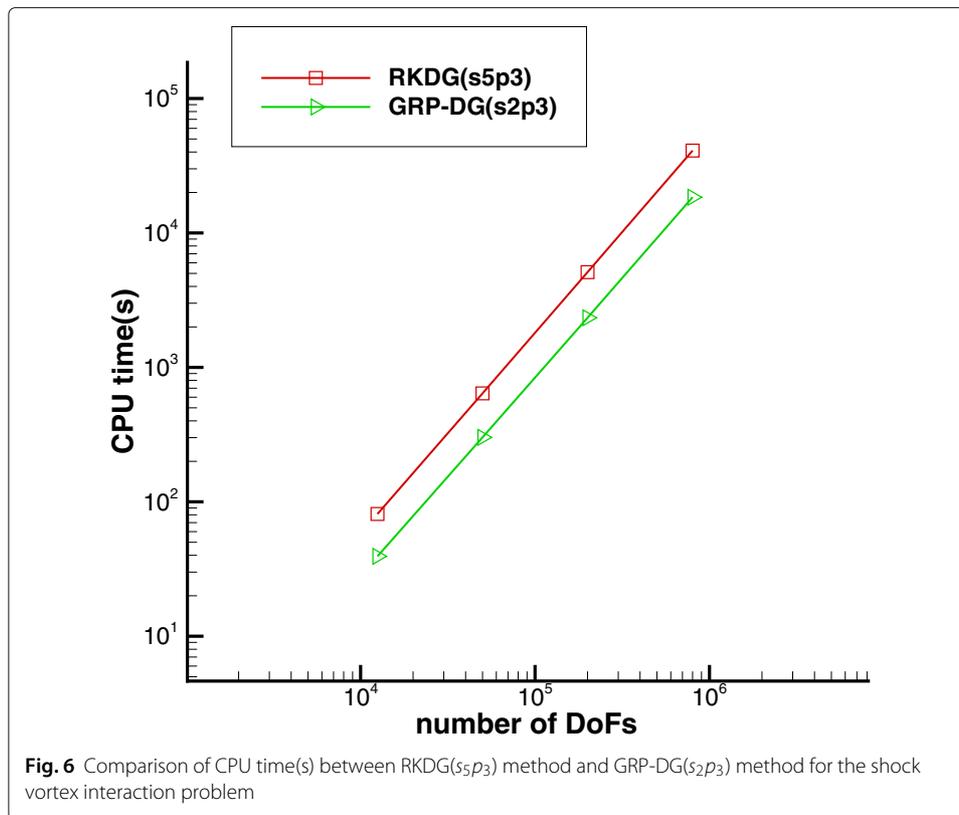
Nevertheless, in practice, the current two-stage fourth order accurate algorithm has the same stability as the second order version: the Courant number is taken above 0.5 except extreme cases such as the large density ration problem. Empirically, the current “2 ⊙ 2” algorithm is more robust than other multi-stage methods.

7.3 Fidelity

In the community of CFD, the *fidelity* is termed for a numerical simulation of very complex problems using a specific algorithm. Since there is no reliable mathematical theory in general supporting the current CFD simulation, the verification of high fidelity appears very valuable. We pursue such studies in the whole process of the current algorithm.

Table 2 Comparison of CPU time(s) between RKDG and GRP-DG methods for shock vortex interaction problem

Methods	50 × 25	100 × 50	200 × 100	400 × 200
GRP-DG(s ₂ p ₃)	39.3	301.6	2339.4	18443.4
RKDG(s ₅ p ₃)	81.2	640.7	5109.3	40999.6



For example, we resolve the associated GRP analytically and use the GRP solution for Hermite-type data reconstruction. In [16] we elaborate the so-called large density ratio problem [50] using the GRP solver. When the current “2 ⊙ 2” algorithm is adopted, quite few grids are needed to obtain satisfactory results, as shown in Fig. 7.

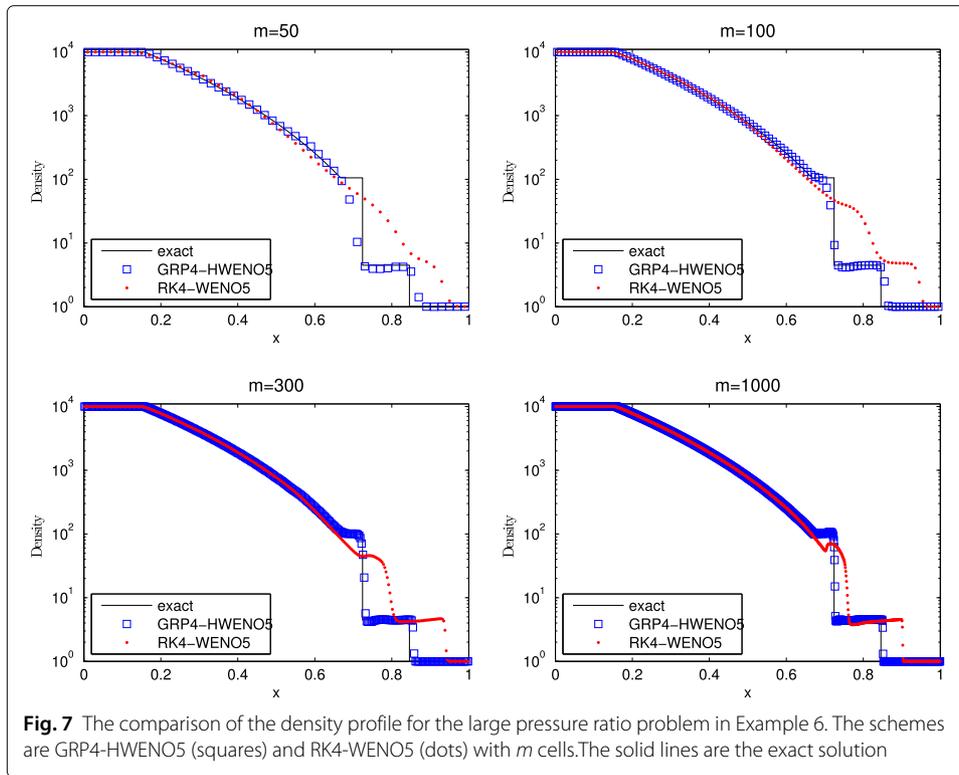
Also readers are recommended to test the benchmark problems in [49], for which all simulations are made in the “2 ⊙ 2” framework using the GKS solver.

8 Conclusion with prospective discussions

It is natural to require the temporal-spatial coupling of a numerical method when simulating compressible fluid flows, for which the GRP solver and the GKS solver are reviewed briefly as the representatives of Lax-Wendroff type flow solvers. The direct embedding into any numerical frameworks, such as the finite volume/DG framework, already results in favorable second order numerical schemes. Interested readers can refer to papers by Jiequan Li and his collaborators for GRP methods (www.ams.org/mathscinet, scholar.google.com or researchgate.net).

As to the “2 ⊙ 2” algorithm itself, it is just at the beginning stage, and many issues are awaiting for our study. Below are some immediate doable problems.

- P_1 What is a good framework for stability analysis?
- P_2 It is valuable to compare and develop multi-stage two-derivative algorithms with arbitrary order of accuracy.
- P_3 Develop implicit “2 ⊙ 2” algorithm with various applications such as detonation simulation.



P_4 Apply this algorithm for the simulation of turbulence flows and other engineering problems.

You are welcome to join this new branch of high order numerical methods for CFD.

9 The interpolation results in subsection 5

This appendix is dedicated to list the interpolation results in Section 5. Recall that we assume $x = 0$ and $x = 1$ are the inflow and outflow boundaries for the IBVP (136) of the one-dimensional scalar conservation laws, respectively. The stencils are denoted in (146).

9.1 Cell averages and cell differences

The reconstructed average of u in I_{-1} and I_{-2} on those stencils are:

$$\begin{aligned}
 \bar{u}_{-1}^{(2)} &= \frac{1}{4} (-6g + 6hf'(g)^{-1}g' + 11\bar{u}_0 - \bar{u}_1), \\
 \bar{u}_{-1}^{(1)} &= hf'(g)^{-1}g' + \bar{u}_0, \\
 \bar{u}_{-1}^{(0)} &= g + \frac{1}{2}hf'(g)^{-1}g', \\
 \bar{u}_{-2}^{(2)} &= \frac{1}{4} (-90g + 42hf'(g)^{-1}g' + 105\bar{u}_0 - 11\bar{u}_1), \\
 \bar{u}_{-2}^{(1)} &= -6gh + 5f'(g)^{-1}g' + 7\bar{u}_0, \\
 \bar{u}_{-2}^{(0)} &= g + \frac{3}{2}hf'(g)^{-1}g'.
 \end{aligned}
 \tag{156}$$

The reconstructed x -difference of u in I_{-1} and I_{-2} on those stencils are:

$$\begin{aligned}
\Delta u_{-1}^{(2)} &= \frac{1}{8h} (66g - 34 h f'(g)^{-1} g' - 73\bar{u}_0 + 7\bar{u}_1), \\
\Delta u_{-1}^{(1)} &= \frac{1}{2h} (6g - 5 h f'(g)^{-1} g' - 6\bar{u}_0), \\
\Delta u_{-1}^{(0)} &= -f'(g)^{-1} g', \\
\Delta u_{-2}^{(2)} &= \frac{1}{8h} (294g - 118 h f'(g)^{-1} g' - 331\bar{u}_0 + 37\bar{u}_1), \\
\Delta u_{-2}^{(1)} &= \frac{1}{2h} (18g - 11 h f'(g)^{-1} g' - 18\bar{u}_0), \\
\Delta u_{-2}^{(0)} &= -f'(g)^{-1} g'.
\end{aligned} \tag{157}$$

9.2 Smoothness indicators

The smoothness indicators on these stencils are listed as follows,

$$\begin{aligned}
\beta^{(2)} &= \frac{1}{80} \left[66516g^2 + 9444(hf'(g)^{-1}g')^2 - 56348f'(g)^{-1}g'h\bar{u}_0 \right. \\
&\quad + 85929\bar{u}_0^2 + 6644f'(g)^{-1}g'h\bar{u}_1 - 20694\bar{u}_0\bar{u}_1 + 1281\bar{u}_1^2 \\
&\quad \left. + 12g(4142f'(g)^{-1}g'h - 12597\bar{u}_0 + 1511\bar{u}_1) \right], \\
\beta^{(1)} &= 48g^2 + 54ghf'(g)^{-1}g' + 16(hf'(g)^{-1}g')^2 \\
&\quad - 96g\bar{u}_0 + 48\bar{u}_0^2 - 54hf'(g)^{-1}g'\bar{u}_0, \\
\beta^{(0)} &= (hf'(g)^{-1}g')^2.
\end{aligned} \tag{158}$$

Acknowledgements

This review paper is based on the joint works with Matania Ben-Artzi, Jian Cheng, Zhifang Du, Xin Lei, Liang Pan, Jin Qi, Yue Wang and Kun Xu, to whom the author expresses his deep thanks.

Funding

This work is supported by NSFC (nos. 11771054, 91852207) and Foundation of LCP.

Availability of data and materials

All data and materials are available from Jiequan Li's research.

Authors' contributions

This is a single author work. The author read and approved the final manuscript.

Competing interests

The author declares that he/she has no competing interests.

Publisher's Note

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

Received: 21 January 2019 Accepted: 21 January 2019

Published online: 21 February 2019

References

- Jiang GS, Shu C-W (1996) Efficient implementation of weighted ENO schemes. *J Comput Phys* 126:202–228
- Cockburn B, Shu C-W (1989) TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. II. general framework. *Math Comput* 52:411–435
- Evans L (2002) *Partial Differential Equations*. Graduate Studies in Mathematics, 19, AMS
- Li J, Du Z (2016) A two-stage fourth order temporal discretization for on the Lax-Wendroff type flow solvers, I. hyperbolic conservation laws. *SIAM J Sci Comput* 38:A3046–A3069
- Butcher JC (1964) On Runge-Kutta processes of high order. *J Aust Math Soc* 4:179–194
- Feng K, Qin M (2010) *Symplectic Geometric Algorithms for Hamiltonian Systems*. Springer, Springer
- Chan PK, Tsai R (2010) On explicit two-derivative Runge-Kutta methods. *Numer Algorithm* 53:171–194
- Christlieb AJ, Gottlieb S, Grant Z, Seal DC (2016) Explicit strong stability preserving multistage two-derivative time-stepping schemes. *J Sci Comput* 68:914–942
- Grant Z, Gottlieb S, Seal DC (2018) A strong stability preserving analysis for multistage two-derivative time-stepping schemes based on Taylor series conditions. [arXiv:1804.10526](https://arxiv.org/abs/1804.10526)
- Seal DC, Guclu Y, Christlieb AJ (2014) High-order multiderivative time integrators for hyperbolic conservation laws. *J Sci Comput* 60:101–140

11. Tsai AYJ, Chan RPK, Wang S (2014) Two-derivative Runge-Kutta methods for PDEs using a novel discretization approach. *Numer Algorith* 65:687–703
12. Lax P, Wendroff B (1960) Systems of conservation laws. *Commun Pur Appl Math* XIII:217–237
13. Harten A (1983) High resolution schemes for hyperbolic conservation laws. *J Comput Phys* 49:357–393
14. Lei X, Li J (2019) Transversal effects and genuine multidimensionality of high order numerical schemes for compressible fluid flows. accepted
15. Li J, Chen G (2006) The generalized Riemann problem method for the shallow water equations with bottom topography. *Int J Numer Methods Eng* 65:834–862
16. Li J, Wang Y (2017) Thermodynamical effects and high resolution methods for compressible fluid flows. *J Comput Phys* 343:340–354
17. Christlieb AJ, Gottlieb S, Grant Z, Seal DC (2016) Explicit strong stability preserving multistage two-derivative time-stepping schemes. *J Sci Comput* 68:914–942
18. Pan L, Xu K, Li Q, Li J (2017) An efficient and accurate two-stage fourth-order gas-kinetic scheme for the Euler and Navier-Stokes equations. *J Comput Phys* 326:197–221
19. Cheng J, Du Z, Lei X, Wang Y, Li J (2019) A two-stage fourth-order discontinuous Galerkin method based on the GRP solver for the compressible Euler equations. *Computer and Fluids* in revision. <https://doi.org/10.1016/j.compfluid.2019.01.025>
20. Landau L, Lifshitz E (1987) *Fluid Dynamics: Course of Theoretical Physics, Vol. 6. A Division of Reed Educational and Professional Publishing Ltd, Butterworth-Heinemann*
21. Ben-Artzi M, Falcovitz J (1984) A second-order Godunov-type scheme for compressible fluid dynamics. *J Comput Phys* 55:1–32
22. Ben-Artzi M, Falcovitz J (2003) *Generalized Riemann Problems in Computational Fluid Dynamics*. Cambridge University Press, Cambridge
23. Ben-Artzi M, Li J, Warnecke G (2006) A direct Eulerian GRP scheme for compressible fluid flows. *J Comput Phys* 218:19–43
24. Ben-Artzi M, Li J (2007) Hyperbolic conservation laws: Riemann invariants and the generalized Riemann problem. *Numer Math* 106:369–425
25. Qian J, Li J, Wang S (2014) The generalized Riemann problems for compressible fluid flows: Towards high order. *J Comput Phys* 259:358–389
26. Balsara DS, Li J, Montecinos GI (2018) An efficient, second order accurate, universal generalized Riemann problem solver based on the HLLI Riemann solver. *J Comput Phys* 375:1238–1269
27. Ben-Artzi M, Falcovitz J (1986) An upwind second-order scheme for compressible duct flows. *SIAM J Sci Stat Comput* 7:744–768
28. Titarev VA, Toro EF (2002) ADER: Arbitrary high order Godunov approach. *J Sci Comput* 17:609–618
29. Li J, Wang Y (2018) The GRP scheme for real materials. Preprint
30. Bouchut F, Bourdarias Ch, Perthame B (1996) A MUSCL method satisfying all the numerical entropy inequalities. *Math Comput* 65:1439–1461
31. LeVeque RJ (2002) *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press
32. Qi J (2017) The numerical modeling and applications of generalized Riemann problem for two-dimensional Euler equations. PhD Thesis, Beijing Normal University
33. Ohwada T (2002) On the construction of kinetic schemes. *J Comput Phys* 177:156–175
34. Bhatnagar PL, Gross EP, Krook M (1954) A model for collision processes in gases I: Small amplitude processes in charged and neutral one-component systems. *Phys Rev* 94:55
35. Prendergast KH, Xu K (1993) Numerical hydrodynamics from gas-kinetic theory. *J Comput Phys* 109:53–66
36. Xu K (2001) A gas-kinetic BGK scheme for the Navier-Stokes equations and its connection with artificial dissipation and Godunov method. *J Comput Phys* 171:289–335
37. Xu K (2010) A unified gas-kinetic scheme for continuum and rarefied flows. *J Comput Phys* 229:7747–7764
38. Xu K (2015) *Direct Modeling for Computational Fluid Dynamics construction and application of unified gas-kinetic schemes*. World Scientific
39. Xu K, Prendergast K (1994) Numerical Navier-Stokes solutions from gas kinetic theory. *J Comput Phys* 114:9–17
40. Qiu J, Shu C-W (2004) Hermite weno schemes and their application as limiters for Runge-Kutta discontinuous Galerkin method: one-dimensional case. *J Comput Phys* 193:115–135
41. Du Z, Li J (2018) A Hermite weno reconstruction for fourth order temporal accurate schemes based on the GRP solver for hyperbolic conservation laws. *J Comput Phys* 355:385–396
42. Ji X, Pan L, Shyy W, Xu K (2018) A compact fourth-order gas-kinetic scheme for the Euler and Navier-Stokes equations. *J Comput Phys* 372:446–472
43. Han E, Li J, Tang H (2011) Accuracy of the adaptive GRP scheme and the simulation of 2-d Riemann problems for compressible Euler equations. *Commun Comput Phys* 10:577–609
44. Du Z, Li J (2018) A two-stage fourth order time-accurate discretization for Lax-Wendroff type flow solvers II. high order numerical boundary conditions. *J Comput Phys* 369:125–147
45. Carpenter M, Gottlieb D, Abarbanel S, Don W-S (1995) The theoretical accuracy of Runge-Kutta time discretizations for the initial boundary value problem: a study of the boundary error. *SIAM J Sci Comput* 16:1241–1252
46. Tan S, Shu C-W (2010) Inverse Lax-Wendroff procedure for numerical boundary conditions of conservation laws. *J Comput Phys* 229:8144–8166
47. Carpenter M, Gottlieb D, Abarbanel S, Don W-S (1989) The theoretical accuracy of Runge-Kutta time discretizations for the initial boundary value problem: a study of the boundary error. *Math Comput* 52:411–435
48. Pathria D (1997) The correct formulation of intermediate boundary conditions for Runge-Kutta time integration of initial boundary value problems. *SIAM J Sci Comput* 18:1255–1266
49. Pan L, Li J, Xu K (2017) A few benchmark test cases for higher-order Euler solvers. *Numer Math Theory Methods Appl* 10:711–736
50. Tang H, Liu T (2006) Note on the conservative schemes for the Euler equations. *J Comput Phys* 218:451–459