A Positivity-Preserving Relaxation Algorithm

Thomas Izgin[1,3*], Hendrik Ranocha[2] and Chi-Wang Shu[3]

[1*]Department of Mathematics, University of Kassel, Heinrich-Plett-Str. 40, 34132, Kassel, Germany.
[2]Institute of Mathematics, Johannes Gutenberg University Mainz, Staudingerweg 9, 55128, Mainz, Germany.
[3]Division of Applied Mathematics, Brown University, Providence, Rhode Island 02906, USA.

*Corresponding author(s). E-mail(s): izgin@mathematik.uni-kassel.de;
Contributing authors: hendrik.ranocha@uni-mainz.de;
chi-wang_shu@brown.edu;

Abstract

We combine Patankar-type methods with suitable relaxation procedures that are capable of ensuring correct dissipation or conservation of functionals such as entropy or energy while producing unconditionally positive and conservative approximations. To that end, we adapt the relaxation algorithm to enforce positivity by using either ideas from the dense output framework when a linear invariant must be preserved, or simply a geometric mean if the only constraint is positivity preservation. The latter merely requires the solution of a scalar nonlinear equation while former results in a coupled linear-nonlinear system of equations. We present sufficient conditions for the solvability of the respective equations. Several applications in the context of ordinary and partial differential equations are presented, and the theoretical findings are validated numerically.

Keywords: Positivity preservation, Relaxation methods, Entropy stability

MSC Classification: 65M06 , 65M08 , 65M20 , 65M22

1 Introduction

We consider initial-value problems (IVPs)

$$\mathbf{u}'(t) = \mathbf{f}(\mathbf{u}(t)), \quad \mathbf{u}(t_0) = \mathbf{u}^0 \in \mathbb{R}^d, \tag{1}$$

either as classical ordinary differential equation (ODE) model on its own, or more typically obtained after discretizing a partial differential equation (PDE) in space. We are interested in two types of structures of the IVP (1). First, many applications require positive solutions, i.e., $\mathbf{u}(t) > \mathbf{0}$ for all $t \geq t_0$ if $\mathbf{u}^0 > \mathbf{0}$, where inequalities are understood component-wise. This occurs, for example, when modeling chemical reactions, population dynamics, or the density of fluids. Second, many problems are equipped with additional functionals of interest, such as Lyapunov functionals, energy, or entropy. We say that the IVP (1) is *dissipative* with respect to a smooth functional η, if $\eta'(\mathbf{u})\mathbf{f}(\mathbf{u}) \leq \mathbf{0}$, i.e.,

$$\frac{\mathrm{d}}{\mathrm{d}t}\eta(\mathbf{u}(t)) \leq 0$$

for all solutions $\mathbf{u}(t)$ of (1). Similarly, (1) is *conservative* with respect to η, if $\eta'(\mathbf{u})\mathbf{f}(\mathbf{u}) = \mathbf{0}$, i.e.,

$$\frac{\mathrm{d}}{\mathrm{d}t}\eta(\mathbf{u}(t)) = 0.$$

In addition to the (typically nonlinear) functional η, many problems also conserve additional linear invariants, such as mass or momentum, which we also want to preserve on the discrete level.

When discretizing (1) in time using a one-step method, we would like to preserve these properties, i.e., we would like to have an unconditionally positive method satisfying

$$\mathbf{u}^0 > \mathbf{0} \implies \mathbf{u}^n > \mathbf{0} \quad \text{for all } n \geq 0.$$

For many positive ODEs/PDEs, avoiding negative approximations is critical; such artifacts can lead to qualitatively incorrect solutions or the total failure of the numerical method [1–4]. Moreover, for dissipative problems, we would like to use a dissipative method that satisfies

$$\eta(\mathbf{u}^n) \leq \eta(\mathbf{u}^{n-1}) \leq \ldots \leq \eta(\mathbf{u}^0). \tag{2}$$

Similarly, a conservative method applied to a conservative problem should satisfy

$$\eta(\mathbf{u}^n) = \eta(\mathbf{u}^{n-1}) = \ldots = \eta(\mathbf{u}^0). \tag{3}$$

For convex η, the implicit Euler method is a well-known example of an unconditionally positive and dissipative method. However, this analysis neglects possible positivity issues that can arise while solving the implicit equations as well as remaining errors of the nonlinear iterative solver.

Concerning positivity, the implicit Euler method is essentially the best method one can use in the class of general linear methods, since any unconditionally positive method can be at most first-order accurate [5]. To address this challenge, several strategies have been proposed:

1. *Clipping techniques*, which forcibly set negative values to zero, either result in a mass-shifting optimization problem or otherwise compromise conservation of linear invariants, and, to date, lack a proof of stability [6].
2. *Projection techniques* [2, 7] can be positive and conserve linear invariants, but they may result in step size constraints and/or reduced accuracy.
3. *Fully implicit, nonlinear methods* [8, 9] can enforce positivity but require costly iterative solvers, which may fail to converge (to a positive solution), and thus, still produce nonphysical results.
4. *Diagonally split Runge–Kutta (DSRK) methods* [10] can be unconditionally positive and with order higher than one. However, they are typically less accurate than the implicit Euler method in practice [11].
5. *Adaptive methods* [3] use root-finding procedures and adapt the time step size. This can be effective, but the resulting schemes are only conditionally positive.
6. *Strong stability preserving (SSP) methods* [12] are positive if the explicit Euler method is positive under a certain time step restriction. However, only the implicit Euler method leads to unconditional positivity, and thus, all other SSP methods are only conditionally positive.
7. *Patankar-type methods* represent a family of explicit or linearly implicit yet nonlinear schemes, which are unconditionally positive and can preserve certain linear invariants [13–17].

In this work, we focus on Patankar-type schemes. The main idea behind them is to modify an existing time-stepping method by introducing nonlinear weights in such a way that the resulting numerical scheme becomes unconditionally positive. The primary challenge lies in designing these weights so that the modified scheme preserves the accuracy of the original (baseline) method. This nonlinear modification is achieved using the so-called *Patankar-trick* [13], which gives this family of methods its name. A notable example is the incorporation of modified Patankar (MP) weights into classical Runge–Kutta (RK) schemes, leading to the development of modified Patankar–Runge–Kutta (MPRK) methods [14, 16, 18], which in addition to being unconditionally positive, are also conservative. Motivated by their strong numerical performance, the Patankar-trick has since been successfully extended to a variety of time integration frameworks, including SSP Runge–Kutta (SSPRK) methods [4, 19], arbitrary high-order Deferred Correction (DeC) schemes [20], generalized BBKS methods [17], GeCo schemes [15], and linear multistep methods [21]. The resulting modified schemes all belong to the broader Patankar-type family, which can themselves be recast as non-standard additive Runge–Kutta (NSARK) methods, see [22, 23].

Concerning the preservation (conservation/dissipation) of functionals η, several results are available for linear schemes such as RK methods applied to linear problems [24–30] and fully-implicit methods [31–35]. There are also positive results on dissipative schemes if the problem is sufficiently dissipative [36–38]. In the general case including conservative problems, however, results are restrictive and include many negative results [39, 40]. Similarly to positivity preservation, postprocessing/projection methods can be used to enforce the desired conservation/dissipation properties of time integration methods [41–45]. In this work, we focus on the relaxation approach

[46–48], which can be used to enforce conservation/dissipation of functionals while preserving all linear invariants. The basic idea of relaxation methods goes back to [49] and [50, pp. 265–266].

Thus, there are several studies and methods devoted to either positivity preservation or the preservation of functionals such as entropy, but to the best of our knowledge, there is no high-order method that can guarantee both properties simultaneously. The main contribution of this work is to design a modified relaxation algorithm capable of simultaneously preserving positivity and conservation/dissipation of functionals. To that end, we first equip unconditionally positivity-preserving NSARK methods with suitable estimates for dissipative entropies by applying the relaxation framework from [48]. While relaxation can be rendered positivity-preserving for dissipative problems with minor adjustments (see Remark 3), entropy-conservative problems require more sophisticated treatment. Leveraging dense output formulae for MPRK methods [51], we propose a modified relaxation step that ensures unconditional positivity. Furthermore, we introduce a bootstrapping technique to achieve arbitrarily high-order accuracy in time for MPRK schemes.

The remainder of the paper is structured as follows. We recall the relaxation technique from [48] in Section 2.1. In Section 2.2 we give a brief introduction to NSARK methods. After that, we explain in Section 3.1 how to apply the relaxation algorithm for NSARK schemes and entropy dissipative problems. The main result is given for the entropy-conservative case, see Section 3.2, where we equip different families of MP schemes with a positivity-preserving relaxation algorithm and present a bootstrapping technique to obtain arbitrary high order (in time) for MPRK schemes. Finally, we present several examples of ordinary and partial differential equations and validate our findings for second- and third-order MP schemes.

2 Preliminaries

In this section, we briefly review relaxation methods to preserve functionals η and nonstandard additive Runge–Kutta (NSARK) methods, which includes Patankar-type methods as a special case.

2.1 Classical Relaxation

One way to guarantee dissipation (2) or conservation (3) of functionals η is the relaxation procedure explained in [48]. We are given a numerical one-step method of order $p \geq 2$ generating approximations $\mathbf{u}^n$ to $\mathbf{u}(t_n)$ with a time step size of Δt. We then have to repeat the following steps, starting with $n = 0$.

1. Define the quantities $(t_{\text{old}}, \mathbf{u}_{\text{old}}, \eta_{\text{old}}) \coloneqq (t_n, \mathbf{u}^n, \eta(\mathbf{u}^n))$ as well as $(t_{\text{new}}, \mathbf{u}_{\text{new}}) \coloneqq (t_{n+1}, \mathbf{u}^{n+1})$.
2. - For dissipative problems (1) compute a suitable estimate

$$\eta_{\text{new}} = \eta(\mathbf{u}_{\text{new}}) + \mathcal{O}(\Delta t^{p+1}), \quad \Delta t \to 0.$$

- For conservative problems we can simply set $\eta_{\text{new}} := \eta_{\text{old}}$, since we arrive at $\eta_{\text{new}} = \eta(\mathbf{u}_{\text{old}}) = \eta(\mathbf{u}(t_{n+1})) = \eta(\mathbf{u}_{\text{new}}) + \mathcal{O}(\Delta t^{p+1})$ by means of an induction over n.

3. Solve the system

$$\begin{pmatrix} t_\gamma^n \\ \mathbf{u}_\gamma^n \\ \eta(\mathbf{u}_\gamma^n) \end{pmatrix} = \begin{pmatrix} t_{\text{old}} \\ \mathbf{u}_{\text{old}} \\ \eta_{\text{old}} \end{pmatrix} + \gamma \begin{pmatrix} t_{\text{new}} - t_{\text{old}} \\ \mathbf{u}_{\text{new}} - \mathbf{u}_{\text{old}} \\ \eta_{\text{new}} - \eta_{\text{old}} \end{pmatrix} \tag{4}$$

   by inserting $\mathbf{u}_\gamma^n$ into the last equation and solving for $\gamma \approx 1$, and then computing t_γ^n and $\mathbf{u}_\gamma^n$ according to the remaining equations.
4. Proceed with the numerical scheme using t_γ^n and $\mathbf{u}_\gamma^n$ instead of t_{n+1} and $\mathbf{u}^{n+1}$.

For dissipative problems, the "suitable estimate η_{new}" must guarantee the discrete dissipativity (2) for the approximations from the relaxation procedure. We will introduce such a suitable estimate for NSARK methods that are based on ARK methods with a non-negative extended Butcher tableau in Section 3.1. For now, let us proceed by revisiting the main results from [48], assuming we have such an η_{new} at hand.

Theorem 1 ([48, Theorem 2.13, Theorem 2.14]) *Consider the relaxation procedure (4) with a numerical method of order p and $\Delta t > 0$ sufficiently small. If*

$$\eta \text{ is convex and } \eta''(\mathbf{u}_{old})(\mathbf{f}(\mathbf{u}_{old}), \mathbf{f}(\mathbf{u}_{old})) \neq 0 \quad \text{or} \tag{5a}$$

$$\eta'(\mathbf{u}_{new}) \frac{\mathbf{u}_{new} - \mathbf{u}_{old}}{\|\mathbf{u}_{new} - \mathbf{u}_{old}\|} = c(\mathbf{u}_{old})\Delta t + \mathcal{O}(\Delta t^2) \text{ with } c \neq 0, \tag{5b}$$

then there exists a unique $\gamma = 1 + \mathcal{O}(\Delta t^{p-1})$ that satisfies (4). Additionally, the relaxation method is of order p, that is $\mathbf{u}_\gamma^n = \mathbf{u}(t_\gamma^n) + \mathcal{O}(\Delta t^{p+1})$. In particular, there exist $\gamma_1, \gamma_2 > 0$ such that

$$r(\gamma) := \eta(\mathbf{u}_{old} + \gamma(\mathbf{u}_{new} - \mathbf{u}_{old})) - \big(\eta_{old} + \gamma(\eta_{new} - \eta_{old})\big) \tag{6}$$

satisfies $r(\gamma_1)r(\gamma_2) < 0$ for $\Delta t > 0$ small enough.

This theorem is the theoretical basis for the existence and uniqueness of the solution of the relaxation procedure (4). Unfortunately, the theorem does not give bounds on Δt for the existence of the solution, so that computations may be rejected due to Δt being too large.

Remark 1 (Issue with positivity) The main issue of positivity-preservation with the above relaxation algorithm is that the update

$$\mathbf{u}_\gamma^n = \mathbf{u}^n + \gamma(\mathbf{u}^{n+1} - \mathbf{u}^n) = \gamma\mathbf{u}^{n+1} + (1-\gamma)\mathbf{u}^n$$

is not necessarily positivity-preserving for $\gamma > 1$, even if the baseline method is positive.

Nevertheless, to overcome this issue, we propose to use *unconditionally positive*[1] time integrators. In the upcoming sections, we introduce the methods of interest for

[1] That is, $\mathbf{u}^n > \mathbf{0}$ component-wise implies $\mathbf{u}^{n+1} > \mathbf{0}$ for all $\Delta t > 0$.

this work, all of which may be recast as so-called non-standard additive Runge–Kutta schemes.

2.2 Non-standard Additive Runge–Kutta Methods

Non-standard additive Runge–Kutta methods (NSARK) methods are applied to an IVP (1), where the right-hand side is split into a sum, that is

$$\mathbf{u}'(t) = \mathbf{f}(\mathbf{u}(t)) = \sum_{\nu=1}^{N} \mathbf{f}^{[\nu]}(\mathbf{u}(t)), \quad \mathbf{u}(t_0) = \mathbf{u}^0 \in \mathbb{R}^d. \tag{7}$$

Already for traditional additive Runge–Kutta (ARK) methods, including Implicit-Explicit (IMEX) Runge–Kutta (RK) methods [52, 53], the main idea is to apply very different RK schemes determined by $\mathbf{A}^{[\nu]} = (a_{ij}^{[\nu]})_{i,j=1,\dots,s}$, $\mathbf{b}^{[\nu]} = (b_1^{[\nu]}, \dots, b_s^{[\nu]})$, $\mathbf{c}^{[\nu]} = (c_1^{[\nu]}, \dots, c_s^{[\nu]})^T$ to the different addends $\mathbf{f}^{[\nu]}$. For internal consistency, we require that the different RK schemes actually do not differ in the abscissa, i.e.

$$c_i = c_i^{[\nu]} = \sum_{j=1}^{s} a_{ij}^{[\nu]} \tag{8}$$

for $i = 1, \dots, s$ and $\nu = 1, \dots, N$, see [54]. However, for autonomous IVPs (7), this has no effect on the resulting ARK method, which in this case reads

$$\begin{aligned} \mathbf{u}^{(i)} &= \mathbf{u}^n + \Delta t \sum_{j=1}^{s} \sum_{\nu=1}^{N} a_{ij}^{[\nu]} \mathbf{f}^{[\nu]}(\mathbf{u}^{(j)}), \quad i = 1, \dots, s, \\ \mathbf{u}^{n+1} &= \mathbf{u}^n + \Delta t \sum_{j=1}^{s} \sum_{\nu=1}^{N} b_j^{[\nu]} \mathbf{f}^{[\nu]}(\mathbf{u}^{(j)}), \end{aligned} \tag{9}$$

and the corresponding extended Butcher tableau is given by

$$\begin{array}{c|c|c|c|c} \mathbf{c} & \mathbf{A}^{[1]} & \mathbf{A}^{[2]} & \cdots & \mathbf{A}^{[N]} \\ \hline & \mathbf{b}^{[1]} & \mathbf{b}^{[2]} & \cdots & \mathbf{b}^{[N]} \end{array}$$

with $\mathbf{c} = (c_1, \dots, c_s)^T$.

NSARK methods now differ from ARK schemes (9) in that their extended Butcher tableau is allowed to also depend on the step size and the solution. In particular,

NSARK methods applied to (7) are of the form

$$\begin{aligned}\mathbf{u}^{(i)} &= \mathbf{u}^n + \Delta t \sum_{j=1}^{s} \sum_{\nu=1}^{N} a_{ij}^{[\nu]}(\mathbf{U}^n, t_n, \Delta t)\mathbf{f}^{[\nu]}(\mathbf{u}^{(j)}), \quad i = 1, \dots, s,\\ \mathbf{u}^{n+1} &= \mathbf{u}^n + \Delta t \sum_{j=1}^{s} \sum_{\nu=1}^{N} b_{j}^{[\nu]}(\mathbf{U}^n, t_n, \Delta t)\mathbf{f}^{[\nu]}(\mathbf{u}^{(j)}),\end{aligned} \tag{10}$$

where $\mathbf{U}^n = (\mathbf{u}^n \,|\, \mathbf{u}^{(1)} \,|\, \dots \,|\, \mathbf{u}^{(s)} \,|\, \mathbf{u}^{n+1}) \in \mathbb{R}^{d \times s+2}$.

In the case of gBBKS [17], Geometric Conservative (GeCo) [15], both of which may be interpreted as NSRK schemes, as well as modified Patankar–Runge–Kutta (MPRK) [16, 18] methods, the same RK scheme is used for the treatment of the different addends in (7) and only the solution-dependent terms vary. For MP strong-stability-preserving RK (MPSSPRK) schemes, the situation is different, see Section 2.2.3. In this work we focus on modified Patankar (MP) schemes in the entropy-conservative case and leave gBBKS and GeCo methods for future works.

2.2.1 Production-Destruction-Rest Systems

The application of modified Patankar (MP) schemes is restricted to production-destruction-rest (PDRS) systems

$$u_k'(t) = r_k^P(\mathbf{u}(t)) - r_k^D(\mathbf{u}(t)) + \sum_{\nu=1}^{d} (p_{k\nu}(\mathbf{u}(t)) - d_{k\nu}(\mathbf{u}(t))), \quad k = 1, \dots, d$$

with $p_{k\nu} = d_{\nu k}$ and $r_k^P, r_k^D, p_{k\nu}, d_{k\nu} \geq 0$ on $\mathbb{R}^d_{>0}$. We note that this is only a formal restriction since every autonomous system with real-valued right-hand sides can be rewritten as such a PDRS [55]. Now, one can recover the function $\mathbf{f}^{[\nu]}$ in (7) and specify the solution-dependent Butcher coefficients. Indeed, according to [23, Remark 2.25], a PDRS can be written in terms of (7) by using the convention $p_{kk} = d_{kk} = 0$ and choosing $N = d + 1$ as well as

$$\begin{aligned}\mathbf{f}^{[N]}(\mathbf{u}(t)) &= (r_1^P(\mathbf{u}(t)), \dots, r_d^P(\mathbf{u}(t)))^T\\ f_k^{[\nu]}(\mathbf{u}(t)) &= \begin{cases} p_{k\nu}(\mathbf{u}(t)), & k \neq \nu, \\ -\left(r_k^D(\mathbf{u}(t)) + \sum_{\mu=1}^{d} d_{k\mu}(\mathbf{u}(t))\right), & k = \nu \end{cases}\end{aligned} \tag{11}$$

for $k, \nu = 1, \dots, d$.

2.2.2 Modified Patankar–Runge–Kutta Schemes

With (11), every MPRK scheme [14, 16, 18] that is based on a single explicit RK method with a non-negative Butcher array can be expressed in terms of an NSARK scheme using

$$
\begin{aligned}
a_{ij}^{[\nu]}(\mathbf{U}^n, t_n, \Delta t) &= a_{ij}\gamma_\nu^{[i]}(\mathbf{U}^n, t_n, \Delta t),\\
b_j^{[\nu]}(\mathbf{U}^n, t_n, \Delta t) &= b_j\delta_\nu(\mathbf{U}^n, t_n, \Delta t),
\end{aligned}
\tag{12}
$$

where

$$
\begin{aligned}
\gamma_\nu^{[i]}(\mathbf{U}^n, t_n, \Delta t) &= \begin{cases} \frac{u_\nu^{(i)}}{\pi_\nu^{(i)}(\mathbf{u}^n, \mathbf{u}^{(1)}, \dots, \mathbf{u}^{(i-1)})}, & \nu < N \\ 1, & \nu = N \end{cases} \quad \text{and}\\
\delta_\nu(\mathbf{U}^n, t_n, \Delta t) &= \begin{cases} \frac{u_\nu^{n+1}}{\sigma_\nu(\mathbf{u}^n, \mathbf{u}^{(1)}, \dots, \mathbf{u}^{(s)})}, & \nu < N \\ 1, & \nu = N \end{cases}
\end{aligned}
\tag{13}
$$

are the so-called *non-standard weights* (NSWs). Here, $\pi_\nu^{(i)}$ and σ_ν denote the so-called *Patankar-weight denominators* (PWDs) and can be chosen for the particular MPRK method to ensure stability and accuracy, see [16, 23] for more insights. If the Butcher array contains negative entries, more care is needed when defining the MPRK method, see e.g. [20].

Example 1 (Second-order Family) The second-order family of MPRK schemes, denoted by MPRK22(α), is given by

$$
\begin{aligned}
u_k^{(1)} &= u_k^n,\\
u_k^{(2)} &= u_k^n + \alpha\Delta t \left(r_k^P(\mathbf{u}^{(1)}) + \sum_{\nu=1}^d p_{k\nu}(\mathbf{u}^{(1)}) \frac{u_\nu^{(2)}}{u_\nu^n} - \left(r_k^D(\mathbf{u}^{(1)}) + \sum_{\nu=1}^d d_{k\nu}(\mathbf{u}^{(1)}) \right) \frac{u_k^{(2)}}{u_k^n} \right),\\
u_k^{n+1} &= u_k^n + \Delta t \sum_{j=1}^2 b_j \left(r_k^P(\mathbf{u}^{(j)}) + \sum_{\nu=1}^d p_{k\nu}(\mathbf{u}^{(j)}) \frac{u_\nu^{n+1}}{(u_\nu^{(2)})^{\frac{1}{\alpha}} (u_\nu^n)^{1-\frac{1}{\alpha}}} \right.\\
&\qquad \left. - \left(r_k^D(\mathbf{u}^{(j)}) + \sum_{\nu=1}^d d_{k\nu}(\mathbf{u}^{(j)}) \right) \frac{u_k^{n+1}}{(u_k^{(2)})^{\frac{1}{\alpha}} (u_k^n)^{1-\frac{1}{\alpha}}} \right)
\end{aligned}
$$

with $k = 1, \dots, d$, $\alpha \geq \frac{1}{2}$ and $b_2 = \frac{1}{2\alpha}$ as well as $b_1 = 1 - b_2$. In terms of the previous notation, we are using the Butcher array

$$
\begin{array}{c|cc}
0 & & \\
\alpha & \alpha & \\
\hline
 & 1 - \frac{1}{2\alpha} & \frac{1}{2\alpha}
\end{array}
$$

and the PWDs

$$
\pi_k^{(2)} = u_k^n, \quad \sigma_k = (u_k^{(2)})^{\frac{1}{\alpha}} (u_k^n)^{1-\frac{1}{\alpha}}.
$$

In this work, we will focus on $\alpha = 1$ as suggested by [56].

Example 2 (Third-order Family) There are two third-order families of MPRK schemes, see [18]. One of them is based on the Butcher array

$$\begin{array}{c|ccc} 0 & & & \\ \alpha & \alpha & & \\ \beta & \frac{3\alpha\beta(1-\alpha)-\beta^2}{\alpha(2-3\alpha)} & \frac{\beta(\beta-\alpha)}{\alpha(2-3\alpha)} & \\ \hline & 1+\frac{2-3(\alpha+\beta)}{6\alpha\beta} & \frac{3\beta-2}{6\alpha(\beta-\alpha)} & \frac{2-3\alpha}{6\beta(\beta-\alpha)} \end{array} \tag{14}$$

see [18] for more details on the domain of α and β. The PWDs are given by

$$\begin{aligned} \pi_\nu^{(2)} &= u_\nu^n, \qquad \pi_\nu^{(3)} = (u_\nu^{(2)})^{\frac{1}{p}} (u_\nu^n)^{1-\frac{1}{p}}, \\ \sigma_k &= u_k^n + \Delta t \sum_{j=1}^{2} \beta_j \left(r_k^P(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} p_{k\nu}(\mathbf{u}^{(j)}) \frac{\sigma_\nu}{(u_\nu^{(2)})^{\frac{1}{a_{21}}} (u_\nu^n)^{1-\frac{1}{a_{21}}}} \right. \\ &\quad \left. - \left(r_k^D(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} d_{k\nu}(\mathbf{u}^{(j)}) \right) \frac{\sigma_k}{(u_k^{(2)})^{\frac{1}{a_{21}}} (u_k^n)^{1-\frac{1}{a_{21}}}} \right) \end{aligned} \tag{15}$$

for $\nu, k = 1, \dots, d$, where $\beta_1 = 1 - \beta_2$, $\beta_2 = \frac{1}{2a_{21}}$, and $p = 3a_{21}\,(a_{31} + a_{32})\,b_3$. Note that $\boldsymbol{\sigma}$ requires the solution of another linear system, which is why this family is denoted by MPRK43I(α, β). In this work we focus on $\alpha = 0.5$ and $\beta = 0.75$.

In any case we point out that the schemes are implicit due to the numerators in (13). Indeed, they are linearly implicit as the PWDs $\pi_\nu^{(i)}$ and σ_ν are required to be independent of the numerator [14, 23]. Consequently, an MPRK scheme can be written in matrix-vector notation as follows.

$$\begin{aligned} \mathbf{M}^{(i)} \mathbf{u}^{(i)} &= \mathbf{u}^n + \Delta t \sum_{j=1}^{i-1} a_{ij} \mathbf{r}^P(\mathbf{u}^{(j)}), \quad i = 1, \dots, s, \\ \mathbf{M} \mathbf{u}^{n+1} &= \mathbf{u}^n + \Delta t \sum_{j=1}^{s} b_j \mathbf{r}^P(\mathbf{u}^{(j)}), \end{aligned} \tag{16}$$

where $\mathbf{r}^P = (r_1^P, \dots, r_d^P)^T$ and $\mathbf{M}^{(i)} = (m_{k\nu}^{(i)})_{1 \le k, \nu \le d}$ with

$$\begin{aligned} m_{kk}^{(i)} &= 1 + \Delta t \sum_{j=1}^{i-1} a_{ij} \left(r_k^D(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} d_{k\nu}(\mathbf{u}^{(j)}) \right) \frac{1}{\pi_k^{(i)}}, \\ m_{k\nu}^{(i)} &= -\Delta t \sum_{j=1}^{i-1} a_{ij} p_{k\nu}(\mathbf{u}^{(j)}) \frac{1}{\pi_\nu^{(i)}}, \quad k \neq \nu \end{aligned} \tag{17}$$

as well as, using $\mathbf{M} = (m_{k\nu})_{1\le k,\nu\le d}$,

$$\begin{aligned} m_{kk} &= 1 + \Delta t \sum_{j=1}^{s} b_j \left(r_k^D(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} d_{k\nu}(\mathbf{u}^{(j)}) \right) \frac{1}{\sigma_k}, \\ m_{k\nu} &= -\Delta t \sum_{j=1}^{s} b_j p_{k\nu}(\mathbf{u}^{(j)}) \frac{1}{\sigma_\nu}, \quad k \neq \nu. \end{aligned} \tag{18}$$

2.2.3 MP Strong-Stability-Preserving-Runge–Kutta Schemes

Although there exist second- and third-order MPSSPRK schemes, see [4, 19], we focus for simplicity on the second-order method and the conservative PDS case. The generalization to non-conservative PDRS is straightforward but complicates the formulae. Also, the consideration of third-order MPSSPRK schemes will be left out for future work. The two-parameter family of second-order MPSSPRK schemes from [4] is given by

$$\begin{aligned} \mathbf{u}^{(1)} =& \mathbf{u}^n, \\ u_i^{(2)} =& u_i^n + \beta\Delta t \left(\sum_{j=1}^{d} p_{ij}(\mathbf{u}^n) \frac{u_j^{(2)}}{u_j^n} - \sum_{j=1}^{d} d_{ij}(\mathbf{u}^n) \frac{u_i^{(2)}}{u_i^n} \right), \\ u_i^{n+1} =& (1-\alpha)u_i^n + \alpha u_i^{(2)} + \Delta t \Bigg(\sum_{j=1}^{d} \left(\beta_{20} p_{ij}(\mathbf{u}^n) + \beta_{21} p_{ij}(\mathbf{u}^{(2)}) \right) \frac{u_j^{n+1}}{(u_j^n)^{1-s}(u_j^{(2)})^s} \\ & - \sum_{j=1}^{d} \left(\beta_{20} d_{ij}(\mathbf{u}^n) + \beta_{21} d_{ij}(\mathbf{u}^{(2)}) \right) \frac{u_i^{n+1}}{(u_i^n)^{1-s}(u_i^{(2)})^s} \Bigg), \end{aligned} \tag{19}$$

where $\beta_{20} = 1 - \frac{1}{2\beta} - \alpha\beta$, $\beta_{21} = \frac{1}{2\beta}$ and $s = \frac{1-\alpha\beta+\alpha\beta^2}{\beta(1-\alpha\beta)}$. There, the free parameters α and β are subject to

$$0 \le \alpha \le 1, \quad \beta > 0, \quad \alpha\beta + \frac{1}{2\beta} \le 1. \tag{20}$$

We refer to the above scheme as MPSSPRK2(α, β). For numerical experiments we use $\alpha = \frac{1}{2}$ and $\beta = 1$ [4].

Substituting the second stage into the update, we can collect production and destruction terms. Hence, in the notation of (11), the solution-dependent coefficients for the conservative PDS case are

$$\begin{aligned} a_{21}^{[\nu]}(\mathbf{U}^n, t_n, \Delta t) &= \beta \frac{u_\nu^{(2)}}{u_\nu^n} \\ b_1^{[\nu]}(\mathbf{U}^n, t_n, \Delta t) &= \alpha\beta \frac{u_\nu^{(2)}}{u_\nu^n} + \beta_{20} \frac{u_\nu^{n+1}}{\sigma_\nu}, \quad b_2^{[\nu]}(\mathbf{U}^n, t_n, \Delta t) = \beta_{21} \frac{u_\nu^{n+1}}{\sigma_\nu}, \end{aligned} \tag{21}$$

where $\sigma_\nu = (u_\nu^n)^{1-s}(u_\nu^{(2)})^s$.

3 Positivity-Preserving Relaxation Technique

In what follows, we adapt the classical relaxation algorithm from Section 2.1 such that it becomes positivity-preserving.

3.1 Entropy Dissipative Case

First of all, we present a suitable estimate η_{new} for a general NSARK scheme. In order to minimize the computational effort, we propose to re-use the computed stage values of the NSARK scheme satisfying (12), that is

$$\begin{aligned}
\mathbf{u}^{(i)} &= \mathbf{u}^n + \Delta t \sum_{j=1}^{s} \sum_{\nu=1}^{N} a_{ij} \gamma_\nu^{[i]}(\mathbf{U}^n, t_n, \Delta t) \mathbf{f}^{[\nu]}(\mathbf{u}^{(j)}), \quad i = 1, \dots, s, \\
\mathbf{u}^{n+1} &= \mathbf{u}^n + \Delta t \sum_{j=1}^{s} \sum_{\nu=1}^{N} b_j \delta_\nu(\mathbf{U}^n, t_n, \Delta t) \mathbf{f}^{[\nu]}(\mathbf{u}^{(j)}), \\
\eta_{\text{new}} &= \eta(\mathbf{u}^n) + \Delta t \sum_{j=1}^{s} b_j (\eta' \mathbf{f})(\mathbf{u}^{(j)}),
\end{aligned} \tag{22}$$

which can be interpreted as computing the numerical approximation of the augmented system

$$\frac{\mathrm{d}}{\mathrm{d}t} \begin{pmatrix} \mathbf{u}(t) \\ \eta(\mathbf{u}(t)) \end{pmatrix} = \sum_{\nu=1}^{N} \underbrace{\begin{pmatrix} \mathbf{f}^{[\nu]}(\mathbf{u}(t)) \\ 0 \end{pmatrix}}_{=:\hat{\mathbf{f}}^{[\nu]}(\mathbf{u}(t))} + \underbrace{\begin{pmatrix} \mathbf{0} \\ (\eta' \mathbf{f})(\mathbf{u}(t)) \end{pmatrix}}_{=:\hat{\mathbf{f}}^{[N+1]}(\mathbf{u}(t))}$$

using an NSARK method with the extended Butcher tableau

$$\begin{array}{c|c|c|c|c} \mathbf{c} & \boldsymbol{\Gamma}_1(\mathbf{U}^n, t_n, \Delta t)\mathbf{A} & \boldsymbol{\Gamma}_2(\mathbf{U}^n, t_n, \Delta t)\mathbf{A} & \cdots & \boldsymbol{\Gamma}_N(\mathbf{U}^n, t_n, \Delta t)\mathbf{A} \\ \hline & \delta_1(\mathbf{U}^n, t_n, \Delta t)\mathbf{b} & \delta_2(\mathbf{U}^n, t_n, \Delta t)\mathbf{b} & \cdots & \delta_N(\mathbf{U}^n, t_n, \Delta t)\mathbf{b} \end{array} \quad \begin{array}{c|c} \mathbf{A} \\ \hline \mathbf{b} \end{array}, \tag{23}$$

where $\boldsymbol{\Gamma}_\nu \coloneqq \operatorname{diag}(\gamma_\nu^{[1]}, \dots, \gamma_\nu^{[s]})$. Assuming that the two corresponding base methods described by the Butcher tableaux

$$\begin{array}{c|c|c|c|c} \mathbf{c} & \boldsymbol{\Gamma}_1(\mathbf{U}^n, t_n, \Delta t)\mathbf{A} & \boldsymbol{\Gamma}_2(\mathbf{U}^n, t_n, \Delta t)\mathbf{A} & \cdots & \boldsymbol{\Gamma}_N(\mathbf{U}^n, t_n, \Delta t)\mathbf{A} \\ \hline & \delta_1(\mathbf{U}^n, t_n, \Delta t)\mathbf{b} & \delta_2(\mathbf{U}^n, t_n, \Delta t)\mathbf{b} & \cdots & \delta_N(\mathbf{U}^n, t_n, \Delta t)\mathbf{b} \end{array} \quad \text{and} \quad \begin{array}{c|c} \mathbf{c} & \mathbf{A} \\ \hline & \mathbf{b} \end{array}$$

both are of p-th order for some $p \in \{2, 3, 4\}$, it can be seen from [22, Theorem 18,Lemma 25,Lemma 26] that the overall scheme (22) is of order p, since the respective order conditions are decoupled with respect to the columns of the tableau (23).

Remark 2 The NSWs from MPSSPRK schemes, see (21), also satisfy (12) after multiplying and dividing by the respective Butcher coefficient.

As a result, we indeed obtain that $\eta_{\text{new}} = \eta(\mathbf{u}_{\text{new}}) + \mathcal{O}(\Delta t^{p+1})$, and additionally,

$$\eta_{\text{new}} \leq \eta(\mathbf{u}^n) \tag{24}$$

whenever $b_j \geq 0$ for $j = 1, \dots, s$. If $b_j < 0$ for some $j = 1, \dots, s$, one can still use Gauß quadrature, as suggested in [48, Page 866] together with the unconditionally positive dense output formulae derived in [51] to obtain the approximations needed for the quadrature formula.

In view of Remark 1, the relaxation technique is in danger of not being positivity-preserving for $\gamma > 1$. The following corollary gives a work around for dissipative problems as we will discuss in the upcoming Remark 3.

Corollary 1 ([48, Pages 882-883]) If η is convex with $\eta''(\mathbf{u}_{\text{old}})(\mathbf{f}(\mathbf{u}_{\text{old}}), \mathbf{f}(\mathbf{u}_{\text{old}})) \neq 0$ then r from (6) is convex and satisfies $r(0) = 0$, $r'(0) < 0$ and $r'(\gamma) > 0$ for all $\gamma \geq 1$ and $\Delta t > 0$ small enough.

Remark 3 (Positivity-preserving relaxation for convex η) Suppose that $\gamma^* > 1$ is the solution to (4), i.e. $r(\gamma^*) = 0$, so that the positivity of the relaxation update $\mathbf{u}_\gamma^n$ is not guaranteed any longer. Because of Corollary 1 we know that $r(\gamma) \leq r(\gamma^*) = 0$ for all $\gamma \in [1, \gamma^*]$. In particular $r(1) \leq 0$ follows, i. e., for $\gamma = 1$ we obtain from (6) the relation

$$\eta(\mathbf{u}_\gamma^n) = \eta(\mathbf{u}_{\text{new}}) \leq \eta_{\text{new}}.$$

This means, that due to (24) only more dissipation will be introduced by using

$$\gamma = \min\{\gamma^*, 1\} \in (0, 1],$$

where γ^* is the solution to (4). But with that choice, $\mathbf{u}_\gamma^n$ is again a convex combination of positive data, and hence, positivity preservation is guaranteed for dissipative problems with a convex η.

3.2 Entropy-Conservative Case

As $\mathbf{u}_\gamma^n$ in (4) is not guaranteed to be positivity-preserving for $\gamma > 1$, see Remark 1, we propose to replace the update formula by a positivity-preserving variant. To indicate this difference in our notation we will write $\mathbf{u}^{n+\gamma}$ for a positivity-preserving approximation rather than $\mathbf{u}_\gamma^n$.

3.2.1 Explicit Positivity-Preserving Procedure

If we are only interested in preserving positivity, a single nonlinear invariant but no further linear invariants, we may apply a Patankar–Runge–Kutta method to guarantee the positivity of the update and combine it with the geometric mean

$$\mathbf{u}^{n+\gamma} = (\mathbf{u}^{n+1})^\gamma (\mathbf{u}^n)^{1-\gamma}. \tag{25}$$

In logarithmic variables, this reduces to

$$\ln(\mathbf{u}^{n+\gamma}) = \ln(\mathbf{u}^n) + \gamma(\ln(\mathbf{u}^{n+1}) - \ln(\mathbf{u}^n)),$$

where we can find a solution to the relaxation problem in logarithmic variables according to the classical theory. Now, if η is convex and non-decreasing in each argument the composition $\eta \circ \exp$ is also convex [57, Section 2.3.4], and hence, also

$$\eta(\mathbf{u}^{n+\gamma}) = \eta_{\text{old}}$$

possesses a positive solution $\gamma = 1 + \mathcal{O}(\Delta t^{p-1})$.

3.2.2 Implicit Positivity-Preserving Procedure for Conservative PDS

One possible candidate for computing $\mathbf{u}^{n+\gamma}$ is to use dense output formulae recently developed in [51], which we briefly recall in the upcoming section.

Positivity-Preserving Dense Output

We first focus on Runge–Kutta methods and the MP variant, but the ideas can be carried out for MPSSPRK schemes in a straightforward manner as we will see.

The main idea is to replace $b_j \in \mathbb{R}$ by a function $\bar{b}_j \colon [0,1] \to \mathbb{R}$ such that

$$u_k^{n+\gamma} = u_k^n + \Delta t \sum_{j=1}^{s} \bar{b}_j(\gamma) \sum_{\nu=1}^{d} \left(r_k^P(\mathbf{u}^{(j)}) - r_k^D(\mathbf{u}^{(j)}) + p_{k\nu}(\mathbf{u}^{(j)}) - d_{k\nu}(\mathbf{u}^{(j)}) \right)$$

approximates $u_k(t^n + \gamma\Delta t)$. We impose

$$\bar{b}_j(0) = 0 \quad \text{and} \quad \bar{b}_j(1) = b_j$$

to recover

$$\mathbf{u}^{n+\gamma} = \begin{cases} \mathbf{u}^n, & \gamma = 0, \\ \mathbf{u}^{n+1}, & \gamma = 1. \end{cases}$$

Example 3 (Second-order dense output for MPRK22(α)) Using $\bar{b}_j(\gamma) = \gamma b_j$ and

$$u_k^{n+\gamma} = u_k^n + \Delta t \sum_{j=1}^{s} \bar{b}_j(\gamma) \left(r_k^P(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} p_{k\nu}(\mathbf{u}^{(j)}) \frac{u_\nu^{n+1}}{\sigma_\nu} - \left(r_k^D(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} d_{k\nu}(\mathbf{u}^{(j)}) \right) \frac{u_k^{n+1}}{\sigma_k} \right)$$

yields a positivity-preserving dense output. Indeed, we find $\mathbf{u}^{n+\gamma} = (1-\gamma)\mathbf{u}^n + \gamma\mathbf{u}^{n+1}$ in this case, which coincides with the relaxation update. For $\gamma > 1$ this is not necessarily positivity-preserving. For our purposes, we want to ensure positivity even for $\gamma > 1$, which can be done

using

$$u_k^{n+\gamma} = u_k^n + \Delta t \sum_{j=1}^{s} \bar{b}_j(\gamma) \left(r_k^P(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} p_{k\nu}(\mathbf{u}^{(j)}) \frac{u_\nu^{n+\gamma}}{\bar{\sigma}_\nu(\gamma)} - \left(r_k^D(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} d_{k\nu}(\mathbf{u}^{(j)}) \right) \frac{u_k^{n+\gamma}}{\bar{\sigma}_k(\gamma)} \right) \tag{26}$$

together with

$$\bar{\sigma}_k(\gamma) = \sigma_k = (u_k^{(2)})^{\frac{1}{\alpha}} (u_k^n)^{1-\frac{1}{\alpha}} \tag{27}$$

or

$$\bar{\sigma}_k(\gamma) = (u_k^{(2)})^{\frac{\gamma}{\alpha}} (u_k^n)^{1-\frac{\gamma}{\alpha}}. \tag{28}$$

Example 4 (Higher order positive dense output for MPRK) In general, one may use $\bar{b}_j(\gamma)$ from the classical dense output formula paired with the update (26). Then, the only quantity to define is $\bar{\boldsymbol{\sigma}}(\gamma)$. According to [51], it is sufficient to use a lower order dense output MPRK scheme for the computation of $\bar{\boldsymbol{\sigma}}(\gamma)$. For instance, we note that third-order MPRK schemes are equipped with

$$\bar{b}_1(\gamma) = \gamma - (1 - b_1)\gamma^2, \quad \bar{b}_j(\gamma) = \gamma^2 b_j, \quad j = 2, \dots, s$$

for the dense output. However, we will discuss a different approach in this work, and thus, omit to also recall $\bar{\boldsymbol{\sigma}}$ from [51].

Example 5 (Second-order dense output for MPSSPRK) Let us incorporate the γ-dependency in (21), which gives

$$b_1^{[\nu]}(\mathbf{U}^n, t_n, \Delta t, \gamma) = \gamma \left(\alpha\beta \frac{u_\nu^{(2)}}{u_\nu^n} + \beta_{20} \frac{u_\nu^{n+\gamma}}{\bar{\sigma}_\nu(\gamma)} \right), \quad b_2^{[\nu]}(\mathbf{U}^n, t_n, \Delta t, \gamma) = \gamma\beta_{21} \frac{u_\nu^{n+\gamma}}{\bar{\sigma}_\nu(\gamma)},$$

where we restrict to the choice $\bar{\sigma}_\nu(\gamma) = (u_\nu^n)^{1-\gamma s} (u_\nu^{(2)})^{\gamma s}$. Then

$$\mathbf{u}^{n+\gamma} = \mathbf{u}^n + \Delta t \sum_{j=1}^{s} \sum_{\nu=1}^{d} b_j(\mathbf{U}^n, t_n, \Delta t, \gamma) \mathbf{f}^{[\nu]}(\mathbf{u}^{(j)}).$$

Remark 4 (Use of dense output for relaxation) As we will show, we can use the dense output from Example 3 for the relaxation algorithm. However, proving the existence of a solution to the relaxation equation for (27) is more complex than for (28), which is due to the respective truncation errors. Moreover, as illustrated in Example 4, the bootstrapping for higher order positive dense output involves higher degree polynomials for $\bar{b}_j$, which may not be positive for $\gamma > 1$. This is crucial since the solvability for the linear systems (16) relies on positive Butcher coefficients. While we could implement a trick [20] to overcome this issue, we rather focus on a different bootstrapping approach to keep the overall algorithm simple.

Preparatory Results for MPRK22(α)

We proceed to develop a relaxation technique using (26)-(27), which is more complicated than using (28) but on the other hand motivates us to derive more general results.

To that end, we note that the scheme with (26)-(27) can be written in matrix-vector notation as

$$\begin{aligned} \mathbf{u}^{(1)} &= \mathbf{u}^n \\ \mathbf{M}^{(2)}(\mathbf{u}^n)\mathbf{u}^{(2)} &= \mathbf{u}^n + \alpha\Delta t \mathbf{r}^P(\mathbf{u}^n) \\ \mathbf{M}_\gamma(\mathbf{u}^n)\mathbf{u}^{n+\gamma} &= \mathbf{u}^n + \gamma\Delta t \sum_{j=1}^{s} b_j \mathbf{r}^P(\mathbf{u}^{(j)}), \quad \gamma > 0, \end{aligned} \tag{29}$$

where $\mathbf{M}^{(2)}$ can be obtained from (17) and

$$\mathbf{M}_\gamma = \gamma(\mathbf{M} - \mathbf{I}) + \mathbf{I} \tag{30}$$

with $\mathbf{M}$ from (18).

Finally, the relaxation step (4) for entropy-conservative problems is now updated to

$$\begin{pmatrix} t_\gamma^n \\ \mathbf{M}_\gamma(\mathbf{u}_{\text{old}})\mathbf{u}^{n+\gamma} \\ \eta(\mathbf{u}^{n+\gamma}) \end{pmatrix} = \begin{pmatrix} t_{\text{old}} \\ \mathbf{u}_{\text{old}} \\ \eta_{\text{old}} \end{pmatrix} + \gamma \begin{pmatrix} t_{\text{new}} - t_{\text{old}} \\ \Delta t \sum_{j=1}^{2} b_j \mathbf{r}^P(\mathbf{u}^{(j)}) \\ 0 \end{pmatrix}, \tag{31}$$

resulting in a coupled linear-nonlinear system for the simultaneous computation of γ and $\mathbf{u}^{n+\gamma}$. Note that if such a $\gamma > 0$ exists, the relaxation method for MPRK22(α) naturally is of the correct order for all γ as we are using an appropriate dense output formula.

Since we allow for a truncation error of $\mathcal{O}(\Delta t^3)$ for the second-order MPRK22(α) scheme, it is beneficial to prove the following

Lemma 1 If $\bar{b}_j(\gamma) = \gamma b_j$ and $\bar{\sigma}_k(\gamma) = \sigma_k$, then the MPRK22($\alpha$) scheme (26) satisfies

$$\mathbf{u}^{n+\gamma} = \mathbf{u}^{n+1} + (\gamma - 1)\Delta t \mathbf{d}_\gamma^n + \mathcal{O}(\Delta t^3), \tag{32}$$

where

$$\begin{aligned} d_{\gamma,k}^n &= \frac{u_k^{n+1} - u_k^n}{\Delta t} + \Delta t\gamma \sum_{j=1}^{s} b_j \left(\sum_{\nu=1}^{d} p_{k\nu}(\mathbf{u}^{(j)}) \frac{f_\nu(\mathbf{u}^n)}{u_\nu^n} - \left(r_k^D(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} d_{k\nu}(\mathbf{u}^{(j)}) \right) \frac{f_k(\mathbf{u}^n)}{u_k^n} \right) \\ &= f_k(\mathbf{u}^n) + \mathcal{O}(\Delta t). \end{aligned} \tag{33}$$

Proof Utilizing [58, Lemma 2, Lemma 3], we observe

$$\frac{u_\nu^{n+\gamma}}{\bar{\sigma}_\nu(\gamma)} = \frac{u_\nu^{n+\gamma}}{(u_\nu^{(2)})^{\frac{1}{\alpha}}(u_\nu^n)^{1-\frac{1}{\alpha}}} = 1+(\gamma-1)\Delta t \frac{f_\nu(\mathbf{u}^n)}{u_\nu^n} + \mathcal{O}(\Delta t^2) = \frac{u_\nu^{n+1}}{\sigma_\nu} + (\gamma-1)\Delta t \frac{f_\nu(\mathbf{u}^n)}{u_\nu^n} + \mathcal{O}(\Delta t^2) \tag{34}$$

as $\frac{u_\nu^{n+1}}{\sigma_\nu} = 1 + \mathcal{O}(\Delta t^2)$ [22]. Substituting this into (26) we receive

$$u_k^{n+\gamma} = u_k^n + \gamma\Delta t \sum_{j=1}^{s} b_j \left(r_k^P(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} p_{k\nu}(\mathbf{u}^{(j)}) \frac{u_\nu^{n+\gamma}}{\bar{\sigma}_\nu(\gamma)} - \left(r_k^D(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} d_{k\nu}(\mathbf{u}^{(j)}) \right) \frac{u_k^{n+\gamma}}{\bar{\sigma}_k(\gamma)} \right)$$

$$
\begin{aligned}
&= u_k^n + \gamma(u_k^{n+1} - u_k^n) \\
&+ \gamma(\gamma-1)\Delta t^2 \sum_{j=1}^{s} b_j \left(\sum_{\nu=1}^{d} p_{k\nu}(\mathbf{u}^{(j)}) \frac{f_\nu(\mathbf{u}^n)}{u_\nu^n} - \left(r_k^D(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} d_{k\nu}(\mathbf{u}^{(j)}) \right) \frac{f_k(\mathbf{u}^n)}{u_k^n} \right) + \mathcal{O}(\Delta t^3) \\
&= u_k^{n+1} + (\gamma-1)(u_k^{n+1} - u_k^n) \\
&+ \gamma(\gamma-1)\Delta t^2 \sum_{j=1}^{s} b_j \left(\sum_{\nu=1}^{d} p_{k\nu}(\mathbf{u}^{(j)}) \frac{f_\nu(\mathbf{u}^n)}{u_\nu^n} - \left(r_k^D(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} d_{k\nu}(\mathbf{u}^{(j)}) \right) \frac{f_k(\mathbf{u}^n)}{u_k^n} \right) + \mathcal{O}(\Delta t^3) \\
&= u_k^{n+1} + (\gamma-1)\Delta t d_{\gamma,k}^n + \mathcal{O}(\Delta t^3).
\end{aligned}
$$

□

Remark 5 (Influence of $\bar{\boldsymbol{\sigma}}$ and application to MPSSPRK) In the situation of Lemma 1, if we use (28) rather than (27), then instead of (34) we obtain

$$
\frac{u_\nu^{n+\gamma}}{\bar{\sigma}_\nu(\gamma)} = 1 + \mathcal{O}(\Delta t^2) = \frac{u_\nu^{n+1}}{\sigma_\nu} + \mathcal{O}(\Delta t^2)
$$

using the same technique, and finally

$$
\mathbf{u}^{n+\gamma} = \mathbf{u}^{n+1} + (\gamma-1)\Delta t \underbrace{\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t}}_{=:\mathbf{d}^n} + \mathcal{O}(\Delta t^3). \tag{35}
$$

Here $\mathbf{d}_\gamma^n = \mathbf{d}^n$ is independent of γ. We note that the PWDs for MPSSPRK2 are similar to the MPRK case, see Section 2.2.3. Indeed, one can show that (35) also holds for the second-order MPSSPRK family.

While the scheme using (26)-(27) is the motivation for the general formulation of our main result in the upcoming section, the scheme (26),(28) will be the basis for the bootstrapping algorithm to obtain higher order, see Section 3.2.2.

Main Result for Entropy Conservation and Positivity Preservation

We assume that the method can be written in the form

$$
\mathbf{u}^{n+\gamma} = \mathbf{u}^{n+1} + (\gamma-1)\Delta t \mathbf{d}_\gamma^n(\Delta t) + \mathcal{O}(\Delta t^{p+1})
$$

with $p \geq 2$ being the order of the baseline method and some suitable $\mathbf{d}_\gamma^n(\Delta t)$ depending on the method. Now, since $\mathbf{d}$ generally also depends on γ, we need the preparatory

Lemma 2 Let $h\colon U \times V \to \mathbb{R}$,

$$
h(\lambda, \gamma) \coloneqq \lambda - (\gamma-1)\Delta t \|\mathbf{d}_\gamma^n(\Delta t)\|_2 \tag{36}
$$

where $U \times V$ is an open neighborhood of $(0, 1)$.

If $\mathbf{d}_\gamma^n(\Delta t)$ is a $\mathcal{C}^1$ map on V w.r.t. γ with $\|\mathbf{d}_1^n(\Delta t)\|_2 \neq 0$ for Δt small enough, then there exist a neighborhood $\widetilde{U}$ of 0 and a continuous function $\widetilde{\gamma}\colon \widetilde{U} \to \mathbb{R}$ such that $h(\lambda, \widetilde{\gamma}(\lambda)) = 0$ for all $\lambda \in \widetilde{U}$ and $\Delta t > 0$ small enough.

Proof We have $h(0,1) = 0$ and

$$\partial_\gamma h(\lambda,\gamma) = -\Delta t\|\mathbf{d}_\gamma^n(\Delta t)\|_2 - (\gamma-1)\Delta t\frac{(\partial_\gamma \mathbf{d}_\gamma^n(\Delta t))^T\mathbf{d}_\gamma^n(\Delta t)}{\|\mathbf{d}_\gamma^n(\Delta t)\|_2}.$$

Thus, $\partial_\gamma h(0,1) = -\Delta t\|\mathbf{d}_1^n(\Delta t)\|_2 \neq 0$ for Δt small enough. The assertion then follows from the implicit function theorem. □

With that, we are positioned to prove the main theorem for the relaxation technique.

Theorem 2 *In the situation of Lemma 2, let*

$$\mathbf{u}^{n+\gamma} = \mathbf{u}^{n+1} + (\gamma-1)\Delta t\mathbf{d}_\gamma^n(\Delta t) + \mathcal{O}(\Delta t^{p+1})$$

with $p \geq 2$ being the order of the baseline method and suppose $\eta \in \mathcal{C}^2$ with

$$\eta'(\mathbf{u}^{n+1})\cdot\frac{\mathbf{d}^n_{\widetilde{\gamma}(\Delta t\mu)}(\Delta t)}{\|\mathbf{d}^n_{\widetilde{\gamma}(\Delta t\mu)}(\Delta t)\|_2} = c(\mathbf{u}^n,\mu)\Delta t + \mathcal{O}(\Delta t^2)$$

for $|\mu|$ small enough and

$$\lim_{\mu\to 0} c(\mathbf{u}^n,\mu) =: \widetilde{c}(\mathbf{u}^n) \neq 0.$$

Then the equation

$$\eta(\mathbf{u}^{n+\gamma}) - \eta_{old} = 0$$

possesses a positive solution γ. Furthermore, if $\|\mathbf{d}_\gamma^n(\Delta t)\|_2 = \mathcal{O}(\Delta t^q)$, then there exists a unique positive solution satisfying $\gamma = 1 + \mathcal{O}(\Delta t^{p-1-q})$.

Proof We set

$$\mathbf{w}_\gamma^n(\Delta t) := \frac{\mathbf{d}_\gamma^n(\Delta t)}{\|\mathbf{d}_\gamma^n(\Delta t)\|_2}$$

and follow the proof of [59, Theorem 2] by analyzing the function

$$z(\Delta t,\mu) := \Delta t^{-2}\left(\eta\left(\mathbf{u}^{n+\widetilde{\gamma}(\Delta t\mu)}\right) - \eta_{\text{old}}\right), \quad \Delta t \neq 0. \tag{37}$$

The idea is to deduce that

$$z(\Delta t,\mu) = \mu c(\mathbf{u}^n,\mu) + \frac{\mu^2}{2}(\mathbf{w}^n_{\widetilde{\gamma}(\Delta t\mu)}(\Delta t))^T H_\eta(\mathbf{u}^{n+1})\mathbf{w}^n_{\widetilde{\gamma}(\Delta t\mu)}(\Delta t) + \mathcal{O}(\Delta t), \tag{38}$$

where H_η denotes the Hessian. Then, since $\widetilde{\gamma}(0) = 1$, we have

$$z(0,\mu) := \lim_{\Delta t\to 0} z(\Delta t,\mu) = \mu\widetilde{c}(\mathbf{u}^n) + \frac{\mu^2}{2}(\mathbf{w}_1^n(0))^T H_\eta(\mathbf{u}^n)\mathbf{w}_1^n(0) \quad \text{and} \quad \partial_\mu z(0,0) = \widetilde{c}(\mathbf{u}^n) \neq 0.$$

According to the proof of [59, Theorem 2], there exist $\Delta t^* > 0$ and a unique function $\mu : [0,\Delta t^*] \to \mathbb{R}$ such that $z(\Delta t,\mu(\Delta t)) = 0$ for all $0 \leq \Delta t \leq \Delta t^*$. Indeed, because of

$$z(\Delta t,0) = \Delta t^{-2}\left(\eta(\mathbf{u}^{n+1}) - \eta_{\text{old}}\right)$$

it can be deduced along the same lines that $\mu = \mathcal{O}(\Delta t^{p-1})$.

To prove (38) we first note that

$$\mathbf{u}^{n+\gamma} = \mathbf{u}^{n+1} + \lambda\mathbf{w}_\gamma^n(\Delta t) + \mathcal{O}(\Delta t^{p+1}), \tag{39}$$

where

$$\lambda := (\gamma - 1)\Delta t \|\mathbf{d}_\gamma^n(\Delta t)\|_2. \tag{40}$$

Next, as $h(\lambda, \gamma) = 0$ we can use Lemma 2 to solve (40) for γ and plug it into (39) resulting in a function of λ only, i.e.

$$\mathbf{u}^{n+\widetilde{\gamma}(\lambda)} = \mathbf{u}^{n+1} + \lambda \mathbf{w}^n_{\widetilde{\gamma}(\lambda)}(\Delta t) + \mathcal{O}(\Delta t^{p+1}).$$

In the following, we denote by $\mathbf{u}(t)$ the exact local solution at t satisfying $\mathbf{u}(t_n) = \mathbf{u}^n$ and recall that we are considering an entropy-conservative problem. Hence, with $\lambda =: \Delta t\mu$ and the assumptions on η' we receive

$$\begin{aligned}
z(\Delta t, \mu) &= \Delta t^{-2}\left(\eta(\mathbf{u}^{n+\widetilde{\gamma}(\Delta t\mu)}) - \eta(\mathbf{u}(t_n + \Delta t))\right) \\
&= \Delta t^{-2}\left(\eta(\mathbf{u}^{n+1} + \Delta t\mu \mathbf{w}^n_{\widetilde{\gamma}(\Delta t\mu)}(\Delta t)) - \eta(\mathbf{u}(t_n + \Delta t))\right) + \mathcal{O}(\Delta t^{p-1}) \\
&= \frac{\eta(\mathbf{u}^{n+1}) - \eta(\mathbf{u}(t_n + \Delta t)) + \Delta t\mu\eta'(\mathbf{u}^{n+1})\mathbf{w}^n_{\widetilde{\gamma}(\Delta t\mu)}(\Delta t)}{\Delta t^2} \\
&\quad + \frac{\mu^2}{2}(\mathbf{w}^n_{\widetilde{\gamma}(\Delta t\mu)}(\Delta t))^T H_\eta(\mathbf{u}^{n+1})\mathbf{w}^n_{\widetilde{\gamma}(\Delta t\mu)}(\Delta t) + \mathcal{O}(\Delta t) \\
&= \mu c(\mathbf{u}^n, \mu) + \frac{\mu^2}{2}(\mathbf{w}^n_{\widetilde{\gamma}(\Delta t\mu)}(\Delta t))^T H_\eta(\mathbf{u}^{n+1})\mathbf{w}^n_{\widetilde{\gamma}(\Delta t\mu)}(\Delta t) + \mathcal{O}(\Delta t).
\end{aligned}$$

Finally, since $\mathcal{O}(\Delta t^{p-1}) = \mu = \frac{\lambda}{\Delta t} = (\gamma - 1)\|\mathbf{d}^n_\gamma(\Delta t)\|_2$ we deduce that $\gamma = 1 + \mathcal{O}(\Delta t^{p-1-q})$. □

Now if we are given such a solution $\gamma = 1 + \mathcal{O}(\Delta t^{p-1})$ we can deduce that the relaxation update is of order p as the following lemma shows.

Lemma 3 If $\mathbf{u}^{n+\gamma} = \mathbf{u}^{n+1} + (\gamma - 1)(\mathbf{u}^{n+1} - \mathbf{u}^n) + \mathcal{O}(\Delta t^{p+1})$ with a p-th order baseline method and $\gamma = 1 + \mathcal{O}(\Delta t^{p-1})$, then

$$\mathbf{u}^{n+\gamma} = \mathbf{u}(t_n + \gamma\Delta t) + \mathcal{O}(\Delta t^{p+1}).$$

Proof This is just Lemma [48, Lemma 2.7], where the relaxation method is perturbed by an additive error of $\mathcal{O}(\Delta t^{p+1})$. □

Bootstrapping Algorithm for Positivity-Preserving Relaxation

The main idea for generalizing the relaxation technique to higher order is to use the observation from Remark 5, where

$$\mathbf{u}^{n+\gamma} = \mathbf{u}^{n+1} + (\gamma - 1)\Delta t \underbrace{\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t}}_{=:\mathbf{d}^n} + \mathcal{O}(\Delta t^{p+1}).$$

Since $\mathbf{d}^n$ now is independent of γ, there is no need for Lemma 2 as we have an explicit expression for $\widetilde{\gamma}$, i.e. $\widetilde{\gamma}(\lambda) = \frac{\lambda + \Delta t\|\mathbf{d}^n\|_2}{\Delta t\|\mathbf{d}^n\|_2}$ and we can apply Theorem 2 giving us $\gamma = 1 + \mathcal{O}(\Delta t^{p-1})$. Also, Remark 4 motivates us to start a bootstrapping algorithm

using the functions $\bar{b}_j(\gamma) = \gamma b_j$ also for higher order. This seems to contradict our results from the theory on dense output, but in combination with relaxation, the issue can be resolved, see Remark 6 below.

Now in view of the following lemma, we can bootstrap the relaxation technique to higher orders.

Lemma 4 Consider a scheme of the form (26) with $\bar{b}_j(\gamma) = \gamma b_j$ and

$$\bar{\boldsymbol{\sigma}}(\gamma) = \mathbf{u}(t_n + \Delta t) + (\gamma - 1)(\mathbf{u}(t_n + \Delta t) - \mathbf{u}(t_n)) + \mathcal{O}(\Delta t^p) \tag{41}$$

for all γ in a neighborhood V of 1. Then

$$\frac{u_\nu^{n+\gamma}}{\bar{\sigma}_\nu(\gamma)} = 1 + \mathcal{O}(\Delta t^p) = \frac{u_\nu^{n+1}}{\sigma_\nu} + \mathcal{O}(\Delta t^p),$$

and in particular,

$$\mathbf{u}^{n+\gamma} = \mathbf{u}^{n+1} + (\gamma - 1)(\mathbf{u}^{n+1} - \mathbf{u}^n) + \mathcal{O}(\Delta t^{p+1})$$

for all $\gamma \in V$.

Remark 6 Before we prove this lemma, we want to stress two things. First, using (41) with $\bar{b}_j(\gamma) = \gamma b_j$ does not result in a higher-order dense output formula, but only guarantees to obtain the desired order at the root γ of $\eta(\mathbf{u}^{n+\gamma}) = \eta_{\text{old}}$, see Theorem 2 and Lemma 3.

Secondly, the bootstrapping process consists of using $\mathbf{u}^{n+\gamma}$ from the scheme of order $p-1$ as the new $\bar{\boldsymbol{\sigma}}(\gamma)$ resulting in a new method of order p. We can start the bootstrapping process by using the second-order MPRK22(α) scheme as a baseline method together with (28). Note that this naturally results in nested functions that depend on γ, which should be kept in mind when implementing the Newton iteration.

Proof of Lemma 4 We prove this claim by induction over $q = 1, \dots, p$ and exploit [23, Lemma 4.6,Lemma 4.8] to justify the implication

$$\mathbf{u}^{n+\gamma} = \bar{\boldsymbol{\sigma}}(\gamma) + \mathcal{O}(\Delta t^q) \Longrightarrow \frac{u_\nu^{n+\gamma}}{\bar{\sigma}_\nu(\gamma)} = 1 + \mathcal{O}(\Delta t^q), \quad \nu = 1, \dots, N.$$

For $q = 1$ we find $\mathbf{u}^{n+\gamma} = \mathbf{u}^n + \mathcal{O}(\Delta t)$ and $\bar{\boldsymbol{\sigma}}(\gamma) = \mathbf{u}^n + \mathcal{O}(\Delta t)$ since $\frac{u_\nu^{n+\gamma}}{\bar{\sigma}_\nu(\gamma)} = \mathcal{O}(1)$ due to [23, Lemma 4.6].

Now suppose that (41) holds with $\mathcal{O}(\Delta t^q)$ and some $q \geq 2$. By the induction hypothesis we have

$$\frac{u_\nu^{n+\gamma}}{\bar{\sigma}_\nu(\gamma)} = 1 + \mathcal{O}(\Delta t^{q-1}) = \frac{u_\nu^{n+1}}{\sigma_\nu} + \mathcal{O}(\Delta t^{q-1})$$

where we used $\frac{u_\nu^{n+1}}{\sigma_\nu} = 1 + \mathcal{O}(\Delta t^p)$ and $p \geq q$ for the last equality, see e.g. [22, Lemma 16]. Substituting this into (26), we see

$$\mathbf{u}^{n+\gamma} = \mathbf{u}^{n+1} + (\gamma - 1)(\mathbf{u}^{n+1} - \mathbf{u}^n) + \mathcal{O}(\Delta t^q).$$

Finally, since (41) holds with $\mathcal{O}(\Delta t^q)$, we deduce

$$\frac{u_\nu^{n+\gamma}}{\bar{\sigma}_\nu(\gamma)} = 1 + \mathcal{O}(\Delta t^q) = \frac{u_\nu^{n+1}}{\sigma_\nu} + \mathcal{O}(\Delta t^q).$$

□

Example 6 (Third-Order Relaxation for Conservative Problems using MPRK) Looking at the third-order MPRK family from Example 2, the relaxation scheme is fully defined by setting

$$\begin{aligned}\bar{\sigma}_k(\gamma) = u_k^n + \gamma\Delta t \sum_{j=1}^{2} \beta_j \Bigg(& r_k^P(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} p_{k\nu}(\mathbf{u}^{(j)}) \frac{\bar{\sigma}_\nu(\gamma)}{(u_\nu^{(2)})^{\frac{\gamma}{a_{21}}} (u_\nu^n)^{1-\frac{\gamma}{a_{21}}}} \\ & - \left(r_k^D(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} d_{k\nu}(\mathbf{u}^{(j)}) \right) \frac{\bar{\sigma}_k(\gamma)}{(u_k^{(2)})^{\frac{\gamma}{a_{21}}} (u_k^n)^{1-\frac{\gamma}{a_{21}}}} \Bigg)\end{aligned} \tag{42}$$

for $k = 1, \dots, d$, $\beta_1 = 1 - \beta_2$, and $\beta_2 = \frac{1}{2a_{21}}$.

Applying Newton's Method

As an illustrative example, we focus on (26) with $\bar{b}_j(\gamma) = \gamma b_j$, i.e.

$$u_k^{n+\gamma} = u_k^n + \Delta t\gamma \sum_{j=1}^{s} b_j \left(r_k^P(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} p_{k\nu}(\mathbf{u}^{(j)}) \frac{u_\nu^{n+\gamma}}{\bar{\sigma}_\nu(\gamma)} - \left(r_k^D(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} d_{k\nu}(\mathbf{u}^{(j)}) \right) \frac{u_k^{n+\gamma}}{\bar{\sigma}_k(\gamma)} \right), \tag{43}$$

and that

$$\begin{aligned}(\mathbf{M}_\gamma)_{kk} &= 1 + \gamma\Delta t \sum_{j=1}^{s} b_j \left(r_k^D(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} d_{k\nu}(\mathbf{u}^{(j)}) \right) \frac{1}{\bar{\sigma}_k(\gamma)}, \\ (\mathbf{M}_\gamma)_{k\nu} &= -\gamma\Delta t \sum_{j=1}^{s} b_j p_{k\nu}(\mathbf{u}^{(j)}) \frac{1}{\bar{\sigma}_\nu(\gamma)}, \quad k \neq \nu.\end{aligned} \tag{44}$$

in (31).

Starting with (43) we deduce

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}\gamma} u_k^{n+\gamma} = {} & \Delta t \sum_{j=1}^{s} \bar{b}_j'(\gamma) \left(r_k^P(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} p_{k\nu}(\mathbf{u}^{(j)}) \frac{u_\nu^{n+\gamma}}{\bar{\sigma}_\nu(\gamma)} - \left(r_k^D(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} d_{k\nu}(\mathbf{u}^{(j)}) \right) \frac{u_k^{n+\gamma}}{\bar{\sigma}_k(\gamma)} \right) \\ & + \Delta t \sum_{j=1}^{s} \bar{b}_j(\gamma) \Bigg(\sum_{\nu=1}^{d} p_{k\nu}(\mathbf{u}^{(j)}) \left(\frac{\frac{\mathrm{d}}{\mathrm{d}\gamma} u_\nu^{n+\gamma}}{\bar{\sigma}_\nu(\gamma)} - \frac{u_\nu^{n+\gamma} \bar{\sigma}_\nu'(\gamma)}{(\bar{\sigma}_\nu(\gamma))^2} \right) \\ & \qquad - \left(r_k^D(\mathbf{u}^{(j)}) + \sum_{\nu=1}^{d} d_{k\nu}(\mathbf{u}^{(j)}) \right) \left(\frac{\frac{\mathrm{d}}{\mathrm{d}\gamma} u_k^{n+\gamma}}{\bar{\sigma}_k(\gamma)} - \frac{u_k^{n+\gamma} \bar{\sigma}_k'(\gamma)}{(\bar{\sigma}_k(\gamma))^2} \right) \Bigg).\end{aligned}$$

To rewrite this in matrix-vector notation, we denote by "$\oslash$" and "$\odot$" the component-wise division and multiplication (Hadamard division and product), respectively. Then, using

$$\mathbf{v}(\gamma) \coloneqq \mathbf{u}^{n+\gamma} \odot \bar{\boldsymbol{\sigma}}'(\gamma) \oslash (\bar{\boldsymbol{\sigma}}(\gamma)),$$

we end up with

$$\mathbf{M}_\gamma(\mathbf{u}^n)\frac{\mathrm{d}}{\mathrm{d}\gamma}\mathbf{u}^{n+\gamma} = \frac{1}{\gamma}(\mathbf{u}^{n+\gamma} - \mathbf{u}^n) + (\mathbf{M}_\gamma(\mathbf{u}^n) - \mathbf{I})\mathbf{v}(\gamma). \tag{45}$$

Note that if $\bar{\boldsymbol{\sigma}}(\gamma) = \boldsymbol{\sigma}$, then $\mathbf{v}(\gamma) = \mathbf{0}$. Also note that the derivative of $\bar{\boldsymbol{\sigma}}$ from (42) itself satisfies an analogue equation to (45) as it represents an MPRK22(α) relaxation method of order 2.

Also, the system for MPSSPRK22 is the same; one only has to plug in the expressions $\mathbf{M}_\gamma$ and $\bar{\boldsymbol{\sigma}}$ for that particular scheme.

4 Numerical Experiments

In this section we apply our new relaxation algorithm to dissipative and conservative problems to validate our theoretical findings and to experimentally test the constraints on Δt for solving the system (31). We note that we use Newton's method for the computation of γ, if not stated otherwise. Also, we use (27) as default for MPRK22(α). Also, we may use a PID controller with parameters $\beta_1 = 0.7$, $\beta_2 = 0.4$, and $\beta_3 = 0$, see [55] for more details. The resulting method is denoted by MPRK22adap. We note that our implementation of the relaxation algorithm, which can be found in our reproducibility repository [60], is also adaptive in the sense that successful relaxation steps increase the Δt by 1% while unsuccessful searches for γ result in a 10% decrease of the time step size. We refer to the repository [60] for the implemented abortion criteria.

4.1 Lotka-Volterra System

The classical Lotka-Volterra system

$$\begin{aligned} u_1'(t) &= 2u_1(t) - u_1(t)u_2(t), d \\ u_2'(t) &= u_1(t)u_2(t) - u_2(t), \quad \mathbf{u}(0) = (2,2)^T \end{aligned}$$

can be written as a non-conservative PDRS with

$$r_1^P = 2u_1, \quad p_{21} = u_1u_2 = d_{12}, \quad r_2^D = u_2.$$

The entropy

$$\eta(\mathbf{u}) = \log(u_1) - u_1 + 2\log(u_2) - u_2$$

is conserved. Since the Lotka-Volterra system has periodic orbits, we expect improved numerical results using relaxation to conserve the entropy [61–63]. Although there are only positivity constraints, η is not non-decreasing for all $\mathbf{u} > \mathbf{0}$ in this example, which is why we use the default relaxation algorithm. As expected, we observe that relaxation improves the error growth of the base method from quadratic to linear, see Figure 1.

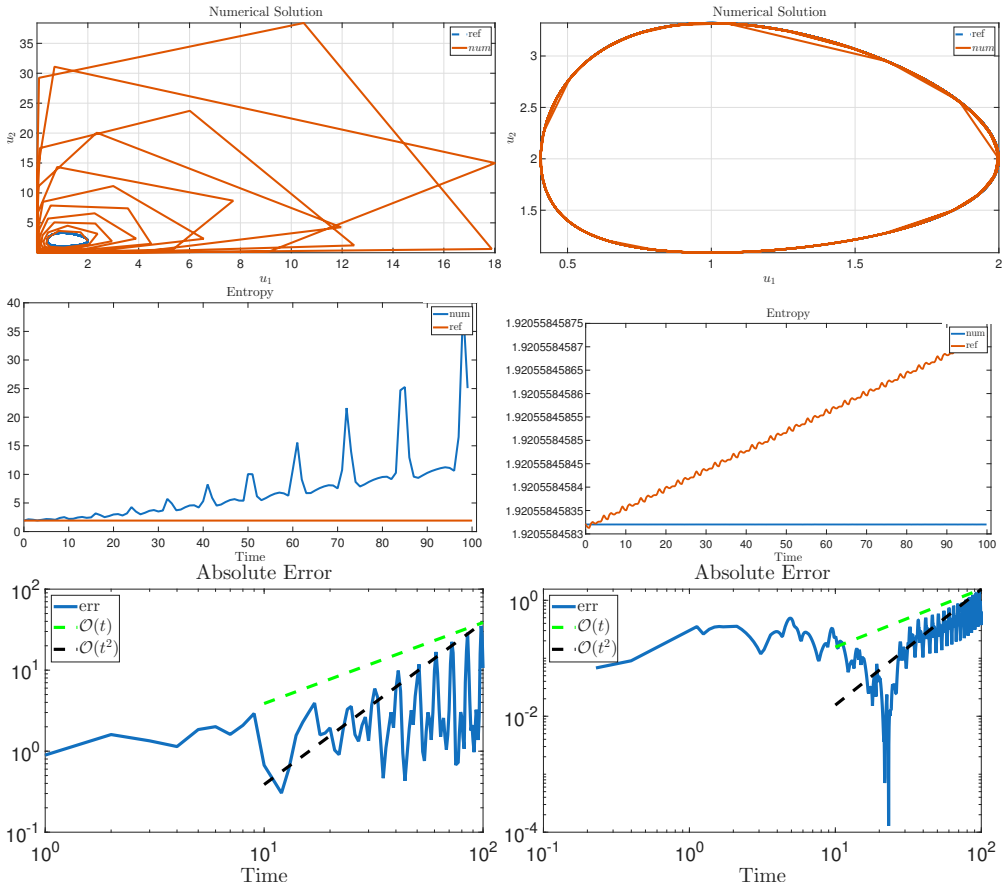


Fig. 1: Numerical solution of Lotka Volterra problem using MPRK22(1) (top) and $\Delta t = 1$. The error is $\mathrm{err}^n = \max(|u_1^n - u_1^{\mathrm{ref},n}|, |u_2^n - u_2^{\mathrm{ref},n}|)$. Left: without relaxation. Right: with relaxation.

4.2 Stratospheric Reaction Problem

The stratospheric reaction problem [2] is a stiff system of ODEs $\mathbf{w}'(t) = \mathbf{f}(t, \mathbf{w}(t))$ describing the interaction of the constituents $\mathbf{w} = (w_1, \ldots, w_6) = (O^{1D}, O, O_3, O_2, NO, NO_2)$. This non-conservative PDS possesses two linear invariants determined by the vectors $\widetilde{\mathbf{n}}_1 = (1, 1, 3, 2, 1, 2)^T$ and $\widetilde{\mathbf{n}}_2 = (0, 0, 0, 0, 1, 1)^T$. In order to apply MPRK schemes to this problem, we scale the corresponding differential equations writing

$$\mathrm{diag}(\widetilde{\mathbf{n}}_1)\mathbf{w}'(t) = \mathrm{diag}(\widetilde{\mathbf{n}_1})\mathbf{f}(t, \mathrm{diag}(\widetilde{\mathbf{n}}_1)^{-1}\,\mathrm{diag}(\widetilde{\mathbf{n}}_1)\mathbf{w}(t)).$$

Hence, introducing $\mathbf{u}(t) = \mathrm{diag}(\widetilde{\mathbf{n}}_1)\mathbf{w}(t)$, the two linear invariants of the differential equations $\mathbf{u}'(t) = \mathbf{f}(t, \mathrm{diag}(\widetilde{\mathbf{n}}_1)^{-1}\mathbf{u}(t))$ are $\mathbf{n}_1 = (1, 1, 1, 1, 1, 1)^T$ and $\mathbf{n}_2 =$

$(0,0,0,0,1,\frac{1}{2})^T$. Moreover, the scaled system takes the form

$$
\begin{aligned}
u_1' &= \tfrac{1}{3}r_5(t,\mathbf{u}) - (r_6(\mathbf{u}) + \tfrac{1}{3}r_7(\mathbf{u})),\\
u_2' &= r_1(t,\mathbf{u}) + \tfrac{1}{3}r_3(t,\mathbf{u}) + r_6(\mathbf{u}) + \tfrac{1}{2}r_{10}(t,\mathbf{u}) - (\tfrac{1}{2}r_2(\mathbf{u}) + \tfrac{1}{3}r_4(\mathbf{u}) + \tfrac{1}{2}r_9(\mathbf{u}) + r_{11}(\mathbf{u})),\\
u_3' &= \tfrac{3}{2}r_2(\mathbf{u}) - (r_3(t,\mathbf{u}) + r_4(\mathbf{u}) + r_5(t,\mathbf{u}) + r_7(\mathbf{u}) + r_8(\mathbf{u})),\\
u_4' &= \tfrac{2}{3}r_3(t,\mathbf{u}) + \tfrac{4}{3}r_4(\mathbf{u}) + \tfrac{2}{3}r_5(t,\mathbf{u}) + \tfrac{4}{3}r_7(\mathbf{u}) + \tfrac{2}{3}r_8(\mathbf{u}) + r_9(\mathbf{u}) - (r_1(t,\mathbf{u}) + r_2(\mathbf{u})),\\
u_5' &= \tfrac{1}{2}r_9(\mathbf{u}) + \tfrac{1}{2}r_{10}(t,\mathbf{u}) - (\tfrac{1}{3}r_8(\mathbf{u})r_{11}(\mathbf{u})),\\
u_6' &= \tfrac{2}{3}r_8(\mathbf{u}) + 2r_{11}(\mathbf{u}) - (r_9(\mathbf{u}) + r_{10}(t,\mathbf{u})),
\end{aligned}
\tag{46}
$$

where

$$
\begin{aligned}
r_1(t,\mathbf{u}) &= k_1(t)u_4, & k_1(t) &= \sigma(T(t))^3 \cdot 2.643 \cdot 10^{-10},\\
r_2(t,\mathbf{u}) &= k_2u_2u_4, & k_2 &= 8.018 \cdot 10^{-17},\\
r_3(t,\mathbf{u}) &= k_3(t)u_3, & k_3(t) &= \sigma(T(t)) \cdot 6.120 \cdot 10^{-4},\\
r_4(t,\mathbf{u}) &= k_4u_2u_3, & k_4 &= 1.576 \cdot 10^{-15},\\
r_5(t,\mathbf{u}) &= k_5(t)u_3, & k_5(t) &= \sigma(T(t))^2 \cdot 1.070 \cdot 10^{-3},\\
r_6(t,\mathbf{u}) &= k_6Mu_1, & k_6 &= 7.110 \cdot 10^{-11}, \qquad M = 8.120 \cdot 10^{16},\\
r_7(t,\mathbf{u}) &= k_7u_1u_3, & k_7 &= 1.200 \cdot 10^{-10},\\
r_8(t,\mathbf{u}) &= k_8u_3u_5, & k_8 &= 6.062 \cdot 10^{-15},\\
r_9(t,\mathbf{u}) &= k_9u_2u_6, & k_9 &= 1.069 \cdot 10^{-11},\\
r_{10}(t,\mathbf{u}) &= k_{10}(t)u_6, & k_{10}(t) &= \sigma(T(t)) \cdot 1.289 \cdot 10^{-2},\\
r_{11}(t,\mathbf{u}) &= k_{11}u_2u_5, & k_{11} &= 10^{-8}
\end{aligned}
$$

as well as

$$
\sigma(T(t)) = \begin{cases} 0.5 + 0.5\cos\left(\pi\left|\frac{2T(t)-T_r-T_2}{T_s-T_r}\right|\frac{2T(t)-T_r-T_2}{T_s-T_r}\right), & T_r \le T(t) \le T_s,\\ 0, & \text{otherwise,} \end{cases}
$$

$$
T(t) = \frac{t}{3600} \mod 24, \quad T_r = 4.5, \quad T_s = 19.5.
$$

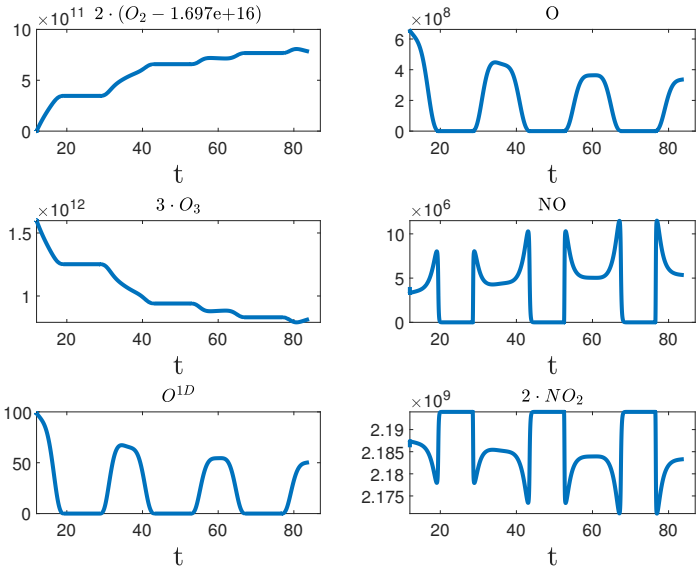


Fig. 2: Reference solution of the scaled stratospheric problem (46) depicted over the time interval $[12, 84]$ in hours.

The non-zero production and destruction terms of the system (46) are given by

$$
\begin{aligned}
&d_{12}(t,\mathbf{u}) = r_6(\mathbf{u}), && d_{14}(t,\mathbf{u}) = \tfrac{1}{3}r_7(\mathbf{u}), && d_{23}(t,\mathbf{u}) = \tfrac{1}{2}r_2(\mathbf{u}),\\
&d_{24}(t,\mathbf{u}) = \tfrac{1}{3}r_4(\mathbf{u}), && d_{25}(t,\mathbf{u}) = \tfrac{1}{2}r_9(\mathbf{u}), && d_{26}(t,\mathbf{u}) = r_{11}(\mathbf{u}),\\
&d_{31}(t,\mathbf{u}) = \tfrac{1}{3}r_5(t,\mathbf{u}), && d_{32}(t,\mathbf{u}) = \tfrac{1}{3}r_3(t,\mathbf{u}), && d_{36}(t,\mathbf{u}) = \tfrac{1}{3}r_8(\mathbf{u}),\\
&d_{34}(t,\mathbf{u}) = \tfrac{2}{3}r_3(t,\mathbf{u}) + r_4(\mathbf{u}) + \tfrac{2}{3}r_5(t,\mathbf{u}) + r_7(\mathbf{u}) + \tfrac{2}{3}r_8(\mathbf{u}),\\
&d_{42}(t,\mathbf{u}) = r_1(t,\mathbf{u}), && d_{43}(t,\mathbf{u}) = r_2(\mathbf{u}), && d_{56}(t,\mathbf{u}) = r_{11}(\mathbf{u}) + \tfrac{1}{3}r_8(\mathbf{u}),\\
&d_{62}(t,\mathbf{u}) = \tfrac{1}{2}r_{10}(t,\mathbf{u}), && d_{64}(t,\mathbf{u}) = r_9(\mathbf{u}), && d_{65}(t,\mathbf{u}) = \tfrac{1}{2}r_{10}(t,\mathbf{u}),
\end{aligned}
$$

and $p_{ij} = d_{ji}$. The solution to this problem will be approximated over the time interval $[12 \cdot 3600, 84 \cdot 3600]$, where a unit of time represents a second. A reference solution of the scaled problem is depicted in Figure 2. As we will see, MPRK schemes do not conserve the second linear invariant, which is why

$$\mathbf{n}_2^T \frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\|\mathbf{u}^{n+1} - \mathbf{u}^n\|} = c(\mathbf{u}^n)\Delta t + \mathcal{O}(\Delta t^2)$$

with $c \neq 0$. Hence, we may use

$$\eta(\mathbf{u}) = \mathbf{n}_2^T \mathbf{u}$$

as an entropy function satisfying (5) to preserve also the second linear invariant with our relaxation technique for conservative problems. As can be seen in Figure 3, using relaxation improves the accuracy of the solution significantly. However, we note that

Newton's method, while working in principle, sometimes fails at finding a solution $\gamma \approx 1$ in our implementation. Indeed, we observed $\gamma \approx 10^{-12}$ and thus decided to use Regula Falsi as a solver.

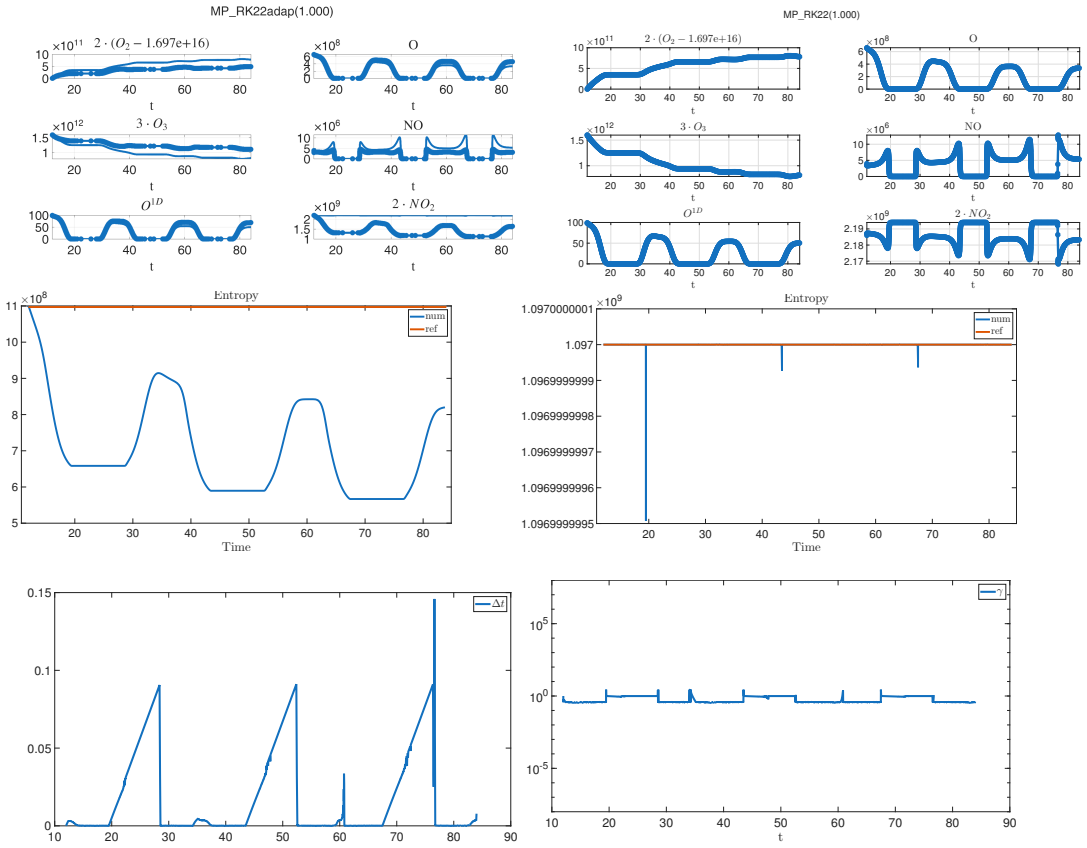


Fig. 3: Numerical solution of stratospheric reaction problem using standard MPRK22adap(1) (left) with `rtol` = `atol` = 10^{-3} and $\Delta t_0 = 0.01h$. Top: without relaxation. Middle: with relaxation (Regula Falsi) and initial $\Delta t = 0.01h$. Bottom: Plots of Δt and γ for the run with relaxation.

4.3 Linear advection

Consider the linear advection equation

$$\partial_t u + \partial_x u = 0 \tag{47}$$

with periodic boundary conditions on $I = [0, 2]$ and a positive initial condition $u^0 > 0$. Then, the solution $u(t, x) = u^0(x - t)$ stays positive. Moreover, every functional of the

form

$$\eta(u(t,I)) = \int_I U\big(u(t,x)\big)\,\mathrm{d}x \tag{48}$$

for an entropy function U is conserved with associated entropy flux $F(u) = U(u)$. Following Tadmor [64], the entropy variables are $w = U'(u)$ and the flux potential is $\psi = wf - F = U'(u)u - U(u)$. The corresponding entropy-conservative numerical flux is

$$f^{\mathrm{num}}(u_-,u_+) = \frac{\psi(u_+)-\psi(u_-)}{w(u_+)-w(u_-)} = \frac{U'(u_+)u_+ - U(u_+) - U'(u_-)u_- + U(u_-)}{U'(u_+)-U'(u_-)}. \tag{49}$$

If $U'(u) \to \infty$ faster than $U(u)$ as $u \to 0$, then the numerical flux goes to zero if one of the states goes to zero. Therefore, the resulting finite volume method

$$\partial_t u_i + \frac{f^{\mathrm{num}}(u_{i-1},u_i) - f^{\mathrm{num}}(u_i,u_{i+1})}{\Delta x} = 0 \tag{50}$$

is positivity-preserving in this case. In particular, the numerical fluxes $f^{\mathrm{num}}(u_-,u_+)$ are always non-negative, resulting in a conservative production-destruction system. Next, we consider several examples.

- The entropy
$$U(u) = u\log(u) - u \tag{51}$$
leads to the entropy-conservative numerical flux
$$f^{\mathrm{num}}(u_-,u_+) = \frac{u_+ - u_-}{\log(u_+)-\log(u_-)} =: \{\{u\}\}_{\log} \tag{52}$$
using the logarithmic mean, see [65, Section 3.2].
- Similarly, the entropy
$$U(u) = -\sqrt{u} \tag{53}$$
leads to the entropy variables $w = -1/(2\sqrt{u})$, the flux potential $\psi = \sqrt{u}/2$, and the entropy-conservative numerical flux
$$f^{\mathrm{num}}(u_-,u_+) = \frac{\sqrt{u_+}-\sqrt{u_-}}{-1/\sqrt{u_+}+1/\sqrt{u_-}} = \sqrt{u_-u_+} =: \{\{u\}\}_{\mathrm{geo}} \tag{54}$$
using the geometric mean.
- Analogously, the entropy
$$U(u) = 1/u \tag{55}$$
leads to the entropy variables $w = -1/u^2$, the flux potential $\psi = -2/u$, and the entropy-conservative numerical flux
$$f^{\mathrm{num}}(u_-,u_+) = \frac{-2/u_+ + 2/u_-}{-1/u_+^2 + 1/u_-^2} = \frac{2u_-u_+}{u_+ + u_-} =: \{\{u\}\}_{\mathrm{harm}} \tag{56}$$
using the harmonic mean.

Please note that positivity preservation for an entropy-conservative method depends on the choice of the entropy function. For example, the standard L^2 entropy

$$U(u) = \frac{u^2}{2} \tag{57}$$

leads to the numerical flux

$$f^{\text{num}}(u_-, u_+) = \frac{1}{2}\frac{u_+^2 - u_-^2}{u_+ - u_-} = \frac{u_- + u_+}{2}, \tag{58}$$

i.e., the standard arithmetic mean. The resulting finite volume discretization

$$\partial_t u_i + \frac{u_{i+1} - u_{i-1}}{2\Delta x} = 0 \tag{59}$$

is the classical second-order central discretization, which is not positivity-preserving.

We use $N_x = 100$ cells and the initial condition

$$u(0, x) = 1.9 \sin(\pi x) + 2, \quad x \in [0, 2]$$

and apply different iterative methods for solving for γ. The respective results are depicted in Figure 4.

4.4 Shallow water equations

The classical shallow water equations

$$\partial_t \underbrace{\begin{pmatrix} h \\ hv \end{pmatrix}}_{=u} + \partial_x \underbrace{\begin{pmatrix} hv \\ hv^2 + \frac{1}{2}gh^2 \end{pmatrix}}_{=f(u)} = 0 \tag{60}$$

have the total energy

$$U(u) = \tfrac{1}{2}hv^2 + \tfrac{1}{2}gh^2 \tag{61}$$

as entropy. The corresponding entropy variables are

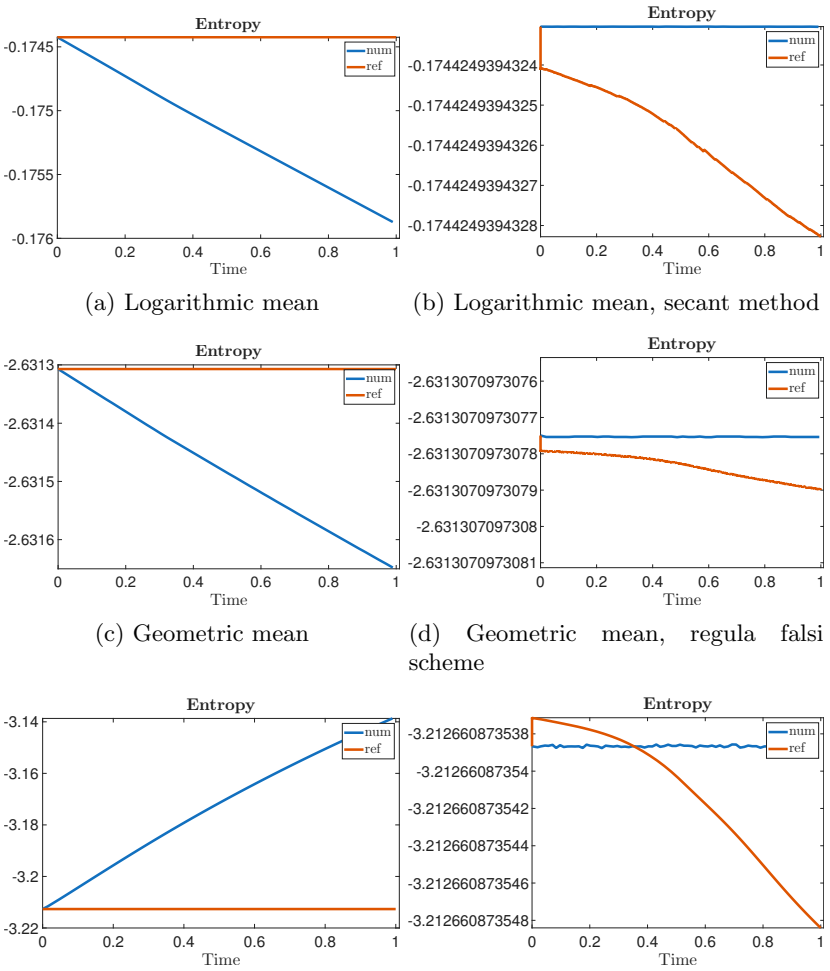
$$w = \begin{pmatrix} gh - \frac{1}{2}v^2 \\ v \end{pmatrix} \tag{62}$$

and the entropy flux potential is

$$\psi = \tfrac{1}{2}gh^2 v. \tag{63}$$

For constant velocity v, the condition for an entropy-conservative numerical flux is

$$0 = [\![w]\!] \cdot f^{\text{num}} - [\![\psi]\!] = g[\![h]\!] f_h^{\text{num}} - \tfrac{1}{2}g[\![h^2]\!]v, \tag{64}$$



(a) Logarithmic mean

(b) Logarithmic mean, secant method

(c) Geometric mean

(d) Geometric mean, regula falsi scheme

(e) Harmonic mean

(f) Harmonic mean, bisection method

Fig. 4: Numerical solution of linear advection equation using MPSSPRK(0.5,1) ((a)-(b)), MPRK43adap(0.5,0.75) ((c)-(d)), and MPRK22adap(1) ((e)-(f)) with `rtol` = `atol` = 10^{-3}, $N = 100$, and $\Delta t_0 = \Delta x$. Left: without relaxation. Right: with relaxation.

where $[\![w]\!] := w_{i+1} - w_i$. Thus, the numerical flux for the water height h is

$$f_h^{\text{num}} = \frac{1}{2}\frac{[\![h^2]\!]}{[\![h]\!]} v = \{\{h\}\} v \tag{65}$$

with the arithmetic mean

$$\{\{h\}\} = \tfrac{1}{2}(h_- + h_+). \tag{66}$$

Similarly to the linear advection equations above, the arithmetic mean does not lead to a positivity-preserving semidiscretization. This proves

Theorem 3 *An entropy-conservative semidiscretization of the shallow water equations is not positivity-preserving.*

One can show a similar result for the polytropic Euler equations with pressure $p \propto \varrho^\gamma$ and $\gamma > 1$. However, the limiting case of the isothermal Euler equations is different and discussed in the next subsection.

4.5 Isothermal Euler equations

The 1D isothermal Euler equations are

$$\partial_t \underbrace{\begin{pmatrix} \varrho \\ \varrho v \end{pmatrix}}_{=\mathbf{u}} + \partial_x \underbrace{\begin{pmatrix} \varrho v \\ \varrho v^2 + c^2 \varrho \end{pmatrix}}_{=\mathbf{f}(\mathbf{u})} = 0, \tag{67}$$

where ϱ is the density, v is the velocity, and c is the speed of sound. We take the total energy

$$U(\mathbf{u}) = \tfrac{1}{2}\varrho v^2 + \tfrac{1}{2}c^2 \varrho \log(\varrho) \tag{68}$$

as (mathematical) entropy. An associated entropy-conservative numerical flux at the interface $i + \frac{1}{2}$ is given by [66]

$$f_\varrho^{\text{num}} = \{\{\varrho\}\}_{\log}\{\{v\}\}, \quad f_{\varrho v}^{\text{num}} = \{\{\varrho\}\}_{\log}\{\{v\}\}^2 + \{\{c^2\varrho\}\}. \tag{69}$$

Since the logarithmic mean goes to zero if one of the states goes to zero, the resulting entropy-conservative finite volume method is unconditionally positive. Even more general, the flux differencing method [31, 64, 67–70] based on diagonal-norm SBP operators. In particular, high-order discontinuous Galerkin spectral element methods (DGSEMs) are positivity-preserving. While we apply the underlying explicit RK method to the second conserved variable together with the standard relaxation algorithm, we use MPRK22 for ϱ, where we use the PDS

$$p_{i+1,i} = d_{i,i+1} = \max\left\{0, f_\varrho^{\text{num}}\right\}, \quad p_{i,i+1} = d_{i+1,i} = -\min\left\{0, f_\varrho^{\text{num}}\right\}$$

for $i = 1, \dots, N-1$, and we take periodic boundary conditions into account for the terms if $i = N$. The study of flux-balanced MPRK schemes introduced in [71] is left for future works. In Figure 5 we solve the Riemann problem (RP)

$$\mathbf{u}_L = (0.8, 10^{-3}), \quad \mathbf{u}_R = (1, 10^{-2})$$

with periodic boundary conditions and final time $t_{\text{end}} = 1$. We note that one should not use an entropy conservative flux for an RP, however, this is a good example that our time integrator maintains the entropy properties of the space discretization.

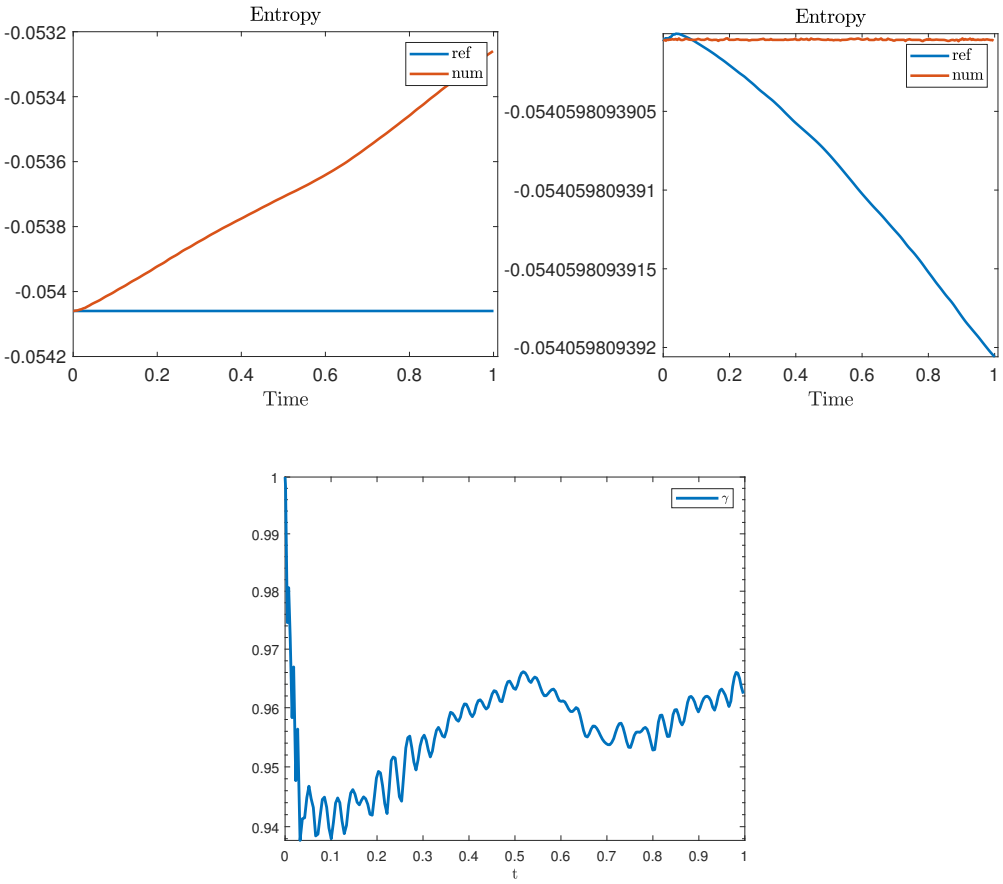


Fig. 5: Numerical solution of isothermal Euler equations with $N_x = 100$ using MPRK22adap(1) with $\texttt{rtol} = \texttt{atol} = 10^{-3}$ and $\Delta t = \Delta x$. Left: without relaxation. Right: with relaxation.

4.6 Porous Medium Equation

The porous medium equation

$$u_t = (u^m)_{xx} = (a(u)u_x)_x, \quad a(u) = mu^{m-1}$$

with a free parameter $m > 1$, see for instance [72], admits a non-negative weak solution

$$u^{(m)}(t,x) = t^{-k}\left[\max\left(1 - \frac{k(m-1)}{2m}\frac{|x|^2}{t^{2k}}, 0\right)\right]^{\frac{1}{m-1}}$$

with $k = \frac{1}{m+1}$, the so-called *Barenblatt* solution [73]. For every $t > 0$, the solution has a compact support $[-\alpha_m(t), \alpha_m(t)]$ where

$$\alpha_m(t) = \sqrt{\frac{2m}{k(m-1)}} t^k.$$

We follow [72] using $u(0,x) = u^{(m)}(1,x)$ as an initial condition. We plot the numerical solution at time $t = 2$ on the spatial domain $[-6, 6]$ using the boundary conditions $u(t, \pm 6) = 0$ for $t > 1$.

We use the second-order space discretization from [74, 75] given by

$$\begin{aligned} f_i(\mathbf{u}(t)) =& \frac{a(u_i(t)) + a(u_{i+1}(t))}{2\Delta x^2} u_{i+1}(t) \\ &- \frac{a(u_{i-1}(t)) + 2a(u_i(t)) + a(u_{i+1}(t))}{2\Delta x^2} u_i(t) \\ &+ \frac{a(u_{i-1}(t)) + a(u_i(t))}{2\Delta x^2} u_{i-1}(t) \end{aligned}$$

for $i = 2, \dots, N$ and

$$f_j(\mathbf{u}(t)) = \frac{a(u_j(t))}{2\Delta x^2} u_j(t), \quad \text{for} \quad j \in \{1, N\}.$$

Next, we consider the convex entropy

$$\eta(\mathbf{u}) = \frac{\Delta x^2}{2} \sum_{i=1}^{N_x} u_i^2,$$

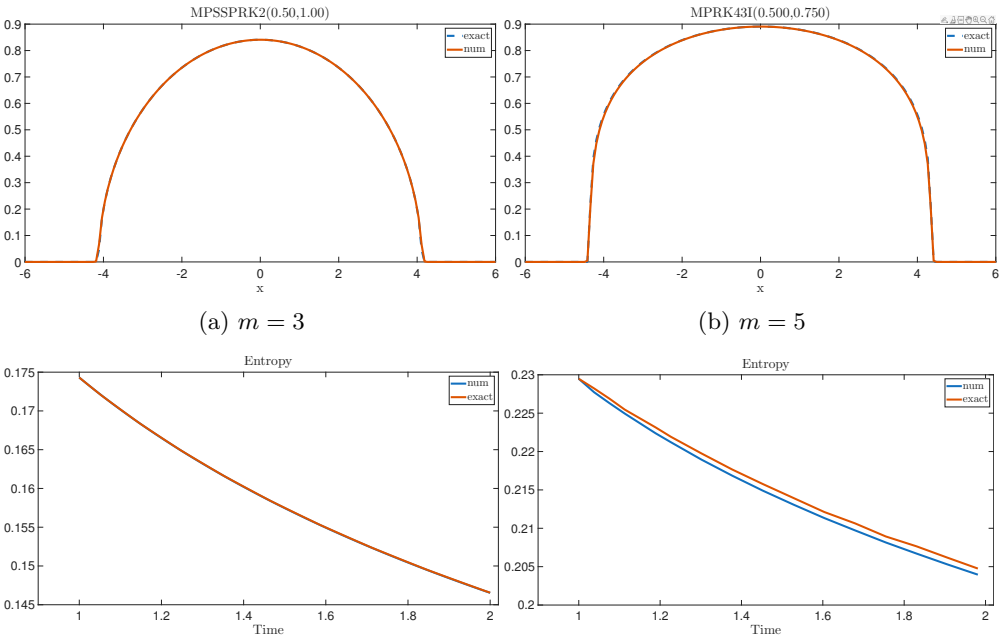
which satisfies

$$\frac{\mathrm{d}}{\mathrm{d}t} \eta(\mathbf{u}(t)) \leq 0$$

for the boundary conditions mentioned above, see [75, Theorem 4.1]. This system of ODEs may be rewritten as a conservative PDS by setting

$$\begin{aligned} p_{i,i+1}(\mathbf{u}) &= \frac{a(u_i) + a(u_{i+1})}{2\Delta x^2} u_{i+1}, & p_{i,i-1}(\mathbf{u}) &= \frac{a(u_{i-1}) + a(u_i)}{2\Delta x^2} u_{i-1}, & i &= 2, \dots, N, \\ p_{1,2}(\mathbf{u}) &= \frac{a(u_2)}{2\Delta x^2} u_2, & p_{N,N-1}(\mathbf{u}) &= \frac{a(u_{N-1})}{2\Delta x^2} u_{N-1}, & d_{i,j} &= p_{j,i}. \end{aligned}$$

According to [72], the cases $m = 3, 5$ are particularly interesting as the numerical solution of the proposed third-order IMEX method in [72, p. 10, eq. (30)] generates negative approximations and which cannot happen with MPRK schemes. Indeed, we observe in Figure 6 that we obtain positive approximations while the relaxation algorithm gives us an entropy estimate. Here, we do not plot γ as it was constantly at 1.



(a) $m = 3$ (b) $m = 5$

(c) $m = 3$ (d) $m = 5$

Fig. 6: Numerical solution of PME equation with $N_x = 160$ using MPSSPRK22(0.5,1) (left) and MPRK43(0.5,0.75) (right) with $\Delta t = \Delta x$ and relaxation.

5 Summary and conclusions

In this work we investigated non-standard additive Runge–Kutta (NSARK) schemes, which include modified Patankar (MP) methods or Geometric Conservative (GeCo) to name a few. Being particularly interested in positivity-preserving methods that are also capable of conserving at least one linear invariant, we answered the question of whether these schemes can be equipped with a relaxation technique that preserves these properties while ensuring entropy stability. We point out that positivity preservation is easy to accomplish for entropy dissipative problems. For entropy conservative problems, where no linear invariant needs to be preserved, one can equip an unconditionally positive method with the geometric mean to compute the relaxation update. If the conservative problem has a linear invariant or one is interested in keeping a conservative PDS part also conservative within the relaxation procedure, we propose to use a linearly implicit formula for the relaxation update, which in turn results in a coupled linear-nonlinear system for the simultaneous computation of γ and $\mathbf{u}^{n+\gamma}$. All techniques can be used for any positivity-preserving method maintaining the order, however, the latter relaxation technique involves a bootstrapping algorithm to achieve higher-order entropy conservative methods preserving a linear invariant.

We have tested our theoretical findings by means of multiple examples of ordinary and partial differential equations. Furthermore, interpreting a linear invariant as

entropy, we were able to preserve both linear invariants of the stiff stratospheric reaction problem using MPRK. We have also tested several flux and entropy pairs for the linear advection equation testing the different iterative solvers for the computation of γ and $\mathbf{u}^{n+\gamma}$. Moreover, we applied our technique also in the context of the isothermal Euler equation guaranteeing the positivity of the density. Finally, we have also tested MPRK and MPSSPRK schemes with the entropy dissipative porous medium equation, where we are also able to avoid negative approximations.

Future research topics include the testing of further NSARK schemes, including the recently developed flux-balanced versions, and the efficiency of the related methods. As some of the NSARK schemes are already proven to be conditionally stable, a thorough stability analysis for these methods is also part of ongoing research.

Acknowledgements.

Declarations

Funding

T. Izgin gratefully acknowledges the financial support by Fulbright Germany. H. Ranocha was supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation, project number 513301895) and the Daimler und Benz Stiftung (Daimler and Benz foundation, project number 32-10/22). C.-W. Shu acknowledges partial support from NSF grant DMS-2309249.

Conflict of interest

Not applicable.

Ethics approval and consent to participate

Not applicable.

Consent for publication

All authors consent to submit for publication.

Data, Materials and Code availability

The source code used in this study is available at [60].

Author contribution

All authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by Thomas Izgin. The first draft of the manuscript was written by Thomas Izgin and all authors commented on as well as wrote on previous versions of the manuscript. All authors read and approved the final manuscript.

References

[1] Bruggeman, J., Burchard, H., Kooi, B.W., Sommeijer, B.: A second-order, unconditionally positive, mass-conserving integration scheme for biochemical systems.

Applied Numerical Mathematics **57**(1), 36–58 (2007) https://doi.org/10.1016/j.apnum.2005.12.001

[2] Sandu, A.: Positive numerical integration methods for chemical kinetic systems. J. Comput. Phys. **170**(2), 589–602 (2001) https://doi.org/10.1006/jcph.2001.6750

[3] Shampine, L.F., Thompson, S., Kierzenka, J.A., Byrne, G.D.: Non-negative solutions of odes. Applied Mathematics and Computation **170**(1), 556–569 (2005) https://doi.org/10.1016/j.amc.2004.12.011

[4] Huang, J., Shu, C.-W.: Positivity-preserving time discretizations for production-destruction equations with applications to non-equilibrium flows. J. Sci. Comput. **78**(3), 1811–1839 (2019)

[5] Bolley, C., Crouzeix, M.: Conservation de la positivité lors de la discrétisation des problèmes d'évolution paraboliques. RAIRO. Analyse numérique **12**(3), 237–245 (1978)

[6] Blanes, Sergio, Iserles, Arieh, Macnamara, Shev: Positivity-preserving methods for ordinary differential equations. ESAIM: M2AN **56**(6), 1843–1870 (2022) https://doi.org/10.1051/m2an/2022042

[7] Nüßlein, S., Ranocha, H., Ketcheson, D.I.: Positivity-preserving adaptive Runge-Kutta methods. Communications in Applied Mathematics and Computational Science **16**(2), 155–179 (2021) https://doi.org/10.2140/camcos.2021.16.155 2005.06268

[8] Hubbard, M.E., Ricchiuto, M., Sármány, D.: Space-time residual distribution on moving meshes. Comput. Math. Appl. **79**(5), 1561–1589 (2020) https://doi.org/10.1016/j.camwa.2019.09.019

[9] Ricchiuto, M.: Contributions to the development of residual discretizations for hyperbolic conservation laws with application to shallow water flows. Habilitation thesis, Université Sciences et Technologies-Bordeaux I, Bordeaux, France (December 2011)

[10] Horváth, Z.: Positivity of Runge–Kutta and diagonally split Runge–Kutta methods. Applied Numerical Mathematics **28**(2-4), 309–326 (1998) https://doi.org/10.1016/S0168-9274(98)00050-6

[11] Macdonald, C.B., Gottlieb, S., Ruuth, S.J.: A Numerical Study of Diagonally Split Runge–Kutta Methods for PDEs with Discontinuities. Journal of Scientific Computing **35**, 89–112 (2008)

[12] Gottlieb, S., Ketcheson, D., Shu, C.-W.: Strong Stability Preserving Runge-Kutta and Multistep Time Discretizations. World Scientific Publishing Co. Pte. Ltd., Hackensack, NJ (2011). https://doi.org/10.1142/7498

[13] Patankar, S.V.: Numerical Heat Transfer and Fluid Flow. Series in computational methods in mechanics and thermal sciences. Hemisphere Pub. Corp. New York, Washington (1980). http://opac.inria.fr/record=b1085925

[14] Burchard, H., Deleersnijder, E., Meister, A.: A high-order conservative Patankar-type discretisation for stiff systems of production-destruction equations. Appl. Numer. Math. **47**(1), 1–30 (2003) https://doi.org/10.1016/S0168-9274(03)00101-6

[15] Martiradonna, A., Colonna, G., Diele, F.: GeCo: Geometric Conservative non-standard schemes for biochemical systems. Appl. Numer. Math. **155**, 38–57 (2020) https://doi.org/10.1016/j.apnum.2019.12.004

[16] Kopecz, S., Meister, A.: On order conditions for modified Patankar-Runge-Kutta schemes. Appl. Numer. Math. **123**, 159–179 (2018)

[17] Ávila, A.I., Kopecz, S., Meister, A.: A comprehensive theory on generalized BBKS schemes. Appl. Numer. Math. **157**, 19–37 (2020)

[18] Kopecz, S., Meister, A.: Unconditionally positive and conservative third order modified Patankar-Runge-Kutta discretizations of production-destruction systems. BIT **58**(3), 691–728 (2018) https://doi.org/10.1007/s10543-018-0705-1

[19] Huang, J., Zhao, W., Shu, C.-W.: A third-order unconditionally positivity-preserving scheme for production-destruction equations with applications to non-equilibrium flows. J. Sci. Comput. **79**(2), 1015–1056 (2019)

[20] Öffner, P., Torlo, D.: Arbitrary high-order, conservative and positivity preserving Patankar-type deferred correction schemes. Appl. Numer. Math. **153**, 15–34 (2020)

[21] Izzo, G., Messina, E., Pezzella, M., Vecchio, A.: Modified patankar linear multistep methods for production-destruction systems. Journal of Scientific Computing **102**(3), 87 (2025)

[22] Izgin, T., Ketcheson, D.I., Meister, A.: Order conditions for runge–kutta-like methods with solution-dependent coefficients. Communications in Applied Mathematics and Computational Science **20-1**, 29–66 (2025) https://doi.org/10.2140/camcos.2025.20.29

[23] Izgin, T.: A unifying theory for runge-kutta-like time integrators: Convergence and stability. PhD thesis, University of Kassel (2024). https://doi.org/10.17170/kobra-202402059522

[24] Tadmor, E.: From semidiscrete to fully discrete: Stability of Runge-Kutta schemes by the energy method II. In: Estep, D.J., Tavener, S. (eds.) Collected Lectures on

the Preservation of Stability Under Discretization. Proceedings in Applied Mathematics, vol. 109, pp. 25–49. Society for Industrial and Applied Mathematics, Philadelphia (2002)

[25] Ranocha, H., Öffner, P.: L_2 stability of explicit Runge-Kutta schemes. Journal of Scientific Computing **75**(2), 1040–1056 (2018) https://doi.org/10.1007/s10915-017-0595-4

[26] Sun, Z., Shu, C.-W.: Stability of the fourth order Runge-Kutta method for time-dependent partial differential equations. Annals of Mathematical Sciences and Applications **2**(2), 255–284 (2017) https://doi.org/10.4310/AMSA.2017.v2.n2.a3

[27] Sun, Z., Shu, C.-W.: Strong stability of explicit Runge-Kutta time discretizations. SIAM Journal on Numerical Analysis **57**(3), 1158–1182 (2019) https://doi.org/10.1137/18M122892X 1811.10680

[28] Achleitner, F., Arnold, A., Jüngel, A.: Necessary and sufficient conditions for strong stability of explicit Runge-Kutta methods. In: Carlen, E., Gonçalves, P., Soares, A.J. (eds.) From Particle Systems to Partial Differential Equations. PSPDE X, Braga, Portugal, June 2022. Springer Proceedings in Mathematics & Statistics, vol. 465, pp. 1–21. Springer, Cham (2024). https://doi.org/10.1007/978-3-031-65195-3_1

[29] Tadmor, E.: On the stability of Runge-Kutta methods for arbitrarily large systems of ODEs. Communications on Pure and Applied Mathematics **78**(4), 821–855 (2025) https://doi.org/10.1002/cpa.22238

[30] Sun, Z., Wei, Y., Wu, K.: On energy laws and stability of Runge-Kutta methods for linear seminegative problems. SIAM Journal on Numerical Analysis **60**(5), 2448–2481 (2022) https://doi.org/10.1137/22M1472218 2201.06501

[31] LeFloch, P.G., Mercier, J.-M., Rohde, C.: Fully discrete, entropy conservative schemes of arbitrary order. SIAM Journal on Numerical Analysis **40**(5), 1968–1992 (2002) https://doi.org/10.1137/S003614290240069X

[32] Friedrich, L., Schnücke, G., Winters, A.R., Fernández, D.C.D.R., Gassner, G.J., Carpenter, M.H.: Entropy stable space-time discontinuous Galerkin schemes with summation-by-parts property for hyperbolic conservation laws. Journal of Scientific Computing **80**(1), 175–222 (2019) https://doi.org/10.1007/s10915-019-00933-2 1808.08218

[33] Burrage, K., Butcher, J.C.: Stability criteria for implicit Runge-Kutta methods. SIAM Journal on Numerical Analysis **16**(1), 46–57 (1979) https://doi.org/10.1137/0716004

[34] Burrage, K., Butcher, J.C.: Non-linear stability of a general class of differential equation methods. BIT Numerical Mathematics **20**(2), 185–203 (1980) https:

//doi.org/10.1007/BF01933191

[35] Dahlby, M., Owren, B., Yaguchi, T.: Preserving multiple first integrals by discrete gradients. Journal of Physics A: Mathematical and Theoretical **44**(30), 305205 (2011) https://doi.org/10.1088/1751-8113/44/30/305205

[36] Higueras, I.: Monotonicity for Runge-Kutta methods: Inner product norms. Journal of Scientific Computing **24**(1), 97–117 (2005) https://doi.org/10.1007/s10915-004-4789-1

[37] Jüngel, A., Milišić, J.-P.: Entropy dissipative one-leg multistep time approximations of nonlinear diffusive equations. Numerical Methods for Partial Differential Equations **31**(4), 1119–1149 (2015) https://doi.org/10.1002/num.21938

[38] Jüngel, A., Schuchnigg, S.: Entropy-dissipating semi-discrete Runge-Kutta schemes for nonlinear diffusion equations. Communications in Mathematical Sciences **15**(1), 27–53 (2017) https://doi.org/10.4310/CMS.2017.v15.n1.a2

[39] Ranocha, H.: On strong stability of explicit Runge-Kutta methods for nonlinear semibounded operators. IMA Journal of Numerical Analysis **41**(1), 654–682 (2021) https://doi.org/10.1093/imanum/drz070 1811.11601

[40] Ranocha, H., Ketcheson, D.I.: Energy stability of explicit Runge-Kutta methods for nonautonomous or nonlinear problems. SIAM Journal on Numerical Analysis **58**(6), 3382–3405 (2020) https://doi.org/10.1137/19M1290346 1909.13215

[41] Shampine, L.F.: Conservation laws and the numerical solution of odes. Computers & Mathematics with Applications **12**(5, Part 2), 1287–1296 (1986) https://doi.org/10.1016/0898-1221(86)90253-1

[42] Grimm, V., Quispel, G.: Geometric integration methods that preserve Lyapunov functions. BIT Numerical Mathematics **45**(4), 709–723 (2005) https://doi.org/10.1007/s10543-005-0034-z

[43] Calvo, M., Hernández-Abreu, D., Montijano, J.I., Rández, L.: On the preservation of invariants by explicit Runge-Kutta methods. SIAM Journal on Scientific Computing **28**(3), 868–885 (2006) https://doi.org/10.1137/04061979X

[44] Calvo, M., Laburta, M., Montijano, J.I., Rández, L.: Projection methods preserving Lyapunov functions. BIT Numerical Mathematics **50**(2), 223–241 (2010) https://doi.org/10.1007/s10543-010-0259-3

[45] Laburta, M., Montijano, J.I., Rández, L., Calvo, M.: Numerical methods for non conservative perturbations of conservative problems. Computer Physics Communications **187**, 72–82 (2015) https://doi.org/10.1016/j.cpc.2014.10.012

[46] Ketcheson, D.I.: Relaxation Runge-Kutta methods: Conservation and stability

for inner-product norms. SIAM Journal on Numerical Analysis **57**(6), 2850–2870 (2019) https://doi.org/10.1137/19M1263662 1905.09847

[47] Ranocha, H., Sayyari, M., Dalcin, L., Parsani, M., Ketcheson, D.I.: Relaxation Runge-Kutta methods: Fully-discrete explicit entropy-stable schemes for the compressible Euler and Navier-Stokes equations. SIAM Journal on Scientific Computing **42**(2), 612–638 (2020) https://doi.org/10.1137/19M1263480 1905.09129

[48] Ranocha, H., Lóczi, L., Ketcheson, D.I.: General relaxation methods for initial-value problems with application to multistep schemes. Numer. Math. **146**(4), 875–906 (2020) https://doi.org/10.1007/s00211-020-01158-4

[49] Sanz-Serna, J.M.: An explicit finite-difference scheme with exact conservation properties. Journal of Computational Physics **47**(2), 199–210 (1982) https://doi.org/10.1016/0021-9991(82)90074-2

[50] Dekker, K., Verwer, J.G.: Stability of Runge-Kutta Methods for Stiff Nonlinear Differential Equations. CWI Monographs, vol. 2. North-Holland, Amsterdam (1984)

[51] Izgin, T.: A boot-strapping technique to design dense output formulae for modified patankar-runge-kutta methods. https://arxiv.org/abs/2406.16718 (2024) arXiv:2406.16718 [math.NA]

[52] Crouzeix, M.: Une méthode multipas implicite-explicite pour l'approximation des équations d'évolution paraboliques. Numer. Math. **35**(3), 257–276 (1980) https://doi.org/10.1007/BF01396412

[53] Ascher, U.M., Ruuth, S.J., Spiteri, R.J.: Implicit-explicit runge-kutta methods for time-dependent partial differential equations. Applied Numerical Mathematics **25**(2), 151–167 (1997) https://doi.org/10.1016/S0168-9274(97)00056-1 . Special Issue on Time Integration

[54] Sandu, A., Günther, M.: A generalized-structure approach to additive Runge-Kutta methods. SIAM J. Numer. Anal. **53**(1), 17–42 (2015) https://doi.org/10.1137/130943224

[55] Izgin, T., Ranocha, H.: Using bayesian optimization to design time step size controllers with application to modified patankar–runge–kutta methods. https://arxiv.org/abs/2312.01796 (2023) arXiv:2312.01796

[56] Torlo, D., Öffner, P., Ranocha, H.: Issues with positivity-preserving Patankar-type schemes. Appl. Numer. Math. **182**, 117–147 (2022) https://doi.org/10.1016/j.apnum.2022.07.014

[57] Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press,

Cambridge, UK (2004)

[58] Kopecz, S., Meister, A.: On the existence of three-stage third-order modified Patankar-Runge-Kutta schemes. Numer. Algorithms **81**(4), 1473–1484 (2019) https://doi.org/10.1007/s11075-019-00680-3

[59] Calvo, M., Laburta, M.P., Montijano, J.I., Rández, L.: Projection methods preserving Lyapunov functions. BIT **50**(2), 223–241 (2010) https://doi.org/10.1007/s10543-010-0259-3

[60] Izgin, T., Ranocha, H., Shu, C.-W.: Reproducibility repository for "A Positivity-Preserving Relaxation Algorithm". https://github.com/IzginThomas/PositiveRelaxation (2026). https://doi.org/10.5281/zenodo.19386973

[61] Cano, B., Sanz-Serna, J.M.: Error growth in the numerical integration of periodic orbits, with application to Hamiltonian and reversible systems. SIAM Journal on Numerical Analysis **34**(4), 1391–1417 (1997) https://doi.org/10.1137/S0036142995281152

[62] Cano, B., Sanz-Serna, J.M.: Error growth in the numerical integration of periodic orbits by multistep methods, with application to reversible systems. IMA Journal of Numerical Analysis **18**(1), 57–75 (1998) https://doi.org/10.1093/imanum/18.1.57

[63] Calvo, M., Laburta, M., Montijano, J.I., Rández, L.: Error growth in the numerical integration of periodic orbits. Mathematics and Computers in Simulation **81**(12), 2646–2661 (2011) https://doi.org/10.1016/j.matcom.2011.05.007

[64] Tadmor, E.: Entropy stability theory for difference approximations of nonlinear conservation laws and related time-dependent problems. Acta Numerica **12**, 451–512 (2003) https://doi.org/10.1017/S0962492902000156

[65] Ranocha, H., Gassner, G.J.: Preventing pressure oscillations does not fix local linear stability issues of entropy-based split-form high-order schemes. Communications on Applied Mathematics and Computation (2021) https://doi.org/10.1007/s42967-021-00148-z 2009.13139

[66] Winters, A.R., Czernik, C., Schily, M.B., Gassner, G.J.: Entropy stable numerical approximations for the isothermal and polytropic Euler equations. BIT Numerical Mathematics **60**(3), 791–824 (2020) https://doi.org/10.1007/s10543-019-00789-w

[67] Tadmor, E.: The numerical viscosity of entropy stable schemes for systems of conservation laws. I. Mathematics of Computation **49**(179), 91–103 (1987) https://doi.org/10.1090/S0025-5718-1987-0890255-3

[68] Fisher, T.C., Carpenter, M.H., Nordström, J., Yamaleev, N.K., Swanson, C.:

Discretely conservative finite-difference formulations for nonlinear conservation laws in split form: Theory and boundary conditions. Journal of Computational Physics **234**, 353–375 (2013) https://doi.org/10.1016/j.jcp.2012.09.026

[69] Ranocha, H.: Comparison of some entropy conservative numerical fluxes for the Euler equations. Journal of Scientific Computing **76**(1), 216–242 (2018) https://doi.org/10.1007/s10915-017-0618-1 1701.02264

[70] Chen, T., Shu, C.-W.: Entropy stable high order discontinuous Galerkin methods with suitable quadrature rules for hyperbolic conservation laws. Journal of Computational Physics **345**, 427–461 (2017) https://doi.org/10.1016/j.jcp.2017.05.025

[71] Izgin, T., Meister, A., Shu, C.-W., Torlo, D.: Flux-balanced patankar-type schemes for the compressible euler equations. arXiv preprint arXiv:2602.14392 (2026). Available at arXiv:2602.14392 [math.NA]

[72] Boscarino, S.: High-order semi-implicit schemes for evolutionary partial differential equations with higher order derivatives. J. Sci. Comput. **96**(1), 11–31 (2023) https://doi.org/10.1007/s10915-023-02235-0

[73] Barenblatt, G.I.: On self-similar motions of a compressible fluid in a porous medium. Akad. Nauk SSSR. Prikl. Mat. Meh. **16**, 679–698 (1952)

[74] Mattsson, K.: Summation by parts operators for finite difference approximations of second-derivatives with variable coefficients. J. Sci. Comput. **51**(3), 650–682 (2012) https://doi.org/10.1007/s10915-011-9525-z

[75] Ranocha, H.: Mimetic properties of difference operators: Product and chain rules as for functions of bounded variation and entropy stability of second derivatives. BIT Numerical Mathematics **59**(2), 547–563 (2019) https://doi.org/10.1007/s10543-018-0736-7 1805.09126