Positivity-preserving time discretizations for production-destruction equations with applications to non-equilibrium flows

Juntao Huang[1] and Chi-Wang Shu[2]

Abstract

In this paper, we construct a family of modified Patankar Runge-Kutta (MPRK) methods, which is conservative and unconditionally positivity-preserving, for production-destruction equations, and derive necessary and sufficient conditions to obtain second-order accuracy. This ordinary differential equation solver is then extended to solve a class of semi-discrete schemes for PDEs. Combining this time integration method with the positivity-preserving finite difference weighted essentially non-oscillatory (WENO) schemes, we successfully obtain a positivity-preserving WENO scheme for non-equilibrium flows. Various numerical tests are reported to demonstrate the effectiveness of the methods.

Keywords: Compressible Euler equations; positivity-preserving; chemical reactions; production-destruction equations; finite difference

[1]Zhou Pei-Yuan Center for Applied Mathematics, Tsinghua University, Beijing 100084, China. E-mail: huangjt13@mails.tsinghua.edu.cn

[2]Division of Applied Mathematics, Brown University, Providence, RI 02912, USA. E-mail: shu@dam.brown.edu. Research supported by ARO grant W911NF-15-1-0226 and NSF grant DMS-1719410.

1 Introduction

We consider the model for non-equilibrium flows without conduction or radiation [20], a system of hyperbolic conservation laws with source terms

$$U_t + F(U)_x = S(U). \tag{1.1}$$

Here U, $F(U)$ and $S(U)$ are column vectors with $m = n_s + 2$ components where n_s is the number of species:

$$\begin{aligned} U &= (\rho_1, \cdots, \rho_{n_s}, \rho u, \rho e_0)^T, \\ F(U) &= (\rho_1 u, \cdots, \rho_{n_s} u, \rho u^2 + p, \rho u e_0 + up)^T, \\ S(U) &= (s_1, \cdots, s_{n_s}, 0, 0)^T, \end{aligned}$$

where ρ_i is the density of the i-th species, u is the velocity and e_0 is the total energy per unit mass of mixture. The total density is defined as $\rho = \sum_{i=1}^{n_s} \rho_i$ and the pressure p is given by

$$p = RT \sum_{i=1}^{n_s} \frac{\rho_i}{M_i},$$

where R is the universal gas constant and M_i is the molar mass of the i-th species. The total energy has the expression:

$$\rho e_0 = \sum_{i=1}^{n_s} \rho_i e_{in,i}(T) + \sum_{i=1}^{n_s} \rho_i h_i^0 + \frac{1}{2} \rho u^2,$$

where $e_{in,i}(T) = C_i T$ is the internal energy of the i-th species with $C_i = 3R/2M_i$ and $5R/2M_i$ for monoatomic species and diatomic species, respectively, and the enthalpies h_i^0 are constants.

The source term $S(U)$ describes the chemical reactions occurring in gas flows which result in changes in the amount of mass of each chemical species. We assume that there are J reactions of the form

$$\nu'_{1,j} X_1 + \nu'_{2,j} X_2 + \cdots + \nu'_{n_s,j} X_{n_s} \rightleftharpoons \nu''_{1,j} X_1 + \nu''_{2,j} X_2 + \cdots + \nu''_{n_s,j} X_{n_s}, \quad j = 1, \cdots, J,$$

where $\nu'_{i,j}$ and $\nu''_{i,j}$ are respectively the stoichiometric coefficients of the reactants and products of the i-th species in the j-th reaction. For non-equilibrium chemistry, the rate of production of species i due to chemical reaction, may be written as

$$s_i = M_i \sum_{j=1}^{J} (\nu''_{i,j} - \nu'_{i,j}) \left(k_{f,j} \prod_{s=1}^{n_s} (\frac{\rho_s}{M_s})^{\nu'_{s,j}} - k_{b,j} \prod_{s=1}^{n_s} (\frac{\rho_s}{M_s})^{\nu''_{s,j}} \right), \quad i = 1, \cdots, n_s.$$

For each reaction j, the forward and backward reaction rates, $k_{f,j}$ and $k_{b,j}$ are known functions of the temperature. Notice that the reactive Euler equations, which are often used to model detonation waves [18], can also be written in this form (1.1). We will turn back to this issue in section 3.3.

The main objective of this work is to develop positivity-preserving schemes for this class of hyperbolic equations. A general framework for constructing high-order positivity-preserving discontinuous Galerkin (DG), finite volume and finite difference schemes was proposed by Zhang and Shu in a series of works [24, 25, 27]. It has been successfully applied to shallow water equations [22, 21], convection-diffusion equations [28], Navier-Stokes equations [23] and so on. This framework was also generalized to compressible Euler equations of gas dynamics with several kinds of source terms in [26]. We remark that almost all these works rely on the idea of convex combinations: one first proves the positivity-preserving property for the first-order explicit Euler forward scheme and then obtain a scheme high-order in time by using strong-stability-preserving (SSP) Runge-Kutta (RK) time discretization [5]. However, for equations with stiff source terms, e.g., the non-equilibrium flows (1.1) considered in this work, the explicit time integration may suffer from very severe restriction on the time step and thus cause large computational cost.

Recently, there are a few works on positivity-preserving schemes for PDEs with stiff source terms. Chertock et al. proposed a class of semi-implicit RK schemes with sign-preserving and well-balanced properties for a particular class of ODEs with stiff terms [2], and applied it to shallow water equations with stiff friction terms [3]. In [8], we constructed a class of second-order positivity-preserving implicit-explicit (IMEX) RK methods for the system of ODEs arising from the semi-discretization of the Kerr-Debye model (a special relaxation system). In [9], we developed a class of exponential SSP high order time integrations with bound-preserving property, and applied it to scalar hyperbolic equations with stiff source terms. Most recently, Hu et al. proposed IMEX schemes [7] for the stiff BGK kinetic equations, which share both the asymptotic-preserving and positivity-preserving properties.

For chemical reactive flows, the chemical source terms are mostly stiff and one usually applies the time-splitting techniques (e.g. Strang's splitting [17]). However, this approach has several limitations. First, the splitting techniques are generally only up to second-order accuracy. It is far more difficult to design splitting schemes with high-order accuracy. Second, after splitting, the ODEs with only the chemical reactions are not easy to solve, due to the stiffness, especially taking the positivity of the numerical solutions into consideration.

In this work, we first ignore the convection terms in (1.1) and only consider a system of ODEs which describes the chemical reactions,

$$\frac{d\rho_i}{dt} = M_i \sum_{j=1}^{J} (\nu''_{i,j} - \nu'_{i,j}) \left(k_{f,j} \prod_{s=1}^{n_s} (\frac{\rho_s}{M_s})^{\nu'_{s,j}} - k_{b,j} \prod_{s=1}^{n_s} (\frac{\rho_s}{M_s})^{\nu''_{s,j}} \right), \quad i = 1, \cdots, n_s. \tag{1.2}$$

Instead of solving (1.2) directly, we move to a larger class of ODEs and concentrate on production-destruction equations which have the form:

$$\frac{dc_i}{dt} = P_i(c) - D_i(c), \quad i = 1, 2, \cdots, N, \tag{1.3}$$

with

$$P_i(c) = \sum_{j=1}^{N} p_{ij}(c), \quad D_i(c) = \sum_{j=1}^{N} d_{ij}(c),$$

and

$$p_{ij}(c) = d_{ji}(c) \geq 0.$$

Here $c = (c_1, c_2, \cdots, c_N)^T$ and c_i denotes the concentration of the i-th component. The production function $p_{ij}(c)$ denotes the rate at which the j-th component transforms into the i-th component, while the destruction function $d_{ij}(c)$ denotes the rate at which the i-th component transforms into the j-th component. The exact solutions to (1.3) share the conservation property, i.e., $\sum_{i=1}^{N} c_i(t)$ remains unchanged with respect to time t. Also, the positivity of the solution is guaranteed as long as the initial condition is positive and $d_{ij}(c) = 0$ for $c_i = 0$ [1]. We remark that the chemical reaction system (1.2) is a special class of production-destruction equations (1.3), see e.g. [4].

There have been many works on constructing numerical schemes for (1.3) which *unconditionally* preserve the conservation and positivity of the solutions. The famous Patankar trick [14] is to modify the explicit Euler forward scheme for (1.3) into

$$c_i^{n+1} = c_i^n + \Delta t(P_i(c^n) - D_i(c^n)\frac{c_i^{n+1}}{c_i^n}), \tag{1.4}$$

to reach the unconditional positivity of the numerical solution. However, this trick loses the conservation property and may fail for some stiff ODEs. For preserving the conservation, in [1], the authors modified the classical Patankar scheme (1.4) and obtained the modified Patankar scheme

$$c_i^{n+1} = c_i^n + \Delta t \left(\sum_j p_{ij}(c^n)\frac{c_j^{n+1}}{c_j^n} - \sum_j d_{ij}(c^n)\frac{c_i^{n+1}}{c_i^n} \right). \tag{1.5}$$

The positivity-preserving property of this scheme can be easily shown by writing (1.5) as a linear system for $c^{n+1} = (c_1^{n+1}, c_2^{n+1}, \cdots, c_N^{n+1})$ of which the coefficient matrix is an M-matrix. It was also generalized to a second-order modified Patankar method [1]

$$c_i^{(1)} = c_i^n + \Delta t \left(\sum_j p_{ij}(c^n)\frac{c_j^{(1)}}{c_j^n} - \sum_j d_{ij}(c^n)\frac{c_i^{(1)}}{c_i^n} \right), \tag{1.6a}$$

$$c_i^{n+1} = c_i^n + \frac{\Delta t}{2} \left(\sum_j (p_{ij}(c^n) + p_{ij}(c^{(1)}))\frac{c_j^{n+1}}{c_j^{(1)}} - \sum_j (d_{ij}(c^n) + d_{ij}(c^{(1)}))\frac{c_i^{n+1}}{c_i^{(1)}} \right). \tag{1.6b}$$

This time integration method was then applied to shallow water equations with wetting and drying in [13]. Recently, Kopecz and Meister formulated a class of modified Patankar Runge-Kutta scheme in a more general form, and obtained second-order [10] and third-order schemes [11]. The second-order modified Patankar Runge-Kutta (MPRK) scheme reads as [10]

$$c_i^{(0)} = c_i^n, \tag{1.7a}$$

$$c_i^{(1)} = c_i^n + \Delta t a_{21} \left(\sum_j p_{ij}(c^{(0)}) \frac{c_j^{(1)}}{\pi_j} - \sum_j d_{ij}(c^{(0)}) \frac{c_i^{(1)}}{\pi_i} \right), \tag{1.7b}$$

$$c_i^{n+1} = c_i^n + \Delta t \left(\sum_j (b_1 p_{ij}(c^{(0)}) + b_2 p_{ij}(c^{(1)})) \frac{c_j^{n+1}}{\sigma_j} - \sum_j (b_1 d_{ij}(c^{(0)}) + b_2 d_{ij}(c^{(1)})) \frac{c_i^{n+1}}{\sigma_i} \right). \tag{1.7c}$$

with parameters $a_{21} = \alpha$, $b_1 = 1 - 1/(2\alpha)$, $b_2 = 1/(2\alpha)$, and $\pi_i = c_i^n$, $\sigma_i = c_i^n (c_i^{(1)}/c_i^n)^{1/\alpha}$, and $\alpha \geq 1/2$.

This class of schemes (1.7) works well for the production-destruction equations (1.3). However, it would fail, if we would like to solve the chemical reaction flow (1.1) *without* using the time-splitting approach, as the convection terms in (1.1) cannot be coupled into the time discretization in (1.7). This is the starting point of our work. Instead of using the RK schemes in the classical form, we apply the RK schemes of the Shu-Osher form [16] and develop another class of modified Patankar Runge-Kutta scheme for (1.3). In order to cover the chemical reactive flows, the convection term is then included in the ODEs solver and the accuracy is analyzed. Combining these time integration methods with the semi-discrete finite difference weighted essentially non-oscillatory (WENO) schemes [12], we successfully obtain the positivity-preserving WENO schemes for (1.1).

The paper is organized as follows. In section 2, we formulate another class of modified Patankar RK scheme (MPRK) with the Shu-Osher form, and then derive the necessary and sufficient conditions for this ODE solver to be second-order accurate. The solver is also generalized to solve semi-discrete schemes for PDEs and the accuracy is analyzed. In section 3, we combine the time integration method with the positivity-preserving finite difference WENO schemes, and discuss the positivity-preserving property by using the limiter. Numerical results for the ODEs and PDEs are presented in section 4. Some concluding remarks are given in section 5.

2 The ODE solver

In this part, we first formulate a class of modified Patankar RK scheme with the Shu-Osher form. Following the procedures in [10], the necessary and sufficient conditions for this ODE solver to be second-order accurate are derived. Afterwards, the solver is generalized to solve semi-discrete schemes for PDEs and the accuracy is analyzed.

The explicit SSP RK scheme for (1.3) written in the Shu-Osher form [16] reads as:

$$c_i^{(0)} = c_i^n, \tag{2.8a}$$

$$c_i^{(1)} = \alpha_{10} c_i^{(0)} + \Delta t \beta_{10} \left(P_i(c^{(0)}) - D_i(c^{(0)}) \right), \tag{2.8b}$$

$$c_i^{n+1} = \alpha_{20} c_i^{(0)} + \Delta t \beta_{20} \left(P_i(c^{(0)}) - D_i(c^{(0)}) \right) + \alpha_{21} c_i^{(1)} + \Delta t \beta_{21} \left(P_i(c^{(1)}) - D_i(c^{(1)}) \right). \tag{2.8c}$$

For (2.8) to be second-order accurate, the conditions on the coefficients are

$$\alpha_{10} = 1, \quad \alpha_{20} + \alpha_{21} = 1, \quad \beta_{20} + \beta_{21} + \alpha_{21}\beta_{10} = 1, \quad \beta_{10}\beta_{21} = \frac{1}{2}. \tag{2.9}$$

Following [10], for preserving conservation and positivity unconditionally, we modify the production and destruction terms in (2.8) and obtain the modified Patankar RK scheme (MPRK)

$$c_i^{(0)} = c_i^n, \tag{2.10a}$$

$$c_i^{(1)} = \alpha_{10} c_i^{(0)} + \Delta t \beta_{10} \left(\sum_j p_{ij}(c^{(0)}) \frac{c_j^{(1)}}{\pi_j} - \sum_j d_{ij}(c^{(0)}) \frac{c_i^{(1)}}{\pi_i} \right), \tag{2.10b}$$

$$\begin{aligned} c_i^{n+1} = {} & \alpha_{20} c_i^{(0)} + \alpha_{21} c_i^{(1)} \\ & + \Delta t \left(\sum_j (\beta_{20} p_{ij}(c^{(0)}) + \beta_{21} p_{ij}(c^{(1)})) \frac{c_j^{n+1}}{\sigma_j} - \sum_j (\beta_{20} d_{ij}(c^{(0)}) + \beta_{21} d_{ij}(c^{(1)})) \frac{c_i^{n+1}}{\sigma_i} \right). \end{aligned} \tag{2.10c}$$

Here $\pi_i, \sigma_i \geq 0$ are functions of $c_i^{(0)}$ and $c_i^{(1)}$ and will be determined later.

Actually, for avoiding solving nonlinear equations in each stage of (2.10), it is required that π_i and σ_i are independent of $(c^{(1)}, c^{n+1})$ and c^{n+1}, respectively. Furthermore, since (2.10) is only a modification of (2.8), it is natural to assume that the constrains (2.9) on the coefficients also hold in (2.10), and π_i and σ_i are some approximations of c_i^n, i.e.,

$$\pi_i = c_i^n + \mathcal{O}(\Delta t), \quad \sigma_i = c_i^n + \mathcal{O}(\Delta t).$$

For simplicity, we choose $\pi_i = c_i^n$.

Remark 2.1. *Although the RK methods in the classical form and the Shu-Osher form are equivalent, they are different after modification (see* (1.7) *and* (2.10)*). The scheme* (2.10) *reduces to* (1.7) *by taking* $\alpha_{21} = 0$*.*

2.1 Necessary condition

In this part, we will derive the necessary condition for (2.10) to be of second-order. We consider a specific class of ODEs: In (1.3), we take, for fixed $I, J \in \{1, 2, \cdots, N\}$ and $I \neq J$,

$$p_{ij}(c) = \begin{cases} \mu c_I, & i = J, \quad j = I, \\ 0, & \text{otherwise}, \end{cases}$$

and

$$d_{ij}(c) = \begin{cases} \mu c_I, & i = I, \quad j = J, \\ 0, & \text{otherwise}, \end{cases}$$

with constant $\mu > 0$. In the following, we will assume that the MPRK scheme (2.10) is of second-order, and apply it to this specific class of ODEs.

By Taylor expansion, the exact solution has the form

$$c_i(t^{n+1}) = c_i(t^n) + \frac{dc_i}{dt}(t^n)\Delta t + \frac{1}{2}\frac{d^2 c_i}{dt^2}(t^n)\Delta t^2 + \mathcal{O}(\Delta t^3).$$

Taking derivative on (1.3) yields

$$\frac{d^2 c_i}{dt^2} = \frac{d}{dt}(P_i(c) - D_i(c)) = \frac{\partial(P_i(c) - D_i(c))}{\partial c}\frac{dc}{dt} = \frac{\partial(P_i(c) - D_i(c))}{\partial c}(P(c) - D(c)),$$

Here $P(c) := (P_1(c), P_2(c), \cdots, P_N(c))^T$ and $D(c) := (D_1(c), D_2(c), \cdots, D_N(c))^T$. Hence, we have

$$c_i(t^{n+1}) = c_i(t^n) + \Delta t(P_i^n - D_i^n) + \frac{\Delta t^2}{2}\frac{\partial(P_i^n - D_i^n)}{\partial c}(P^n - D^n) + \mathcal{O}(\Delta t^3).$$

Since the solver is second-order accurate, we obtain

$$\alpha_{20}c_i^n + \alpha_{21}c_i^{(1)} + \Delta t\left(\sum_j(\beta_{20}p_{ij}^n + \beta_{21}p_{ij}^{(1)})\frac{c_j^{n+1}}{\sigma_j} - \sum_j(\beta_{20}d_{ij}^n + \beta_{21}d_{ij}^{(1)})\frac{c_i^{n+1}}{\sigma_i}\right)$$
$$= c_i^n + \Delta t(P_i^n - D_i^n) + \frac{\Delta t^2}{2}\frac{\partial(P_i^n - D_i^n)}{\partial c}(P^n - D^n) + \mathcal{O}(\Delta t^3),$$

and subsequently

$$\alpha_{21}(c_i^{(1)} - c_i^n) + \Delta t\left(\sum_j(\beta_{20}p_{ij}^n + \beta_{21}p_{ij}^{(1)})\frac{c_j^{n+1}}{\sigma_j} - \sum_j(\beta_{20}d_{ij}^n + \beta_{21}d_{ij}^{(1)})\frac{c_i^{n+1}}{\sigma_i}\right)$$
$$- \Delta t(P_i^n - D_i^n) - \frac{\Delta t^2}{2}\frac{\partial(P_i^n - D_i^n)}{\partial c}(P^n - D^n) = \mathcal{O}(\Delta t^3),$$

by the order condition $\alpha_{20} + \alpha_{21} = 1$. Set $i = I$, and write I as i,

$$\alpha_{21}(c_i^{(1)} - c_i^n) - \Delta t(\beta_{20}D_i^n + \beta_{21}D_i^{(1)})\frac{c_i^{n+1}}{\sigma_i} + \Delta t D_i^n - \frac{\Delta t^2}{2}\frac{\partial D_i^n}{\partial c}D^n = \mathcal{O}(\Delta t^3). \qquad (2.11)$$

Note that (2.11) holds for any $i = 1, 2\cdots, N$, since I is chosen arbitrary.

Using (2.10b), we have

$$c_i^{(1)} = c_i^n - \Delta t\beta_{10}D_i^n\frac{c_i^{(1)}}{\pi_i} = c_i^n - \Delta t\beta_{10}\mu c_i^n\frac{c_i^{(1)}}{\pi_i}.$$

Substitute it into the left hand side of (2.11):

$$- \alpha_{21}\beta_{10}\Delta t\mu c_i^n\frac{c_i^{(1)}}{\pi_i} - \Delta t(\beta_{20}\mu c_i^n + \beta_{21}\mu c_i^{(1)})\frac{c_i^{n+1}}{\sigma_i} + \Delta t\mu c_i^n - \frac{\Delta t^2}{2}\mu^2 c_i^n,$$
$$= - \alpha_{21}\beta_{10}\Delta t\mu c_i^n\frac{c_i^{(1)}}{\pi_i} - \Delta t\left(\beta_{20}\mu c_i^n + \beta_{21}\mu(c_i^n - \Delta t\beta_{10}\mu c_i^n\frac{c_i^{(1)}}{\pi_i})\right)\frac{c_i^{n+1}}{\sigma_i} + \Delta t\mu c_i^n - \frac{\Delta t^2}{2}\mu^2 c_i^n,$$
$$= - \mu\Delta t c_i^n\left(\alpha_{21}\beta_{10}\frac{c_i^{(1)}}{\pi_i} + (\beta_{20} + \beta_{21})\frac{c_i^{n+1}}{\sigma_i} - 1\right) + \frac{1}{2}\mu^2\Delta t^2 c_i^n\left(\frac{c_i^{(1)}}{\pi_i}\frac{c_i^{n+1}}{\sigma_i} - 1\right),$$

where we have used the order condition $\beta_{21}\beta_{10} = \frac{1}{2}$. Now we have

$$-\mu\Delta t c_i^n\left(\alpha_{21}\beta_{10}\frac{c_i^{(1)}}{\pi_i} + (\beta_{20} + \beta_{21})\frac{c_i^{n+1}}{\sigma_i} - 1\right) + \frac{1}{2}\mu^2\Delta t^2 c_i^n\left(\frac{c_i^{(1)}}{\pi_i}\frac{c_i^{n+1}}{\sigma_i} - 1\right) = \mathcal{O}(\Delta t^3).$$

Since the constant $\mu > 0$ is arbitrary, we reach the following two constraints with the aid of Lemma 3.2 in [10]:

$$\alpha_{21}\beta_{10}\frac{c_i^{(1)}}{\pi_i} + (\beta_{20} + \beta_{21})\frac{c_i^{n+1}}{\sigma_i} - 1 = \mathcal{O}(\Delta t^2), \qquad (2.12)$$

and

$$\frac{c_i^{(1)}}{\pi_i}\frac{c_i^{n+1}}{\sigma_i} - 1 = \mathcal{O}(\Delta t). \tag{2.13}$$

The second constrain (2.13) is automatically fulfilled since $\pi_i = c_i^n$ and $\sigma_i = c_i^n + \mathcal{O}(\Delta t)$.

Substitute $\pi_i = c_i^n$ into the first stage (2.10b):

$$\begin{aligned} c_i^{(1)} &= \alpha_{10} c_i^n + \Delta t \beta_{10} \left(\sum_j p_{ij}^n \frac{c_j^{(1)}}{c_j^n} - \sum_j d_{ij}^n \frac{c_i^{(1)}}{c_i^n} \right), \\ &= c_i^n + \Delta t \beta_{10} \left(\sum_j p_{ij}^n \frac{c_j^n + \mathcal{O}(\Delta t)}{c_j^n} - \sum_j d_{ij}^n \frac{c_i^n + \mathcal{O}(\Delta t)}{c_i^n} \right), \\ &= c_i^n + \Delta t \beta_{10} (P_i^n - D_i^n) + \mathcal{O}(\Delta t^2). \end{aligned}$$

and thus

$$\frac{c_i^{(1)}}{\pi_i} = \frac{c_i^{(1)}}{c_i^n} = 1 + \Delta t \beta_{10} \frac{P_i^n - D_i^n}{c_i^n} + \mathcal{O}(\Delta t^2).$$

The first constrain (2.12) is simplified:

$$\begin{aligned} \alpha_{21}\beta_{10} \left(1 + \Delta t \beta_{10} \frac{P_i^n - D_i^n}{c_i^n} \right) + (\beta_{20} + \beta_{21}) \frac{c_i^{n+1}}{\sigma_i} - 1 = \mathcal{O}(\Delta t^2), \\ \alpha_{21}\beta_{10}^2 \Delta t \frac{P_i^n - D_i^n}{c_i^n} + (\beta_{20} + \beta_{21}) \frac{c_i^{n+1}}{\sigma_i} = \beta_{20} + \beta_{21} + \mathcal{O}(\Delta t^2), \end{aligned}$$

and we solve out:

$$\frac{c_i^{n+1}}{\sigma_i} = 1 - \frac{\alpha_{21}\beta_{10}^2}{\beta_{20} + \beta_{21}} \Delta t \frac{P_i^n - D_i^n}{c_i^n} + \mathcal{O}(\Delta t^2).$$

Thus, the expression of σ_i is derived:

$$\begin{aligned} \sigma_i =& \frac{c_i^{n+1}}{\frac{c_i^{n+1}}{\sigma_i}}, \\ =& \frac{c_i^{n+1}}{1 - \frac{\alpha_{21}\beta_{10}^2}{\beta_{20}+\beta_{21}} \Delta t \frac{P_i^n - D_i^n}{c_i^n} + \mathcal{O}(\Delta t^2)}, \\ =& (c_i^n + \Delta t (P_i^n - D_i^n))(1 + \frac{\alpha_{21}\beta_{10}^2}{\beta_{20} + \beta_{21}} \Delta t \frac{P_i^n - D_i^n}{c_i^n}) + \mathcal{O}(\Delta t^2), \\ =& c_i^n + (1 + \frac{\alpha_{21}\beta_{10}^2}{\beta_{20} + \beta_{21}}) \Delta t (P_i^n - D_i^n) + \mathcal{O}(\Delta t^2). \end{aligned}$$

We finally reach the relation

$$\sigma_i = c_i^n + (1 + \frac{\alpha_{21}\beta_{10}^2}{\beta_{20} + \beta_{21}}) \Delta t (P_i^n - D_i^n) + \mathcal{O}(\Delta t^2). \tag{2.14}$$

2.2 Sufficient condition

In this part, we prove that the necessary condition (2.14) is also sufficient.

First, we do expansion for $c_i^{(1)}$ in the first stage (2.10b)

$$\begin{aligned} c_i^{(1)} =& c_i^n + \Delta t \beta_{10} (\sum_j p_{ij}^n \frac{c_j^{(1)}}{c_j^n} - \sum_j d_{ij}^n \frac{c_i^{(1)}}{c_i^n}), \\ =& c_i^n + \Delta t \beta_{10} (P_i^n - D_i^n) + \mathcal{O}(\Delta t^2). \end{aligned}$$

Iterate once,

$$\begin{aligned} c_i^{(1)} =& c_i^n + \Delta t \beta_{10} (\sum_j p_{ij}^n \frac{c_j^n + \Delta t \beta_{10} (P_j^n - D_j^n)}{c_j^n} - \sum_j d_{ij}^n \frac{c_i^n + \Delta t \beta_{10} (P_i^n - D_i^n)}{c_i^n}) + \mathcal{O}(\Delta t^3), \\ =& c_i^n + \Delta t \beta_{10} (P_i^n - D_i^n) + \Delta t^2 \beta_{10}^2 (\sum_j p_{ij}^n \frac{P_j^n - D_j^n}{c_j^n} - \sum_j d_{ij}^n \frac{P_i^n - D_i^n}{c_i^n}) + \mathcal{O}(\Delta t^3). \end{aligned}$$

By Taylor expansion for $\phi^{(1)} := \phi(c^{(1)})$ with $\phi = p_{ij}$ or d_{ij}, we have

$$\phi^{(1)} = \phi(c^{(1)}) = \phi(c^n) + \frac{\partial \phi^n}{\partial c}(c^{(1)} - c^n) + \mathcal{O}(\Delta t^2), \tag{2.15}$$

and thus,

$$\begin{aligned} \beta_{20}\phi^n + \beta_{21}\phi^{(1)} =& (\beta_{20} + \beta_{21})\phi^n + \beta_{21} \frac{\partial \phi^n}{\partial c}(c^{(1)} - c^n) + \mathcal{O}(\Delta t^2) \\ =& (\beta_{20} + \beta_{21})\phi^n + \Delta t \beta_{21} \beta_{10} \frac{\partial \phi^n}{\partial c}(P^n - D^n) + \mathcal{O}(\Delta t^2). \end{aligned}$$

Subsequently, we have expansion for c_i^{n+1}:

$$\begin{aligned} c_i^{n+1} =& \alpha_{20} c_i^n + \alpha_{21} (c_i^n + \Delta t \beta_{10} (P_i^n - D_i^n)) + \Delta t (\beta_{20} + \beta_{21})(P_i^n - D_i^n) + \mathcal{O}(\Delta t^2), \\ =& (\alpha_{20} + \alpha_{21}) c_i^n + (\alpha_{21}\beta_{10} + \beta_{20} + \beta_{21}) \Delta t (P_i^n - D_i^n) + \mathcal{O}(\Delta t^2), \\ =& c_i^n + \Delta t (P_i^n - D_i^n) + \mathcal{O}(\Delta t^2). \end{aligned}$$

Iterate once,

$$\begin{aligned} \frac{c_i^{n+1}}{\sigma_i} =& \frac{c_i^n + \Delta t (P_i^n - D_i^n) + \mathcal{O}(\Delta t^2)}{c_i^n + (1 + \frac{\alpha_{21}\beta_{10}^2}{\beta_{20} + \beta_{21}}) \Delta t (P_i^n - D_i^n) + \mathcal{O}(\Delta t^2)}, \\ =& 1 - \Delta t \frac{\alpha_{21}\beta_{10}^2}{\beta_{20} + \beta_{21}} \frac{P_i^n - D_i^n}{c_i^n} + \mathcal{O}(\Delta t^2). \end{aligned}$$

and thus,

$$
\begin{aligned}
&\sum_j(\beta_{20}p_{ij}(c^{(0)})+\beta_{21}p_{ij}(c^{(1)}))\frac{c_j^{n+1}}{\sigma_j}-\sum_j(\beta_{20}d_{ij}(c^{(0)})+\beta_{21}d_{ij}(c^{(1)}))\frac{c_i^{n+1}}{\sigma_i}\\
=&\sum_j\left((\beta_{20}+\beta_{21})p_{ij}^n+\Delta t\beta_{21}\beta_{10}\frac{\partial p_{ij}^n}{\partial c}(P^n-D^n)\right)\left(1-\Delta t\frac{\alpha_{21}\beta_{10}^2}{\beta_{20}+\beta_{21}}\frac{P_j^n-D_j^n}{c_j^n}\right)\\
&-\sum_j\left((\beta_{20}+\beta_{21})d_{ij}^n+\Delta t\beta_{21}\beta_{10}\frac{\partial d_{ij}^n}{\partial c}(P^n-D^n)\right)\left(1-\Delta t\frac{\alpha_{21}\beta_{10}^2}{\beta_{20}+\beta_{21}}\frac{P_i^n-D_i^n}{c_i^n}\right)+\mathcal{O}(\Delta t^2),\\
=&(\beta_{20}+\beta_{21})(P_i^n-D_i^n)+\Delta t\beta_{21}\beta_{10}\frac{\partial(P_i^n-D_i^n)}{\partial c}(P^n-D^n)\\
&-\Delta t\alpha_{21}\beta_{10}^2\left(\sum_j p_{ij}^n\frac{P_j^n-D_j^n}{c_j^n}-\sum_j d_{ij}^n\frac{P_i^n-D_i^n}{c_i^n}\right)+\mathcal{O}(\Delta t^2).
\end{aligned}
$$

Substituting the above relation into the final stage (2.10c) yields:

$$
\begin{aligned}
c_i^{n+1}=&\alpha_{20}c_i^{(0)}+\alpha_{21}\left(c_i^n+\Delta t\beta_{10}(P_i^n-D_i^n)+\Delta t^2\beta_{10}^2(\sum_j p_{ij}^n\frac{P_j^n-D_j^n}{c_j^n}-\sum_j d_{ij}^n\frac{P_i^n-D_i^n}{c_i^n})\right)\\
&+\Delta t((\beta_{20}+\beta_{21})(P_i^n-D_i^n)+\Delta t\beta_{21}\beta_{10}\frac{\partial(P_i^n-D_i^n)}{\partial c}(P^n-D^n)\\
&-\Delta t\alpha_{21}\beta_{10}^2(\sum_j p_{ij}^n\frac{P_j^n-D_j^n}{c_j^n}-\sum_j d_{ij}^n\frac{P_i^n-D_i^n}{c_i^n}))+\mathcal{O}(\Delta t^3),\\
&=c_i^n+\Delta t(P_i^n-D_i^n)+\frac{\Delta t^2}{2}\frac{\partial(P_i^n-D_i^n)}{\partial c}(P^n-D^n)+\mathcal{O}(\Delta t^3).
\end{aligned}
$$

Now the second-order accuracy has been proved.

2.3 MPRK schemes

We have shown that the relation (2.14) is a necessary and sufficient condition for (2.10) to be second-order accurate. In the following, we will derive an explicit expression of σ_i for (2.14) to be satisfied. We try to take

$$
\sigma_i=(c_i^{(1)})^s(c_i^n)^{1-s} \tag{2.16}
$$

with s a constant to be undetermined. Thanks to the expansion of $c_i^{(1)}$, we have

$$
\begin{aligned}
\sigma_i&=(c_i^n+\Delta t\beta_{10}(P_i^n-D_i^n)+\mathcal{O}(\Delta t^2))^s(c_i^n)^{1-s},\\
&=c_i^n(1+\Delta t\beta_{10}\frac{P_i^n-D_i^n}{c_i^n}+\mathcal{O}(\Delta t^2))^s,
\end{aligned}
$$

$$= c_i^n(1 + \Delta t\beta_{10}s\frac{P_i^n - D_i^n}{c_i^n} + \mathcal{O}(\Delta t^2)),$$
$$= c_i^n + \Delta t\beta_{10}s(P_i^n - D_i^n) + \mathcal{O}(\Delta t^2).$$

To satisfy the constrain (2.14), it is only required that

$$s = \frac{1 + \frac{\alpha_{21}\beta_{10}^2}{\beta_{20}+\beta_{21}}}{\beta_{10}} = \frac{\beta_{20} + \beta_{21} + \alpha_{21}\beta_{10}^2}{\beta_{10}(\beta_{20} + \beta_{21})}. \tag{2.17}$$

Now we have a family of schemes

$$c_i^{(0)} = c_i^n, \tag{2.18a}$$

$$c_i^{(1)} = \alpha_{10}c_i^{(0)} + \Delta t\beta_{10}\left(\sum_j p_{ij}(c^{(0)})\frac{c_j^{(1)}}{c_j^n} - \sum_j d_{ij}(c^{(0)})\frac{c_i^{(1)}}{c_i^n}\right), \tag{2.18b}$$

$$\begin{aligned} c_i^{n+1} &= \alpha_{20}c_i^{(0)} + \alpha_{21}c_i^{(1)} \\ &+ \Delta t\left(\sum_j(\beta_{20}p_{ij}(c^{(0)}) + \beta_{21}p_{ij}(c^{(1)}))\frac{c_j^{n+1}}{(c_j^{(1)})^s(c_j^{(0)})^{1-s}} - \sum_j(\beta_{20}d_{ij}(c^{(0)}) + \beta_{21}d_{ij}(c^{(1)}))\frac{c_i^{n+1}}{(c_i^{(1)})^s(c_i^{(0)})^{1-s}}\right). \end{aligned} \tag{2.18c}$$

with the coefficients satisfying the conditions (2.9) and $s = \frac{\beta_{20}+\beta_{21}+\alpha_{21}\beta_{10}^2}{\beta_{10}(\beta_{20}+\beta_{21})}$.

Note that there are four constraints in (2.9) and six parameters $(\alpha_{10}, \alpha_{20}, \alpha_{21}, \beta_{10}, \beta_{20}, \beta_{21})$. Hence, there are two free parameters. Set

$$\alpha_{21} = \alpha, \quad \beta_{10} = \beta,$$

and express other parameters in terms of α and β by (2.9):

$$\alpha_{10} = 1, \quad \alpha_{20} = 1 - \alpha, \quad \alpha_{21} = \alpha,$$
$$\beta_{10} = \beta, \quad \beta_{20} = 1 - \frac{1}{2\beta} - \alpha\beta, \quad \beta_{21} = \frac{1}{2\beta}.$$

and subsequently,

$$s = \frac{1 - \alpha\beta + \alpha\beta^2}{\beta(1 - \alpha\beta)}.$$

For guaranteeing the non-negativity of these parameters $(\alpha_{10}, \alpha_{20}, \alpha_{21}, \beta_{10}, \beta_{20}, \beta_{21})$, it is required that

$$0 \le \alpha \le 1, \quad \beta > 0, \quad \alpha\beta + \frac{1}{2\beta} \le 1.$$

We summarize the above results in the following theorem:

Theorem 2.1 (MPRK schemes)**.** *With the parameters satisfying*

$$\alpha_{10} = 1, \quad \alpha_{20} = 1 - \alpha, \quad \alpha_{21} = \alpha,$$
$$\beta_{10} = \beta, \quad \beta_{20} = 1 - \frac{1}{2\beta} - \alpha\beta, \quad \beta_{21} = \frac{1}{2\beta},$$
$$s = \frac{1 - \alpha\beta + \alpha\beta^2}{\beta(1 - \alpha\beta)},$$

and

$$0 \le \alpha \le 1, \quad \beta > 0, \quad \alpha\beta + \frac{1}{2\beta} \le 1,$$

the MPRK scheme (2.18) *is second-order accurate. Moreover, it is conservative:*

$$\sum_{i=1}^{N} c_i^n = \sum_{i=1}^{N} c_i^{(1)} = \sum_{i=1}^{N} c_i^{n+1},$$

and unconditionally positivity-preserving: if $c_i^n \ge 0$ for $i = 1, 2, \cdots, N$, then $c_i^{n+1} \ge 0$ for $i = 1, 2, \cdots, N$ and all $\Delta t > 0$.

Remark 2.2. *Our scheme* (2.18) *is a generalization of the schemes in [1, 10]. By taking $\alpha = 0$ in* (2.18)*, it reduces to the scheme in [10]. If we further set $\beta = 1$, it reduces to the scheme in [1].*

Remark 2.3. *If we take $\alpha = \frac{1}{2}$ and $\beta = 1$, the coefficients of the optimal SSP RK method are recovered:*

$$\alpha_{10} = \beta_{10} = 1, \quad \alpha_{20} = \alpha_{21} = \beta_{21} = \frac{1}{2}, \quad \beta_{20} = 0,$$

and accordingly $s = 2$.

Remark 2.4. *The form of σ_i in* (2.18) *is not the only possible choice. Following [10], we can also take a convex combination of $(c_i^{(1)})^s (c_i^n)^{1-s}$:*

$$\sigma_i = \lambda (c_i^{(1)})^{s_1} (c_i^n)^{1-s_1} + (1 - \lambda)(c_i^{(1)})^{s_2} (c_i^n)^{1-s_2}.$$

with $0 \le \lambda \le 1$. To satisfy the condition (2.14)*, the relations of parameters λ, s_1, s_2 can be derived in the same approach. For simplicity, we do not investigate this issue in detail and will only focus on* (2.18) *in the following.*

2.4 Extension to semi-discrete schemes

To cover the semi-discrete scheme for the PDEs, we formulate a system of ODEs in the following form:

$$\frac{dc_{k,i}}{dt} = F_{k,i}(c) + P_{k,i}(c) - D_{k,i}(c), \quad k = 1, \cdots, M, \quad i = 1, \cdots, N. \tag{2.19}$$

Here $c_{k,i} = c_{k,i}(t)$ denotes the "concentration" of the i-th species at the k-th grid point, N and M denote the number of species and nodes, respectively. The vector c is the collection of all unknown variables with the form $c := (c_{11}, c_{12}, \cdots, c_{1N}, \cdots, c_{M1}, c_{M2}, \cdots, c_{MN})^T$ and is of length $M \times N$. $F_{k,i} = F_{k,i}(c)$ denotes the contributions of the convection terms after spatial discretizations in the PDEs. The production and destruction terms are $P_{k,i} = P_{k,i}(c) = \sum_{j=1}^{N} p_{k,i,j}(c)$ and $D_{k,i} = D_{k,i}(c) = \sum_{j=1}^{N} d_{k,i,j}(c)$ which satisfy

$$p_{k,i,j}(c) = d_{k,j,i}(c), \quad \forall i, j, k \quad \text{and} \quad c \geq 0.$$

We make the following assumption on (2.19):

Assumption 2.1. The Euler forward method for the convection term satisfies the positivity-preserving property: if $c^n_{k,i} \geq 0$ for all k, i, then

$$c^n_{k,i} + \Delta t F_{k,i}(c^n) \geq 0,$$

for all k, i and $\Delta t \leq \Delta t_0$.

Remark 2.5. *This family of ODEs* (2.19) *covers the semi-discrete finite difference WENO scheme for chemical reacting flows* (1.1) *by setting the production and destruction terms to be zero in the momentum and energy equations. The variable $c_{k,i}$ does not necessarily denote the concentration or density in this case.*

We generalize the MPRK scheme (2.18) by incorporating the convection term $F_{k,i}$ using the Euler forward discretizations in each stage and arrive at

$$c^{(0)}_{k,i} = c^n_{k,i}, \tag{2.20a}$$

$$c_{k,i}^{(1)} = \alpha_{10} c_{k,i}^{(0)} + \Delta t \beta_{10} F_{k,i}(c^{(0)}) + \Delta t \beta_{10} \left(\sum_j p_{k,i,j}(c^{(0)}) \frac{c_{k,j}^{(1)}}{c_{k,j}^{(0)}} - \sum_j d_{k,i,j}(c^{(0)}) \frac{c_{k,i}^{(1)}}{c_{k,i}^{(0)}} \right), \quad (2.20b)$$

$$\begin{aligned} c_{k,i}^{n+1} &= \alpha_{20} c_{k,i}^{(0)} + \alpha_{21} c_{k,i}^{(1)} + \Delta t(\beta_{20} F_{k,i}(c^{(0)}) + \beta_{21} F_{k,i}(c^{(1)})) \\ &+ \Delta t(\sum_j (\beta_{20} p_{k,i,j}(c^{(0)}) + \beta_{21} p_{k,i,j}(c^{(1)})) \frac{c_{k,j}^{n+1}}{(c_{k,j}^{(1)})^s (c_{k,j}^{(0)})^{1-s}} \\ &- \sum_j (\beta_{20} d_{k,i,j}(c^{(0)}) + \beta_{21} d_{k,i,j}(c^{(1)})) \frac{c_{k,i}^{n+1}}{(c_{k,i}^{(1)})^s (c_{k,i}^{(0)})^{1-s}}). \end{aligned} \quad (2.20c)$$

where the parameters are the same with those in Theorem 2.1, Obviously, the scheme (2.20) satisfies the positivity-preserving property if the time step satisfies

$$\Delta t \le \min\{\frac{\alpha_{10}}{\beta_{10}}, \frac{\alpha_{20}}{\beta_{20}}, \frac{\alpha_{21}}{\beta_{21}}\} \Delta t_0.$$

Next, we show that the scheme (2.20) is second-order accurate. First, we do expansion for $c_{k,i}^{(1)}$ in the first stage (2.20b):

$$\begin{aligned} c_{k,i}^{(1)} &= c_{k,i}^n + \Delta t \beta_{10} F_{k,i}^n + \Delta t \beta_{10} (\sum_j p_{k,i,j}^n \frac{c_{k,j}^{(1)}}{c_{k,j}^n} - \sum_j d_{k,i,j}^n \frac{c_{k,i}^{(1)}}{c_{k,i}^n}), \\ &= c_{k,i}^n + \Delta t \beta_{10} (F_{k,i}^n + P_{k,i}^n - D_{k,i}^n) + \mathcal{O}(\Delta t^2). \end{aligned}$$

Iterate once,

$$\begin{aligned} c_{k,i}^{(1)} &= c_{k,i}^n + \Delta t \beta_{10} F_{k,i}^n \\ &+ \Delta t \beta_{10} (\sum_j p_{k,i,j}^n \frac{c_{k,j}^n + \Delta t \beta_{10}(F_{k,j}^n + P_{k,j}^n - D_{k,j}^n)}{c_{k,j}^n} - \sum_j d_{k,i,j}^n \frac{c_{k,i}^n + \Delta t \beta_{10}(F_{k,i}^n + P_{k,i}^n - D_{k,i}^n)}{c_{k,i}^n}) + \mathcal{O}(\Delta t^3), \\ &= c_{k,i}^n + \Delta t \beta_{10} (F_{k,i}^n + P_{k,i}^n - D_{k,i}^n) + \Delta t^2 \beta_{10}^2 (\sum_j p_{k,i,j}^n \frac{F_{k,j}^n + P_{k,j}^n - D_{k,j}^n}{c_{k,j}^n} - \sum_j d_{k,i,j}^n \frac{F_{k,i}^n + P_{k,i}^n - D_{k,i}^n}{c_{k,i}^n}) \\ &+ \mathcal{O}(\Delta t^3). \end{aligned}$$

By doing expansion for c_i^{n+1} up to $\mathcal{O}(\Delta t)$ in the second stage (2.20c), we have

$$\begin{aligned} c_{k,i}^{n+1} &= \alpha_{20} c_{k,i}^n + \alpha_{21}(c_{k,i}^n + \Delta t \beta_{10}(F_{k,i}^n + P_{k,i}^n - D_{k,i}^n)) + \Delta t(\beta_{20} + \beta_{21})(F_{k,i}^n + P_{k,i}^n - D_{k,i}^n) + \mathcal{O}(\Delta t^2), \\ &= c_i^n + \Delta t(F_{k,i}^n + P_{k,i}^n - D_{k,i}^n) + \mathcal{O}(\Delta t^2), \end{aligned}$$

and then

$$\frac{c_{k,i}^{n+1}}{(c_{k,i}^{(1)})^s (c_{k,i}^{(0)})^{1-s}} = \frac{c_{k,i}^n + \Delta t(F_{k,i}^n + P_{k,i}^n - D_{k,i}^n) + \mathcal{O}(\Delta t^2)}{(c_{k,i}^n + \Delta t \beta_{10}(F_{k,i}^n + P_{k,i}^n - D_{k,i}^n) + \mathcal{O}(\Delta t^2))^s (c_{k,i}^n)^{1-s}},$$

$$
\begin{aligned}
&= \frac{c^n_{k,i} + \Delta t(F^n_{k,i} + P^n_{k,i} - D^n_{k,i}) + \mathcal{O}(\Delta t^2)}{c^n_{k,i} + \Delta t s \beta_{10}(F^n_{k,i} + P^n_{k,i} - D^n_{k,i}) + \mathcal{O}(\Delta t^2)},\\
&= \frac{c^n_{k,i} + \Delta t(F^n_{k,i} + P^n_{k,i} - D^n_{k,i}) + \mathcal{O}(\Delta t^2)}{c^n_{k,i} + (1 + \frac{\alpha_{21}\beta_{10}^2}{\beta_{20}+\beta_{21}})\Delta t(F^n_{k,i} + P^n_{k,i} - D^n_{k,i}) + \mathcal{O}(\Delta t^2)},\\
&= 1 - \Delta t \frac{\alpha_{21}\beta_{10}^2}{\beta_{20} + \beta_{21}} \frac{F^n_{k,i} + P^n_{k,i} - D^n_{k,i}}{c^n_{k,i}} + \mathcal{O}(\Delta t^2).
\end{aligned}
$$

Define the vector $\psi := (\psi_{11}, \psi_{12}, \cdots, \psi_{1N}, \cdots, \psi_{M1}, \psi_{M2}, \cdots, \psi_{MN})$ for $\psi = F, P, D$. Expansion for $\phi^{(1)} = \phi(c^{(1)})$ with $\phi = p_{k,i,j}$, $d_{k,i,j}$ or $F_{k,i}$:

$$
\begin{aligned}
\beta_{20}\phi^n + \beta_{21}\phi^{(1)} &= \beta_{20}\phi^n + \beta_{21}(\phi^n + \frac{\partial \phi^n}{\partial c}(c^{(1)} - c^n)) + \mathcal{O}(\Delta t^2)\\
&= (\beta_{20} + \beta_{21})\phi^n + \beta_{21}\frac{\partial \phi^n}{\partial c}(c^{(1)} - c^n) + \mathcal{O}(\Delta t^2)\\
&= (\beta_{20} + \beta_{21})\phi^n + \Delta t \beta_{21}\beta_{10}\frac{\partial \phi^n}{\partial c}(F^n + P^n - D^n) + \mathcal{O}(\Delta t^2).
\end{aligned}
$$

Substituting the above relation into the main part of the right hand side of the second stage (2.20c) yields

$$
\begin{aligned}
&\sum_j (\beta_{20} p_{k,i,j}(c^{(0)}) + \beta_{21} p_{k,i,j}(c^{(1)}))\frac{c^{n+1}_{k,j}}{\sigma_{k,j}} - \sum_j (\beta_{20} d_{k,i,j}(c^{(0)}) + \beta_{21} d_{k,i,j}(c^{(1)}))\frac{c^{n+1}_{k,i}}{\sigma_{k,i}}\\
=&\sum_j \left((\beta_{20} + \beta_{21}) p^n_{k,i,j} + \Delta t \beta_{21}\beta_{10} \frac{\partial p^n_{k,i,j}}{\partial c}(F^n + P^n - D^n)\right)\left(1 - \Delta t \frac{\alpha_{21}\beta_{10}^2}{\beta_{20}+\beta_{21}} \frac{F^n_{k,j} + P^n_{k,j} - D^n_{k,j}}{c^n_{k,j}}\right)\\
&- \sum_j \left((\beta_{20} + \beta_{21}) d^n_{k,i,j} + \Delta t \beta_{21}\beta_{10} \frac{\partial d^n_{k,i,j}}{\partial c}(F^n + P^n - D^n)\right)\left(1 - \Delta t \frac{\alpha_{21}\beta_{10}^2}{\beta_{20}+\beta_{21}} \frac{F^n_{k,i} + P^n_{k,i} - D^n_{k,i}}{c^n_{k,i}}\right)\\
&+ \mathcal{O}(\Delta t^2),\\
=&(\beta_{20} + \beta_{21})(P^n_{k,i} - D^n_{k,i}) + \Delta t \beta_{21}\beta_{10} \frac{\partial (P^n_{k,i} - D^n_{k,i})}{\partial c}(F^n + P^n - D^n)\\
&- \Delta t \alpha_{21}\beta_{10}^2 \left(\sum_j p^n_{k,i,j} \frac{F^n_{k,j} + P^n_{k,j} - D^n_{k,j}}{c^n_{k,j}} - \sum_j d^n_{k,i,j} \frac{F^n_{k,i} + P^n_{k,i} - D^n_{k,i}}{c^n_{k,i}} \right) + \mathcal{O}(\Delta t^2).
\end{aligned}
$$

Finally, we simplify the second stage (2.20c):

$$
\begin{aligned}
c^{n+1}_{k,i} &= \alpha_{20} c^{(0)}_{k,i} + \alpha_{21}(c^n_{k,i} + \Delta t \beta_{10}(F^n_{k,i} + P^n_{k,i} - D^n_{k,i})\\
&\quad + \Delta t^2 \beta_{10}^2 (\sum_j p^n_{k,i,j} \frac{F^n_{k,j} + P^n_{k,j} - D^n_{k,j}}{c^n_{k,j}} - \sum_j d^n_{k,i,j} \frac{F^n_{k,i} + P^n_{k,i} - D^n_{k,i}}{c^n_{k,i}}))\\
&\quad + \Delta t((\beta_{20} + \beta_{21}) F^n_{k,i} + \Delta \beta_{21}\beta_{10} \frac{\partial F^n_{k,i}}{\partial c}(F^n + P^n - D^n))
\end{aligned}
$$

$$
\begin{aligned}
&+\Delta t((\beta_{20}+\beta_{21})(P^n_{k,i}-D^n_{k,i})+\Delta t\beta_{21}\beta_{10}\frac{\partial(P^n_{k,i}-D^n_{k,i})}{\partial c}(F^n+P^n-D^n)\\
&-\Delta t\alpha_{21}\beta_{10}^2(\sum_j p^n_{k,i,j}\frac{F^n_{k,j}+P^n_{k,j}-D^n_{k,j}}{c^n_{k,j}}-\sum_j d^n_{ij}\frac{F^n_{k,i}+P^n_{k,i}-D^n_{k,i}}{c^n_{k,i}}))+\mathcal{O}(\Delta t^3),\\
&=c^n_i+\Delta t(F^n_{k,i}+P^n_{k,i}-D^n_{k,i})+\frac{\Delta t^2}{2}\frac{\partial(F^n_{k,i}+P^n_{k,i}-D^n_{k,i})}{\partial c}(F^n+P^n-D^n)+\mathcal{O}(\Delta t^3).
\end{aligned}
$$

The second-order accuracy has been proved. We have the following results:

Theorem 2.2. *The MPRK scheme* (2.20) *is second-order accurate for the system of ODEs* (2.19)*. It is conservative:*

$$
\begin{aligned}
&\sum_{k,i} c^{(1)}_{k,i}=\sum_{k,i} c^n_{k,i}+\Delta t\beta_{10}\sum_{k,i} F_{k,i}(c^n),\\
&\sum_{k,i} c^{n+1}_{k,i}=\alpha_{20}\sum_{k,i} c^n_{k,i}+\alpha_{21}\sum_{k,i} c^{(1)}_{k,i}+\Delta t\sum_{k,i}(\beta_{20}F_{k,i}(c^n)+\beta_{21}F_{k,i}(c^{(1)})),
\end{aligned}
$$

where the summation of the convection terms will vanish if periodic boundaries are imposed. It is positivity-preserving: if $c^n_{k,i}\geq 0$ for all k,i, then $c^{(1)}_{k,i}\geq 0$ and $c^{n+1}_{k,i}\geq 0$ for all k,i and $\Delta t\leq \min\{\frac{\alpha_{10}}{\beta_{10}},\frac{\alpha_{20}}{\beta_{20}},\frac{\alpha_{21}}{\beta_{21}}\}\Delta t_0$.

Remark 2.6. *Note that the condition for the MPRK scheme* (2.20) *to be positivity-preserving does not depend on the production and destruction terms $P_{k,i}$ and $D_{k,i}$, but only relies on the explicit SSP RK solver for the convection part. This is highly desirable for solving Euler equations with stiff chemical reaction sources. The CFL conditions for positivity-preserving schemes developed in [26, 18] all depend on the stiffness of the source terms.*

3 Positivity-preserving finite difference WENO schemes

In this section, we first review the positivity-preserving finite difference WENO scheme [27]. Then we apply our ODE solver (2.20) to reactive Euler equations [18] and Euler equations with three species reactions [20] and discuss the positivity-preserving properties.

3.1 The finite difference WENO scheme for hyperbolic conservation laws

We first review the finite difference WENO scheme for hyperbolic conservation laws *without* source terms [15]

$$U_t + F(U)_x = 0, \tag{3.21}$$

where U is the unknown variable vector and $F(U)$ is the physical flux. Consider a uniform mesh with node x_i. Define $x_{i+\frac{1}{2}} = \frac{1}{2}(x_i + x_{i+1})$, $\Delta x = x_{i+1} - x_i$ and $I_i = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$. Denote the point values at x_i and time level n by W_i^n. The finite difference scheme with high order spatial discretization and Euler forward time discretization solving (3.21) has the form

$$W_i^{n+1} = W_i^n - \frac{\Delta t}{\Delta x}(\hat{F}_{i+\frac{1}{2}} - \hat{F}_{i-\frac{1}{2}}), \tag{3.22}$$

where $\frac{1}{\Delta x}(\hat{F}_{i+\frac{1}{2}} - \hat{F}_{i-\frac{1}{2}})$ should be a high order approximation to $F(W)_x$ at $x = x_i$.

If there exists a function $H(x)$ depending on the mesh size Δx such that

$$F(x) = \frac{1}{\Delta x}\int_{x-\Delta x/2}^{x+\Delta x/2} H(\xi)d\xi,$$

the we call F and H a reconstruction pair and denote them by

$$H = R_{\Delta x}(F), \quad F = R_{\Delta x}^{-1}(H).$$

Then $F_x = \frac{1}{\Delta x}(H(x+\frac{\Delta x}{2}) - H(x-\frac{\Delta x}{2}))$. Thus, if $\hat{F}_{j+\frac{1}{2}}$ is a high order accurate approximation of $H(x_{i+\frac{1}{2}})$, then $\frac{1}{\Delta x}(\hat{F}_{i+\frac{1}{2}} - \hat{F}_{i-\frac{1}{2}})$ will be a high order approximation of $F(W)_x$ at $x = x_i$.

We use the global Lax-Friedrichs splitting,

$$F_{\pm}(W) = W \pm \frac{F(W)}{\alpha},$$

Here α is the maximum eigenvalue of the Jacobian matrix $\frac{\partial F(U)}{\partial U}$, and the maximum is taken over all the grid points at time level n. For Euler equations of gas dynamics, $\alpha = \max(|u|+c)$ where u and c are the velocity and the speed of sound. For clarity, the notation $\pm$ on the subscripts denotes the positive and negative parts of the splitting flux, and that on the superscripts denotes different stencils in the WENO reconstructions for cell I_i and I_{i+1}. We also introduce the notation $L_{i+\frac{1}{2}}$ and $R_{i+\frac{1}{2}}$ which denote the left and right eigenvector matrix of $\frac{\partial F(U)}{\partial U}$ taking value at $W_{i+\frac{1}{2}}$ and satisfy $L_{i+\frac{1}{2}} = R^{-1}_{i+\frac{1}{2}}$. Here $W_{i+\frac{1}{2}}$ is some "average" of W_i and W_{i+1}. For simplicity, we take $W_{i+\frac{1}{2}} = \frac{1}{2}(W_i + W_{i+1})$. The finite difference WENO scheme are formulated as follows:

At each fixed $x_{i+\frac{1}{2}}$,

- Denote $H_\pm = R_{\Delta x}(F_\pm)$, then we have the cell averages $(\bar{H}_\pm)_i^n = F_\pm(W_i^n)$. Transform all the cell averages $(\bar{H}_\pm)_j^n$ for j in a neighborhood of i to the local characteristic field by setting

$$(\bar{V}_\pm)_j^n = L_{i+\frac{1}{2}}(\bar{H}_\pm)_j^n.$$

- Perform the WENO reconstruction for each component of $(\bar{V}_+)_j^n$ to obtain approximations of the point value of the function $L_{i+\frac{1}{2}}H_+$ at the point $x_{i+\frac{1}{2}}$ on the left side and denote them as $(V_+)^-_{i+\frac{1}{2}}$. Also, perform the WENO reconstruction for each component of $(\bar{V}_-)_j^n$ to obtain approximations of the point value of the function $L_{i+\frac{1}{2}}H_-$ at the point $x_{i+\frac{1}{2}}$ on the right side and denote them as $(V_-)^+_{i+\frac{1}{2}}$.

- Transform back into physical space by

$$(H_+)^-_{i+\frac{1}{2}} = R_{i+\frac{1}{2}}(V_+)^-_{i+\frac{1}{2}}, \quad (H_-)^+_{i+\frac{1}{2}} = R_{i+\frac{1}{2}}(V_-)^+_{i+\frac{1}{2}}.$$

- Form the flux by

$$\hat{F}_{i+\frac{1}{2}} = \frac{\alpha}{2}((H_+)^-_{i+\frac{1}{2}} - (H_-)^+_{i+\frac{1}{2}}).$$

Finally, we get the conservative scheme

$$W_i^{n+1} = W_i^n - \lambda(\hat{F}_{i+\frac{1}{2}} - \hat{F}_{i-\frac{1}{2}}), \tag{3.23}$$

with $\lambda = \Delta t/\Delta x$.

3.2 Positivity-preserving finite difference WENO schemes

In this part, we briefly review the positivity-preserving finite difference WENO schemes in [27]. For (1.1), we define the set of admissible states G:

$$G = \{U = (\rho_1, \cdots, \rho_{n_s}, m, E)^T | \quad \rho_i > 0, i = 1, \cdots, n_s, \quad p > 0\}. \tag{3.24}$$

Define $\tilde{e} := \sum_{i=1}^{n_s} \rho_i e_{in,i}(T)$. Then $\tilde{e}$ as a function of U is convex. The convexity of G is obtained by noticing $G = \{U| \quad \rho_i > 0, i = 1, \cdots, n_s, \quad \tilde{e} > 0\}$. See also Lemma 2.1 in [26] for details.

Next, we present the sufficient condition in [27] for the scheme to keep $W_i^{n+1} \in G$ provided $W_i^n \in G$.

$$\begin{aligned} W_i^{n+1} &= W_i^n - \lambda(\hat{F}_{i+\frac{1}{2}} - \hat{F}_{i-\frac{1}{2}}), \\ &= \frac{1}{2}((\bar{H}_+)_i^n + (\bar{H}_-)_i^n) - \frac{\lambda\alpha}{2}((H_+)_{i+\frac{1}{2}}^- - (H_-)_{i+\frac{1}{2}}^+ - (H_+)_{i-\frac{1}{2}}^- + (H_-)_{i-\frac{1}{2}}^+), \\ &:= \frac{1}{2}T^+ + \frac{1}{2}T^-, \end{aligned}$$

where

$$\begin{aligned} T^+ &= (\bar{H}_+)_i^n - \lambda\alpha((H_+)_{i+\frac{1}{2}}^- - (H_+)_{i-\frac{1}{2}}^-), \\ T^- &= (\bar{H}_-)_i^n + \lambda\alpha((H_-)_{i+\frac{1}{2}}^+ - (H_-)_{i-\frac{1}{2}}^+). \end{aligned}$$

Then, the cell averages $\bar{H}_+$ and $\bar{H}_-$ are split into a convex combination of values at quadrature points and the following theorem can be proved:

Theorem 3.1 (Zhang and Shu [27]). *Under the CFL condition $\alpha\lambda \le \hat{w}_1$, if $q_i^{+,*}$, $(H_+)_{i+\frac{1}{2}}^-$, $(H_+)_{i-\frac{1}{2}}^-$, $q_i^{-,*}$, $(H_-)_{i+\frac{1}{2}}^+$, $(H_-)_{i-\frac{1}{2}}^+ \in G$, then the finite difference scheme will be positivity-preserving, i.e., $W_i^{n+1} \in G$, where*

$$q_i^{+,*} = \frac{1}{1-\hat{w}_N}((\bar{H}_+)_i^n - \hat{w}_N(H_+)_{i+\frac{1}{2}}^-), \quad q_i^{-,*} = \frac{1}{1-\hat{w}_1}((\bar{H}_-)_i^n - \hat{w}_1(H_-)_{i-\frac{1}{2}}^+). \tag{3.25}$$

$\hat{w}_1 = \hat{w}_N$ are the quadrature weights for the two end points in Gauss-Lobatto quadrature rules.

Note that $\hat{w}_1 = \hat{w}_N = 1/6$ for the third-order WENO schemes and $1/12$ for the fifth-order WENO schemes.

To enforce the sufficient conditions in Theorem 3.1, the positivity-preserving limiter is applied. Here, we slightly modify the simple and robust limiter in [18] since the pressure is no longer a convex function of W in our model. Let $(\bar{H}_+)_i^n = ((\bar{\rho}_1)_i, \cdots, (\bar{\rho}_{n_s})_i, \bar{m}_i, \bar{E}_i)^T$, $(H_+)_{i+\frac{1}{2}}^- = ((\rho_1)_{i+\frac{1}{2}}^-, \cdots, (\rho_{n_s})_{i+\frac{1}{2}}^-, m_{i+\frac{1}{2}}^-, E_{i+\frac{1}{2}}^-)^T$, $q_i^{+,*} = ((\rho_1)_i^*, \cdots, (\rho_{n_s})_i^*, m_i^*, E_i^*)$. The limiter is presented as follows:

- Set up small parameters $\epsilon_s = \min_i\{10^{-13}, (\bar{\rho}_s)_i\}$ for $s = 1, \cdots, n_s$.

- For each cell I_i, modify the densities: for $s = 1, \cdots, n_s$,

$$(\hat{\rho}_s)_{i+\frac{1}{2}}^- = \theta_1((\rho_s)_{i+\frac{1}{2}}^- - (\bar{\rho}_s)_i) + (\bar{\rho}_s)_i, \quad \theta_1 = \min\{\left|\frac{(\bar{\rho}_s)_i - \epsilon_s}{(\bar{\rho}_s)_i - (\rho_s)_{\min}}\right|, 1\}, \tag{3.26}$$

  where $(\rho_s)_{\min} := \min\{(\rho_s)_{i+\frac{1}{2}}^-, (\rho_s)_i^*\}$. Then denote $(\hat{H}_+)_{i+\frac{1}{2}}^- := ((\hat{\rho}_1)_{i+\frac{1}{2}}^-, \cdots, (\hat{\rho_{n_s}})_{i+\frac{1}{2}}^-, m_{i+\frac{1}{2}}^-, E_{i+\frac{1}{2}}^-)^T$ and $\hat{q}_i^{+,*} := \frac{1}{1-\hat{w}_N}((\bar{H}_+)_i^n - \hat{w}_N(\hat{H}_+)_{i+\frac{1}{2}}^-)$.

- Then we modify $\tilde{e}$. For convenience, let $q^1 = (\hat{H}_+)_{i+\frac{1}{2}}^-$ and $q^2 = \hat{q}_i^{+,*}$. For $m = 1, 2$, if $\tilde{e}(q^m) \geq 0$, set $t^m = 1$; otherwise, set

$$t^m = \frac{\tilde{e}((\bar{H}_+)_i^n)}{\tilde{e}((\bar{H}_+)_i^n) - \tilde{e}(q^m)}. \tag{3.27}$$

  Then modify

$$(\tilde{H}_+)_{i+\frac{1}{2}}^- = \theta_2((\hat{H}_+)_{i+\frac{1}{2}}^- - (\bar{H}_+)_i^n) + (\bar{H}_+)_i^n, \quad \theta_2 = \min\{t_1, t_2\}. \tag{3.28}$$

  Similarly, we get the revised point value $(\tilde{H}_-)_{i-\frac{1}{2}}^+$. Then we have the modified WENO scheme with the numerical flux replaced by

$$\hat{F}_{i+\frac{1}{2}} = \frac{\alpha}{2}((\tilde{H}_+)_{i+\frac{1}{2}}^- - (\tilde{H}_-)_{i+\frac{1}{2}}^+). \tag{3.29}$$

It is straightforward to extend the positivity-preserving finite difference scheme for one-dimension to multi-space dimensions. The CFL condition for preserving the positivity for 2D case is replaced by

$$\Delta t(\frac{a_1}{\Delta x} + \frac{a_2}{\Delta y}) \leq \hat{w}_1, \tag{3.30}$$

where $a_1 = \max\{|u|+c\}$ and $a_2 = \max\{|v|+c\}$, with u, v are velocities in x and y directions, c the speed of sound. For the time integration, the SSP high order RK time discretization [5] will keep the validity of Theorem 3.1 since G is convex.

3.3 Reactive Euler equations

We consider the reactive Euler equations which are often used to model the detonation waves [18] in 2D case:

$$U_t + F(U)_x + G(U)_y = S(U), \tag{3.31}$$

with

$$\begin{aligned} U &= (\rho, m, n, E, \rho Y), \\ F(U) &= (m, \rho u^2 + p, \rho uv, (E+p)u, \rho uY), \\ G(U) &= (n, \rho uv, \rho v^2 + p, (E+p)v, \rho vY), \\ S(U) &= (0, 0, 0, 0, \omega), \end{aligned}$$

and

$$m = \rho u, \quad n = \rho v, \quad E = \frac{1}{2}\rho(u^2+v^2) + \frac{p}{\gamma - 1} + \rho q Y. \tag{3.32}$$

Here $q > 0$ is the heat release of reaction, γ is the specific heat ratio and $0 \le Y \le 1$ denotes the reactant mass fraction. The source term is assumed to be in an Arrhenius form

$$\omega = -\tilde{K}\rho Y e^{-\tilde{T}/T}, \tag{3.33}$$

where $T = p/\rho$ is the temperature, $\tilde{T} > 0$ is the activation constant temperature and $\tilde{K} > 0$ is a constant.

To fit into our framework, we rewrite (3.31) in an equivalent form. The unknown variables and the corresponding physical fluxes and source terms are replaced by

$$\begin{aligned} U &= (\rho Y, \rho Z, m, n, E), \\ F(U) &= (\rho u Y, \rho u Z, \rho u^2 + p, \rho uv, (E+p)u), \end{aligned}$$

$$G(U) = (\rho vY, \rho vZ, \rho uv, \rho v^2 + p, (E+p)v),$$
$$S(U) = (\omega, -\omega, 0, 0, 0),$$

where Z denotes the unreacted mass fraction. The admissible set G is defined as

$$G = \{U = (\rho Y, \rho Z, m, n, E) | \rho Y \geq 0, \rho Z \geq 0, p > 0\}.$$

The semi-discrete finite difference WENO scheme for the equivalent PDEs reads as

$$\frac{dW_{i,j}}{dt} = -\frac{1}{\Delta x}(\hat{F}_{i+\frac{1}{2},j} - \hat{F}_{i-\frac{1}{2},j}) - \frac{1}{\Delta y}(\hat{G}_{i,j+\frac{1}{2}} - \hat{G}_{i,j-\frac{1}{2}}) + S(W_{i,j}), \tag{3.34}$$

with $W_{i,j} := (W_{i,j,1}, W_{i,j,2}, W_{i,j,3}, W_{i,j,4}, W_{i,j,5}) = ((\rho Y)_{i,j}, (\rho Z)_{i,j}, m_{i,j}, n_{i,j}, E_{i,j})$. This system of ODEs can be written in the form (2.19) by splitting the source terms into

$$S_i(W) = P_i(W) - D_i(W), \tag{3.35}$$

with the production and destruction terms

$$p_{2,1}(W) = d_{1,2}(W) = -\omega \geq 0, \tag{3.36}$$

and $p_{m,n}(W)$, $d_{m,n}(W)$ vanish for other set of $1 \leq m, n \leq 5$. Then we apply our ODE solver (2.20) and obtain

$$\begin{aligned} W_{i,j}^{(1)} =& \alpha_{10} W_{i,j}^n - \beta_{10}\left(\frac{\Delta t}{\Delta x}(\hat{F}^n_{i+\frac{1}{2},j} - \hat{F}^n_{i-\frac{1}{2},j}) + \frac{\Delta t}{\Delta y}(\hat{G}^n_{i,j+\frac{1}{2}} - \hat{G}^n_{i,j-\frac{1}{2}})\right) \\ &+ \Delta t \beta_{10}(\omega^n \frac{W^{(1)}_{i,j,1}}{W^n_{i,j,1}}, -\omega^n \frac{W^{(1)}_{i,j,1}}{W^n_{i,j,1}}, 0, 0, 0)^T, \end{aligned} \tag{3.37a}$$

$$\begin{aligned} W_{i,j}^{n+1} =& \alpha_{20} W_{i,j}^n + \alpha_{21} W_{i,j}^{(1)} \\ &- \beta_{20}\left(\frac{\Delta t}{\Delta x}(\hat{F}^n_{i+\frac{1}{2},j} - \hat{F}^n_{i-\frac{1}{2},j}) + \frac{\Delta t}{\Delta y}(\hat{G}^n_{i,j+\frac{1}{2}} - \hat{G}^n_{i,j-\frac{1}{2}})\right) \\ &- \beta_{21}\left(\frac{\Delta t}{\Delta x}(\hat{F}^{(1)}_{i+\frac{1}{2},j} - \hat{F}^{(1)}_{i-\frac{1}{2},j}) + \frac{\Delta t}{\Delta y}(\hat{G}^{(1)}_{i,j+\frac{1}{2}} - \hat{G}^{(1)}_{i,j-\frac{1}{2}})\right) \\ &+ \Delta t((\beta_{20}\omega^n + \beta_{21}\omega^{(1)}) \frac{W^{n+1}_{i,j,1}}{(W^{(1)}_{i,j,1})^s (W^n_{i,j,1})^{1-s}}, -(\beta_{20}\omega^n + \beta_{21}\omega^{(1)}) \frac{W^{n+1}_{i,j,1}}{(W^{(1)}_{i,j,1})^s (W^n_{i,j,1})^{1-s}}, 0, 0, 0)^T. \end{aligned} \tag{3.37b}$$

By introducing auxiliary variables $V_{i,j}^{(1)}$ and $V_{i,j}^{n+1}$, we rewrite (3.37) into an equivalent form

$$V_{i,j}^{(1)} = \alpha_{10} W_{i,j}^n - \beta_{10}\left(\frac{\Delta t}{\Delta x}(\hat{F}^n_{i+\frac{1}{2},j} - \hat{F}^n_{i-\frac{1}{2},j}) + \frac{\Delta t}{\Delta y}(\hat{G}^n_{i,j+\frac{1}{2}} - \hat{G}^n_{i,j-\frac{1}{2}})\right), \tag{3.38a}$$

$$W_{i,j}^{(1)} = V_{i,j}^{(1)} + \Delta t \beta_{10}(\omega^n \frac{W_{i,j,1}^{(1)}}{W_{i,j,1}^n}, -\omega^n \frac{W_{i,j,1}^{(1)}}{W_{i,j,1}^n}, 0, 0, 0)^T, \tag{3.38b}$$

$$\begin{aligned} V_{i,j}^{n+1} = &\alpha_{20} W_{i,j}^n + \alpha_{21} W_{i,j}^{(1)} - \beta_{20}\left(\frac{\Delta t}{\Delta x}(\hat{F}^n_{i+\frac{1}{2},j} - \hat{F}^n_{i-\frac{1}{2},j}) + \frac{\Delta t}{\Delta y}(\hat{G}^n_{i,j+\frac{1}{2}} - \hat{G}^n_{i,j-\frac{1}{2}})\right) \\ &- \beta_{21}\left(\frac{\Delta t}{\Delta x}(\hat{F}^{(1)}_{i+\frac{1}{2},j} - \hat{F}^{(1)}_{i-\frac{1}{2},j}) + \frac{\Delta t}{\Delta y}(\hat{G}^{(1)}_{i,j+\frac{1}{2}} - \hat{G}^{(1)}_{i,j-\frac{1}{2}})\right), \end{aligned} \tag{3.38c}$$

$$W_{i,j}^{n+1} = V_{i,j}^{n+1} + \Delta t((\beta_{20}\omega^n + \beta_{21}\omega^{(1)}) \frac{W_{i,j,1}^{n+1}}{(W_{i,j,1}^{(1)})^s (W_{i,j,1}^n)^{1-s}}, -(\beta_{20}\omega^n + \beta_{21}\omega^{(1)}) \frac{W_{i,j,1}^{n+1}}{(W_{i,j,1}^{(1)})^s (W_{i,j,1}^n)^{1-s}}, 0, 0, 0)^T. \tag{3.38d}$$

At time level n, given $W_{i,j}^n \in G$ for all i, j, then the positivity of densities and pressure of $V_{i,j}^{(1)}$ in the first stage (3.38a) is guaranteed by the limiter in section 3.2. In the second stage (3.38b), the positivity of densities of $W_{i,j}^{(1)}$ is trivial. In addition, note that the density ρY of $W_{i,j}^{(1)}$ is no greater than that of $V_{i,j}^{(1)}$, i.e., $W_{i,j,1}^{(1)} \le V_{i,j,1}^{(1)}$, due to the fact that $\omega^n \le 0$. Then the positivity of pressure of $W_{i,j}^{(1)}$ is guaranteed by noticing that $p = (\gamma - 1)(E - \frac{1}{2}\frac{m^2}{\rho} - q\rho Y)$. In the third and the fourth stage, the positivity of densities and pressure of $V_{i,j}^{n+1}$ and $W_{i,j}^{n+1}$ can be preserved by the same argument.

3.4 Euler equations with three species reactions

We consider the three species model with a more general equation of state in [20]

$$U = (\rho_1, \rho_2, \rho_3, \rho u, E)^T,$$
$$F(U) = (\rho_1 u, \rho_2 u, \rho_3 u, \rho u^2 + p, (E+p)u)^T,$$
$$S(U) = (2M_1\omega, -M_2\omega, 0, 0, 0).$$

and

$$\rho = \sum_{s=1}^3 \rho_s, \quad p = RT\sum_{s=1}^3 \frac{\rho_s}{M_s}, \quad E = \sum_{s=1}^3 \rho_s e_s(T) + \rho_1 h_1^0 + \frac{1}{2}\rho u^2. \tag{3.39}$$

the internal energy $e_s(T) = 3RT/2M_s$ and $5RT/2M_s$ for monoatomic and diatomic species respectively. The rate of chemical reaction is given by

$$\omega = \left(k_f(T)\frac{\rho_2}{M_2} - k_b(T)(\frac{\rho_1}{M_1})^2 \right) \sum_{s=1}^{3} \frac{\rho_s}{M_s} \tag{3.40}$$

$$k_f = CT^{-2}e^{-E/T}, \quad k_b = k_f/\exp(b_1 + b_2 \log z + b_3 z + b_4 z^2 + b_5 z^3), \quad z = 10000/T \tag{3.41}$$

where b_i, C and E are constants which can be found in [26].

The semi-discrete finite difference WENO scheme for the equivalent PDEs reads as

$$\frac{dW_i}{dt} = -\frac{1}{\Delta x}(\hat{F}_{i+\frac{1}{2}} - \hat{F}_{i-\frac{1}{2}}) + S(W_i), \tag{3.42}$$

with $W_i := (W_{i,1}, W_{i,2}, W_{i,3}, W_{i,4}, W_{i,5}) = ((\rho_1)_i, (\rho_2)_i, (\rho_3)_i, m_i, E_i)$. For this model, we split the source terms $2M_1\omega$ into two parts:

$$2M_1\omega = \omega_+ - \omega_-,$$

with the positive part

$$\omega_+ = 2M_1 k_f(T)\frac{\rho_2}{M_2}\sum_{s=1}^{3}\frac{\rho_s}{M_s} \geq 0,$$

and the negative part

$$\omega_- = 2M_1 k_b(T)(\frac{\rho_1}{M_1})^2\sum_{s=1}^{3}\frac{\rho_s}{M_s} \geq 0.$$

This system of ODEs can be written in the form (2.19) by splitting the source terms into

$$S_i(W) = P_i(W) - D_i(W), \tag{3.43}$$

with the production and destruction terms

$$p_{2,1}(W) = d_{1,2}(W) = \omega_- \geq 0, \quad p_{1,2}(W) = d_{2,1}(W) = \omega_+ \geq 0, \tag{3.44}$$

and $p_{m,n}(W)$, $d_{m,n}(W)$ vanish for other set of $1 \leq m, n \leq 5$. Then we apply our ODE solver (2.20) and obtain

$$W_i^{(1)} = \alpha_{10}W_i^n - \beta_{10}\frac{\Delta t}{\Delta x}(\hat{F}^n_{i+\frac{1}{2}} - \hat{F}^n_{i-\frac{1}{2}})$$

$$+\Delta t\beta_{10}(\omega_+^n\frac{W_{i,2}^{(1)}}{W_{i,2}^n}-\omega_-^n\frac{W_{i,1}^{(1)}}{W_{i,1}^n},-\omega_+^n\frac{W_{i,2}^{(1)}}{W_{i,2}^n}+\omega_-^n\frac{W_{i,1}^{(1)}}{W_{i,1}^n},0,0,0)^T, \tag{3.45a}$$

$$\begin{aligned}W_i^{n+1}=&\alpha_{20}W_i^n+\alpha_{21}W_i^{(1)}-\beta_{20}\frac{\Delta t}{\Delta x}(\hat{F}_{i+\frac{1}{2}}^n-\hat{F}_{i-\frac{1}{2}}^n)-\beta_{21}\frac{\Delta t}{\Delta x}(\hat{F}_{i+\frac{1}{2}}^{(1)}-\hat{F}_{i-\frac{1}{2}}^{(1)})\\&+\Delta t((\beta_{20}\omega_+^n+\beta_{21}\omega_+^{(1)})\frac{W_{i,2}^{n+1}}{(W_{i,2}^{(1)})^s(W_{i,2}^n)^{1-s}}-(\beta_{20}\omega_-^n+\beta_{21}\omega_-^{(1)})\frac{W_{i,1}^{n+1}}{(W_{i,1}^{(1)})^s(W_{i,1}^n)^{1-s}},\\&-(\beta_{20}\omega_+^n+\beta_{21}\omega_+^{(1)})\frac{W_{i,2}^{n+1}}{(W_{i,2}^{(1)})^s(W_{i,2}^n)^{1-s}}+(\beta_{20}\omega_-^n+\beta_{21}\omega_-^{(1)})\frac{W_{i,1}^{n+1}}{(W_{i,1}^{(1)})^s(W_{i,1}^n)^{1-s}},0,0,0)^T.\end{aligned} \tag{3.45b}$$

For this model, our scheme (3.45) could only preserve the positivity of the internal energy $(E-\frac{m^2}{2\rho})$, but could not guarantee the positivity of the pressure. Actually, for this chemical reacting flow, the pressure is

$$p=\frac{\frac{\rho_1}{M_1}+\frac{\rho_2}{M_2}+\frac{\rho_3}{M_3}}{\frac{3\rho_1}{2M_1}+\frac{5\rho_2}{2M_2}+\frac{5\rho_3}{2M_3}}(E-\rho_1h_1^0-\frac{m^2}{2\rho}), \tag{3.46}$$

which may become negative due to the existence of $\rho_1h_1^0$. We also remark that this difficulty seems to be essential, since even the first-order splitting with the exact evolution in time for the ODEs part does not necessarily preserve the positivity of the pressure.

In our numerical experiments, if there exists negative pressure in the calculation, we just take the absolute value of the pressure when calculating the sound of speed. Moreover, we do not enforce the positivity of pressure in the limiter in section 3.2.

To conclude this section, we give several remarks below:

Remark 3.1. *We use the finite difference method as the spatial discretization and not the finite volume and DG schemes. The reason is that, for the finite volume and DG schemes, the semi-discrete scheme in general could not preserve the original form of the source.*

Remark 3.2. *We also remark that our ODE solver can be applied to convection-diffusion-reaction equations by treating the convection and diffusion terms explicitly in the time integration, as long as the reaction terms can be written in the production-destruction form.*

4 Numerical tests

In this part, the numerical results will be presented. We first discuss the convergence order for the non-stiff problems and the performance on the stiff problems in solving the ODEs. Then we move to reactive Euler equations and Euler equations with three species reactions and general equation of state. Since the second-order SSP RK method is linearly unstable when coupled with the fifth-order WENO spatial discretization [19], we adopt the third-order finite difference WENO scheme proposed in [12]. To ensure the sufficient conditions of the positivity of solutions, the CFL number is set to be 1/6. We also take a similar strategy as in [18]: if a preliminary calculation to the next time step produces negative density or pressure, then recalculate from the time step n with half of the previous time step.

4.1 ODEs

Example 4.1 (linear case)**.** The linear test case is

$$\frac{dc_1}{dt} = c_2 - ac_1,$$
$$\frac{dc_2}{dt} = ac_1 - c_2,$$

with constant $a > 0$, and initial value

$$c_1(0) = c_1^0, \quad c_2(0) = c_2^0. \tag{4.47}$$

The exact solution is

$$c_1(t) = (1 + b\exp(-(a+1)t))c_1^\infty, \quad c_2(t) = c_1^0 + c_2^0 - c_1(t), \tag{4.48}$$

with the parameters c_1^∞ and b determined by

$$c_1^\infty = \frac{c_1^0 + c_2^0}{a+1}, \quad b = \frac{c_1^0}{c_1^\infty} - 1. \tag{4.49}$$

In the numerical experiment, we take $c_1^0 = 4.5$, $c_2^0 = 3.2$, $a = 2.7$ and the final time $t = 1$. The errors between numerical solutions and exact solutions at the final time are listed in Table 4.1. The second-order accuracy is clearly observed.

Example 4.2 (nonlinear case)**.** To test the accuracy of our solver (2.20), we make up a non-stiff nonlinear problem:

$$\begin{aligned}\frac{dc_1}{dt} &= F_1(c) - \frac{c_1c_2}{c_1+1},\\ \frac{dc_2}{dt} &= F_2(c) + \frac{c_1c_2}{c_1+1} - ac_2,\\ \frac{dc_3}{dt} &= F_3(c) + ac_2.\end{aligned}$$

where $(F_1(c), F_2(c), F_3(c))$ denotes "convection terms".

To express this system of ODEs in the form of production-destruction equations, we set

$$p_{21}(c) = d_{21}(c) = \frac{c_1c_2}{c_1+1}, \quad p_{32}(c) = d_{32}(c) = ac_2, \tag{4.50}$$

and $p_{ij} = d_{ij} = 0$ for other sets of i, j. In the numerical examples, the initial conditions are set as $c_1^0 = 9.98$, $c_2 = 0.01$ and $c_3 = 0.01$. The convection terms are

$$(F_1(c), F_2(c), F_3(c)) = (c_1c_2c_3, \frac{c_3}{c_2}, c_1c_2c_3^2). \tag{4.51}$$

The final time is $t = 1$ and the parameter $a = 1$. The reference solutions are generated by the fourth-order explicit Runge-Kutta method with small enough time step. The errors are listed in Table 4.2, which shows second-order accuracy.

Example 4.3 (stiff case)**.** One of the most prominent examples of the stiff ODEs is the Robertson test case, which describes the chemical reactions [6]:

$$\frac{dc_1}{dt} = 10^4c_2c_3 - 4\times10^{-2}c_1,$$

Table 4.1: Example 4.1: Error table for linear ODEs.

Δt	error	order
1/20	1.20e-03	-
1/40	3.07e-04	1.97
1/80	7.78e-05	1.98
1/160	1.96e-05	1.99
1/320	4.91e-06	1.99

$$\frac{dc_2}{dt} = 4 \times 10^{-2} c_1 - 10^4 c_2 c_3 - 3 \times 10^7 c_2^2,$$
$$\frac{dc_3}{dt} = 3 \times 10^7 c_2^2,$$

with initial values $c_i(0) = c_i^0$ for $i = 1, 2, 3$.

For this problem, the production and destruction terms are

$$p_{12}(c) = d_{21}(c) = 10^4 c_2 c_3, \quad p_{21}(c) = d_{21}(c) = 4 \times 10^{-2} c_1, \quad p_{32}(c) = d_{23}(c) = 3 \times 10^7 c_2^2, \tag{4.52}$$

and $p_{ij} = d_{ij} = 0$ for other sets of i, j. In the numerical simulations, we take $c_1^0 = 1$ and $c_2^0 = c_3^0 = 0$. Following [10], the time step size in the k-th time step is chosen as $\Delta t_k = 2^{k-1} \Delta t_1$ with the initial time step size $\Delta t_1 = 10^{-6}$. The small initial time step was set to obtain an adequate resolution of the component c_2 in the starting time interval. The reference solutions are generated by the fourth-order explicit Runge-Kutta method with small enough time step. To visualize the evolution of c_2, it is multiplied by 10^4 in Fig. 4.1. From Fig. 4.1, we observe the excellent accuracy of our scheme in the case of a highly stiff problem.

4.2 Reactive Euler equations

Example 4.4 (Detonation diffraction problems). We test the detonation diffraction in this example. The same parameters and initial conditions with [18] are applied here. The initial conditions are, if $x < 0.5$, then $(\rho, u, v, E, Y) = (11, 6.18, 0, 970, 1)$; otherwise, $(\rho, u, v, E, Y) = (1, 0, 0, 55, 1)$. The boundary conditions are reflective except that at $x = 0$, $(\rho, u, v, E, Y) = (11, 6.18, 0, 970, 1)$. The terminal time is $t = 0.6$. The parameters are

Table 4.2: Example 4.2: Error table for nonlinear ODEs.

Δt	error	order
1/20	1.35e-03	-
1/40	3.18e-04	2.09
1/80	7.69e-05	2.05
1/160	1.89e-05	2.03
1/320	4.68e-06	2.01

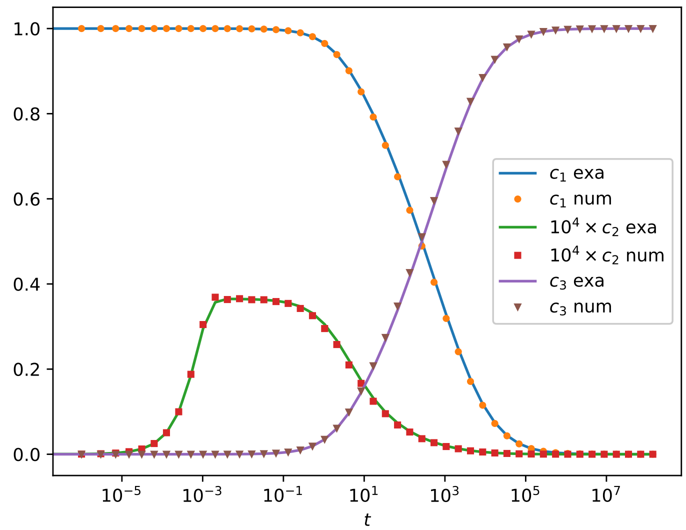


Figure 4.1: Example 4.3: Time evolution of c_i, $i = 1, 2, 3$.

$\gamma = 1.2$, $q = 50$, $\tilde{T} = 50$ and $\tilde{K} = 2566.4$. The numerical results with $\Delta x = \Delta y = 1/48$ are shown in Figure 4.2, which are comparable to the results in [18].

In order to demonstrate the advantage of our method over the explicit RK method, we make up another test case. Since the effect of the source term is not strong in the above detonation problem, we artificially take a smaller ignition temperature $\tilde{T} = 2$, a coarse mesh size $\Delta x = \Delta y = 1/24$ and a smaller terminal time $t = 0.35$, while the other parameters and conditions are the same with the above test case. The contour maps of density and pressure are presented in Figure 4.3, where our results are comparable to the results with the explicit SSP RK2 method. The time steps in the calculation are 806 and 2146 for our method and the explicit RK method, respectively. This indicates that our method behaves much better than the explicit RK method in terms of computational efficiency.

Example 4.5 (Multiple obstacles)**.** Following Example 4.6 in [18], we design a numerical test with multiple obstacles. The locations of the obstacles are different from those in [18]. In our test, the location of the first obstacle is $[1, 3] \times [0, 3]$ and the second one is $[5, 10] \times [0, 5]$. The uniform mesh can be easily applied. The initial condition is, if $x^2 + y^2 \leq 0.36$, then

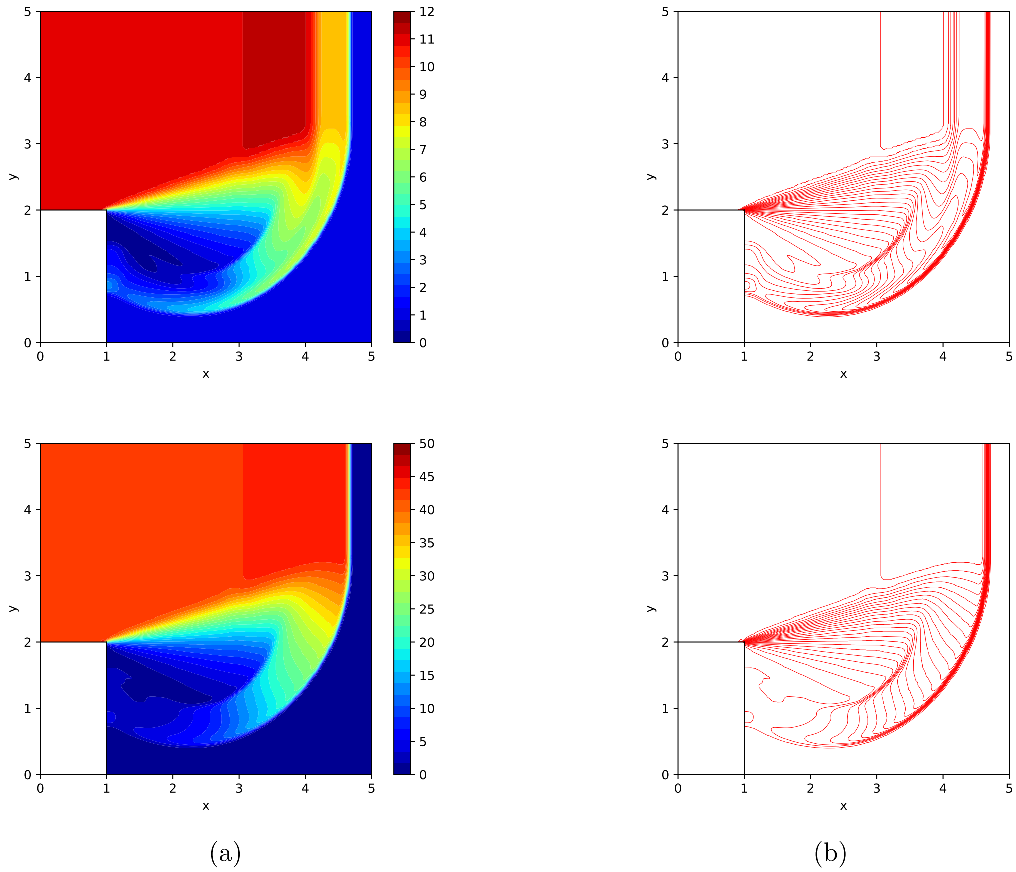
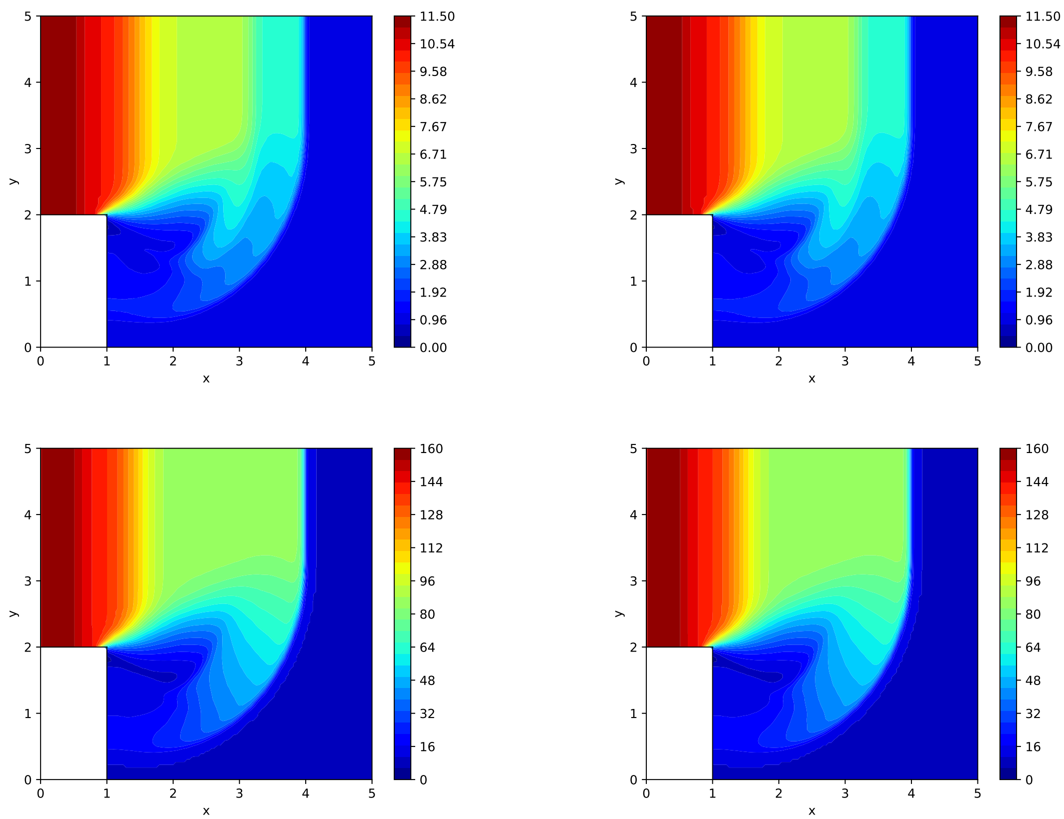
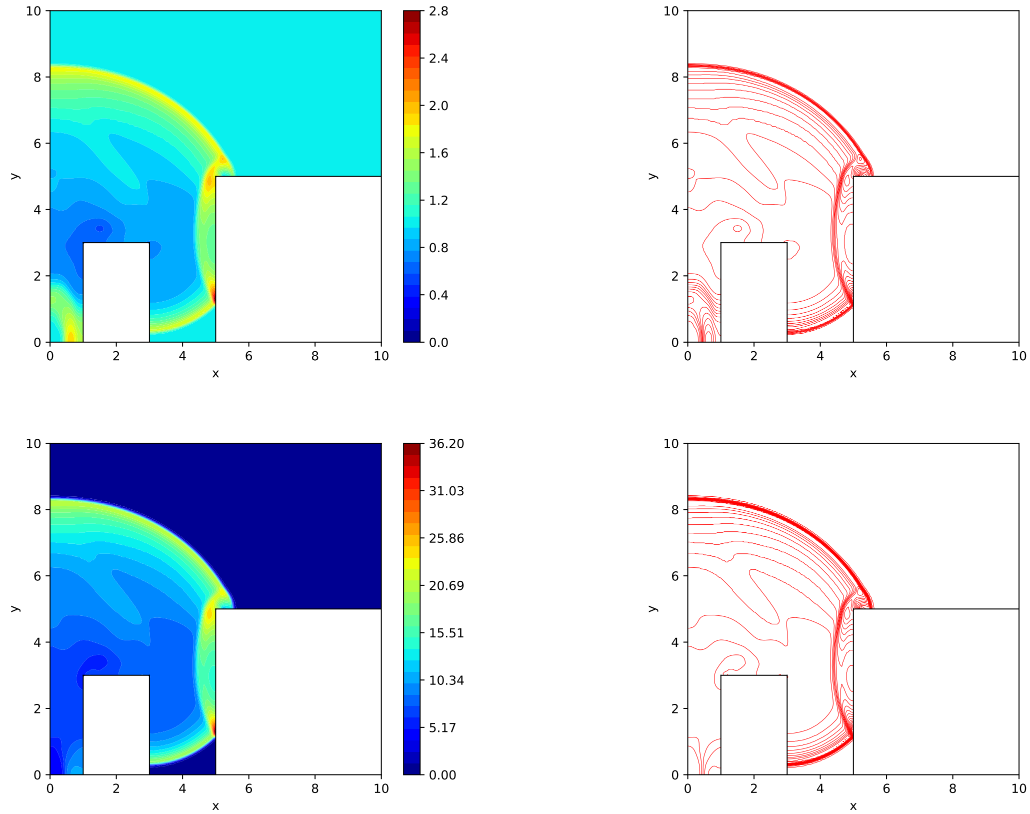


Figure 4.2: Example 4.4: Detonation diffraction problems. Top: colored contour map and contour line of density; bottom: colored contour map and contour line of pressure.



(a) (b)

Figure 4.3: Example 4.4: Detonation diffraction problems: our method vs. explicit SSP RK2 method. Top: colored contour maps of density; bottom: colored contour maps of pressure. Left: our method; right: explicit SSP RK2.



(a) (b)

Figure 4.4: Example 4.5: Multiple obstacles. Top: colored contour map and contour line of density; bottom: colored contour map and contour line of pressure.

$(\rho, u, v, E, Y) = (7, 0, 0, 200, 0)$; otherwise, $(\rho, u, v, E, Y) = (1, 0, 0, 55, 1)$. The boundary conditions are reflective for any boundaries. The parameters are set as $\gamma = 1.2$, $q = 50$, $\tilde{T} = 20$, $\tilde{K} = 2410.2$. The colored contour map and the contour line of the density and pressure with the mesh size $\Delta x = \Delta y = 1/20$ are presented in Figure 4.4.

4.3 Euler equations with three species reactions and general equation of state

Example 4.6. We solve the three species model of the one-dimensional Euler system with a more general equation of state in [20, 27], with the same parameters as in [20, 27]. The parameters are $M_1 = 0.016$, $M_2 = 0.032$, $M_3 = 0.028$, $h_1^0 = 1.558 \times 10^7$, $R = 8.31447215$, $C_0 = 2.9 \times 10^{17}$ m^3, $E_0 = 59750$ K, and $b_1 = 2.855$, $b_2 = 0.988$, $b_3 = -6.181$, $b_4 = -0.023$,

$b_5 = -0.001$. The eigenvalues of the Jacobian are $(u, u, u, u - c, u + c)$ where $c = \sqrt{\frac{\gamma p}{\rho}}$ with $\gamma = 1 + \frac{p}{T \sum_{s=1}^{3} \rho_s e_s'(T)}$. The initial conditions are: the densities ρ_1, ρ_2 and ρ_3 are $5.251896311257204 \times 10^{-5}$, $3.748071704863518 \times 10^{-5}$, $2.962489471973072 \times 10^{-4}$ on the left, and $8.341661837019181 \times 10^{-8}$, $9.455418692098664 \times 10^{-11}$, $2.748909430004963 \times 10^{-7}$ on the right. The velocities are zero. The pressures are 1000 on the left and 1 on the right. The final time is $t = 0.0001$. The profiles of densities, velocity and pressure are presented in Figure 4.5, where the converged solutions are observed.

5 Concluding remarks

In this paper, we have developed a class of time integration methods for the production-destruction equations, which is conservative and unconditionally positivity-preserving. The necessary and sufficient conditions for the methods to be second-order accurate are derived. This ODE solver is then extended to cover a class of semi-discrete schemes for PDEs and successfully applied to finite difference WENO schemes for non-equilibrium flows. We have tested the third-order WENO scheme with the positivity limiter coupled with the time integration method on a variety of numerical examples. Generalizations to third-order schemes constitute our ongoing work.

Acknowledgements

We would like to thank Xiangxiong Zhang from Purdue University and Tao Xiong from Xiamen University for many fruitful discussions.

References

[1] H. Burchard, E. Deleersnijder, and A. Meister. A high-order conservative Patankar-type discretisation for stiff systems of production–destruction equations. *Applied Numerical Mathematics*, 47(1):1–30, 2003.

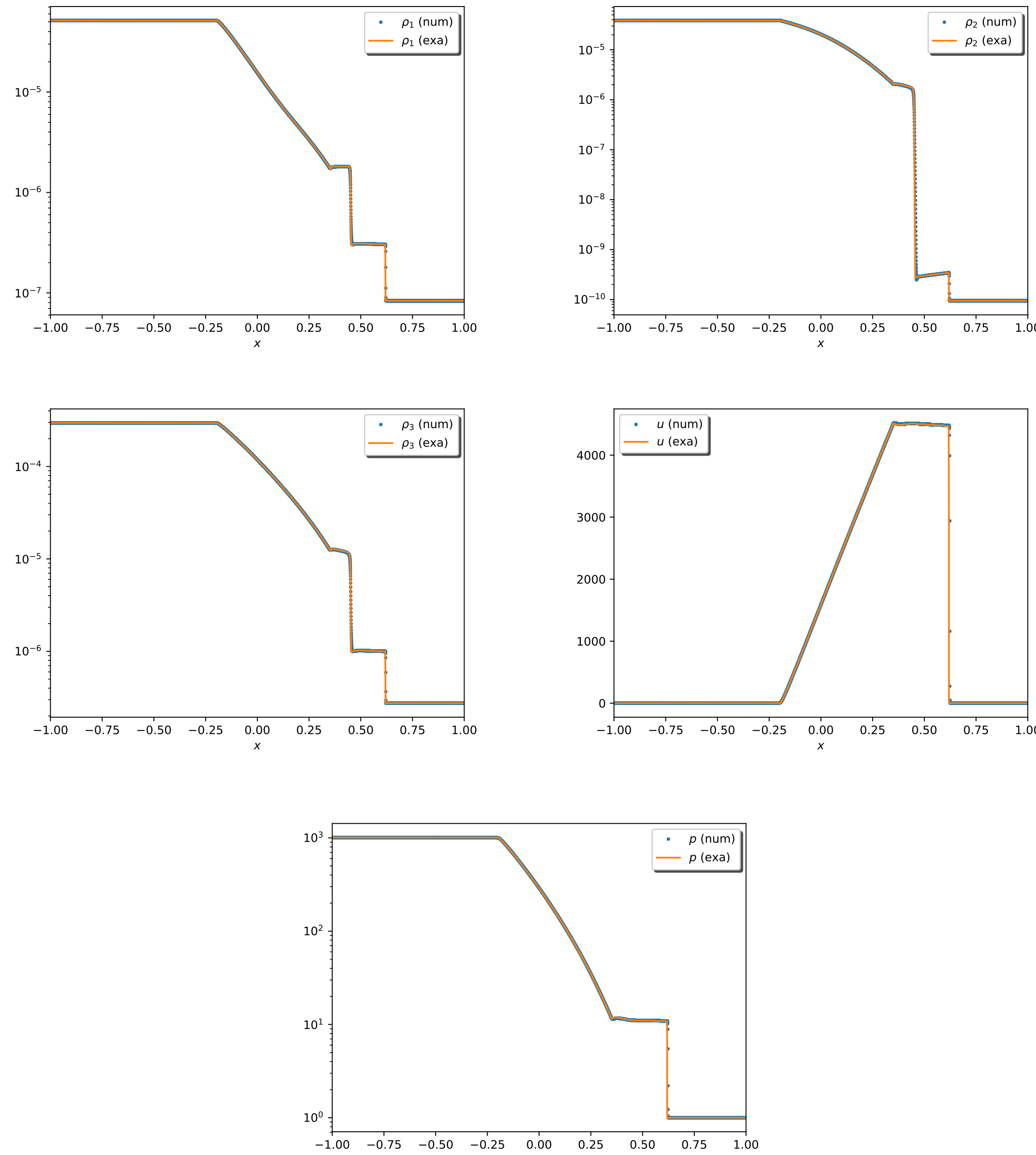


Figure 4.5: Example 4.6: Three species reaction problem at $t = 0.0001$. The solid lines are the reference solutions with $\Delta x = 2/8000$. Symbols are the numerical solutions with $\Delta x = 2/4000$.

[2] A. Chertock, S. Cui, A. Kurganov, and T. Wu. Steady state and sign preserving semi-implicit Runge–Kutta methods for ODEs with stiff damping term. *SIAM Journal on Numerical Analysis*, 53(4):2008–2029, 2015.

[3] A. Chertock, S. Cui, A. Kurganov, and T. Wu. Well-balanced positivity preserving central-upwind scheme for the shallow water system with friction terms. *International Journal for Numerical Methods in Fluids*, 78(6):355–383, 2015.

[4] L. Formaggia and A. Scotti. Positivity and conservation properties of some integration schemes for mass action kinetics. *SIAM Journal on Numerical Analysis*, 49(3):1267–1288, 2011.

[5] S. Gottlieb, C.-W. Shu, and E. Tadmor. Strong stability-preserving high-order time discretization methods. *SIAM review*, 43(1):89–112, 2001.

[6] E. Hairer and G. Wanner. Solving ordinary differential equations II: stiff and differential-algebraic problems. 8(3), 1996.

[7] J. Hu, R. Shu, and X. Zhang. Asymptotic-preserving and positivity-preserving implicit-explicit schemes for the stiff BGK equation. *SIAM Journal on Numerical Analysis*, 56(2):942–973, 2018.

[8] J. Huang and C.-W. Shu. A second-order asymptotic-preserving and positivity-preserving discontinuous Galerkin scheme for the Kerr–Debye model. *Mathematical Models and Methods in Applied Sciences*, 27(03):549–579, 2017.

[9] J. Huang and C.-W. Shu. Bound-preserving modified exponential Runge-Kutta discontinuous Galerkin methods for scalar hyperbolic equations with stiff source terms. *Journal of Computational Physics*, 361, 2018.

[10] S. Kopecz and A. Meister. On order conditions for modified Patankar-Runge-Kutta schemes. *Applied Numerical Mathematics*, 123:159–179, 2018.

[11] S. Kopecz and A. Meister. Unconditionally positive and conservative third order modified Patankar–Runge–Kutta discretizations of production–destruction systems. *BIT Numerical Mathematics*, to appear. DOI: 10.1007/s10543-018-0705-1.

[12] X.-D. Liu, S. Osher, and T. Chan. Weighted Essentially Non-oscillatory Schemes. *Journal of Computational Physics*, 115(1):200–212, 1994.

[13] A. Meister and S. Ortleb. On unconditionally positive implicit time integration for the DG scheme applied to shallow water flows. *International Journal for Numerical Methods in Fluids*, 76(2):69–94, 2014.

[14] S. Patankar. *Numerical Heat Transfer and Fluid Flow.* CRC press, 1980.

[15] C.-W. Shu. *Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws*, pages 325–432. Springer Berlin Heidelberg, Berlin, Heidelberg, 1998.

[16] C.-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of Computational Physics*, 77(2):439–471, 1989.

[17] G. Strang. On the construction and comparison of difference schemes. *SIAM Journal on Numerical Analysis*, 5(3):506–517, 1968.

[18] C. Wang, X. Zhang, C.-W. Shu, and J. Ning. Robust high order discontinuous Galerkin schemes for two-dimensional gaseous detonations. *Journal of Computational Physics*, 231(2):653–665, 2012.

[19] R. Wang and R. J. Spiteri. Linear instability of the fifth-order WENO method. *SIAM Journal on Numerical Analysis*, 45(5):1871–1901, 2007.

[20] W. Wang, C.-W. Shu, H. Yee, and B. Sjögreen. High-order well-balanced schemes and applications to non-equilibrium flow. *Journal of Computational Physics*, 228(18):6682–6702, 2009.

[21] Y. Xing and X. Zhang. Positivity-preserving well-balanced discontinuous Galerkin methods for the shallow water equations on unstructured triangular meshes. *Journal of Scientific Computing*, 57(1):19–41, 2013.

[22] Y. Xing, X. Zhang, and C.-W. Shu. Positivity-preserving high order well-balanced discontinuous Galerkin methods for the shallow water equations. *Advances in Water Resources*, 33(12):1476–1493, 2010.

[23] X. Zhang. On positivity-preserving high order discontinuous Galerkin schemes for compressible navier–stokes equations. *Journal of Computational Physics*, 328:301–343, 2017.

[24] X. Zhang and C.-W. Shu. On maximum-principle-satisfying high order schemes for scalar conservation laws. *Journal of Computational Physics*, 229(9):3091–3120, 2010.

[25] X. Zhang and C.-W. Shu. On positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations on rectangular meshes. *Journal of Computational Physics*, 229(23):8918–8934, 2010.

[26] X. Zhang and C.-W. Shu. Positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations with source terms. *Journal of Computational Physics*, 230(4):1238–1248, 2011.

[27] X. Zhang and C.-W. Shu. Positivity-preserving high order finite difference WENO schemes for compressible Euler equations. *Journal of Computational Physics*, 231(5):2245–2258, 2012.

[28] Y. Zhang, X. Zhang, and C.-W. Shu. Maximum-principle-satisfying second order discontinuous Galerkin schemes for convection–diffusion equations on triangular meshes. *Journal of Computational Physics*, 234:295–316, 2013.