SCIENCE CHINA
Mathematics

• ARTICLES •

March 2025 Vol. 68 No. 3: 729–760
https://doi.org/10.1007/s11425-023-2268-0

High-order finite volume methods for solving compressible multicomponent flows

Feng Zheng[1] & Jianxian Qiu[2,*]

[1]*School of Mathematics and Statistics, Fujian Key Laboratory of Mathematical Analysis and Application, Fujian Normal University, Fuzhou* 350117*, China;*
[2]*School of Mathematical Sciences and Fujian Provincial Key Laboratory of Mathematical Modeling & High-Performance Scientific Computing, Xiamen University, Xiamen* 361005*, China*

*Email: fzbz*200808-31@163*.com, jxqiu@xmu.edu.cn*

Received August 20, 2023; accepted February 19, 2024; published online September 25, 2024

Abstract In this paper, we propose a high-order finite volume method for solving multicomponent fluid problems. Our method couples the quasi-conservative form with the reconstruction of conservative variables in a characteristic manner. The source term and numerical fluxes are carefully designed to maintain the pressure and velocity equilibrium for the interface-only problem and preserve the equilibrium of physical parameters in a single-component fluid. These ingredients enable our scheme to achieve both high-order accuracy in the smooth region and the high resolution in the discontinuity region of the solution. Extensive numerical tests are performed to verify the high resolution and accuracy of the scheme.

Keywords finite volume, high-order, multicomponent flows

MSC(2020) 35L65, 65M08, 76M12

Citation: Zheng F, Qiu J X. High-order finite volume methods for solving compressible multicomponent flows. Sci China Math, 2025, 68: 729–760, https://doi.org/10.1007/s11425-023-2268-0

1 Introduction

Numerical simulation for multicomponent flows is an important branch in computational fluid dynamics and plays a significant role in aerospace, chemical engineering, biomedical engineering, hydraulic engineering, and other fields. There are three main approaches to this problem: the Lagrangian method, the arbitrary Lagrangian-Eulerian (ALE) method, and the Eulerian method.

The Lagrangian method was first introduced by Neumann and Richtmyer [52]. In the Lagrangian method, the grid can dynamically change with the fluid evolution during the computation. Consequently, we can achieve a sharp interface by tracking the nodes at the interface. However, if the interface undergoes significant deformation, this method may cause grid distortion and entanglement, eventually leading to code crashes [8].

To overcome such problems in the Lagrangian method, Hirt et al. [22] proposed the ALE method. This method allows for artificial control of grid velocity and optimization of the grid to solve large deformation problems. Due to its excellent performance, the ALE method has been continuously improved, for

* Corresponding author

example, for multi-material flows [6, 7, 32], for hydrodynamics and magnetohydrodynamics (MHD) [16], and for climate simulation [48]. However, if the deformation of multi-material problems is further increased, topological changes may occur, and the ALE method may still suffer from difficulties.

Compared with the Lagrangian method and ALE method, the Eulerian method is designed on a fixed grid, so there is no need to worry about mesh deformation. The Eulerian method can be divided into two categories: one is interface-tracking methods, for example, coupling with the cut-cell method [12, 23, 28, 31, 36], and coupling with the ghost fluid method (GFM) [15, 29, 54]; the other one is interface-capturing methods, for example, solving the four equations [27], solving the five equations [9, 10, 17, 18], solving the six equations [42], and solving the seven equations [40]. The interface-capturing method uses a unique set of equations that combine the multicomponent flows governed by the Euler equations with the fluid composition described by the transport equations. Compared with the interface-tracking method, the interface-capturing method is easier to be implemented and generalized to high-dimensional cases. It allows interfaces to be arbitrarily large, complex deformations, and dynamically created. Additionally, it can preserve the conservation of the total mass, momentum, and energy of the system. Due to these advantages, the interface-capturing method has become one of the popular methods for solving multicomponent flow problems [19].

How to design a satisfying finite volume interface-capturing method for solving the system that combines the governing equations with the transport equations is a tough job. Numerical experiments have shown that spurious oscillations will appear if the scheme is not designed carefully [34]. These oscillations already exist in first-order calculations and are hard to eliminate by using the high-order method [26]. Abgrall [1] carefully analyzed the reason causing oscillations and proposed a way to eliminate them. The key ingredients are two parts: on one hand, the transport equations should be designed in advection form; on the other hand, the transport equations should be consistent with the governing equations such that the system can maintain the equilibrium of velocity and pressure across the interface. Shyue [44–46] extended the method to more general equations of state. Abgrall and Saurel [4, 40] further extended the method to the multiphase case. Allaire et al. [5] introduced a new five-equation model and designed a suitable scheme. More details can be found in [41]. However, due to numerical dissipation, we may still suffer from oscillation when extending the aforementioned numerical method to the high-order ones [33]. The oscillation cannot be suppressed by high-order non-oscillatory methods, such as essentially non-oscillatory (ENO) and weighted ENO (WENO) methods. In [25], Johnsen and Colonius suggested utilizing the quasi-conservative form coupled with the reconstruction of primitive variables can eliminate the spurious oscillation no matter whether WENO techniques are implemented in a componentwise or characteristic way. Following this idea, researchers have designed many satisfying schemes to solve the multicomponent flows problem [11, 21, 53, 55]. They are all based on finite volume methods, but none of them can be considered to be truly high-order schemes for two reasons. Firstly, in these methods, primitive variables are reconstructed to maintain the equilibrium of velocity and pressure, but they are obtained in a low-order way. Secondly, advection equations are rewritten into a conservation law with a source term to enhance performance, but this source term is not handled with a high-order of accuracy. As a result, although these finite volume methods can achieve high-order accuracy for some linear problems, this is not the case for nonlinear ones. Therefore, it is necessary to develop finite volume methods that can handle multicomponent fluid problems with truly high-order accuracy.

In this paper, we propose a new high-order finite volume method for solving the multicomponent fluids problem. Our method can be regarded as an extension of the method presented in [25]. As for the method presented in [25], due to its low-order calculation of primitive variables and source terms, it is not truly a high-order method. Our goal is to develop a scheme that can achieve high-order accuracy in the smooth case, especially when the velocity or pressure is non-equilibrium, while also maintaining its essentially non-oscillatory property in non-smooth regions. Firstly, we couple the quasi-conservative form with the reconstruction of conservative variables in a characteristic manner. This approach maintains the equilibrium of the pressure and velocity for the interface-only problem, eliminates spurious oscillation along the interface, and achieves high-order accuracy during the reconstruction phase. Secondly, we employ integration by parts and accurate numerical integration in a high-order manner for

the source term, allowing our scheme to achieve high-order accuracy in the smooth region. We introduce a new WENO method to reconstruct the numerical solution. The new WENO method is similar to the WENO method which was presented by Zhu and Qiu [61], termed as WENO-ZQ, which uses one big stencil and two small stencils. It enables the method to maintain accuracy in smooth regions and improve the resolution in discontinuous regions. Specifically, the polynomial within a given cell can be reconstructed by using the new WENO method, simplifying the computation of the finite volume method. Furthermore, we carefully design our numerical flux to preserve physical parameters in single-component fluids. This property further enhances the robustness of the scheme. Finally, both variables are evolved by using Runge-Kutta time discretization. We performed extensive numerical tests for both one- and two-dimensional problems to verify the high resolution and accuracy of our scheme. It is noted that there are other schemes available for solving compressible flow problems, such as the arbitrary high-order derivative Riemann (ADER) problem, the residual distribution (RD) schemes, the discontinuous Galerkin (DG) method, the Hermite WENO (HWENO) method, and the semi-Lagrangian (SL) methods. Details can be found in [2, 3, 13, 14, 30, 38, 39, 50, 51, 57–60].

The rest of this paper is organized as follows. In Section 2, we describe the detailed steps of our scheme in a one-dimensional case. In Section 3, we perform numerical experiments to verify the numerical accuracy and efficiency of the scheme. In Section 4, we give a conclusion. In Appendix A, we describe our scheme in a two-dimensional case.

2 The framework for the one-dimensional case

We consider the following governing five-equation system to solve multicomponent problems with the stiffened gas equation of state [44]. This system contains three conservative equations and two advective equations, i.e.,

$$\begin{cases} \rho_t + (\rho u)_x = 0, \\ (\rho u)_t + (\rho u^2 + p)_x = 0, \\ E_t + (u(E+p))_x = 0, \\ \left(\dfrac{1}{\gamma-1}\right)_t + u\left(\dfrac{1}{\gamma-1}\right)_x = 0, \\ \left(\dfrac{\gamma p_\infty}{\gamma-1}\right)_t + u\left(\dfrac{\gamma p_\infty}{\gamma-1}\right)_x = 0. \end{cases} \tag{2.1}$$

In [44], we obtain the left and right eigenvector matrices of such a Jacobian matrix as

$$R = \begin{pmatrix} \frac{1}{c^2} & \frac{1}{c^2} & \frac{1}{c^2} & 0 & 0 \\ \frac{u-c}{c^2} & \frac{u}{c^2} & \frac{u+c}{c^2} & 0 & 0 \\ \frac{1}{2}(\frac{u}{c})^2 + \frac{1}{\gamma-1} - \frac{u}{c} & \frac{1}{2}(\frac{u}{c})^2 & \frac{1}{2}(\frac{u}{c})^2 + \frac{1}{\gamma-1} + \frac{u}{c} & p & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{2.2}$$

and

$$L = \begin{pmatrix} \frac{u(u\gamma+2c-u)}{4} & -\frac{u\gamma+c-u}{2} & \frac{\gamma-1}{2} & -\frac{p(\gamma-1)}{2} & -\frac{\gamma-1}{2} \\ \frac{-\gamma u^2+2c^2+u^2}{2} & u(\gamma-1) & 1-\gamma & p(\gamma-1) & \gamma-1 \\ -\frac{u(-u\gamma+2c+u)}{4} & \frac{-u\gamma+c+u}{2} & \frac{\gamma-1}{2} & -\frac{p(\gamma-1)}{2} & -\frac{\gamma-1}{2} \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \tag{2.3}$$

Here, ρ is the density, u is the velocity, E is the total energy, p is the pressure, γ and p_∞ are the usual ratio of specific heats and prescribed physical constant, respectively, and c is the sound speed. Furthermore,

due to the stiffened gas equation of state, we have the following equation:

$$E=\frac{1}{2}\rho u^2+\frac{p+\gamma p_\infty}{\gamma-1},\quad c=\sqrt{\frac{\gamma(p+p_\infty)}{\rho}}.$$

Physically, the parameters γ and p_∞ are used to describe the properties of materials and can be determined from laboratory experiments via an empirical fit. For example, for air we usually set $\gamma=1.4$ and $p_\infty=0$, for helium we usually set $\gamma=1.67$ and $p_\infty=0$, and for water we can set $\gamma=7.15$ and $p_\infty=3.309\times10^8$ [15, 29].

For simplicity, the computational domain is divided into N uniform cells. We denote the cell by $I_j=[x_{j-\frac{1}{2}},x_{j+\frac{1}{2}}]$, its cell center by $x_j=(x_{j-\frac{1}{2}}+x_{j+\frac{1}{2}})/2$, and its cell size by $\Delta x=x_{j+\frac{1}{2}}-x_{j-\frac{1}{2}}$. Furthermore, we take

$$U=\left(\rho,\rho u,E,\frac{1}{\gamma-1},\frac{\gamma p_\infty}{\gamma-1}\right)^{\mathrm{T}},$$
$$F(U)=\left(\rho u,\rho u^2+p,u(E+p),\frac{u}{\gamma-1},\frac{u\gamma p_\infty}{\gamma-1}\right)^{\mathrm{T}},$$

and

$$S(U)=\left(0,0,0,\frac{1}{\gamma-1},\frac{\gamma p_\infty}{\gamma-1}\right)^{\mathrm{T}},$$

and then we can rewrite the system (2.1) into

$$U_t+F(U)_x=S(U)u_x,\tag{2.4}$$

where u_x refers to the derivative of velocity. We integrate (2.4) over the cell I_j. Next, to handle the source term, we employ the integration by part, a technique used in [9, 10, 25]. This allows us to obtain the following semidiscretization form:

$$\begin{aligned}&\frac{\partial U_j(t)}{\partial t}+\frac{1}{\Delta x}(F(U(x_{j+\frac{1}{2}},t))-F(U(x_{j-\frac{1}{2}},t)))\\&\quad=\frac{1}{\Delta x}(S(U(x_{j+\frac{1}{2}},t))u(x_{j+\frac{1}{2}},t)-S(U(x_{j-\frac{1}{2}},t))u(x_{j-\frac{1}{2}},t))-\frac{1}{\Delta x}\int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}}S(U(x,t))_xu(x,t)dx,\end{aligned}\tag{2.5}$$

where $U_j(t)=\frac{1}{\Delta x}\int_{x_{j-1/2}}^{x_{j+1/2}}U(x,t)dx$. Then, we approximate (2.5) by the following formulation:

$$\frac{\partial U_j}{\partial t}+\frac{1}{\Delta x}(\hat{F}_{j+\frac{1}{2}}-\hat{F}_{j-\frac{1}{2}})=\frac{1}{\Delta x}\left(S(U^-_{j+\frac{1}{2}})\hat{u}_{j+\frac{1}{2}}-S(U^+_{j-\frac{1}{2}})\hat{u}_{j-\frac{1}{2}}-\Delta x\sum_{k=1}^{4}w_kS(U)_{x_{G_k}}u_{G_k}\right).\tag{2.6}$$

U_j refers to the numerical approximation to the cell average $U_j(t)$, and $\hat{F}_{j+\frac{1}{2}}$ and $\hat{u}_{j+\frac{1}{2}}$ represent the numerical flux evaluated at the interface $x_{j+1/2}$. The expression for the numerical flux is defined by $\hat{F}_{j+\frac{1}{2}}=\hat{F}(U^-_{j+\frac{1}{2}},U^+_{j+\frac{1}{2}})$ and $\hat{u}_{j+\frac{1}{2}}=\hat{u}(u^-_{j+\frac{1}{2}},u^+_{j+\frac{1}{2}})$. In this paper, we choose the HLLC numerical flux (see [25, 49])

$$\hat{F}_{j+\frac{1}{2}}=\frac{1+\mathrm{sgn}(s^*)}{2}[F_L+s^-(U_{*L}-U_L)]+\frac{1-\mathrm{sgn}(s^*)}{2}[F_R+s^+(U_{*R}-U_R)],\tag{2.7}$$

where the intermediate state is defined as

$$U_{*k}=\chi_{*k}\begin{pmatrix}\rho_k\\ \rho_ks_*\\ E_k+(s_*-u_k)(\rho_ks_*+\frac{p_k}{s_k-u_k})\\ \frac{1}{\gamma_k-1}\\ \frac{\gamma_kp_{\infty_k}}{\gamma_k-1}\end{pmatrix},\quad \chi_{*k}=\frac{s_k-u_k}{s_k-s_*},$$

where $k = L, R$. w_L refers to $w^-_{j+\frac{1}{2}}$ and w_R refers to $w^+_{j+\frac{1}{2}}$. The wave speeds are given by

$$s^- = \min(0, s_L), \quad s^+ = \max(0, s_R), \quad s_L = \min((u-c)^{\mathrm{ROE}}, u_L - c_L), \quad s_R = \max((u+c)^{\mathrm{ROE}}, u_R + c_R),$$

where $(u-c)^{\mathrm{ROE}}$ and $(u+c)^{\mathrm{ROE}}$ are the velocities obtained from an intermediate state based on the Roe average. The intermediate wave speed is computed by

$$s_* = \frac{p_R - p_L + \rho_L u_L(s_L - u_L) - \rho_R u_R(s_R - u_R)}{\rho_L(s_L - u_L) - \rho_R(s_R - u_R)}.$$

As to the velocity in the source term, we define

$$\hat{u}_{j+\frac{1}{2}} = \frac{1 + \mathrm{sgn}(s^*)}{2}[u_L + s^-(\chi_{*L} - 1)] + \frac{1 - \mathrm{sgn}(s^*)}{2}[u_R + s^+(\chi_{*R} - 1)]. \tag{2.8}$$

In (2.6), w_k and G_k are Gauss-Lobatto quadrature points and coefficients, i.e.,

$$\begin{aligned} &G_1 = x_{j-\frac{1}{2}}, \quad G_2 = x_{j-\frac{\sqrt{5}}{10}}, \quad G_3 = x_{j+\frac{\sqrt{5}}{10}}, \quad G_4 = x_{j+\frac{1}{2}}, \\ &\omega_1 = \frac{1}{12}, \quad \omega_2 = \frac{5}{12}, \quad \omega_3 = \frac{5}{12}, \quad \omega_4 = \frac{1}{12}. \end{aligned} \tag{2.9}$$

$U^{\pm}_{j+\frac{1}{2}}$, U_{G_k}, and u_{xG_k} are numerical approximation to the solutions and derivatives at the $x_{j+\frac{1}{2}}$ or G_k respectively, which are obtained by the WENO reconstruction method described in Subsection 2.1.

Then, we can rewrite the semi-discrete system (2.6) as $\mathrm{U}_t = \mathcal{L}(\mathrm{U})$, where $\mathcal{L}$ denotes the operator of the spatial discretization, and we use the third-order total variation diminishing (TVD) Runge-Kutta time discretization [43] to solve the semi-discrete form (2.6):

$$\begin{cases} \mathrm{U}^{(1)} = \mathrm{U}^n + \Delta t \mathcal{L}(\mathrm{U}^n), \\ \mathrm{U}^{(2)} = \dfrac{3}{4}\mathrm{U}^n + \dfrac{1}{4}(\mathrm{U}^{(1)} + \Delta t \mathcal{L}(\mathrm{U}^{(1)})), \\ \mathrm{U}^{n+1} = \dfrac{1}{3}\mathrm{U}^n + \dfrac{2}{3}(\mathrm{U}^{(2)} + \Delta t \mathcal{L}(\mathrm{U}^{(2)})). \end{cases} \tag{2.10}$$

2.1 The WENO reconstruction

Now, we list a detailed flowchart of the new WENO reconstruction method.

Step 1. Given one big stencil $S_0 = \{I_{i-2}, I_{i-1}, I_i, I_{i+1}, I_{i+2}\}$, two small stencils $S_1 = \{I_{i-1}, I_i\}$ and $S_2 = \{I_i, I_{i+1}\}$, and three middle stencils $S_3 = \{I_{i-2}, I_{i-1}, I_i\}$, $S_4 = \{I_{i-1}, I_i, I_{i+1}\}$, and $S_5 = \{I_i, I_{i+1}, I_{i+2}\}$, we need to construct polynomials $p_0(x)$, $p_1(x)$, $p_2(x)$, $p_3(x)$, $p_4(x)$, and $p_5(x)$ such that

$$\begin{aligned} &\frac{1}{\Delta x}\int_{I_{i+l}} p_0(x)dx = \overline{u}_{i+l}, \quad l = -2, -1, 0, 1, 2, \\ &\frac{1}{\Delta x}\int_{I_{i+l}} p_1(x)dx = \overline{u}_{i+l}, \quad l = -1, 0, \\ &\frac{1}{\Delta x}\int_{I_{i+l}} p_2(x)dx = \overline{u}_{i+l}, \quad l = 0, 1, \\ &\frac{1}{\Delta x}\int_{I_{i+l}} p_3(x)dx = \overline{u}_{i+l}, \quad l = -2, -1, 0, \\ &\frac{1}{\Delta x}\int_{I_{i+l}} p_4(x)dx = \overline{u}_{i+l}, \quad l = -1, 0, 1, \\ &\frac{1}{\Delta x}\int_{I_{i+l}} p_5(x)dx = \overline{u}_{i+l}, \quad l = 0, 1, 2. \end{aligned}$$

Step 2. We compute the smoothness indicators, denoted by $\beta_0, \beta_1, \beta_2, \beta_3, \beta_4$, and β_5, respectively. The smoothness indicators are based on the formula in [24]:

$$\beta_m = \frac{1}{\Delta x}\sum_{k=1}^{r}\int_{I_i}\left(\Delta x^k \frac{\partial^k}{\partial x^k} p_m(x)\right)^2 dx, \quad m = 0, 1, 2, 3, 4, 5,$$

where $r=4$ for $p_0(x)$, $r=1$ for $p_1(x)$ and $p_2(x)$, and $r=2$ for $p_3(x)$, $p_4(x)$, and $p_5(x)$.

Step 3. We take the linear weights as

$$r_0=0.2,\quad r_1=0.4,\quad r_2=0.4,$$

and

$$s_0=0.7,\quad s_3=0.1,\quad s_4=0.1,\quad s_5=0.1.$$

Then, we can rewrite $p_0(x)$ as

$$p_0(x)=\left(1-(1-r_0)(1-s_0)^2\left(1-\frac{s_0}{s_0}\right)^2\right)p_0(x)+(1-s_0)^2\left(1-\frac{s_0}{s_0}\right)^2(r_1p_1(x)+r_2p_2(x)). \tag{2.11}$$

Step 4. We compute the nonlinear weights

$$\omega_k=\frac{\overline{\omega}_k}{\overline{\omega}_0+\overline{\omega}_1+\overline{\omega}_2},\quad \overline{\omega}_k=\frac{r_k}{(\beta_k+\varepsilon)^2},\quad k=0,1,2,$$

and

$$\mu_0=\frac{\overline{\mu}_0}{\overline{\mu}_0+\overline{\mu}_3+\overline{\mu}_4+\overline{\mu}_5},\quad \overline{\mu}_k=\frac{s_k}{(\beta_k+\varepsilon)^2},\quad k=0,3,4,5,$$

where $\varepsilon=10^{-12}$ to avoid dividing by zero. Then, we have

$$p(x)=\left(1-(1-\omega_0)(1-\mu_0)^2\left(1-\frac{\mu_0}{s_0}\right)^2\right)p_0(x)+(1-\mu_0)^2\left(1-\frac{\mu_0}{s_0}\right)^2(\omega_1p_1(x)+\omega_2p_2(x)) \tag{2.12}$$

and

$$p'(x)=\left(1-(1-\omega_0)(1-\mu_0)^2\left(1-\frac{\mu_0}{s_0}\right)^2\right)p_0'(x)+(1-\mu_0)^2\left(1-\frac{\mu_0}{s_0}\right)^2(\omega_1p_1'(x)+\omega_2p_2'(x)). \tag{2.13}$$

Now, we give a brief analysis of the method.

When the solution is smooth in the stencil S_0, through the Taylor expansion analysis, we have

$$\mu_0=s_0+\mathcal{O}(\Delta x^2).$$

It implies that the method can realize the fifth-order accuracy, i.e.,

$$\begin{aligned}p(x)-u(x)&=(1-\mu_0)^2\left(1-\frac{\mu_0}{s_0}\right)^2(\omega_1(p_1(x)-u(x))+\omega_2(p_2(x)-u(x)))\\&\quad+\left(1-(1-\omega_0)(1-\mu_0)^2\left(1-\frac{\mu_0}{s_0}\right)^2\right)(p_0(x)-u(x))\\&=\mathcal{O}(\Delta x^4)(\omega_1\mathcal{O}(\Delta x^2)+\omega_2\mathcal{O}(\Delta x^2))+(1-(1-\omega_0)\mathcal{O}(\Delta x^4))\mathcal{O}(\Delta x^5)\\&=\mathcal{O}(\Delta x^5).\end{aligned}$$

When the solution is discontinuous in the stencil S_0, we have $\beta_0=\mathcal{O}(1)$. If the solution is smooth in the stencil S_m, then $\beta_m=\mathcal{O}(\Delta x^2)$. As to the nonlinear weights μ_0 and ω_m, we have $\mu_0=\mathcal{O}(\Delta x^4)$ and $\omega_m=\mathcal{O}(\Delta x^4)$ when the solution is discontinuous in the stencil S_m, and $\omega_m=\mathcal{O}(1)$ when the solution is smooth in the stencil S_m. Therefore, the method maintains the ENO property.

Remark 1. We provide a simple explanation of the equation (2.12). The equation (2.13) is similar. The final reconstruction (2.12) is a convex combination of $p_0(x)$, $p_1(x)$, and $p_2(x)$. We expect $p_0(x)$ to play the main role when the solution is smooth in the big stencil, while $p_1(x)$ or $p_2(x)$ becomes the major component to maintain the ENO property when the solution is rough in the big stencil. If we simply define the (2.12) as $\omega_0p_0(x)+\omega_1p_1(x)+\omega_2p_2(x)$, it can clearly maintain the ENO property. However, the scheme would degrade its optimal fifth-order accuracy due to insufficient accuracy of nonlinear weights

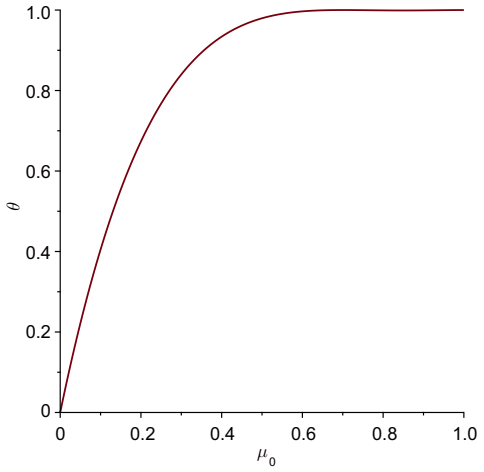

Figure 1 (Color online) The plot of parameter θ

ω_k and the low-order approximation of $p_1(x)$ and $p_2(x)$. The final reconstructions (2.12) can be rewritten as

$$p(x) = \theta p_0(x) + (1-\theta)(\omega_0 p_0(x) + \omega_1 p_1(x) + \omega_2 p_2(x)), \tag{2.14}$$

where

$$\theta = 1 - (1-\mu_0)^2 \left(1 - \frac{\mu_0}{s_0}\right)^2. \tag{2.15}$$

Therefore, (2.12) can be seen as a correction of $\omega_0 p_0(x) + \omega_1 p_1(x) + \omega_2 p_2(x)$. Figure 1 shows the plot of the parameter θ. From Figure 1 and (2.12), we can observe the property of the parameter θ.

- $\theta \to 0$, when $\mu_0 \to 0$.
- $\theta = 1$, when $\mu_0 = 1$.
- $\theta = 1 + \mathcal{O}(\Delta x^4)$, when $\mu_0 = s_0 + \mathcal{O}(\Delta x^2)$.
- $0 < \theta \leqslant 1$.

Property 1 ensures the second term in (2.14) plays the major role when the solution in the big stencil is discontinuous, while the properties 2 and 3 ensure that the first term in (2.14) becomes the major component and maintains the accuracy at the same time when the solution in the big stencil is smooth. Property 4 ensures the stability of the WENO reconstruction. As we know, $p_1(x)$ and $p_2(x)$ will become much smoother near the critical point than $p_0(x)$ which would lead to a degradation in the optimal order of the parameter μ_0. Hence, we introduce $p_3(x)$, $p_4(x)$, $p_5(x)$, and $p_0(x)$ to compute the parameter μ_0 in (2.15) in order to ensure the accuracy of the parameter μ_0.

2.2 Algorithm

Now, we list the algorithm of the high-order finite volume method for the multicomponent fluid problem (see Algorithm 1).

2.3 Properties of the schemes

The proposed high-order finite volume scheme has attractive features for the simulation of multicomponent flows. The two major properties are listed in the following propositions. For the convenience of proof, we introduce the following notations:

$$m = \rho u, \quad \Gamma = \frac{1}{\gamma - 1}, \quad \Pi = \frac{\gamma p_\infty}{\gamma - 1}.$$

Proposition 1. *The high-order finite volume scheme* (2.6) *with the numerical fluxes* (2.7) *and* (2.8) *preserves the equilibrium of the constants* γ *and* p_∞ *in a single-component fluid.*

Algorithm 1 Algorithm of the scheme

1. At interface $x_{j+1/2}$, compute the Roe average state $U_{j+1/2}$.
2. Compute the right eigenvectors and the left eigenvectors following (2.3) and (2.2), and denote them by
$$R = R(U_{j+1/2}), \quad L = L(U_{j+1/2}).$$
3. Project all the values in the potential stencil of the WENO method to the local characteristic fields, and denote them by
$$W_i = LU_i, \quad i = j-2, \ldots, j+3.$$
4. Based on the principle of proximity, for each component of the $\{W_i\}_{j-2}^{j+3}$, perform the WENO method at Gauss-Lobatto quadrature points G_3 and G_4 in the cell I_j and G_1 and G_2 in cell I_{j+1}, denote them by
$$W_{G_1 j+1}, \quad W_{G_2 j+1}, \quad W_{G_3 j}, \quad W_{G_4 j}.$$
5. Project back into physical space by using
$$U_{G_{j+k}} = RW_{G_{j+k}}, \quad G = G_1, G_2, G_3, G_4, \quad k = 1 \quad \text{for } G_1 \text{ and } G_2 \text{ and } k = 0 \text{ for } G_3 \text{ and } G_4.$$
6. Perform the WENO method to obtain $S(U)_{x_{G_k}}$ $(k = 1, 2, 3, 4)$.
7. Compute $\hat{u}_{j\pm\frac{1}{2}}$, $\hat{F}_{j\pm\frac{1}{2}}$, $S(U^{\mp}_{j\pm\frac{1}{2}})$, and $S(U)_{x_{G_k}}$ in (2.6) and form the scheme.
8. Evolve the scheme by the TVD Runge-Kutta method (2.10).

Proof. In order to prove this property, we only need to verify that the parameters Γ and Π can maintain equilibrium during the evolution. Without loss of generality, we give the proof for the equation of Γ, and the proof for the equation of Π is similar. The scheme for Γ is given as

$$\frac{\partial \Gamma_j}{\partial t} + \frac{1}{\Delta x}(\hat{u\Gamma}_{j+\frac{1}{2}} - \hat{u\Gamma}_{j-\frac{1}{2}}) = \frac{1}{\Delta x}\left(\Gamma^-_{j+\frac{1}{2}}\hat{u}_{j+\frac{1}{2}} - \Gamma^+_{j-\frac{1}{2}}\hat{u}_{j-\frac{1}{2}} - \Delta x \sum_k w_k \Gamma_{x_{G_k}} u_{G_k}\right). \tag{2.16}$$

Due to the equilibrium of constant γ, we have

$$\hat{u\Gamma}_{j\pm\frac{1}{2}} = \Gamma \hat{u}_{j\pm\frac{1}{2}}, \quad \Gamma^{\pm}_{j\mp\frac{1}{2}} = \Gamma, \quad \Gamma_{x_{G_k}} = 0.$$

Therefore, the terms in the brackets in the equation (2.16) cancel each other. Then, we have

$$\frac{\partial \Gamma_j}{\partial t} = 0.$$

It indicates that Γ can maintain equilibrium when the solution evolves. □

Proposition 2. *The high-order finite volume scheme* (2.6) *with the numerical fluxes* (2.7) *and* (2.8) *preserves the equilibrium of the pressure and velocity for the interface-only problem.*

Proof. The same as before, we only need to verify that the velocity u and pressure p can maintain equilibrium during the evolution.

First, we implement the WENO method in a characteristic way with the conservative variables. We assume that the conservative variables $U_j = (\rho_j, m_j, E_j, \Gamma_j, \Pi_j)$ maintains the equilibrium of the pressure and velocity. Then, we have the following equality:

$$m_j = \rho_j \overline{u}, \quad E_j = \frac{1}{2}\rho_j \overline{u}^2 + \overline{p}\Gamma_j + \Pi_j,$$

where $\overline{u}$ and $\overline{p}$ denote the equilibrium values of the velocity and pressure, respectively.

We use the Roe average to linearize the right and left eigenvector matrices in (2.2) and (2.3) at interface $x = x_{j+\frac{1}{2}}$ of the cells, i.e.,

$$\overline{R} = \begin{pmatrix} \frac{1}{\overline{c}^2} & \frac{1}{\overline{c}^2} & \frac{1}{\overline{c}^2} & 0 & 0 \\ \frac{\overline{u}-\overline{c}}{\overline{c}^2} & \frac{\overline{u}}{\overline{c}^2} & \frac{\overline{u}+\overline{c}}{\overline{c}^2} & 0 & 0 \\ \frac{1}{2}(\frac{\overline{u}}{\overline{c}})^2 + \frac{1}{\overline{\gamma}-1} - \frac{\overline{u}}{\overline{c}} & \frac{1}{2}(\frac{\overline{u}}{\overline{c}})^2 & \frac{1}{2}(\frac{\overline{u}}{\overline{c}})^2 + \frac{1}{\overline{\gamma}-1} + \frac{\overline{u}}{\overline{c}} & \overline{p} & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

and

$$\overline{L}=\begin{pmatrix} \frac{\overline{u}(\overline{u}\ \overline{\gamma}+2\overline{c}-\overline{u})}{4} & -\frac{\overline{u}\ \overline{\gamma}+\overline{c}-\overline{u}}{2} & \frac{\overline{\gamma}-1}{2} & -\frac{\overline{p}(\overline{\gamma}-1)}{2} & -\frac{\overline{\gamma}-1}{2} \\ \frac{-\overline{\gamma}\ \overline{u}^2+2\overline{c}^2+\overline{u}^2}{2} & \overline{u}(\overline{\gamma}-1) & 1-\overline{\gamma}\ \overline{p}(\overline{\gamma}-1) & \overline{\gamma}-1 & \\ -\frac{\overline{u}(-\overline{u}\ \overline{\gamma}+2\overline{c}+\overline{u})}{4} & \frac{-\overline{u}\ \overline{\gamma}+\overline{c}+\overline{u}}{2} & \frac{\overline{\gamma}-1}{2} & -\frac{\overline{p}(\overline{\gamma}-1)}{2} & -\frac{\overline{\gamma}-1}{2} \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

We project conservative variables

$$U_j=\left(\rho_j,\rho_j\overline{u},\frac{1}{2}\rho_j\overline{u}^2+\overline{p}\Gamma_j+\Pi_j,\Gamma_j,\Pi_j\right)^{\mathrm{T}}$$

into the characteristic space

$$W_j=\overline{L}U_j=\begin{pmatrix} \frac{\overline{u}(\overline{u}\ \overline{\gamma}+2\overline{c}-\overline{u})}{4} & -\frac{\overline{u}\ \overline{\gamma}+\overline{c}-\overline{u}}{2} & \frac{\overline{\gamma}-1}{2} & -\frac{\overline{p}(\overline{\gamma}-1)}{2} & -\frac{\overline{\gamma}-1}{2} \\ \frac{-\overline{\gamma}\ \overline{u}^2+2\overline{c}^2+\overline{u}^2}{2} & \overline{u}(\overline{\gamma}-1) & 1-\overline{\gamma}\ \overline{p}(\overline{\gamma}-1) & \overline{\gamma}-1 & \\ -\frac{\overline{u}(-\overline{u}\ \overline{\gamma}+2\overline{c}+\overline{u})}{4} & \frac{-\overline{u}\ \overline{\gamma}+\overline{c}+\overline{u}}{2} & \frac{\overline{\gamma}-1}{2} & -\frac{\overline{p}(\overline{\gamma}-1)}{2} & -\frac{\overline{\gamma}-1}{2} \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} \rho_j \\ \rho_j\overline{u} \\ \frac{1}{2}\rho_j\overline{u}^2+\overline{p}\Gamma_j+\Pi_j \\ \Gamma_j \\ \Pi_j \end{pmatrix}=\begin{pmatrix} 0 \\ \overline{c}^2\rho_j \\ 0 \\ \Gamma_j \\ \Pi_j \end{pmatrix}.$$

After the WENO reconstruction, we denote the obtained variables by

$$W^{\pm}_{j+\frac{1}{2}}=\begin{pmatrix} 0 \\ \overline{c}^2w_2 \\ 0 \\ w_4 \\ w_5 \end{pmatrix}^{\pm}_{j+\frac{1}{2}}.$$

Then, we project the obtained variables back into the physical space. We have

$$\begin{aligned} U^{\pm}_{j+\frac{1}{2}}=\overline{R}W^{\pm}_{j+\frac{1}{2}}&=\begin{pmatrix} \frac{1}{\overline{c}^2} & \frac{1}{\overline{c}^2} & \frac{1}{\overline{c}^2} & 0 & 0 \\ \frac{\overline{u}-\overline{c}}{\overline{c}^2} & \frac{\overline{u}}{\overline{c}^2} & \frac{\overline{u}+\overline{c}}{\overline{c}^2} & 0 & 0 \\ \frac{1}{2}(\frac{\overline{u}}{\overline{c}})^2+\frac{1}{\overline{\gamma}-1}-\frac{\overline{u}}{\overline{c}} & \frac{1}{2}(\frac{\overline{u}}{\overline{c}})^2 & \frac{1}{2}(\frac{\overline{u}}{\overline{c}})^2+\frac{1}{\overline{\gamma}-1}+\frac{\overline{u}}{\overline{c}} & \overline{p} & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} 0 \\ \overline{c}^2w_2 \\ 0 \\ w_4 \\ w_5 \end{pmatrix}^{\pm}_{j+\frac{1}{2}} \\ &=\begin{pmatrix} w_2 \\ w_2\overline{u} \\ \frac{1}{2}w_2\overline{u}^2+\overline{p}w_4+w_5 \\ w_4 \\ w_5 \end{pmatrix}^{\pm}_{j+\frac{1}{2}}. \end{aligned}$$

Then, $U^{\pm}_{j+\frac{1}{2}}$ are used to compute the HLLC flux $\hat{F}_{j+\frac{1}{2}}$. Due to the equilibrium of the pressure and velocity, we have

$$u_L=u_R=\overline{u},\quad p_L=p_R=\overline{p},\quad \hat{u}_{j\pm\frac{1}{2}}=\overline{u}.$$

Therefore, we have

$$s_*=\overline{u},\quad U_{*k}=U_k,$$

and

$$\hat{F}_{j+\frac{1}{2}}=\frac{1+\mathrm{sgn}(s^*)}{2}F_L+\frac{1-\mathrm{sgn}(s^*)}{2}F_R=\begin{pmatrix}\tilde{\rho}\overline{u}\\ \tilde{\rho}\overline{u}^2+\overline{p}\\ \overline{u}(\frac{1}{2}\tilde{\rho}\overline{u}^2+\overline{p}\tilde{\Gamma}+\tilde{\Pi}+\overline{p})\\ \tilde{\Gamma}\\ \tilde{\Pi}\end{pmatrix},$$

where $\tilde{q}=\frac{1+\mathrm{sgn}(s^*)}{2}q_L+\frac{1-\mathrm{sgn}(s^*)}{2}q_R$ and $q=\rho,\Gamma,\Pi$.

As to the source term, due to the equilibrium of the velocity, we have

$$\begin{aligned}&S(U^-_{j+\frac{1}{2}})\hat{u}_{j+\frac{1}{2}}-S(U^+_{j-\frac{1}{2}})\hat{u}_{j-\frac{1}{2}}-\Delta x\sum_k w_kS(U)_{x_{G_k}}u_{G_k}\\ &=S(U^-_{j+\frac{1}{2}})\overline{u}-S(U^+_{j-\frac{1}{2}})\overline{u}-\Delta x\overline{u}\sum_k w_kS(U)_{x_{G_k}}\\ &=S(U^-_{j+\frac{1}{2}})\overline{u}-S(U^+_{j-\frac{1}{2}})\overline{u}-\overline{u}\int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}}S(U)_x dx\\ &=(S(U^-_{j+\frac{1}{2}})\overline{u}-S(U^+_{j-\frac{1}{2}})\overline{u})-\overline{u}(S(U^-_{j+\frac{1}{2}})-S(U^+_{j-\frac{1}{2}}))\\ &=0,\end{aligned}\tag{2.17}$$

where from the second to third equations, we use the fact that the four-point Gauss-Lobatto quadrature rule is exact for the polynomial degree up to five. Therefore, we have

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ \rho u\\ E\\ \Gamma\\ \Pi\end{pmatrix}_j=-\frac{1}{\Delta x}\left(\begin{pmatrix}\tilde{\rho}\overline{u}\\ \tilde{\rho}\overline{u}^2+\overline{p}\\ \overline{u}(\frac{1}{2}\tilde{\rho}\overline{u}^2+\overline{p}\tilde{\Gamma}+\tilde{\Pi}+\overline{p})\\ \overline{u}\tilde{\Gamma}\\ \overline{u}\tilde{\Pi}\end{pmatrix}_{j+\frac{1}{2}}-\begin{pmatrix}\tilde{\rho}\overline{u}\\ \tilde{\rho}\overline{u}^2+\overline{p}\\ \overline{u}(\frac{1}{2}\tilde{\rho}\overline{u}^2+\overline{p}\tilde{\Gamma}+\tilde{\Pi}+\overline{p})\\ \overline{u}\tilde{\Gamma}\\ \overline{u}\tilde{\Pi}\end{pmatrix}_{j-\frac{1}{2}}\right).$$

From the density and momentum equations, we have

$$\frac{\partial(\rho u)_j}{\partial t}=\overline{u}\frac{\partial\rho_j}{\partial t},$$

which means that the velocity will remain uniform when the solution evolves.

Now, we consider the pressure by comparing the energy equation with density and momentum equations, which are given as

$$\frac{\partial(p\Gamma+\Pi)_j}{\partial t}=-\frac{1}{\Delta x}((\overline{p}\tilde{\Gamma}_{j+\frac{1}{2}}\overline{u}+\tilde{\Pi}_{j+\frac{1}{2}}\overline{u})-(\overline{p}\tilde{\Gamma}_{j+\frac{1}{2}}\overline{u}+\tilde{\Pi}_{j+\frac{1}{2}}\overline{u})).$$

By resorting to the Γ equation and the Π equation, we can further simplify the above equation and obtain

$$\frac{\partial(p\Gamma)_j}{\partial t}=\overline{p}\frac{\partial\Gamma_j}{\partial t},$$

which means that the pressure will remain uniform when the solution evolves. □

Remark 2. The one-dimensional framework can be extended to a two-dimensional case. We provide the detailed steps in Appendix A.

3 The numerical test

In this section, we show the numerical result using the reconstruction of conservative variables by the finite volume method (denoted by 'FVCV') with the Courant-Friedrichs-Lewy (CFL) number set to 0.5. For comparison, we also list the computational result using reconstruction of primitive variables by finite volume method (denoted by 'FVPV') in the one-dimensional case [25].

Example 1. We consider the following accuracy test:

$$\rho(x,0)=1+0.2\sin(\pi x),\quad u(x,0)=1,\quad p(x,0)=1.$$

We choose

$$\frac{1}{\gamma(x,0)-1}=\frac{1}{\gamma_L-1}\left(\frac{1}{2}+\frac{1}{2}\sin(\pi x)\right)+\frac{1}{\gamma_R-1}\left(\frac{1}{2}-\frac{1}{2}\sin(\pi x)\right),$$

and

$$\frac{\gamma(x,0)p_\infty(x,0)}{\gamma(x,0)-1}=\frac{\gamma_L p_{\infty L}}{\gamma_L-1}\left(\frac{1}{2}+\frac{1}{2}\sin(\pi x)\right)+\frac{\gamma_R p_{\infty R}}{\gamma_R-1}\left(\frac{1}{2}-\frac{1}{2}\sin(\pi x)\right),$$

where $\gamma_L=1.4$, $\gamma_R=4$, and $p_{\infty L}=0$, $p_{\infty R}=1$. The computational domain is $[0,2]$. Periodic boundary conditions are used in this test. The exact solution of ρ is

$$\rho(x,t)=1+0.2\sin(\pi(x-t)).$$

We set the final time $t=2$. The results and a comparison with FVPV are listed in Table 1. We can see that our method achieves the designed fifth-order accuracy, and the numerical errors are smaller than the ones obtained by the FVPV.

Example 2. We consider the artificial accuracy test. We choose the following initial conditions:

$$\rho(x,0)=\frac{1+0.2\sin(x)}{2\sqrt{\gamma}},\quad u(x,0)=\sqrt{\gamma}\rho(x,0),\quad p(x,0)=\rho(x,0)^\gamma,\quad \gamma(x,0)=3,\quad p_\infty(x,0)=0.$$

The computational domain is $[0,2\pi]$. Periodic boundary conditions are used in this test. By the special choice of the parameter γ, initial conditions, and boundary conditions, we can verify that $2\sqrt{\gamma}\rho(x,t)$ is the exact solution of the following Burgers equation:

$$\mu_t+\frac{1}{2}(\mu^2)_x=0,\quad \mu(x,0)=1+0.2\sin(x).$$

The velocity, pressure, γ, and p_∞ satisfy the following relations:

$$u(x,t)=\sqrt{\gamma}\rho(x,t),\quad p(x,t)=\rho(x,t)^\gamma,\quad \gamma(x,0)=3,\quad p_\infty(x,0)=0.$$

It is easy to verify that the solution of the Burgers equation above is smooth up to time $T=5$. We set the final time as $T=3$. At this time, the solution is still smooth. We list the error and numerical accuracy order in Table 2. We can see that the FVPV can only achieve second-order accuracy due to the velocity and pressure being in non-equilibrium. However, our method can achieve the expected fifth-order of accuracy. We also plot the γ and p_∞ at the final time using $N=320$ (see Figure 2). The base of γ and p_∞ have been subtracted. We can see that the method can reach a round-off error.

Table 1 The accuracy test for the Euler system in 1D

	FVCV				FVPV			
Mesh size	L_∞ error	Order	L_1 error	Order	L_∞ error	Order	L_1 error	Order
10	2.66E−03		1.30E−03		9.33E−03		5.79E−03	
20	6.66E−05	5.32	3.97E−05	5.04	5.03E−04	4.21	2.86E−04	4.34
40	2.07E−06	5.01	1.27E−06	4.97	1.76E−05	4.84	8.94E−06	5.00
80	6.44E−08	5.00	3.98E−08	4.99	5.54E−07	4.99	2.78E−07	5.01
160	2.00E−09	5.01	1.25E−09	5.00	1.57E−08	5.14	8.60E−09	5.02
320	6.23E−11	5.01	3.89E−11	5.00	4.41E−10	5.15	2.58E−10	5.06

Table 2 The accuracy test for the density in 1D

	FVCV				FVPV			
Mesh size	L_∞ error	Order	L_1 error	Order	L_∞ error	Order	L_1 error	Order
10	2.21E−02		9.50E−03		3.08E−02		1.02E−02	
20	5.60E−03	1.98	1.57E−03	2.60	9.48E−03	1.70	1.59E−03	2.68
40	1.44E−03	1.95	1.57E−04	3.32	2.50E−03	1.92	3.64E−04	2.13
80	2.70E−04	2.42	1.47E−05	3.42	4.70E−04	2.41	7.54E−05	2.27
160	6.15E−06	5.45	3.08E−07	5.58	1.04E−04	2.18	1.76E−05	2.10
320	2.00E−07	4.95	8.94E−09	5.11	2.57E−05	2.01	4.35E−06	2.02

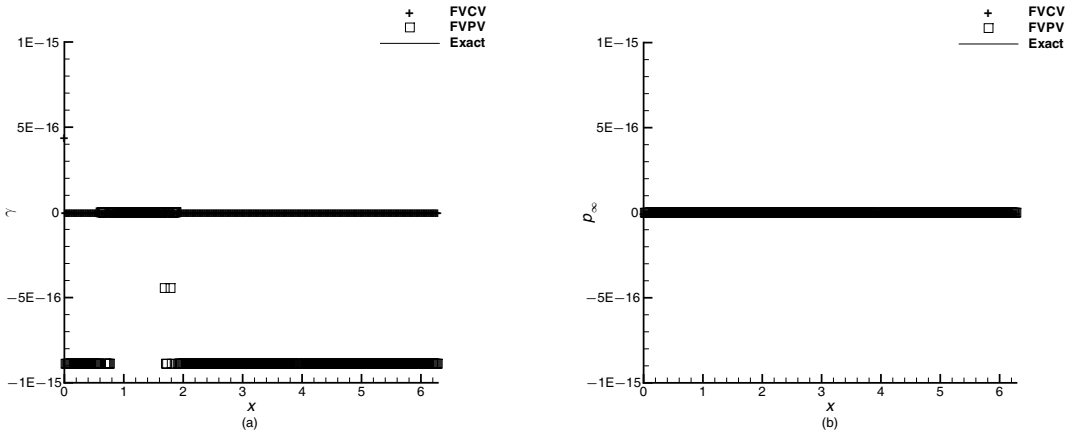

Figure 2 Numerical results for Example 2. From (a)–(b): figures of γ and p_∞. The base $\gamma_0 = 3$ and $p_{\infty 0} = 0$

Example 3. We solve a Riemann problem consisting of a single contact discontinuity in gas dynamics, i.e.,

$$(\rho, u, p, \gamma, p_\infty) = \begin{cases} (1, 1, 1, 1.4, 0), & x < 0.5, \\ (0.125, 1, 1, 4, 1), & x \geqslant 0.5. \end{cases}$$

We set the computational domain as $[0, 1]$, the periodic condition, the final time $T = 2$, and $N = 200$. Figure 3 shows the results. The base velocity and pressure have been subtracted. From the figures, we can see that our method can obtain comparable results with the one obtained by the FVPV method.

Example 4. This example is taken from [37]. We consider the following initial conditions:

$$(\rho, u, p, \gamma, p_\infty) = \begin{cases} (1.3333, 0.3535\sqrt{10^5}, 1.5 \times 10^5, 1.4, 0), & x < 0.05, \\ (1.0, 0, 10^5, 1.4, 0), & 0.05 \leqslant x < 0.5, \\ (0.1379, 0, 10^5, 5/3, 0), & x \geqslant 0.5. \end{cases}$$

The computational domain is $[0, 1]$. We compute the solution of this problem to $T = 0.0012$ with $N = 200$ and show the final result in Figure 4. From the figures, we can see that our method matches the exact solution well.

Example 5. This example is taken from [37]. We consider the following initial conditions:

$$(\rho, u, p, \gamma, p_\infty) = \begin{cases} (4.3333, 3.2817\sqrt{10^5}, 15 \times 10^5, 1.4, 0), & x < 0.05, \\ (1.0, 0, 10^5, 1.4, 0), & 0.05 \leqslant x < 0.5, \\ (3.1538, 0, 10^5, 1.249, 0), & x \geqslant 0.5. \end{cases}$$

The computational domain is $[0, 1]$. We compute the solution of this problem to $T = 0.0007$ with $N = 200$ and show the final result in Figure 5. From the figures, we can see that our method can obtain satisfying results.

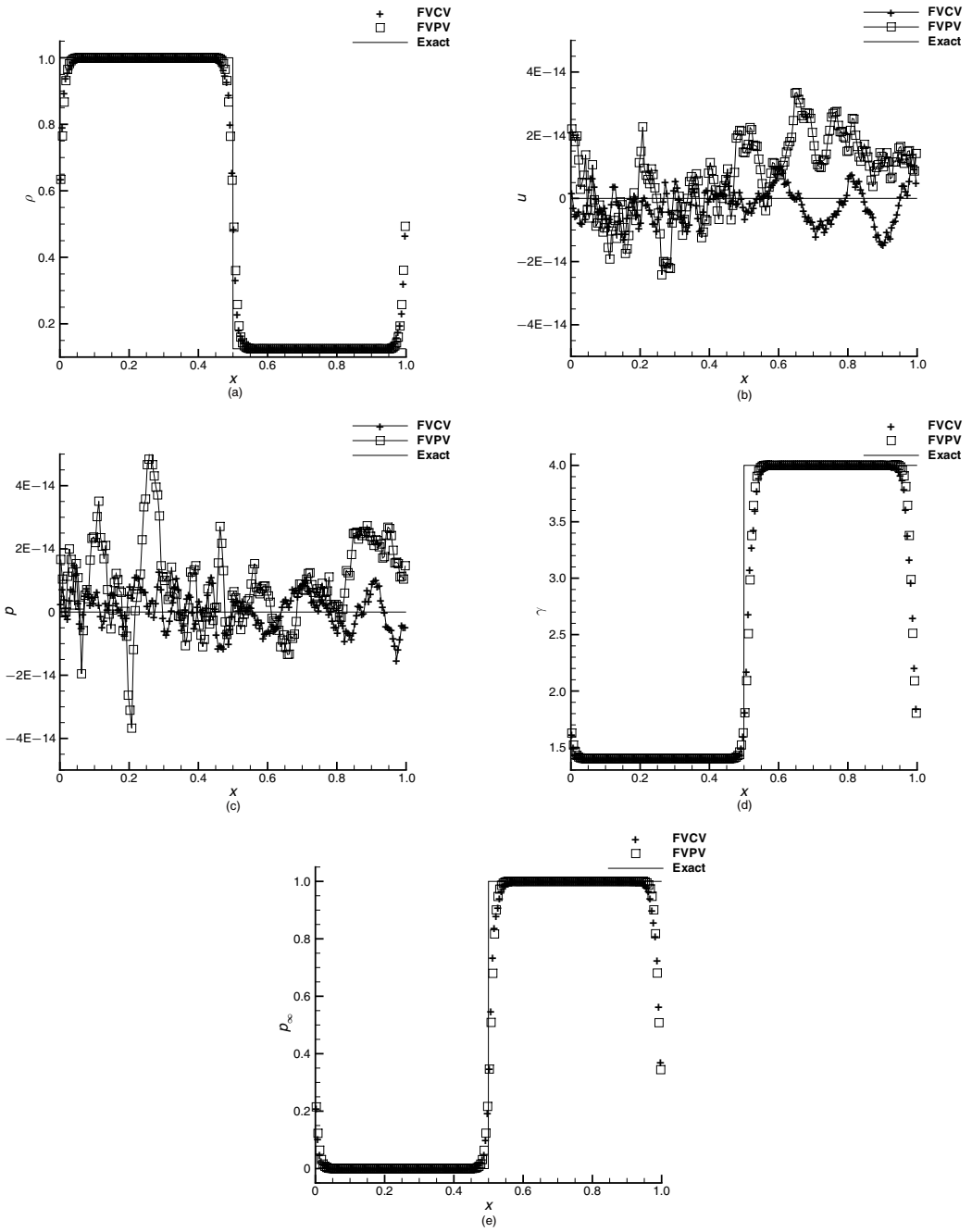

Figure 3 Numerical results for Example 3. From (a)–(e): figures of density, velocity, pressure, γ, and p_∞. The base $u_0 = 1$ and $p_0 = 1$

Example 6. This example is taken from [56]. Now, we consider the following similar test:

$$(\rho, u, p, \gamma, p_\infty) = \begin{cases} (1, 0, 1000.0, 1.4, 0), & x < 0.1, \\ (1, 0, 0.01, 1.4, 0), & 0.1 \leqslant x < 0.5, \\ (1, 0, 0.01, 5/3, 0), & 0.5 \leqslant x < 0.9, \\ (1, 0, 100.0, 5/3, 0), & x \geqslant 0.9. \end{cases}$$

The computational domain is $[0, 1]$. We compute the solution of this problem to $T = 0.038$ with $N = 200$. The reference solution is obtained by the same method with $N = 10000$. We show the final results in Figure 6. From the figures, we can see that our method can obtain satisfying results.

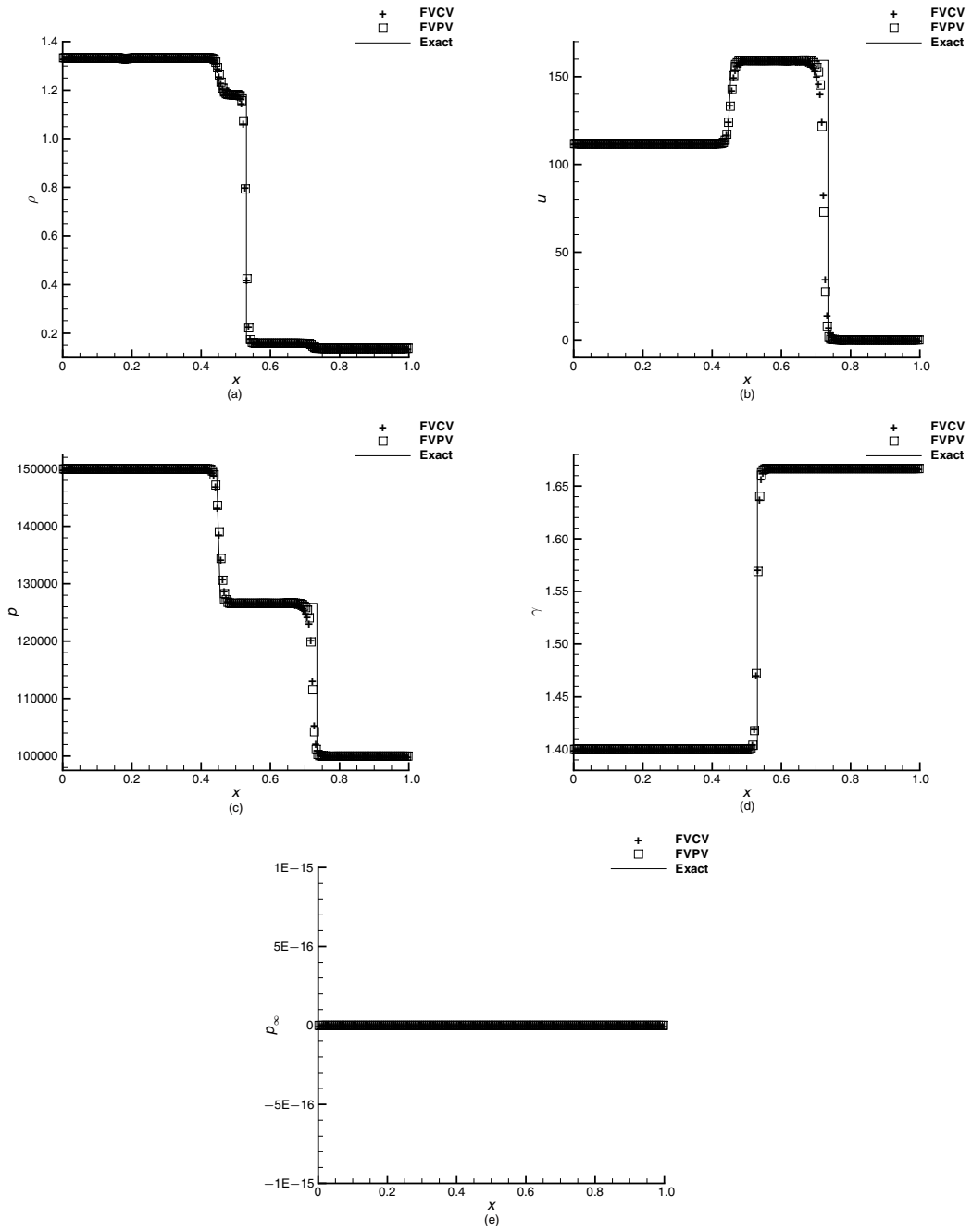

Figure 4 Numerical results for Example 4. From (a)–(e): figures of density, velocity, pressure, γ, and p_∞

Example 7. We consider the following initial conditions:

$$(\rho, u, p, \gamma, p_\infty) = \begin{cases} (1270, 0, 8 \times 10^8, 1.4, 0), & x < 0.5, \\ (1000, 0, 10^5, 7.15, 3.309 \times 10^8), & x \geqslant 0.5. \end{cases}$$

The example is taken from [37]. The domain is $[0, 1]$, the grid number is $N = 200$, and the final time is $T = 0.00016$. We list the results using the present method and the one in [25] in Figure 7. It is clear that both methods can capture the features of the exact solution very well.

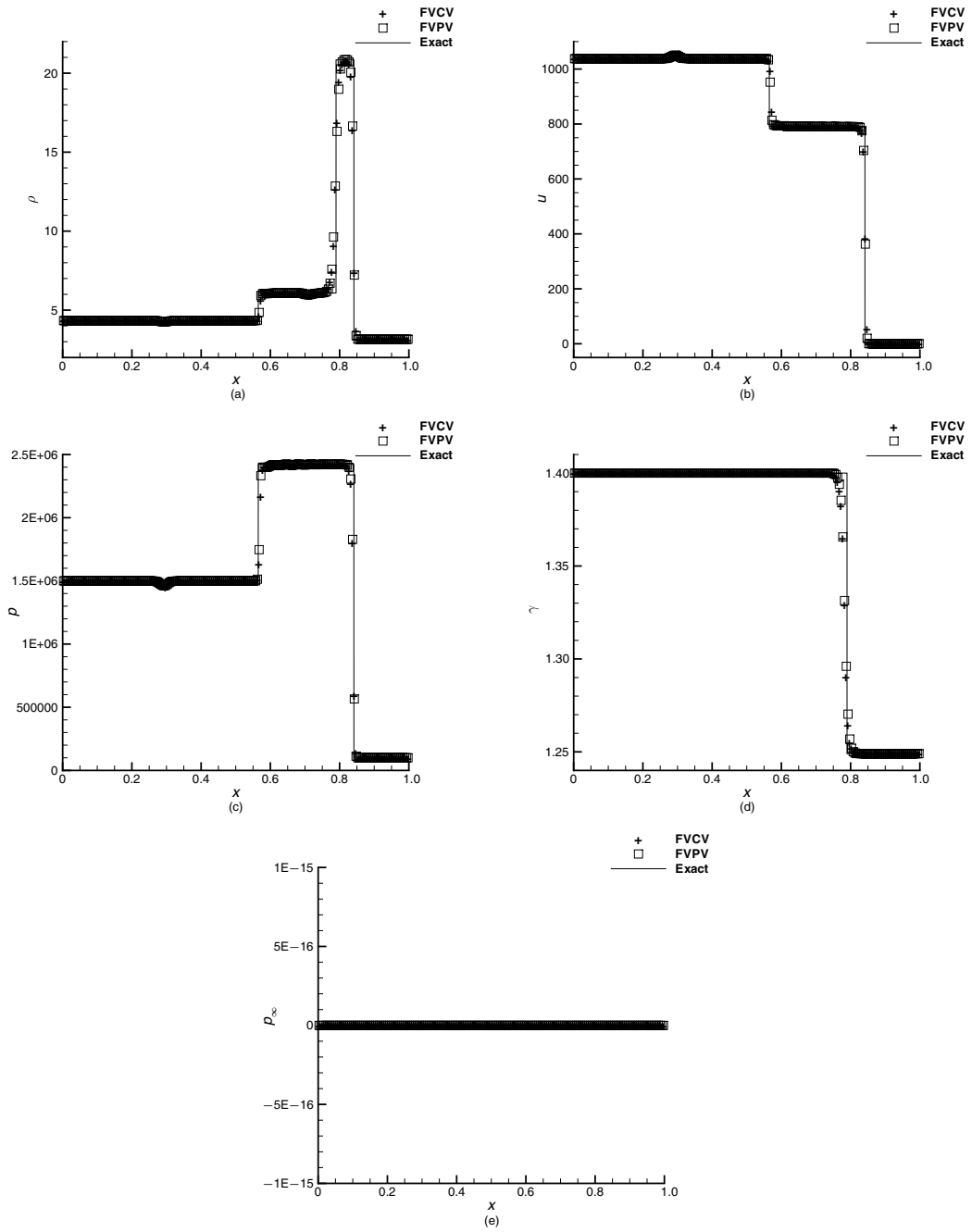

Figure 5 Numerical results for Example 5. From (a)–(e): figures of density, velocity, pressure, γ, and p_∞

Example 8. We consider the following initial conditions:

$$(\rho, u, p, \gamma, p_\infty) = \begin{cases} (1630, 0, 7.81 \times 10^9, 1.4, 0), & x < 0.5, \\ (1000, 0, 10^5, 7.15, 3.309 \times 10^8), & x \geqslant 0.5. \end{cases}$$

The example is taken from [37]. The domain is $[0, 1]$ and the grid number $N = 200$. We list the results using both the current method and the one in [25] at time $t = 0.0001$ in Figure 8. From the figures, we can see that both methods obtain the correct interface location and high resolution.

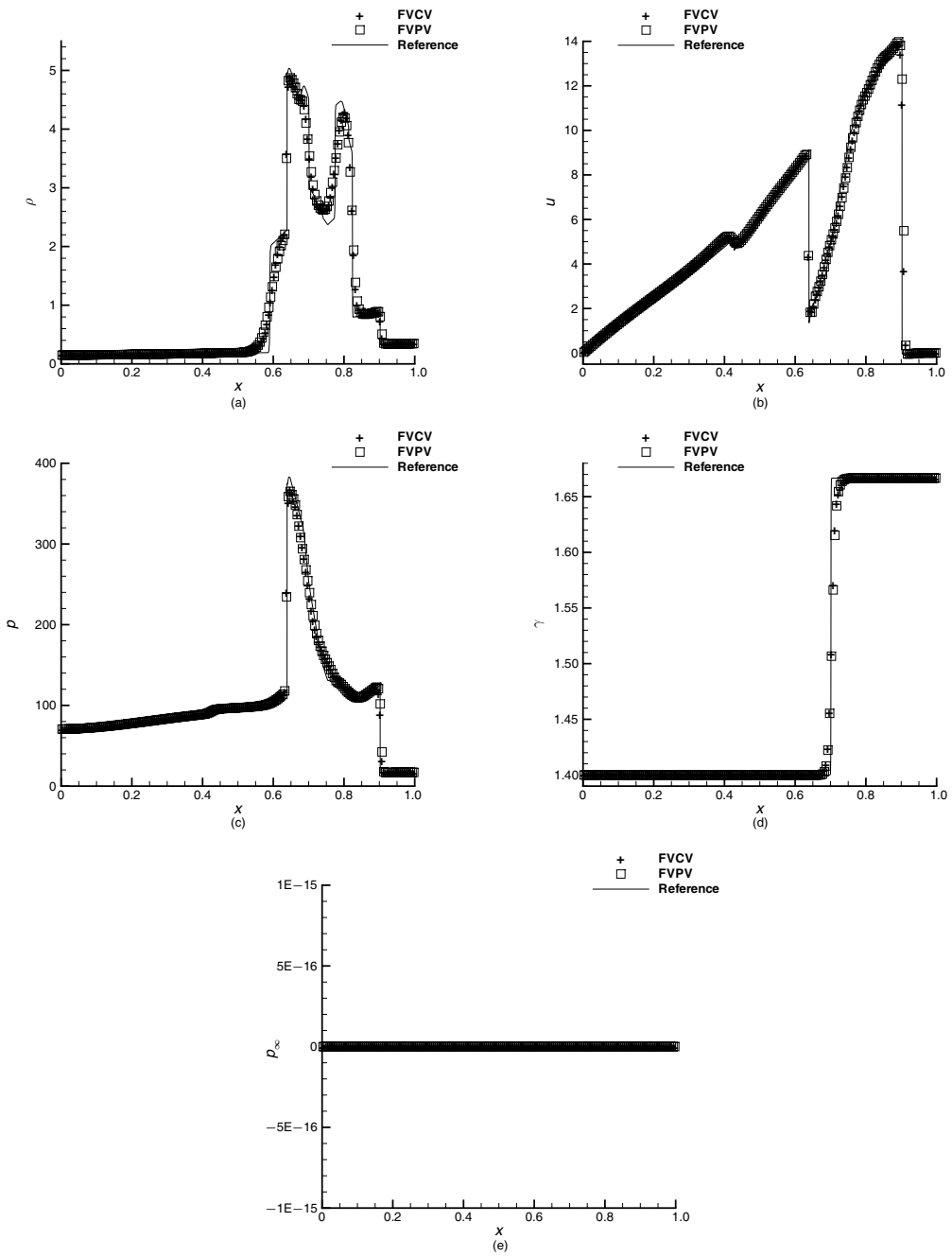

Figure 6 Numerical results for Example 6. From (a)–(e): figures of density, velocity, pressure, γ, and p_∞

Example 9. We consider the following initial conditions:

$$(\rho, u, p, \gamma, p_\infty) = \begin{cases} (1037.3620, 0.1897, 1000, 7.15, 3309), & x < 0.05, \\ (1000, 0, 1, 7.15, 3309), & 0.05 \leqslant x < 0.5, \\ (1, 0, 1, 1.4, 0), & x \geqslant 0.5. \end{cases}$$

The example is taken from [29]. The domain is $[0, 1]$ and the grid number is $N = 200$. In this case, a very strong rarefaction wave is reflected back into the water, which leads to the pressure being more than 500 times smaller than the incident shock pressure. We show the results at time $t = 0.15$ in Figure 9. From the figures, we can see that both methods can capture the main features of the solution well.

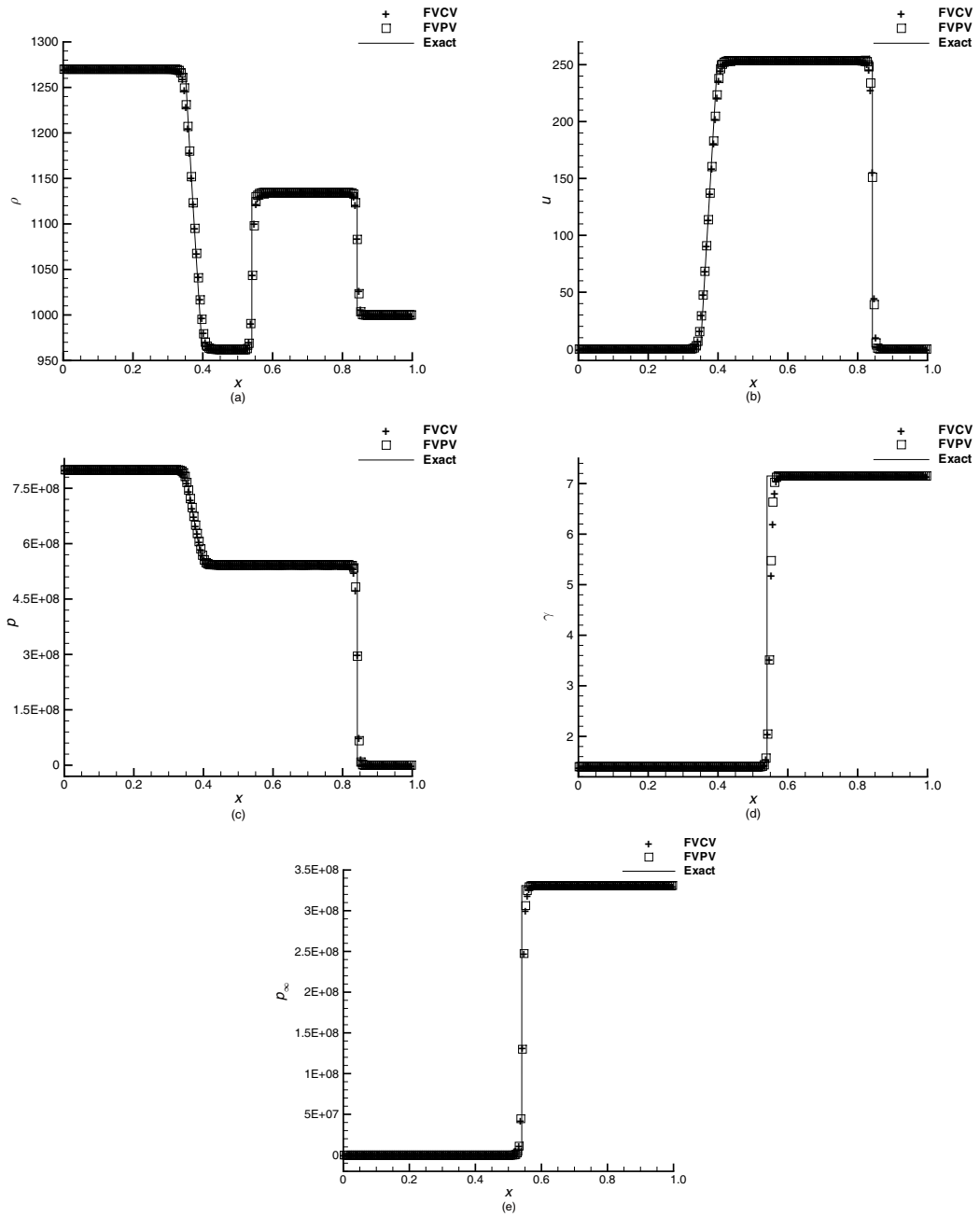

Figure 7 Numerical results for Example 7. From (a)–(e): figures of density, velocity, pressure, γ, and p_∞

Example 10. We consider the following initial conditions:

$$(\rho, u, p, \gamma, p_\infty) = \begin{cases} (5.9652, 28.8362, 1000, 1.4, 0), & x < 0.05, \\ (1, 0, 1, 1.4, 0), & 0.05 \leqslant x < 0.5, \\ (1000, 0, 1, 7.15, 3309), & x \geqslant 0.5. \end{cases}$$

The example is taken from [29]. The domain is $[0, 1]$ and the grid number is $N = 200$. In this case, a very strong shock wave is reflected back into the air. We plot the numerical results at time $t = 0.05$. From Figure 10 and zoom-in figure, it is evident that our method performs much better than the approach presented in [25] as it yields superior results and closely matches the exact solution.

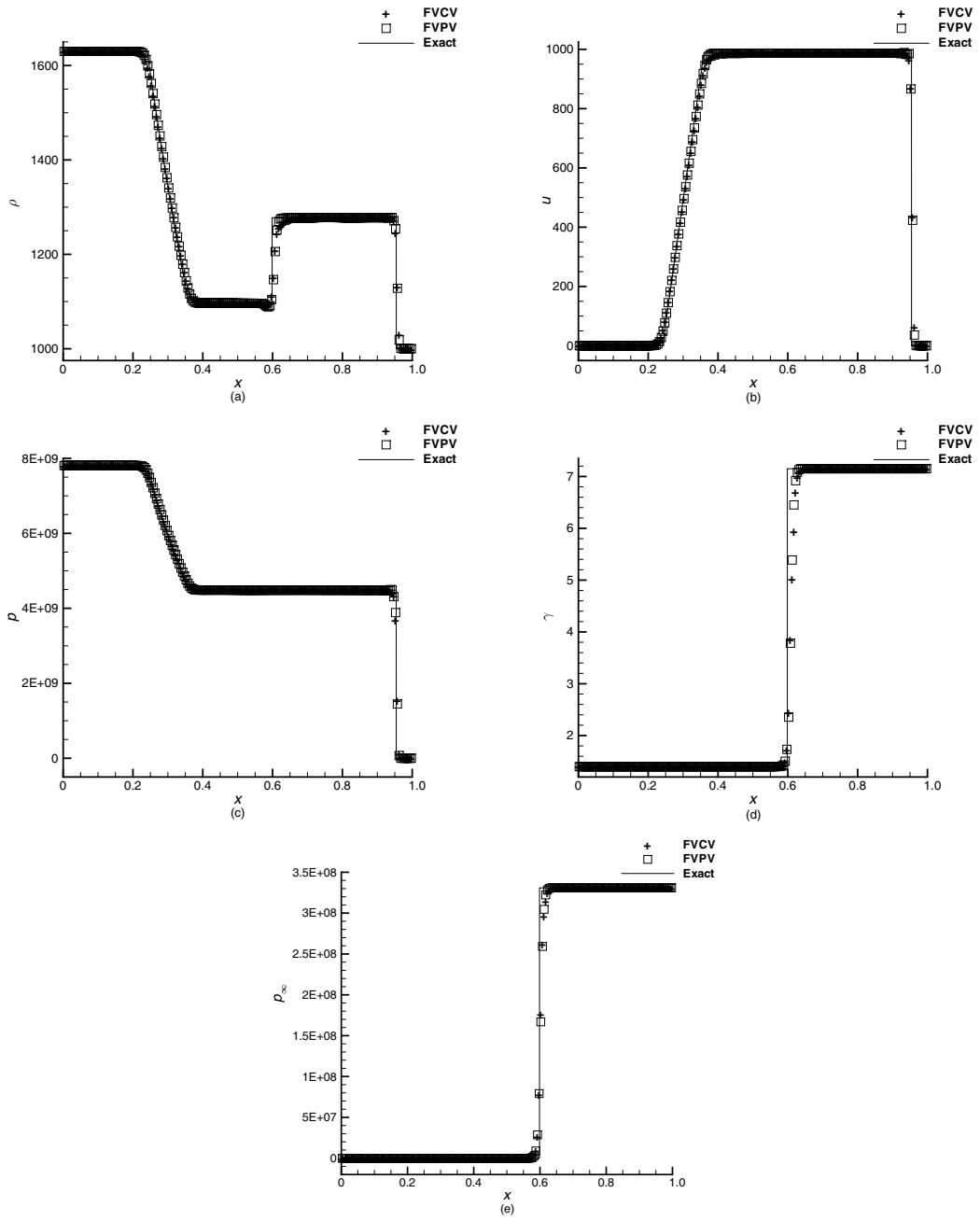

Figure 8 Numerical results for Example 8. From (a)–(e): figures of density, velocity, pressure, γ, and p_∞

Example 11. We consider the following two initial conditions:

$$I:(\rho,u,p,\gamma,p_\infty)=\begin{cases}(1000,0,10^9,4.4,6\times10^8), & x<0.5,\\ (50,0,10^5,1.4,0), & x\geqslant 0.5,\end{cases}$$

$$II:(\rho,u,p,\gamma,p_\infty)=\begin{cases}(1000,0,10^9,4.4,6\times10^8), & x<0.5,\\ (1,0,10^5,1.4,0), & x\geqslant 0.5.\end{cases}$$

The domain is $[-0.2,1]$ and the grid number is $N=200$. In this case, very strong shock waves are reflected back into the air. We plot the numerical results at time $t=0.0002$. It is a tough test, and many methods fail. From Figures 11 and 12, we can see that both methods exhibit overshoots or undershoots in the plots

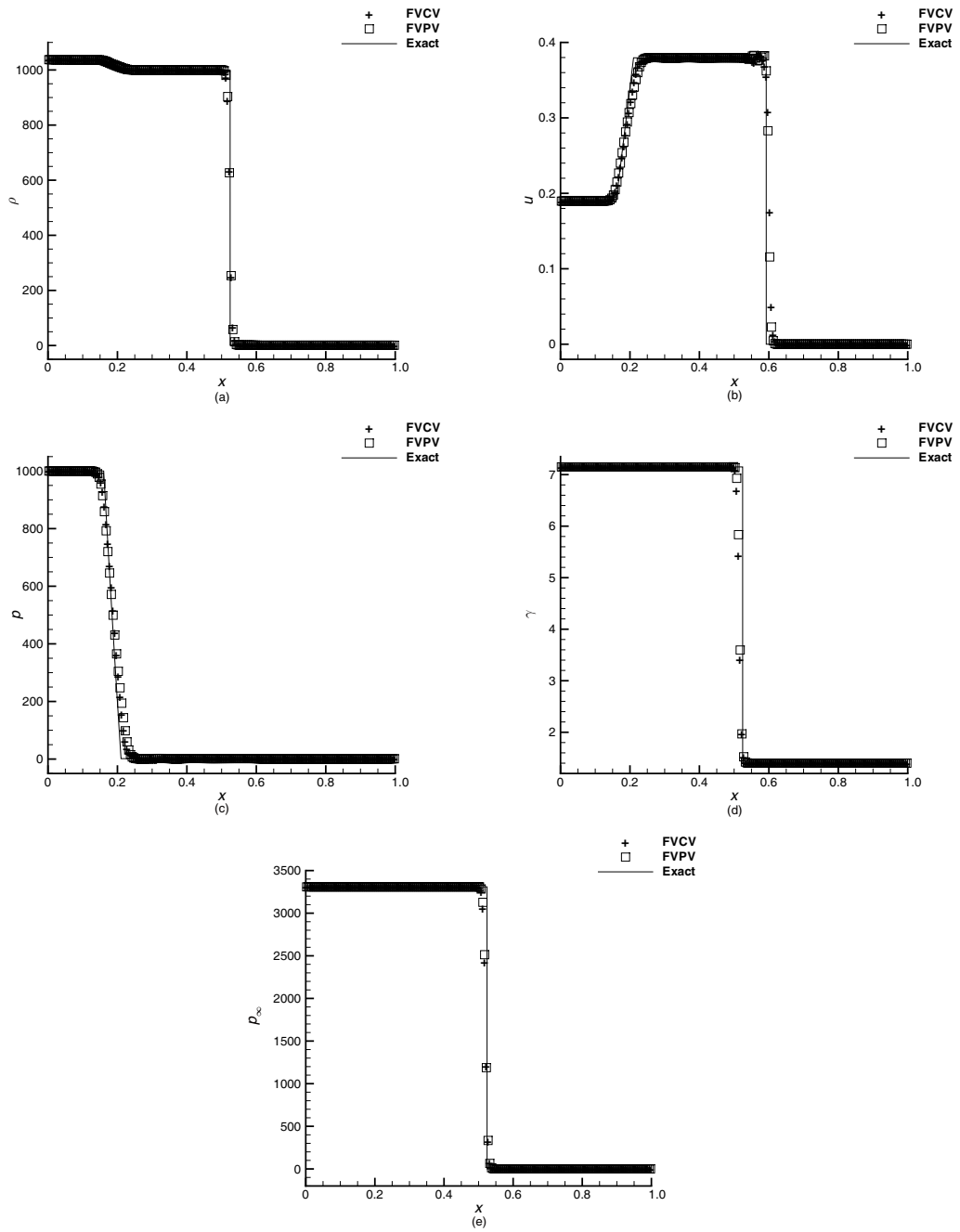

Figure 9 Numerical results for Example 9. From (a)–(e): figures of density, velocity, pressure, γ, and p_∞

of density velocity and pressure. Additionally, The shock locations in density and velocity plots are not well captured.

Example 12. We consider the following accuracy test:

$$\rho(x,y,0)=1+0.2\sin(\pi(x+y)),\quad u(x,y,0)=1,\quad v(x,y,0)=1,\quad p(x,y,0)=1.$$

We choose

$$\frac{1}{\gamma(x,y,0)-1}=\frac{1}{\gamma_L-1}\left(\frac{1}{2}+\frac{1}{2}\sin(\pi(x+y))\right)+\frac{1}{\gamma_R-1}\left(\frac{1}{2}-\frac{1}{2}\sin(\pi(x+y))\right)$$

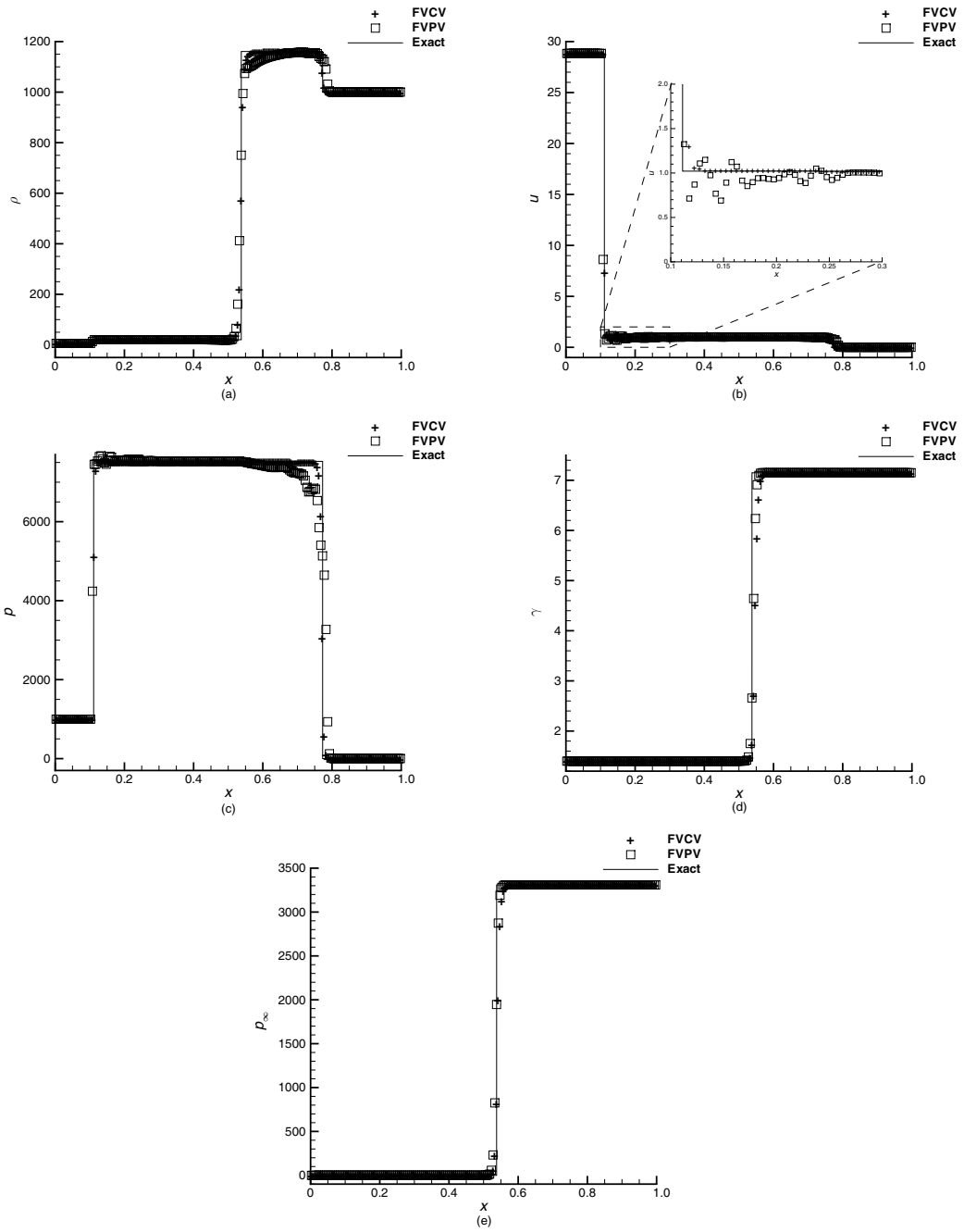

Figure 10 Numerical results for Example 10. From (a)–(e): figures of density, velocity, pressure, γ, and p_∞

and

$$\frac{\gamma(x,y,0)p_\infty(x,y,0)}{\gamma(x,y,0)-1}=\frac{\gamma_L p_{\infty L}}{\gamma_L-1}\left(\frac{1}{2}+\frac{1}{2}\sin(\pi(x+y))\right)+\frac{\gamma_R p_{\infty R}}{\gamma_R-1}\left(\frac{1}{2}-\frac{1}{2}\sin(\pi(x+y))\right),$$

where $\gamma_L=1.4$, $\gamma_R=4$ and $p_{\infty L}=0$, $p_{\infty R}=1$. The computational domain is $[0,2]\times[0,2]$. Periodic boundary conditions are used in this test. The exact solution of ρ is

$$\rho(x,t)=1+0.2\sin(\pi(x+y-2t)).$$

We set the final time $t=1$. We list the error in Table 3. We can also see that our method achieves the designed fifth-order accuracy.

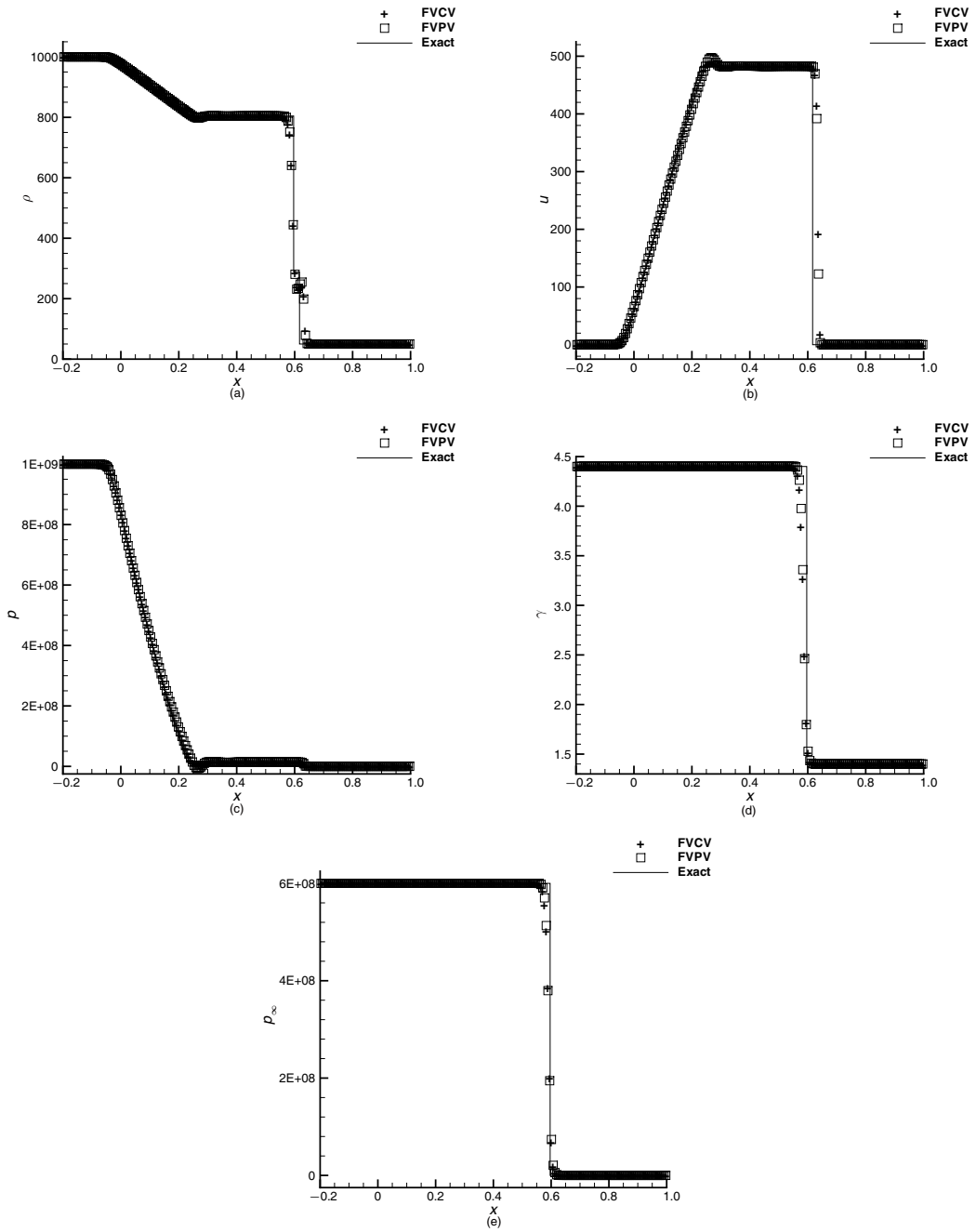

Figure 11 Numerical results for the initial condition I in Example 11. From (a)–(e): figures of density, velocity, pressure, γ, and p_∞

Example 13. We consider the 2D artificial accuracy test. We take $\gamma = 3$ and $p_\infty = 0$. Then, the system (2.1) becomes a single-component problem. Furthermore, we choose the following special initial conditions:

$$\rho(x,y,0) = \frac{1+0.2\sin(\frac{x+y}{2})}{\sqrt{2\gamma}}, \quad u(x,y,0) = v(x,y,0) = \sqrt{\frac{\gamma}{2}}\rho(x,y,0), \quad p(x,y,0) = \rho(x,y,0)^\gamma.$$

The computational domain is $[0,4\pi]\times[0,4\pi]$. Periodic boundary conditions are used in this test. By the special choice of parameter γ and p_∞, initial conditions, and boundary conditions, we can verify that

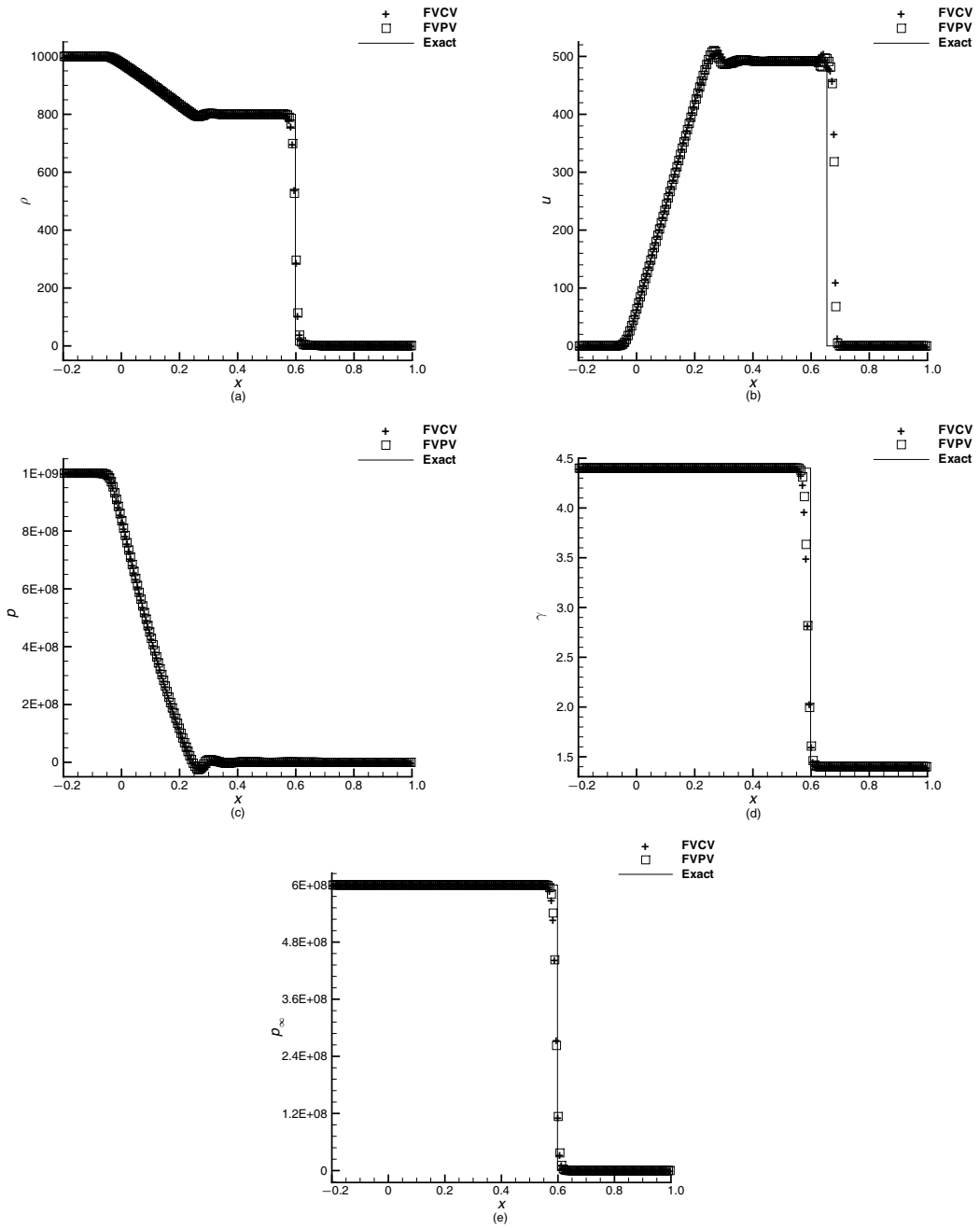

Figure 12 Numerical results for the initial condition II in Example 11. From (a)–(e): figures of density, velocity, pressure, γ, and p_∞

Table 3 The accuracy test for the Euler system in 2D

Mesh size	L_∞ error	Order	L_2 error	Order	L_1 error	Order
10× 10	2.24E−03		1.54E−03		1.33E−03	
20× 20	6.79E−05	5.04	4.46E−05	5.11	3.99E−05	5.06
40× 40	2.08E−06	5.03	1.41E−06	4.98	1.27E−06	4.97
80× 80	6.43E−08	5.01	4.42E−08	5.00	3.98E−08	4.99
160× 160	2.00E−09	5.01	1.38E−09	5.00	1.25E−09	5.00
320× 320	6.22E−11	5.00	4.32E−11	5.00	3.89E−11	5.00

$\sqrt{2\gamma}\rho(x,y,t)$ is the exact solution of the following Burgers equation:

$$\mu_t + \frac{1}{2}(\mu^2)_x + \frac{1}{2}(\mu^2)_y = 0, \quad \mu(x,y,0) = 1 + 0.2\sin\left(\frac{x+y}{2}\right).$$

The velocity and pressure satisfy the relation

$$u(x,y,t) = v(x,y,t) = \sqrt{\frac{\gamma}{2}}\rho(x,y,t),$$
$$p(x,y,t) = \rho(x,y,t)^\gamma.$$

It is easy to verify that the solution of the above Burgers equation is smooth up to time $T = 5$. We set the final time $T = 3$. At this time, the solution is still smooth. We list the error and numerical accuracy order in Table 4. We can see that our method can achieve the designed fifth-order accuracy.

Example 14. We consider the single-mode Richtmyer-Meshkov instability, which has been studied in [9,35]. The schematic for this problem is given in Figure 13.

The computational domain is $[0,4]\times[0,1]$. The initial position of the air-SF_6 interface is located at $x = 2.9 - 0.1\sin(2\pi(y+0.25))$. A shock wave with the Mach number 1.24 at $x = 3.2$ in air is moving from right to left. It collides with the interface and triggers the instability. The initial conditions are

$$(\rho,u,v,p,\gamma,p_\infty) = \begin{cases} (5.04,0,0,1.0,1.093,0), & SF_6, \\ (1.0,0,0,1.0,1.4,0), & \text{pre-shock in air}, \\ (1.4112,-0.4275,0,1.6272,1.4,0), & \text{post-shock in air}. \end{cases}$$

The non-reflection boundary conditions are applied in the x-direction, and periodical boundary conditions are used in the y-direction. In Figure 14, we show that the density solution contours at different times using 1200×300 cells. We can see that when the shock moves from the light into the heavy gas region, it generates vorticity and rollups due to the initial perturbation of the interface. In Figure 15, the history of the leading edges for the bubble x_b and spike x_s along with the thickness of the mixing layer defined as $h_{\text{mix}} = x_s - x_b$ are presented for three different grid resolutions: 300×75, 600×150, and 1200×300. The results are consistent across the three different grid sizes.

Table 4 The accuracy test for the density in 2D

Mesh size	L_∞ error	Order	L_2 error	Order	L_1 error	Order
10× 10	2.33E−02		1.13E−02		8.67E−03	
20× 20	1.04E−02	1.17	2.64E−03	2.10	1.39E−03	2.64
40× 40	1.64E−03	2.66	4.32E−04	2.61	2.13E−04	2.71
80× 80	2.27E−04	2.85	4.24E−05	3.35	1.64E−05	3.70
160× 160	6.61E−06	5.10	1.01E−06	5.39	3.50E−07	5.55
320× 320	2.00E−07	5.04	3.06E−08	5.05	9.17E−09	5.26

Figure 13 Schematic for Example 14

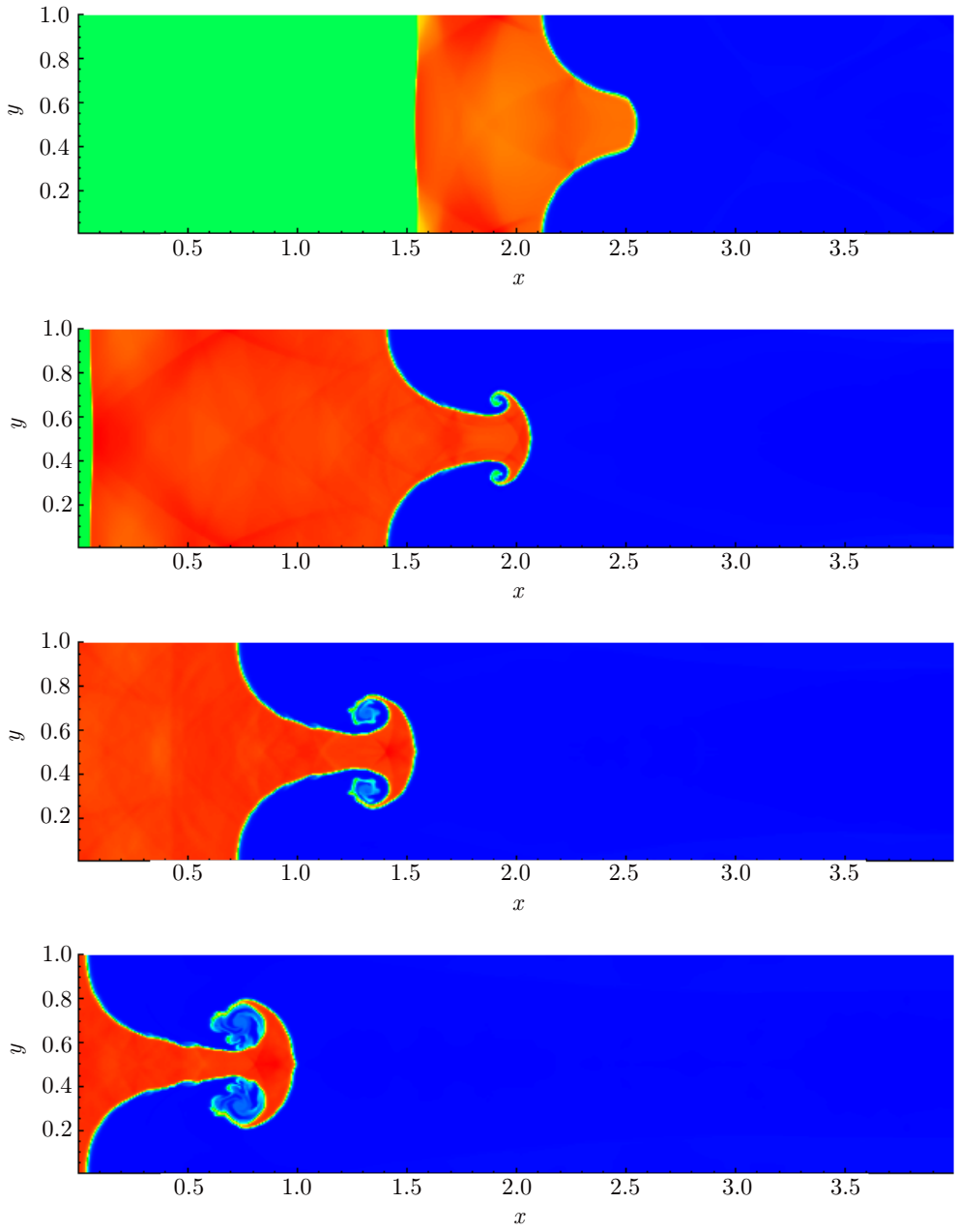

Figure 14 (Color online) Density contours for Example 14. From top to bottom: $t = 2.31$s, $t = 4.62$s, $t = 6.93$s, and $t = 9.24$s

Example 15. We consider a weak shock in the air interacting with a bubble of Refrigerant-22 (R_{22}). This problem has been studied experimentally in [20] and numerically in [15, 35, 53]. The schematic for this problem is given in Figure 16. The computational domain is $[0, 0.295] \times [0, 0.089]$. A R_{22} bubble with 0.05 diameter is initially located at $(0.16, 0.0445)$. A planar shock in the air with the Mach number 1.22 moves from right to left and strikes the bubble. The initial conditions are

$$(\rho, u, v, p, \gamma, p_\infty) = \begin{cases} (3.1539, 0, 0, 10^5, 1.249, 0), & R_{22}, \\ (1, 0, 0, 10^5, 1.4, 0), & \text{pre-shock in air}, \\ (1.3764, -124.8241, 0, 1.5698 \times 10^5, 1.4, 0), & \text{post-shock in air}. \end{cases}$$

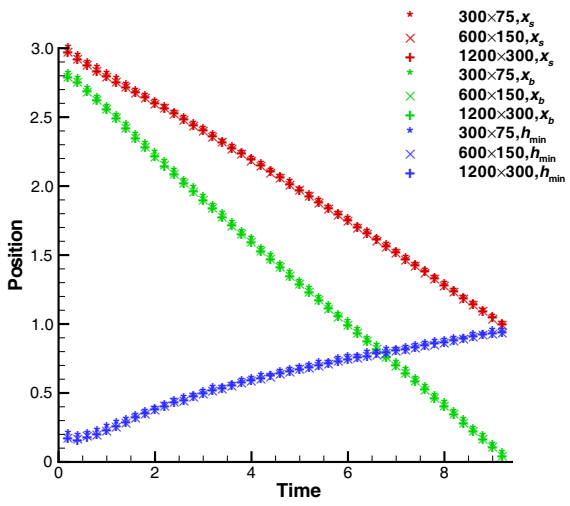

Figure 15 (Color online) Positions of the bubble and spike and thickness of the mixing layer for Example 14

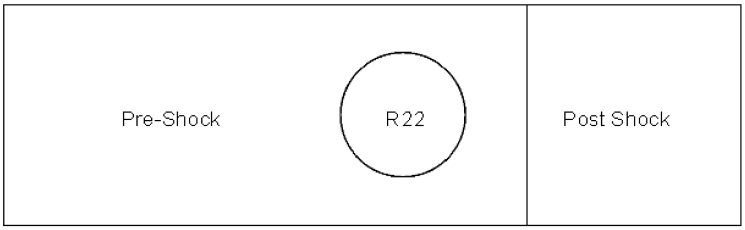

Figure 16 Schematic for Example 15

The non-reflection boundary conditions are applied in the x-direction, and periodic boundary conditions are used in the y-direction. We show that the density solution numerical schlieren images at different times using 2360×712 cells, which is approximately 400 cells per initial bubble's diameter. From Figures 17 and 18, we can see that our simulations are comparable with those in [35], especially in resolving the shape of the bubble, the transmitted, reflection wave, the interface instabilities, and the R_{22} jet.

Example 16. We consider a model underwater explosion problem. This problem has been studied in [35, 47]. The schematic for this problem is given in Figure 19. The computational domain is $[-2, 2] \times [-1.5, 1]$. A horizontal air-water interface is located at $y = 0$. The explosion zone is represented as a highly pressurized gas bubble in water with the center $(x_0, y_0) = (0, -0.3)$ and the radius $r_0 = 0.12$. The initial conditions are

$$(\rho, u, v, p, \gamma, p_\infty) = \begin{cases} (1.225, 0, 0, 101325, 1.4, 0), & \text{air}_1, \\ (1250, 0, 0, 10^9, 1.4, 0), & \text{air}_2, \\ (1000, 0, 0, 101325, 4.4, 6 \times 10^8), & \text{water}. \end{cases}$$

The non-reflection boundary conditions are applied at the top, left, and right boundaries of the computational domain. Reflection boundary conditions are used at the bottom. In Figure 20, we show the density and pressure solution numerical schlieren images at different times using 800×500 cells. Initially, all the fluids are in an equilibrium state. Due to the pressure difference between the fluids, it results in the formation of out-going shock waves and in-going rarefaction waves. The interface between gas and water separates the two fluids and changes shape from a circular to an oval shape after diffraction through the nearby air-water interface. As time evolves, the gas bubble continues to rise upwards, causing

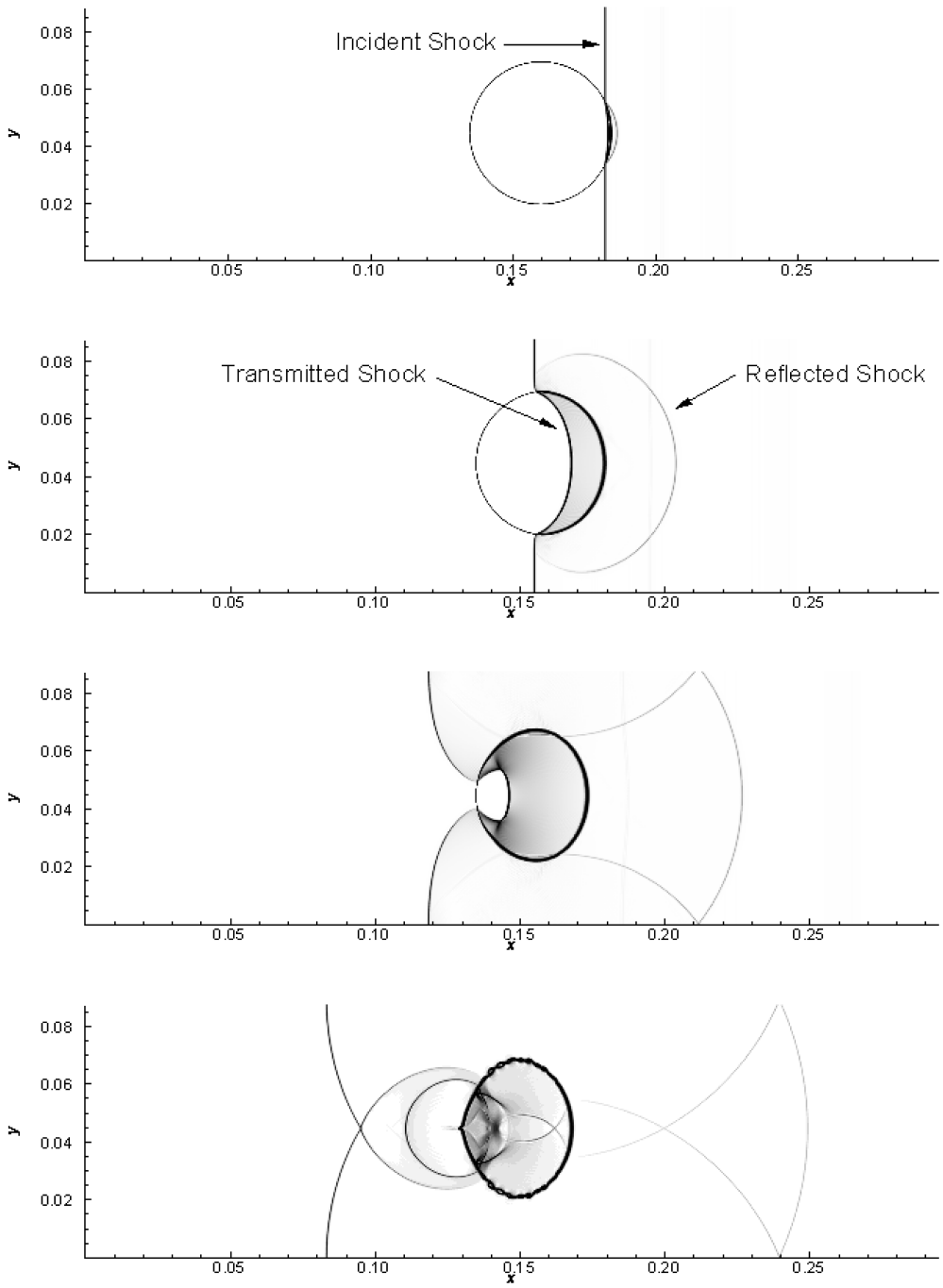

Figure 17 Numerical schlieren images for Example 15. From top to bottom: $t = 60\mu$s, $t = 120\mu$s, $t = 200\mu$s, and $t = 280\mu$s

further deformation of the horizontal air-water interface. Compared with the numerical results in [35,47], our scheme successfully captures all the major features and presents a satisfying result.

4 Conclusion

In this paper, a high-order finite volume method is constructed for solving multicomponent fluid problems. The key is that the conservative variables instead of primitive variables are reconstructed in a

characteristic manner, and the source term and numerical fluxes are designed properly. These ingredients enable our scheme to achieve high-order accuracy in the smooth region and the high resolution when discontinuities appear. We have proved that the scheme can maintain the equilibrium of the pressure and velocity for the interface-only problem and preserve the equilibrium of physical parameters in a single-component fluid. Extensive numerical tests have been performed to verify the high resolution and high accuracy of the scheme.

The research on coupling our scheme with a more complex equation of state is ongoing.

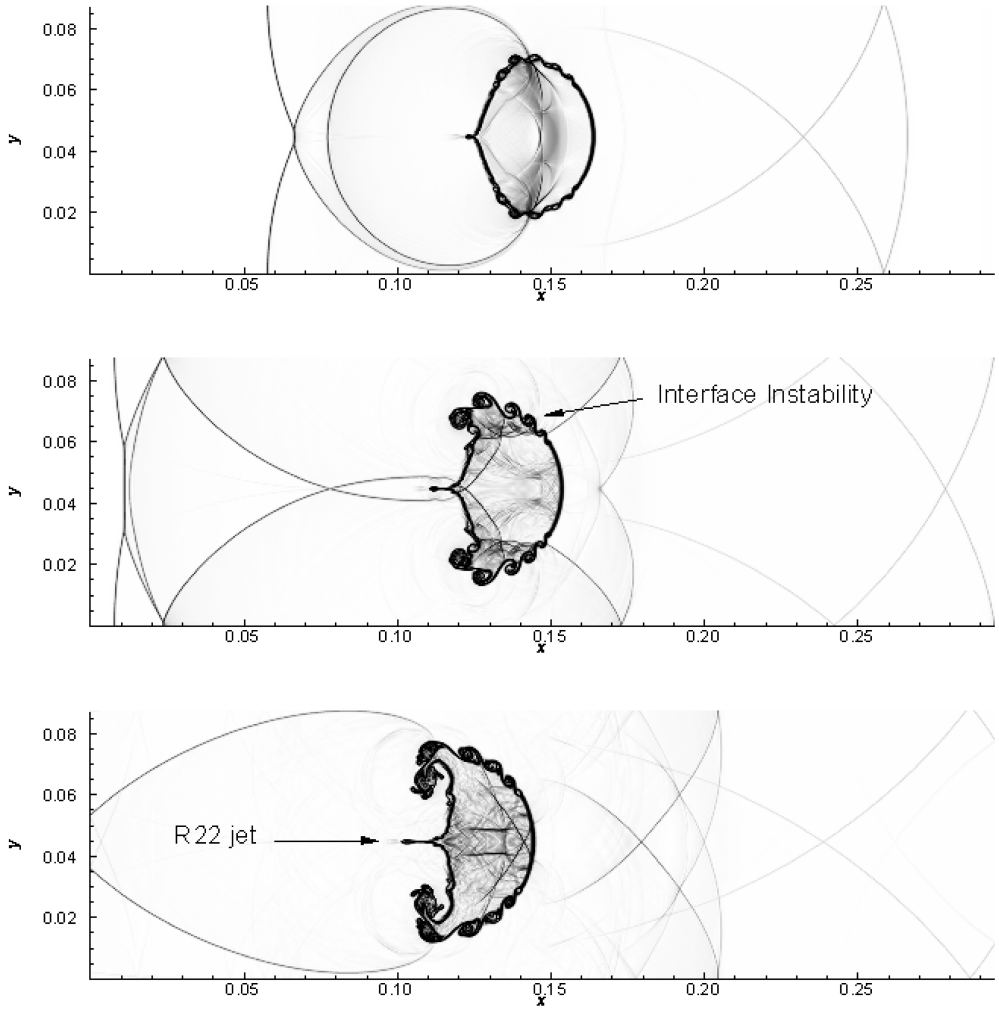

Figure 18 Numerical schlieren images for Example 15. From top to bottom: $t = 340\mu$s, $t = 460\mu$s, and $t = 560\mu$s

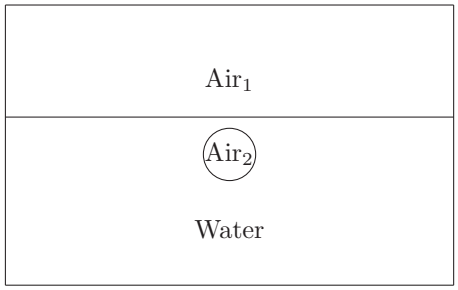

Figure 19 Schematic for Example 16

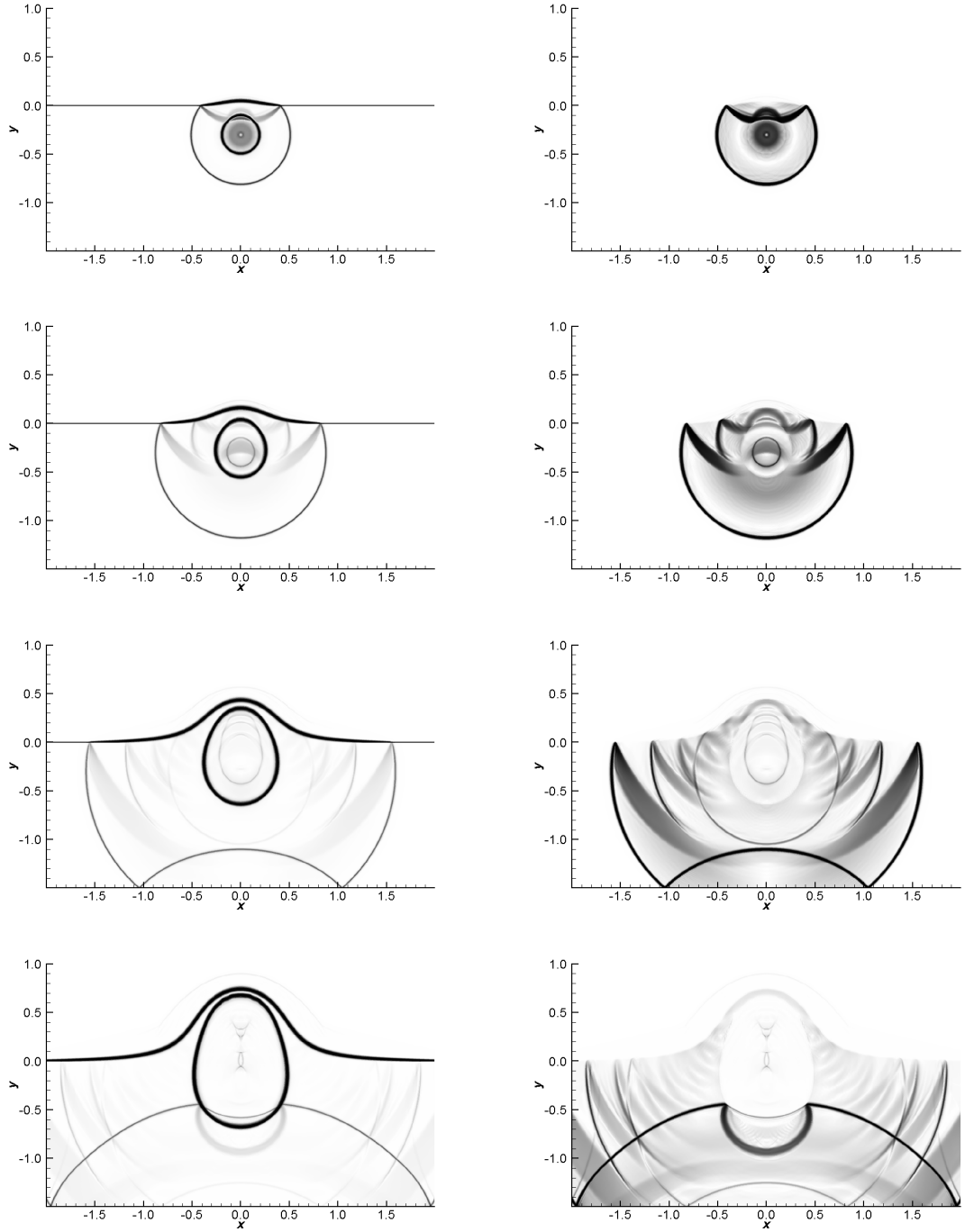

Figure 20 Numerical schlieren images for Example 16. Left: numerical schlieren images for density. Right: numerical schlieren images for pressure. From top to bottom: $t = 0.2$ms, $t = 0.4$ms, $t = 0.8$ms, and $t = 1.2$ms

Acknowledgements The first author was supported by National Natural Science Foundation of China (Grant No. 12101128). The second author was supported by Natural Science Foundation of Fujian Province (Grant No. 2023J02003) and National Natural Science Foundation of China (Grant No. 12071392). The authors thank the referees for their helpful suggestions to improve the paper.

References

1 Abgrall R. How to prevent pressure oscillations in multicomponent flow calculations: A quasi conservative approach. J Comput Phys, 1996, 125: 150–160
2 Abgrall R. Essentially non-oscillatory residual distribution schemes for hyperbolic problems. J Comput Phys, 2006, 214: 773–808
3 Abgrall R, Barth T. Residual distribution schemes for conservation laws via adaptive quadrature. SIAM J Sci Comput, 2002, 24: 732–769
4 Abgrall R, Saurel R. Discrete equations for physical and numerical compressible multiphase mixtures. J Comput Phys, 2003, 186: 361–396
5 Allaire G, Clerc S, Kokh S. A five-equation model for the simulation of interfaces between compressible fluids. J Comput Phys, 2002, 181: 577–616
6 Anbarlooei H R, Mazaheri K. Moment of fluid interface reconstruction method in multi-material arbitrary Lagrangian Eulerian (MMALE) algorithms. Comput Methods Appl Mech Engrg, 2009, 198: 3782–3794
7 Barlow A J, Maire P H, Rider W J, et al. Arbitrary Lagrangian-Eulerian methods for modeling high-speed compressible multimaterial flows. J Comput Phys, 2016, 322: 603–665
8 Cheng J, Shu C W. Positivity-preserving Lagrangian scheme for multi-material compressible flow. J Comput Phys, 2014, 257: 143–168
9 Cheng J, Zhang F, Liu T. A discontinuous Galerkin method for the simulation of compressible gas-gas and gas-water two-medium flows. J Comput Phys, 2020, 403: 109059
10 Cheng J, Zhang F, Liu T. A quasi-conservative discontinuous Galerkin method for solving five equation model of compressible two-medium flows. J Sci Comput, 2020, 85: 12
11 Coralic V, Colonius T. Finite-volume WENO scheme for viscous compressible multicomponent flows. J Comput Phys, 2014, 274: 95–121
12 Deng X L, Li M. Simulating compressible two-medium flows with sharp-interface adaptive Runge-Kutta discontinuous Galerkin methods. J Sci Comput, 2018, 74: 1347–1368
13 Dumbser M, Enaux C, Toro E F. Finite volume schemes of very high order of accuracy for stiff hyperbolic balance laws. J Comput Phys, 2008, 227: 3971–4001
14 Dumbser M, Hidalgo A, Zanotti O. High order space-time adaptive ADER-WENO finite volume schemes for non-conservative hyperbolic systems. Comput Methods Appl Mech Engrg, 2014, 268: 359–387
15 Fedkiw R P, Aslam T, Merriman B, et al. A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method). J Comput Phys, 1999, 152: 457–492
16 Gaburro E, Boscheri W, Chiocchetti S, et al. High order direct arbitrary-Lagrangian-Eulerian schemes on moving Voronoi meshes with topology changes. J Comput Phys, 2020, 407: 109167
17 Gu Y, Gao Z, Hu G, et al. A robust high order alternative WENO scheme for the five-equation model. J Sci Comput, 2021, 88: 12
18 Gu Y, Gao Z, Hu G, et al. High order finite difference alternative WENO scheme for multi-component flows. J Sci Comput, 2021, 89: 52
19 Ha C T, Kim D H, Park W G, et al. A compressive interface-capturing scheme for computation of compressible multi-fluid flows. Comput Fluids, 2017, 152: 164–181
20 Haas J F, Sturtevant B. Interaction of weak shock waves with cylindrical and spherical gas inhomogeneities. J Fluid Mech, 1987, 181: 41–76
21 He Z, Tian B, Zhang Y, et al. Characteristic-based and interface-sharpening algorithm for high-order simulations of immiscible compressible multi-material flows. J Comput Phys, 2017, 333: 247–268
22 Hirt C W, Amsden A A, Cook J L. An arbitrary Lagrangian-Eulerian computing method for all flow speeds. J Comput Phys, 1974, 14: 227–253
23 Hu X Y, Khoo B C, Adams N A, et al. A conservative interface method for compressible flows. J Comput Phys, 2006, 219: 553–578
24 Jiang G S, Shu C W. Efficient implementation of weighted ENO schemes. J Comput Phys, 1996, 126: 202–228
25 Johnsen E, Colonius T. Implementation of WENO schemes in compressible multicomponent flow problems. J Comput Phys, 2006, 219: 715–732
26 Karni S. Multicomponent flow calculations by a consistent primitive algorithm. J Comput Phys, 1994, 112: 31–43
27 Li S, Luo D, Qiu J, et al. A one-stage high-order gas-kinetic scheme for multi-component flows with interface-sharpening technique. J Comput Phys, 2023, 490: 112318
28 Lin J Y, Shen Y, Ding H, et al. Simulation of compressible two-phase flows with topology change of fluid-fluid interface by a robust cut-cell method. J Comput Phys, 2017, 328: 140–159
29 Liu T G, Khoo B C, Yeo K S. Ghost fluid method for strong shock impacting on material interface. J Comput Phys, 2003, 190: 651–681
30 Liu X, Zhang D, Meng X, et al. Superconvergence of local discontinuous Galerkin methods with generalized alternating

fluxes for 1D linear convection-diffusion equations. Sci China Math, 2021, 64: 1305–1320
31 Luo J, Hu X Y, Adams N A. A conservative sharp interface method for incompressible multiphase flows. J Comput Phys, 2015, 284: 547–565
32 Marboeuf A, Claisse A, Le Tallec P. Conservative and entropy controlled remap for multi-material ALE simulations with space-staggered schemes. J Comput Phys, 2019, 390: 66–92
33 Marquina A, Mulet P. A flux-split algorithm applied to conservative models for multicomponent compressible flows. J Comput Phys, 2003, 185: 120–138
34 Mulder W, Osher S, Sethian J A. Computing interface motion in compressible gas dynamics. J Comput Phys, 1992, 100: 209–228
35 Nourgaliev R R, Dinh T N, Theofanous T G. Adaptive characteristics-based matching for compressible multifluid dynamics. J Comput Phys, 2006, 213: 500–529
36 Pan S, Han L, Hu X, et al. A conservative interface-interaction method for compressible multi-material flows. J Comput Phys, 2018, 371: 870–895
37 Qiu J, Liu T, Khoo B C. Runge-Kutta discontinuous Galerkin methods for compressible two-medium flow simulations: One-dimensional case. J Comput Phys, 2007, 222: 353–373
38 Qiu J M, Christlieb A. A conservative high order semi-Lagrangian WENO method for the Vlasov equation. J Comput Phys, 2010, 229: 1130–1149
39 Qiu J M, Shu C W. Conservative high order semi-Lagrangian finite difference WENO methods for advection in incompressible flow. J Comput Phys, 2011, 230: 863–889
40 Saurel R, Abgrall R. A multiphase Godunov method for compressible multifluid and multiphase flows. J Comput Phys, 1999, 150: 425–467
41 Saurel R, Pantano C. Diffuse-interface capturing methods for compressible two-phase flows. Annu Rev Fluid Mech, 2018, 50: 105–130
42 Saurel R, Petitpas F, Berry R A. Simple and efficient relaxation methods for interfaces separating compressible fluids, cavitating flows and shocks in multiphase mixtures. J Comput Phys, 2009, 228: 1678–1712
43 Shu C W, Osher S. Efficient implementation of essentially non-oscillatory shock-capturing schemes. J Comput Phys, 1988, 77: 439–471
44 Shyue K M. An efficient shock-capturing algorithm for compressible multicomponent problems. J Comput Phys, 1998, 142: 208–242
45 Shyue K M. A fluid-mixture type algorithm for compressible multicomponent flow with van der Waals equation of state. J Comput Phys, 1999, 156: 43–88
46 Shyue K M. A fluid-mixture type algorithm for compressible multicomponent flow with Mie-Grüneisen equation of state. J Comput Phys, 2001, 171: 678–707
47 Shyue K M. A wave-propagation based volume tracking method for compressible multicomponent flow in two space dimensions. J Comput Phys, 2006, 215: 219–244
48 Smolarkiewicz P K, Margolin L G. MPDATA: A finite-difference solver for geophysical flows. J Comput Phys, 1998, 140: 459–480
49 Toro E F. Riemann Solvers and Numerical Methods for Fluid Dynamics. Berlin: Springer, 2009
50 Toro E F, Millington R C, Nejad L A M. Towards very high order Godunov schemes. In: Godunov Methods Theory and Applications. New York: Kluwer/Plenum, 2001, 907–940
51 Villedieu N, Quintino T, Ricchiuto M, et al. Third order residual distribution schemes for the Navier-Stokes equations. J Comput Phys, 2011, 230: 4301–4315
52 VonNeumann J, Richtmyer R D. A method for the numerical calculation of hydrodynamic shocks. J Appl Phys, 1950, 21: 232–237
53 Wang B, Xiang G, Hu X Y. An incremental-stencil WENO reconstruction for simulation of compressible two-phase flows. Int J Multiph Flow, 2018, 104: 20–31
54 Wang C W, Liu T G, Khoo B C. A real ghost fluid method for the simulation of multimedium compressible flow. SIAM J Sci Comput, 2006, 28: 278–302
55 Wang Q, Deiterding R, Pan J, et al. Consistent high resolution interface-capturing finite volume method for compressible multi-material flows. Comput Fluids, 2020, 202: 104518
56 Woodward P, Colella P. The numerical simulation of two-dimensional fluid flow with strong shocks. J Comput Phys, 1984, 54: 115–173
57 Xiong T, Qiu J M, Xu Z, et al. High order maximum principle preserving semi-Lagrangian finite difference WENO schemes for the Vlasov equation. J Comput Phys, 2014, 273: 618–639
58 Zhang H, Wu B, Meng X. Analysis of the local discontinuous Galerkin method with generalized fluxes for one-dimensional nonlinear convection-diffusion systems. Sci China Math, 2023, 66: 2641–2664
59 Zhao Z, Qiu J. An oscillation-free Hermite WENO scheme for hyperbolic conservation laws. Sci China Math, 2024, 67: 431–454

60 Zhu J, Qiu J. A class of the fourth order finite volume Hermite weighted essentially non-oscillatory schemes. Sci China Ser A, 2008, 51: 1549–1560
61 Zhu J, Qiu J. A new fifth order finite difference WENO scheme for solving hyperbolic conservation laws. J Comput Phys, 2016, 318: 110–121

Appendix A The framework for the two-dimensional case

For simplicity, the computational domain is equally divided, i.e.,

$$a = x_{1/2} < x_{3/2} < \cdots < x_{N_x+1/2} = b, \quad c = y_{1/2} < y_{3/2} < \cdots < y_{N_y+1/2} = d.$$

The cell is denoted by

$$I_{i,j} = [x_{i-1/2}, x_{i+1/2}] \times [y_{j-1/2}, y_{j+1/2}],$$

the cell center is denoted by (x_i, y_j), and the mesh size is denoted by

$$\Delta x = x_{i+1/2} - x_{i-1/2}, \quad \Delta y = y_{j+1/2} - y_{j-1/2}.$$

Furthermore, we take

$$\begin{aligned}
U &= \left(\rho, \rho u, \rho v, E, \frac{1}{\gamma-1}, \frac{\gamma p_\infty}{\gamma-1}\right)^{\mathrm{T}},\\
F(U) &= \left(\rho u, \rho u^2 + p, \rho v u, u(E+p), \frac{u}{\gamma-1}, \frac{u\gamma p_\infty}{\gamma-1}\right)^{\mathrm{T}},\\
G(U) &= \left(\rho v, \rho u v, \rho v^2 + p, v(E+p), \frac{v}{\gamma-1}, \frac{v\gamma p_\infty}{\gamma-1}\right)^{\mathrm{T}},
\end{aligned}$$

and

$$S(U) = \left(0, 0, 0, 0, \frac{1}{\gamma-1}, \frac{\gamma p_\infty}{\gamma-1}\right)^{\mathrm{T}},$$

and then we can rewrite the system (2.1) into

$$U_t + F(U)_x + G(U)_y = S(U)(u_x + v_y), \tag{A.1}$$

where u_x and v_y refer to the derivative of velocities u and v in the x and y directions. We integrate (A.1) over the cell $I_{i,j}$, and then we obtain the following semidiscretization form:

$$\begin{aligned}
&\frac{\partial U(t)_{i,j}}{\partial t} + \frac{1}{\Delta x \Delta y}\left(\int_{y_{j-1/2}}^{y_{j+1/2}} F(U(x_{i+\frac{1}{2}}, y, t))dy - \int_{y_{j-1/2}}^{y_{j+1/2}} F(U(x_{i-\frac{1}{2}}, y, t))dy\right)\\
&\quad + \frac{1}{\Delta x \Delta y}\left(\int_{x_{i-1/2}}^{x_{i+1/2}} G(U(x, y_{j+\frac{1}{2}}, t))dx - \int_{x_{i-1/2}}^{x_{i+1/2}} G(U(x, y_{j-\frac{1}{2}}, t))dx\right)\\
&\qquad = \frac{1}{\Delta x \Delta y}\left(\int_{y_{j-1/2}}^{y_{j+1/2}} S(U(x_{i+\frac{1}{2}}, y, t))u(x_{i+\frac{1}{2}}, y, t)dy - \int_{y_{j-1/2}}^{y_{j+1/2}} S(U(x_{i-\frac{1}{2}}, y, t))u(x_{i-\frac{1}{2}}, y, t)dy\right)\\
&\qquad\quad + \frac{1}{\Delta x \Delta y}\left(\int_{x_{i-1/2}}^{x_{i+1/2}} S(U(x, y_{j+\frac{1}{2}}, t))v(x, y_{j+\frac{1}{2}}, t)dx - \int_{x_{i-1/2}}^{x_{i+1/2}} S(U(x, y_{j-\frac{1}{2}}, t))v(x, y_{j-\frac{1}{2}}, t)dx\right)\\
&\qquad\quad - \frac{1}{\Delta x \Delta y}\int_{x_{i-1/2}}^{x_{i+1/2}}\int_{y_{j-1/2}}^{y_{j+1/2}} (S(U(x,y,t))_x u(x,y,t) + S(U(x,y,t))_y v(x,y,t))dxdy,
\end{aligned} \tag{A.2}$$

where

$$U(t)_{i,j} = \frac{1}{\Delta x \Delta y}\int_{x_{i-1/2}}^{x_{i+1/2}}\int_{y_{j-1/2}}^{y_{j+1/2}} U(x,y,t)dxdy.$$

Then, we approximate (A.2) by the following formulation:

$$\frac{\partial U_{i,j}}{\partial t} + \frac{1}{\Delta x}\sum_{k=1}^{4}(\hat{F}_{i+\frac{1}{2},G_k} - \hat{F}_{i-\frac{1}{2},G_k}) + \frac{1}{\Delta y}\sum_{k=1}^{4}(\hat{G}_{G_k,j+\frac{1}{2}} - \hat{G}_{G_k,j-\frac{1}{2}})$$

$$
\begin{aligned}
&= \frac{1}{\Delta x}\sum_{k=1}^{4}(S(U^{-}_{i+\frac{1}{2},G_k})\hat{u}_{i+\frac{1}{2},G_k} - S(U^{+}_{i-\frac{1}{2},G_k})\hat{u}_{i-\frac{1}{2},G_k}) \\
&\quad + \frac{1}{\Delta y}\sum_{k=1}^{4}(S(U^{-}_{G_k,j+\frac{1}{2}})\hat{v}_{G_k,j+\frac{1}{2}} - S(U^{+}_{G_k,j-\frac{1}{2}})\hat{v}_{G_k,j-\frac{1}{2}}) \\
&\quad - \Bigg(\sum_{k=1}^{4}\sum_{l=1}^{4} w_k w_l (S(U)_{x_{G_k,G_l}} u_{G_k,G_l} + S(U)_{y_{G_k,G_l}} v_{G_k,G_l})\Bigg).
\end{aligned} \tag{A.3}
$$

Here, $U_{i,j}$ refers to the numerical approximation to the cell average $U(t)_{i,j}$, and $\hat{F}_{i+\frac{1}{2},G_k}$, $\hat{G}_{G_k,j+\frac{1}{2}}$, $\hat{u}_{i+\frac{1}{2},G_k}$, and $\hat{v}_{G_k,j+\frac{1}{2}}$ represent the HLLC numerical flux evaluated at the Gauss-Lobatto quadrature points. The left and right states of the flux are obtained by the WENO method in the dimension by dimension way. Then, we also rewrite the semi-discrete system (A.3) as $\mathrm{U}_t = \mathcal{L}(\mathrm{U})$, and evolved by the third-order TVD Runge-Kutta time discretization [43].