MATHEMATICS OF COMPUTATION
Volume 86, Number 304, March 2017, Pages 747–769
http://dx.doi.org/10.1090/mcom/3115
Article electronically published on June 2, 2016

EXPLICIT STRONG STABILITY PRESERVING MULTISTEP RUNGE–KUTTA METHODS

CHRISTOPHER BRESTEN, SIGAL GOTTLIEB, ZACHARY GRANT, DANIEL HIGGS, DAVID I. KETCHESON, AND ADRIAN NÉMETH

Abstract. High-order spatial discretizations of hyperbolic PDEs are often designed to have strong stability properties, such as monotonicity. We study explicit multistep Runge–Kutta strong stability preserving (SSP) time integration methods for use with such discretizations. We prove an upper bound on the SSP coefficient of explicit multistep Runge–Kutta methods of order two and above. Numerical optimization is used to find optimized explicit methods of up to five steps, eight stages, and tenth order. These methods are tested on the linear advection and nonlinear Buckley-Leverett equations, and the results for the observed total variation diminishing and/or positivity preserving time-step are presented.

1. Introduction

The present work is motivated by the numerical solution of hyperbolic conservation laws, which in one dimension take the form $U_t + f(U)_x = 0$. Their solution is complicated by the fact that the exact solutions may develop discontinuities. For this reason, significant effort has been expended on finding spatial discretizations that can handle discontinuities [12]. Once the spatial derivative is discretized, we obtain the system of ODEs

$$u_t = F(u), \tag{1}$$

where u is a vector of approximations to U: $u_j \approx U(x_j)$. This system of ODEs can then be evolved in time using standard methods. The spatial discretizations used to approximate $f(U)_x$ are carefully designed so that when (1) is evolved in time using the forward Euler method the solution at time u^n satisfies the strong stability property

$$\|u^n + \Delta t F(u^n)\| \leq \|u^n\| \quad \text{under the step size restriction } 0 \leq \Delta t \leq \Delta t_{\text{FE}}. \tag{2}$$

Here and throughout, $\|\cdot\|$ represents a norm, semi-norm, or convex functional, determined by the design of the spatial discretization. For example, if the spatial discretization is total variation diminishing (TVD), then we would like (2) to hold with respect to the TVD semi-norm. If the spatial discretization is positivity

Received by the editor September 3, 2014 and, in revised form, July 2, 2015 and September 18, 2015.

2010 *Mathematics Subject Classification.* Primary 65M20.

This research was supported by AFOSR grant number FA-9550-12-1-0224 and KAUST grant FIC/2010/05.

preserving, we would like to have (2) with, for instance,

$$\|u\| = \begin{cases} 0 & \text{if } u_j \geq 0 \text{ for all } j, \\ -\min_j u_j & \text{otherwise.} \end{cases}$$

We assume that the spatial discretization satisfies the desired property when coupled with the forward Euler time discretization, and we use Δt_{FE} to denote the largest step size under which this condition can be guaranteed. But in practice we want to use a higher-order time integrator rather than forward Euler, while still ensuring that the strong stability property

$$\|u^{n+1}\| \leq \|u^n\| \tag{3}$$

is satisfied.

In [31] it was observed that some Runge–Kutta methods can be decomposed into convex combinations of forward Euler steps, and so any convex functional property satisfied by forward Euler will be *preserved* by these higher-order time discretizations, generally under a different time-step restriction. This approach was used to develop second- and third-order Runge–Kutta methods that preserve the strong stability properties of the spatial discretizations developed in that work. In fact, this approach also guarantees that the intermediate stages in a Runge–Kutta method satisfy the strong stability property as well.

For multistep methods, where the solution value u^{n+1} at time t^{n+1} is computed from previous solution values $u^{n-k+1}, \ldots, u^n$, we say that a k-step numerical method is *strong stability preserving* (SSP) if there exists some $\mathcal{C} > 0$ such that

$$\|u^{n+1}\| \leq \max\left\{\|u^n\|, \|u^{n-1}\|, \ldots, \|u^{n-k+1}\|\right\} \tag{4}$$

for any time-step

$$0 \leq \Delta t \leq \mathcal{C}\Delta t_{\mathrm{FE}}, \tag{5}$$

where Δt_{FE} is the largest possible step size such that (2) holds. An explicit multistep method of the form

$$u^{n+1} = \sum_{i=1}^{k} \left(\alpha_i u^{n+1-i} + \Delta t \beta_i F(u^{n+1-i})\right) \tag{6}$$

has $\sum_{i=1}^{k} \alpha_i = 1$ for consistency, so if all the coefficients are non-negative ($\alpha_i, \beta_i \geq 0$), the method can be written as convex combinations of forward Euler steps:

$$u^{n+1} = \sum_{i=1}^{k} \alpha_i \left(u^{n+1-i} + \frac{\beta_i}{\alpha_i}\Delta t F(u^{n+1-i})\right).$$

Clearly, if the forward Euler condition (2) holds, then the solution obtained by the multistep method (6) is strong stability preserving under the time-step restriction (5) with $\mathcal{C} = \min_i \frac{\alpha_i}{\beta_i}\Delta t_{\mathrm{FE}}$, where if any β_i is equal to zero, the corresponding ratio is considered infinite [31].

The convex combination approach has also been applied to obtain sufficient conditions for strong stability for *implicit* Runge–Kutta methods and *implicit* linear multistep methods. Furthermore, it has been shown that these conditions are not only sufficient but necessary as well [8, 9, 15, 16]. Much research on SSP methods

focuses on finding high-order time discretizations with the largest allowable time-step $\Delta t \leq \mathcal{C}\Delta t_{\mathrm{FE}}$. Our aim is to maximize the *SSP coefficient* $\mathcal{C}$ of the method, relative to the number of function evaluations at each time-step (typically the number of stages of a method). For this purpose we define the *effective SSP coefficient* $\mathcal{C}_{\mathrm{eff}} = \frac{\mathcal{C}}{s}$ where s is the number of stages. This value allows us to compare the efficiency of explicit methods of a given order.

Explicit Runge–Kutta methods with positive SSP coefficient cannot be more than fourth-order accurate [21, 30], while explicit SSP linear multistep methods of high-order accuracy must use very many steps, and therefore impose large storage requirements [12, 23]. These characteristics have led to the design of explicit methods with multiple steps and multiple stages in the search for higher-order SSP methods with large effective SSP coefficients. In [13] Gottlieb et al. considered a class of two-step, two-stage methods. Huang [17] considered two-stage hybrid methods with many steps, and found methods of up to seventh order (with seven steps) with reasonable SSP coefficients. Constantinescu and Sandu [5] found multistep multistage methods with up to four stages and four steps, with a focus on finding SSP methods with order up to four. Multistep Runge–Kutta SSP methods with order as high as twelve have been developed in [26] and numerous similar works by the same authors, using sufficient conditions for monotonicity and focusing on a single set of parameters in each work. Spijker [32] developed a complete theory for strong stability preserving multistep multistage methods and found new second-order and third-order methods with optimal SSP coefficients. In [20], Spijker's theory (including necessary and sufficient conditions for monotonicity) is applied to two-step Runge–Kutta methods to develop two-step multistage explicit methods with optimized SSP coefficients. In the present work we present a general application of the same theory to multistep Runge–Kutta (MSRK) methods with more steps. We determine necessary and sufficient conditions for strong stability preservation and prove sharp upper bounds on $\mathcal{C}$ for second-order methods. We also find and test optimized methods with up to five steps and up to tenth order. The approach we employ ensures that the intermediate stages of each method also satisfy a strong stability property.

In Section 2 we extend the order conditions and SSP conditions from two-step Runge–Kutta methods [20] to MSRK methods with arbitrary numbers of steps and stages. In Section 3 we recall an upper bound on $\mathcal{C}$ for general linear methods of order one and prove a new, sharp upper bound on $\mathcal{C}$ for general linear methods of order two. These bounds are important to our study because the explicit MSRK methods we consider are a subset of the class of general linear methods. In Section 4 we formulate and numerically solve the problem of determining methods with the largest $\mathcal{C}$ for a given order and number of stages and steps. We present the effective SSP coefficients of optimized methods of up to five steps and tenth order, thus surpassing the order-eight barrier established in [20] for two-step methods. Most of the methods we find have higher effective SSP coefficients than methods previously found, though in some cases we had trouble with the optimization subroutines for higher orders. Section 4 concludes with a set of recommended methods. Finally, in Section 5 we explore how well the recommended methods perform in practice on a series of well-established test problems. We highlight the need for higher-order methods and the behavior of these methods in terms of strong stability and positivity preservation.

2. SSP multistep Runge–Kutta methods

In this work we study methods in the class of multistep Runge–Kutta methods with optimal strong stability preservation properties. These multistep Runge–Kutta methods are a simple generalization of Runge–Kutta methods to include the numerical solution at previous steps. These methods are Runge–Kutta methods in the sense that they compute multiple stages based on the initial input; however, they use the previous k solution values $u^{n-k+1}, u^{n-k}, \ldots, u^{n-1}, u^n$ to compute the solution value u^{n+1}.

A class of two-step Runge–Kutta methods was studied in [20]. Here we study the generalization of that class to an arbitrary number of steps:

$$y_1^n = u^n, \tag{7a}$$

$$y_i^n = \sum_{l=1}^{k} d_{il} u^{n-k+l} + \Delta t \sum_{l=1}^{k-1} \hat{a}_{il} F(u^{n-k+l}) + \Delta t \sum_{j=1}^{i-1} a_{ij} F(y_j^n), \quad 2 \le i \le s, \tag{7b}$$

$$u^{n+1} = \sum_{l=1}^{k} \theta_l u^{n-k+l} + \Delta t \sum_{l=1}^{k-1} \hat{b}_l F(u^{n-k+l}) + \Delta t \sum_{j=1}^{s} b_j F(y_j^n). \tag{7c}$$

Here the values u^{n-k+j} denote the previous steps and y_j^n are intermediate stages used to compute the next solution value u^{n+1}. The form (7) is convenient for identifying the computational cost of the method: it is evident that s new function evaluations are needed to progress from u^n to u^{n+1}.

To study the strong stability preserving properties of method (7), we write it in the form [32]

$$\mathbf{w} = \mathbf{S}\mathbf{x} + \Delta t \mathbf{T}\mathbf{f}. \tag{8}$$

To accomplish this, we stack the last k steps into a column vector:

$$\mathbf{x} = \left[u^{n-k+1}; u^{n-k+2}; \ldots; u^{n-1}; u^n\right].$$

We define a column vector of length $k+s$ that contains these steps and the stages:

$$\mathbf{w} = \left[u^{n-k+1}; u^{n-k+2}; \ldots; u^{n-1}; y_1 = u^n; y_2; \ldots; y_s; u^{n+1}\right],$$

and another column vector containing the derivative of each element of $\mathbf{w}$:

$$\mathbf{f} = \left[F\left(u^{n-k+1}\right); F\left(u^{n-k+2}\right); \ldots; F\left(u^{n-1}\right), F\left(y_1\right); \ldots; F\left(y_s\right); F\left(u^{n+1}\right)\right]^{\mathrm{T}}.$$

Here we have used the semi-colon to denote (as in MATLAB) vertical concatenation of vectors. Thus, each of the above is a column vector.

Now the method (7) can be written in the matrix-vector form (8) where the matrices $\mathbf{S}$ and $\mathbf{T}$ are

$$\mathbf{S} = \begin{pmatrix} \mathbf{I}_{(k-1)\times(k-1)} & \mathbf{0}_{1\times(k-1)} \\ \mathbf{D} & \\ & \boldsymbol{\theta}^{\mathrm{T}} \end{pmatrix}, \quad \mathbf{T} = \begin{pmatrix} \mathbf{0} & \mathbf{0} & 0 \\ \hat{\mathbf{A}} & \mathbf{A} & \mathbf{0} \\ \hat{\mathbf{b}}^{\mathrm{T}} & \mathbf{b}^{\mathrm{T}} & 0 \end{pmatrix}. \tag{9}$$

The matrices $\mathbf{D}, \mathbf{A}, \hat{\mathbf{A}}$ and the vectors $\boldsymbol{\theta}, \hat{\mathbf{b}}, \mathbf{b}$ contain the coefficients $d_{il}, \hat{a}_{il}, a_{ij}$ and $\theta_l, \hat{b}_l, b_j$ from (7); note that the first row of $\mathbf{D}$ is $(0, 0, \ldots, 0, 1)$ and the first row of $\mathbf{A}, \hat{\mathbf{A}}$ is identically zero. Consistency requires that

$$\sum_{l=1}^{k} \theta_l = 1, \tag{10a}$$

$$\sum_{l=1}^{k} d_{il} = 1, \qquad 1 \le i \le s, \tag{10b}$$

so $\mathbf{Se} = \mathbf{e}$ where $\mathbf{e}$ is a column vector with all entries equal to unity (see [32, Section 2.1.1]).

In the next two subsections we use representation (8) to study monotonicity properties of the method (7). The results in these subsections are a straightforward generalization of the corresponding results in [20], and so the discussion below is brief and the interested reader is referred to [20] for more detail.

2.1. **A review of the SSP property for multistep Runge–Kutta methods.** To write (8) as a linear combination of forward Euler steps, we add the term $r\mathbf{Tw}$ to both sides of (8), obtaining

$$(\mathbf{I} + r\mathbf{T})\,\mathbf{w} = \mathbf{Sx} + r\mathbf{T}\left(\mathbf{w} + \frac{\Delta t}{r}\mathbf{f}\right).$$

We now left-multiply both sides by $(\mathbf{I} + r\mathbf{T})^{-1}$ (which exists since $\mathbf{T}$ is strictly lower-triangular) to obtain

$$\begin{aligned}\mathbf{w} &= (\mathbf{I} + r\mathbf{T})^{-1}\mathbf{Sx} + r(\mathbf{I} + r\mathbf{T})^{-1}\mathbf{T}\left(\mathbf{w} + \frac{\Delta t}{r}\mathbf{f}\right)\\ &= \mathbf{Rx} + \mathbf{P}\left(\mathbf{w} + \frac{\Delta t}{r}\mathbf{f}\right),\end{aligned} \tag{11}$$

where

$$\mathbf{P} = r(\mathbf{I} + r\mathbf{T})^{-1}\mathbf{T}, \quad \mathbf{R} = (\mathbf{I} + r\mathbf{T})^{-1}\mathbf{S} = (\mathbf{I} - \mathbf{P})\mathbf{S}. \tag{12}$$

In consequence of the consistency condition (10), the row sums of $[\mathbf{R}\ \mathbf{P}]$ are each equal to one:

$$\mathbf{Re} + \mathbf{Pe} = (\mathbf{I} - \mathbf{P})\mathbf{Se} + \mathbf{Pe} = \mathbf{e} - \mathbf{Pe} + \mathbf{Pe} = \mathbf{e}.$$

Thus if $\mathbf{R}$ and $\mathbf{P}$ have no negative entries, then each stage w_i is a convex combination of the inputs x_j and the forward Euler quantities $w_j + (\Delta t/r)F(w_j)$. It is then simple to show (following [32]) that any strong stability property of the forward Euler method is preserved by the method (8) under the time-step restriction $\Delta t \le \mathcal{C}(\mathbf{S}, \mathbf{T})\Delta t_{\mathrm{FE}}$ where $\mathcal{C}(\mathbf{S}, \mathbf{T})$ is defined as

$$\mathcal{C}(\mathbf{S}, \mathbf{T}) = \sup_r \left\{ r : (\mathbf{I} + r\mathbf{T})^{-1} \text{ exists and } \mathbf{P} \ge 0, \mathbf{R} \ge 0 \right\}.$$

Hence the SSP coefficient of method (11) is greater than or equal to $\mathcal{C}(\mathbf{S}, \mathbf{T})$. In fact, following [32, Remark 3.2]) we conclude that if the method is row-irreducible, then the SSP coefficient is, in fact, exactly equal to $\mathcal{C}(\mathbf{S}, \mathbf{T})$. (For the definition of row reducibility, see [32, Remark 3.2]) or [20]).

By applying [32, Theorem 2.2], one finds that method (7) has positive SSP coefficient if and only if the following conditions hold:

$$\mathbf{0} \le \mathbf{D} \le \mathbf{1}, \qquad 0 \le \theta \le 1, \tag{13a}$$

$$\mathbf{A} \ge \mathbf{0}, \qquad \hat{\mathbf{A}} \ge \mathbf{0}, \tag{13b}$$

$$\mathbf{b} \ge \mathbf{0}, \qquad \hat{\mathbf{b}} \ge \mathbf{0}, \tag{13c}$$

$$\mathrm{Inc}(\mathbf{TS}) \le \mathrm{Inc}(\mathbf{S}), \qquad \mathrm{Inc}(T^2) \le \mathrm{Inc}(T), \tag{13d}$$

where $\mathbf{S}, \mathbf{T}$ are defined in (9).

2.2. **Order conditions.** In [20] we derived order conditions for methods of the form (7) with two steps. Other authors had already derived order conditions for more general classes of two-step multistage methods, but our conditions in [20] are simpler. Our conditions extend in a simple way to method (7) with any number of steps. For convenience, we rewrite (7) in the form

$$\mathbf{y}^n = \tilde{\mathbf{D}}\mathbf{u}^n + \Delta t \tilde{\mathbf{A}}\mathbf{f}^n, \tag{14a}$$

$$u^{n+1} = \boldsymbol{\theta}^{\mathrm{T}}\mathbf{u}^n + \Delta t \tilde{\mathbf{b}}^{\mathrm{T}}\mathbf{f}^n, \tag{14b}$$

where

$$\tilde{\mathbf{D}} = \begin{pmatrix} \mathbf{I}_{(k-1)\times(k-1)} & \mathbf{0}_{1\times(k-1)} \\ & \mathbf{D} \end{pmatrix}, \quad \tilde{\mathbf{A}} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \hat{\mathbf{A}} & \mathbf{A} \end{pmatrix}, \quad \tilde{\mathbf{b}} = \begin{pmatrix} \hat{\mathbf{b}} & \mathbf{b} \end{pmatrix}, \tag{15}$$

and $\mathbf{y}^n = [u^{n-k+1}; u^{n-k+2}; \ldots; u^{n-1}; y_1^n; \ldots; y_s^n]$ and $\mathbf{f}^n = F(\mathbf{y}^n)$ are the vector of stage values and stage derivatives, respectively, and $\mathbf{u}^n = [u^{n-k+1}, u^{n-k+2}, \ldots, u^n]$ is the vector of previous step values.

The derivation of the order conditions closely follows Section 3 of [20] with the following changes: (1) the vector $\mathbf{d}$ is replaced by the matrix $\mathbf{D}$; (2) the scalar θ is replaced by the vector $\boldsymbol{\theta}$; and (3) the vector $\boldsymbol{l} = (k-1, k-2, \ldots, 1, 0)^T$ appears in place of the number 1 in the expression for the stage residuals, which are thus:

$$\boldsymbol{\tau}_j = \frac{1}{j!}\left(\mathbf{c}^j - \tilde{\mathbf{D}}(-\boldsymbol{l})^j\right) - \frac{1}{(j-1)!}\tilde{\mathbf{A}}\mathbf{c}^{j-1},$$

$$\tau_j = \frac{1}{j!}\left(1 - \boldsymbol{\theta}^{\mathrm{T}}(-\boldsymbol{l})^j\right) - \frac{1}{(j-1)!}\tilde{\mathbf{b}}^{\mathrm{T}}\mathbf{c}^{j-1},$$

where $\mathbf{c} = \tilde{\mathbf{A}}\mathbf{e} - \tilde{\mathbf{D}}\boldsymbol{l}$ and exponents are to be interpreted element-wise. The derivation of the order conditions is identical to that in [20] except for these changes.

A method is said to have stage order q if $\boldsymbol{\tau}_j$ and τ_j vanish for all $j \le q$. A method is said to be DJ-reducible if it includes one or more stages whose value does not affect u^{n+1}. The following result is an extension of Theorem 2 in [20].

Theorem 1. *Any DJ-irreducible MSRK method* (7) *of order p with positive SSP coefficient has stage order at least $\lfloor \frac{p-1}{2} \rfloor$.*

Proof. Let a DJ-irreducible method of order p with positive SSP coefficient be given. We show first that the weights $\mathbf{b}$ are strictly positive and then that this implies the satisfaction of the stage order conditions.

From (13c) we have already that $\mathbf{b} \geq 0$. Now suppose that $b_j = 0$ for some j. Then the second part of (13d) implies that

$$\sum_i b_i a_{ij} = 0.$$

Since also $a_{ij} \geq 0$ (see (13b)), this implies that either b_i or a_{ij} vanishes for each value of i. This would imply that the method is DJ-reducible. Therefore, we have $\mathbf{b} > 0$ for any SSP method.

Now we show that the stage order conditions are satisfied up to order $\lfloor \frac{p-1}{2} \rfloor$. First, notice that the first $k-1$ components of $\boldsymbol{\tau}_j$ vanish identically (reflecting the fact that these "stages" correspond to previous step values) and that τ_j vanishes for $1 \leq j \leq p$ (since the method has order p). Next, observe that the order conditions for order p include the condition

$$\tilde{\mathbf{b}}^T \boldsymbol{\tau}_k^2 = 0, \qquad k = 1, 2, \ldots, \left\lfloor \frac{p-1}{2} \right\rfloor.$$

This implies that

$$\boldsymbol{\tau}_k^2 = 0, \qquad k = 1, 2, \ldots, \left\lfloor \frac{p-1}{2} \right\rfloor.$$

□

For RK and two-step RK methods, this relation between order and stage order leads to an order barrier for explicit SSP methods. For MSRK methods with three steps, we have not yet been able to deduce such an order barrier; as we will see, at least order ten is possible.

Note that the approach used in [20], which is based on the work of Albrecht [1], produces a set of order conditions that are *equivalent* to the set of conditions derived using B-series. However, the two sets have different equations. Albrecht's approach has two advantages over that based on B-series in the present context. First, it leads to algebraically simpler conditions that are almost identical in appearance to those for one-step RK methods. Second, it leads to conditions in which the residuals $\boldsymbol{\tau}_j$ appear explicitly. As a result, very many of the order conditions are *a priori* satisfied by methods with high stage order, due to Theorem 1. This simplifies the numerical optimization problem that is formulated in Section 4.

3. Upper bounds on the SSP coefficient

In this section we present upper bounds on the SSP coefficient of general linear methods of first- and second-order. These upper bounds apply to all explicit multistep multistage methods, not just those of form (7). They are obtained by considering a relaxed optimization problem. Specifically, we consider monotonicity and order conditions for methods applied to linear problems only.

Given a function $\psi : \mathbb{R} \to \mathbb{R}$, let $R(\psi)$ denote the *radius of absolute monotonicity*:

$$R(\psi) = \sup\{r \geq 0 \;\mid\; \psi^{(j)}(z) \text{ exists and is non-negative for all } z \in [-r, 0] \text{ and all } j = 0, 1, 2, \ldots\}. \tag{16}$$

Here the $\psi^{(j)}(z)$ denotes the jth derivative of ψ at z. Any explicit k-step, s-stage general linear method applied to the linear, scalar ODE $u'(t) = \lambda u$ results in an iteration of the form

$$u^{n+1} = \psi_1(z)u^n + \psi_2(z)u^{n-1} + \cdots + \psi_k(z)u^{n-k+1}, \tag{17}$$

where $z = \Delta t\lambda$ and $\{\psi_1, \ldots, \psi_k\}$ are polynomials of degree at most s. The method is strong stability preserving for linear problems under the step size restriction $\Delta t \le \overline{R}(\psi_1, \ldots, \psi_k)\Delta t_{\text{FE}}$ where

$$\overline{R}(\psi_1, \ldots, \psi_k) = \min_i R(\psi_i). \tag{18}$$

The constant $\overline{R}(\psi_1, \ldots, \psi_k)$ is commonly referred to as the *threshold factor* [33]. We also refer to the *optimal threshold factor*

$$R_{s,k,p} = \sup\left\{\overline{R}(\psi_1, \ldots, \psi_k) \;\middle|\; (\psi_1, \ldots, \psi_k) \in \Pi_{s,k,p}\right\} \tag{19}$$

where $\Pi_{s,k,p}$ denotes the set of all stability functions of k-step, s-stage methods satisfying the (linear) order conditions up to order p. Clearly the SSP coefficient of any s-stage, k-step, order p MSRK method is no greater than the corresponding $R_{s,k,p}$. Optimal values of $R_{s,k,p}$ are given in [19].

The following result is proved in Section 2.3 of [11].

Theorem 2. *The threshold factor of a first-order accurate explicit s-stage general linear method is at most s.*

Methods consisting of s iterated forward Euler steps achieve this bound (with both threshold factor and SSP coefficient equal to s). Clearly it provides an upper bound on the threshold factor and SSP coefficient also for methods of higher order. For second-order methods, a tighter bound is given in the next theorem. We will see in Section 4 that it is sharp, even over the smaller class of MSRK methods.

Theorem 3. *For any $s > 0, k > 1$ the optimal threshold factor for explicit s-stage, k-step, second-order general linear methods is*

$$R_2 := \frac{(k-2)s + \sqrt{(k-2)^2s^2 + 4s(s-1)(k-1)}}{2(k-1)}. \tag{20}$$

This is attained by taking

$$\psi_1(z) = \frac{kR_2}{s - R_2 + kR_2}\left(1 + \frac{z}{R_2}\right)^s,$$
$$\psi_k(z) = \frac{s - R_2}{s - R_2 + kR_2},$$

and letting all other polynomials ψ_i vanish.

Proof. It is convenient to write the stability polynomials in the form

$$\psi_i(z) = \sum_{j=0}^{s} \gamma_{ij}\left(1 + \frac{z}{r}\right)^j \tag{21}$$

where we assume $r \in [0, \overline{R}(\psi_1, \ldots, \psi_k)]$, which implies

$$\gamma_{ij} \ge 0. \tag{22}$$

By using Taylor series in (17), one finds that the conditions for second-order accuracy are:

$$\sum_{i=1}^{k}\sum_{j=0}^{s}\gamma_{ij}=1, \tag{23a}$$

$$\sum_{i=1}^{k}\sum_{j=0}^{s}\gamma_{ij}(j+(k-i)r)=kr, \tag{23b}$$

$$\sum_{i=1}^{k}\sum_{j=0}^{s}\gamma_{ij}((k-i)^2r^2+2(k-i)jr+j(j-1))=k^2r^2. \tag{23c}$$

The claimed value of the optimal threshold factor (20) is achieved by taking

$$\begin{aligned}&\gamma_{1s}=\frac{kR_2}{s-R_2+kR_2},\quad \gamma_{k0}=\frac{s-R_2}{s-R_2+kR_2},\\ &\gamma_{ij}=0 \text{ for all } (i,j)\notin\{(1,s),(k,0)\},\end{aligned} \tag{24}$$

which satisfy conditions (23a)–(23c) with $r=R_2$ and also satisfy condition (22) since $R_2<s$. Thus R_2 serves as a lower bound on the optimal threshold factor. We now prove that R_2 is also an upper bound on the threshold factor.

Suppose that some coefficients γ satisfy conditions (22) and (23) for some positive $r=R$. We show that $R\le R_2$. If $R\le\frac{s-1}{k}$, then $R\le R_2$. So from now on we assume $R>\frac{s-1}{k}$.

Multiply (23b) by kr and subtract (23c) from the result to obtain

$$\sum_{i=1}^{k}\sum_{j=0}^{s}\gamma_{ij}(i(k-i)r^2-(k-2i)jr-j(j-1))=0, \tag{25}$$

which is a quadratic equation for r. We denote its coefficients by:

$$a(\gamma)=+\sum_{i=1}^{k}\sum_{j=0}^{s}\gamma_{ij}i(k-i), \tag{26a}$$

$$b(\gamma)=-\sum_{i=1}^{k}\sum_{j=0}^{s}\gamma_{ij}(k-2i)j, \tag{26b}$$

$$c(\gamma)=-\sum_{i=1}^{k}\sum_{j=0}^{s}\gamma_{ij}j(j-1). \tag{26c}$$

From (22) and the definitions of $a(\gamma)$ and $c(\gamma)$ we have that $a(\gamma)\ge0$ and $c(\gamma)\le0$. We show that $a(\gamma)>0$. Suppose to the contrary that $a(\gamma)=0$. Then we have $\gamma_{ij}=0$ for all $i\ne k$, and thus (25) simplifies to

$$\sum_{j=0}^{s}\gamma_{kj}j(kR-(j-1))=0. \tag{27}$$

It follows that $\gamma_{kj}=0$ for all $j\ne0$, since $kR-(j-1)>0$, but this contradicts (23b). Hence $a(\gamma)>0$.

Since $a(\gamma) > 0$ and $c(\gamma) \leq 0$, the largest root of (25) is a non-negative number that can be expressed as

$$r(\gamma) = \frac{-b(\gamma)}{2a(\gamma)} + \sqrt{\left(\frac{-b(\gamma)}{2a(\gamma)}\right)^2 + \frac{-c(\gamma)}{a(\gamma)}}. \tag{28}$$

Indeed, since $r(\gamma)$ is the only non-negative root of (25), we have $r(\gamma) = R$. We now show that $r(\gamma) \leq R_2$.

Let new coefficients γ^* be given by

$$\gamma^*_{ij} := \gamma_{ij} \qquad \text{for } i \neq k, \tag{29a}$$

$$\gamma^*_{kj} := 0 \qquad \text{for all } 0 \leq j \leq s, \tag{29b}$$

and then renormalize γ^* so that (23a) holds. Then coefficients γ^* satisfy condition (22), $a(\gamma^*) > 0$ and $c(\gamma^*) \leq 0$. Observe that $r(\gamma^*) \geq r(\gamma)$ since differentiating $r(\gamma)$ with respect to γ_{kj} yields

$$\frac{\partial}{\partial \gamma_{kj}} r(\gamma) = \frac{-kj}{2a(\gamma)} + \frac{2b(\gamma)kj + 4a(\gamma)j(j-1)}{4a(\gamma)\sqrt{b(\gamma)^2 - 4a(\gamma)c(\gamma)}} = \frac{-r(\gamma)kj + j(j-1)}{\sqrt{b(\gamma)^2 - 4a(\gamma)c(\gamma)}}, \tag{30}$$

which is zero for $j = 0$ and negative for all $j \neq 0$ when we have $\frac{s-1}{k} < r(\gamma)$.

Next let

$$\gamma^{**}_{ij} := \gamma^*_{ij} + \gamma^*_{k-i,j} \qquad \text{for all } 1 \leq i < \tfrac{k}{2}, \tag{31a}$$

$$\gamma^{**}_{ij} := \gamma^*_{ij} \qquad \text{for } i = \tfrac{k}{2}, \tag{31b}$$

$$\gamma^{**}_{ij} := 0 \qquad \text{for all } \tfrac{k}{2} < i \leq k. \tag{31c}$$

Clearly these γ^{**} satisfy conditions (22) and (23a); furthermore we have $a(\gamma^{**}) = a(\gamma^*)$, $c(\gamma^{**}) = c(\gamma^*)$ and $-b(\gamma^{**}) \geq |b(\gamma^*)|$. Hence $r(\gamma^{**}) \geq r(\gamma^*)$.

Finally, let γ^{***} be given by

$$\gamma^{***}_{1s} := 1, \tag{32a}$$

$$\gamma^{***}_{ij} := 0 \qquad \text{for all } (i,j) \neq (1,s). \tag{32b}$$

On the one hand we have that $r(\gamma^{***}) = R_2$. On the other hand since coefficients γ^{**} satisfy (22) and (23a), and since $\gamma^{**}_{kj} = 0$ for all j, we have $0 < a(\gamma^{***}) \leq a(\gamma^{**})$ and $-b(\gamma^{***}) \geq -b(\gamma^{**})$ and $-c(\gamma^{***}) \geq -c(\gamma^{**})$. Together with $-b(\gamma^{**}) \geq 0$ these imply that $r(\gamma^{***}) \geq r(\gamma^{**})$. Thus we have

$$R = r(\gamma) \leq r(\gamma^*) \leq r(\gamma^{**}) \leq r(\gamma^{***}) = R_2.$$

So far, we have shown that the optimal second-order threshold factor is given by (20). We easily check that if $R_2 > 0$ but $\gamma_{ij} \neq 0$ for some $(i,j) \notin \{(1,s),(k,0)\}$, then at least one of the above inequalities is sharp, and thus $R < R_2$. It follows that for $R_2 > 0$ the optimal threshold factor is uniquely achieved by (24). □

4. Optimized explicit MSRK methods

In this section we present an optimization problem for finding MSRK methods with the largest possible SSP coefficient. This optimization problem is implemented in a MATLAB code and solved using the `fmincon` function for optimization (code is available at our website [10]). This implementation recovers the known optimal methods of first- and second-order mentioned above. For high-order methods with large numbers of stages and steps, numerical solution of the optimization problem is difficult due to the number of coefficients and constraints. Despite the extensive numerical optimization searches, we do not claim that all of the methods found are truly optimal; we refer to them only as *optimized*. The second-order methods are known to be optimal because they achieve the upper bounds presented in Section 3. Other methods are known to be optimal because they are equivalent to those found by certified computations in earlier work [5].

In Section 4.2 we present the effective SSP coefficients of the optimized methods. The coefficients $d_{il}, \hat{a}_{il}, a_{ij}, \theta_l, \hat{b}_l$ and b_j can be downloaded (as MATLAB files) from [10]. The SSP coefficients of methods known to be optimal are printed in boldface in the corresponding tables. The coefficients of methods that are known not to be optimal (e.g. when better methods have been found in the literature) are printed in the table in a light grey. We chose to include these to show the issues with the performance of the optimizer. We discuss these issues in the relevant sections below.

A major issue in the implementation and the performance of the optimized time integrators is the choice of starting methods to obtain the initial k step values. Typically exact values are not available, and we recommend the use of many small steps of a lower order SSP method to generate the starting values. A discussion of starting procedures appears in [20].

4.1. The optimization problem.

Based on the results above, the problem of finding optimal SSP multistep Runge–Kutta methods can be formulated algebraically. We wish to find coefficients $\mathbf{S}$ and $\mathbf{T}$ (corresponding to (9)) that maximize the value of r subject to the following conditions:

(1) $r(\mathbf{I} + r\mathbf{T})^{-1}\mathbf{T} \geq 0$ and $(\mathbf{I} + r\mathbf{T})^{-1}\mathbf{S} \geq 0$, where the inequalities are understood component-wise.
(2) $\mathbf{S}$ and $\mathbf{T}$ satisfy the relevant order conditions.

This is a non-convex, nonlinear constrained optimization problem in many variables.

This problem was used to formulate a MATLAB optimization code that uses `fmincon`. We ran this extensively, and when needed used methods with a lower number of steps as starting values. We note that for a large number of coefficients and constraints, this optimization process was slow and seemed to get stuck in local minima.

4.2. Effective SSP coefficients of the optimized methods.

We now discuss the optimized SSP coefficients among methods with prescribed order, number of stages, and number of steps. For a given order, the SSP coefficient is larger for methods with more stages, and usually the effective SSP coefficient is also larger. Comparing optimized SSP coefficients among classes of methods with the same

number of stages and order but different number of steps, we see the following behavior:

- For methods of even order, the SSP coefficient increases monotonically with k, and the marginal increase from k to $k+1$ is smaller for larger k.
- For methods of odd order up to five, for a large enough number of stages there exists k_0 such that optimized methods never use more than k_0 steps (hence the optimized SSP coefficient remains the same as the allowed number of steps is increased beyond k_0). The value of k_0 depends on the order and number of stages.

  We observed two variants of this behavior where the optimization routine converged to a method with fewer steps than allowed or when additional steps do not lead to better SSP coefficients. In either case, the optimal methods have a smaller number of steps than one would naively expect from a glance at the table. To avoid this confusion, we denote such methods with an asterisk on the effective SSP coefficient.

This behavior seems to generalize that seen for multistep methods [23]. The behavior described for odd orders is observed here up to order five. Since the value of k_0 increases with p, we expect that a study including larger k values would show the same behavior for optimized methods of higher (odd) order as well. Overall, the effective SSP coefficient tends to increase more quickly with the number of stages than with the number of steps. Where relevant, we compare the methods we found to those of [5, 17, 24, 25, 28].

4.2.1. *Second-order methods.* We have already given a characterization of the optimal second-order methods in Theorem 3 above. These methods were first found by numerical optimization; observation of their structure led to Theorem 3. Let $Q = 2(k-1)R_{s,k,2}$ where $R_{s,k,2}$ is given in (20)

$$R_{s,k,2} = \frac{(k-2)s + \sqrt{(k-2)^2 s^2 + 4s(s-1)(k-1)}}{2(k-1)}.$$

The non-zero coefficients of these methods are (compare (24)):

$$\begin{aligned} d_{ik} &= 1, && 1 \le i \le s, \\ b_j = \beta &:= \frac{kQ}{s(k-1)\left(2(s-1)+Q\right)}, && 1 \le j \le s, \\ a_{ij} &= \frac{1}{R_{s,k,2}}, && 1 \le j < i \le s, \\ \theta_k &= \frac{k-\beta s}{k-1}, && \theta_1 = 1 - \theta_k. \end{aligned}$$

The SSP coefficients of these methods, which are known exactly, provide an upper bound on the SSP coefficient for higher-order methods.

4.2.2. *Third-order methods.* The effective SSP coefficients of optimized third-order methods are shown in Table 1 and plotted in Figure 1(a). All methods with four or more stages turn out to be two-step methods (i.e., $k_0 = 2$ for this case). These methods are denoted with an asterisk. For $s = 3$, there is no advantage to increasing the number of steps beyond $k_0 = 3$, and for $s = 2$, $k_0 = 4$. Note that although

we report only values up to five steps, this pattern was verified up to eight steps. All methods up to $k = 4, s = 4$ are optimal (to two decimal places) according to results of [5], and the $\mathcal{C}_{\text{eff}}$ values for $(s, k) = (2, 2), (3, 2), (2, 3)$ are provably optimal because they achieve the optimal values $R_{s,k,3}$ given in [19].

TABLE 1. $\mathcal{C}_{\text{eff}}$ for third-order methods

$s\backslash k$	2	3	4	5
2	**0.36603**	**0.55643**	**0.57475***	0.57475*
3	**0.55019**	**0.57834***	**0.57834***	0.57834*
4	**0.57567**	**0.57567***	**0.57567***	0.57567*
5	0.59758	0.59758*	0.59758*	0.59758*
6	0.62946	0.62946*	0.62946*	0.62946*
7	0.64051	0.64051*	0.64051*	0.64051*
8	0.65284	0.65284*	0.65284*	0.65284*
9	0.67220	0.67220*	0.67220*	0.67220*
10	0.68274	0.68274*	0.68274*	0.68274*

TABLE 2. $\mathcal{C}_{\text{eff}}$ for fourth-order methods

$s\backslash k$	2	3	4	5
2	–	**0.24767**	**0.34085**	0.39640
3	**0.28628**	**0.38794**	**0.45515**	0.48741
4	**0.39816**	**0.46087**	**0.48318**	0.49478
5	0.47209	0.50419	0.50905	0.51221
6	0.50932	0.51214	0.51425	0.51550
7	0.53436	0.53552	0.53610	0.53646
8	0.56151	0.56250	0.56317	0.56362
9	0.58561	0.58690	0.58871	0.58927
10	0.61039	0.61415	0.61486	0.61532

4.2.3. *Fourth-order methods.* Effective coefficients are given in Figure 1(b) and Table 2. All methods up to $k = 4, s = 4$ are optimal (to two decimal places) according to the certified optimization performed in [5]. The $(2, 5, 4)$ method we found has an SSP coefficient that matches that of [17].

4.2.4. *Fifth-order methods.* The effective SSP coefficients of the fifth-order methods are displayed in Figure 1(c) and Table 3. For lower number of stages ($s = 2, \ldots, 5$) the effective SSP coefficients increase with increasing number of steps. However, for $s = 6$ and above, we observe that once k gets large enough, we do not necessarily get an improvement in the SSP coefficient by adding more steps. These methods are denoted with an asterisk.

Although the optimized SSP coefficient is a strictly increasing function of the number of stages, in some cases the effective SSP coefficient decreases. Our $(s, k) = (2, 4)$ and $(s, k) = (2, 5)$ methods have effective SSP coefficients that match the ones in [17]. Our $(s, k) = (8, 2), (3, 3), (3, 4), (3, 5)$, and $(7, 3)$ methods have effective SSP coefficients that match those in [24, 25, 28].

TABLE 3. $\mathcal{C}_{\text{eff}}$ for fifth-order methods

$s\backslash k$	2	3	4	5
2	–	–	0.18556	0.26143
3	–	0.21267	0.33364	0.38735
4	0.21354	0.34158	0.38436	0.39067
5	0.32962	0.38524	0.40054	0.40461
6	0.38489	0.40386	0.40456	0.40456*
7	0.41826	0.42619	0.42619*	0.42619*
8	0.44743	0.44743*	0.44743*	0.44743*
9	0.43794	0.43806	0.43806*	0.43806*
10	0.42544	0.43056	0.43098	0.43098*

TABLE 4. $\mathcal{C}_{\text{eff}}$ for sixth-order methods

$s\backslash k$	2	3	4	5
2	–	–	–	0.10451
3	–	0.00971	0.11192	0.21889
4	–	0.17924	0.27118	0.31639
5	–	0.27216	0.32746	0.34142
6	0.09928	0.32302	0.33623	0.34453
7	0.18171	0.34129	0.34899	0.35226
8	0.24230	0.33951	0.34470	0.34680
9	0.28696	0.34937	0.34977	0.35033
10	0.31992	0.35422	0.35643	0.35665

4.2.5. *Sixth-order methods.* Effective SSP coefficients of optimized sixth-order methods are given in Figure 1(d) and Table 4. Once again, the effective SSP coefficient occasionally decreases with increasing stage number. Our $(s, k) = (2, 5)$ method has an effective SSP coefficient that matches the one in [17], and our values for $(s, k) = (8, 3), (8, 4)$, and $(8, 5)$ improve upon the values obtained in [28]. Our values for $(s, k) = (7, 3), (7, 4)$ match those of [25], and our $(s, k) = (7, 5)$ value improves on that in [25]. The $(s, k) = (3, 4), (3, 5)$ values illustrate the challenges of using our general numerical optimization formulation for this problem: we were not able to match the methods in [24] from a "cold start" (with random initial guesses). However, converting their methods to our form we were able to replicate their results while tightening the optimizer parameters `TolCon, TolFun` and `TolX` from 10^{-12} in their work to 10^{-14}. This suggests that the approach used in [24] which focuses on one set of parameters at a time may make the optimization problem more manageable. However, this same approach was used in [25] and led to an $(s, k) = (7, 5)$ method that had a smaller SSP coefficient than that found with our approach.

4.2.6. *Seventh-order methods.* Coefficients are given in Figure 1(e) and Table 5. Compared to the seven-step two-stage method in [17], which has $\mathcal{C} = 0.234$ and $\mathcal{C}_{\text{eff}} = 0.117$, our five-step methods with $s \geq 3$, four-step with $k \geq 5$, three-step with $k \geq 6$ and two-step with $k \geq 9$ all have larger effective SSP coefficient. Our $(7, 4), (7, 5), (3, 5)$ methods have SSP coefficients that match those in [25] and [24], while our $(7, 3)$ and $(8, 3), (8, 4), (8, 5)$ have larger SSP coefficients that those in [25] and [28].

TABLE 5. $\mathcal{C}_{\text{eff}}$ for seventh-order methods

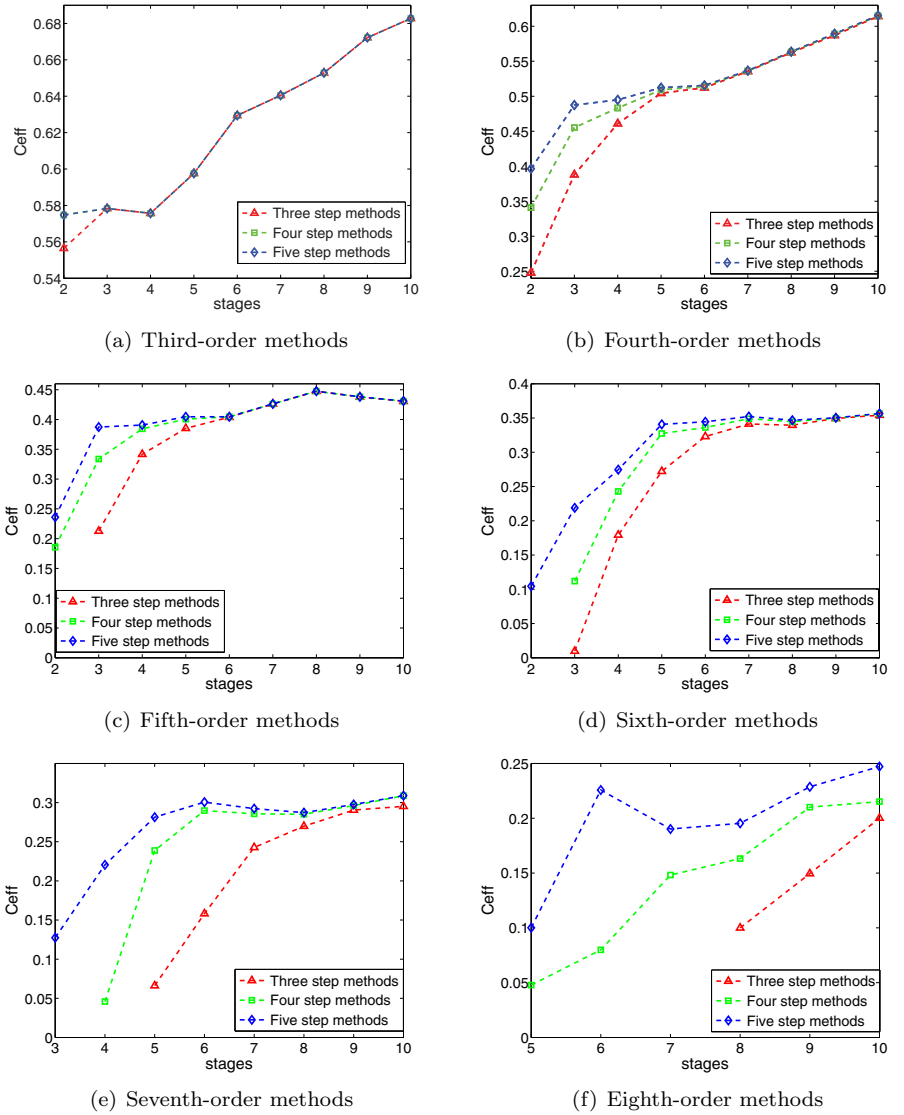
$s\backslash k$	2	3	4	5
2	–	–	–	–
3	–	–	–	0.12735
4	–	–	0.04584	0.22049
5	–	0.06611	0.23887	0.28137
6	–	0.15811	0.28980	0.30063
7	–	0.24269	0.28562	0.29235
8	–	0.26988	0.28517	0.28715
9	0.12444	0.29046	0.29616	0.29759
10	0.17857	0.29522	0.30876	0.30886

TABLE 6. $\mathcal{C}_{\text{eff}}$ for eighth-order methods

$s\backslash k$	2	3	4	5
2	–	–	–	–
3	–	–	–	–
4	–	–	–	–
5	–	–	0.04781	0.10007
6	–	–	0.07991	0.22574
7	–	–	0.14818	0.22229
8	–	0.09992	0.16323	0.19538
9	–	0.14948	0.21012	0.23826
10	–	0.20012	0.21517	0.24719

4.2.7. *Eighth-order methods.* Explicit eighth-order two-step RK methods found in [20] require at least 11 stages and have $\mathcal{C}_{\text{eff}} \leq 0.078$. Much larger values of $\mathcal{C}_{\text{eff}}$ can be achieved with fewer stages by using additional steps, as shown in Figure 1(f) and Table 6. The best method has $\mathcal{C}_{\text{eff}} \approx 0.247$; to achieve the same efficiency with a linear multistep method requires the use of more than thirty steps [19]. Once again, due to the number of coefficients and constraints this was a difficult optimization problem, and we were not able to converge to the best methods from a "cold start". This is evident in our $(s,k) = (7,4), (7,5), (8,3), (8,4), (8,5)$ methods, which have a smaller SSP coefficient than those in [25, 28]. However, converting the methods in [27] to our form we were able to replicate their results while tightening the optimizer parameters `TolCon, TolFun` and `TolX` from 10^{-12} in their work to 10^{-14}.

4.2.8. *Ninth- and tenth-order methods.* For orders higher than eight, finding practical multistep or Runge–Kutta methods is a challenge even when the SSP property is not required. Numerical optimization of such high-order MSRK methods is computationally intensive, so we have restricted our search to a few combinations of stage and step number. Explicit two-step RK methods with positive SSP coefficient and order nine do not exist [20]. In contrast, with three-step methods we can achieve at least order ten. Investigating ninth-order methods with four steps, we obtain an $(s,k) = (8,4)$ method with $\mathcal{C}_{\text{eff}} = 0.1276$, and an $(s,k) = (9,4)$ method with $\mathcal{C}_{\text{eff}} = 0.1766$. We also found a $(9,5)$ method with $\mathcal{C}_{\text{eff}} = 0.1883$. By comparison, a multistep method requires 23 steps for $\mathcal{C}_{\text{eff}} = 0.116$ and 28 steps for $\mathcal{C}_{\text{eff}} = 0.175$.



(a) Third-order methods

(b) Fourth-order methods

(c) Fifth-order methods

(d) Sixth-order methods

(e) Seventh-order methods

(f) Eighth-order methods

FIGURE 1. Effective SSP coefficients of optimized methods.

These methods also compare favorably to the $(s, k) = (8, 5)$ method in [27], which has $\mathcal{C}_{\text{eff}} = 0.153$.

Searching for tenth-order methods, we obtained an $(s, k) = (20, 3)$ with $\mathcal{C}_{\text{eff}} = 0.0917$ and a $(k, s) = (8, 6)$ method with $\mathcal{C}_{\text{eff}} = 0.0839$. While these methods have small effective coefficients, they demonstrate that it is possible to find tenth-order SSP methods with much fewer than the 22 steps required for linear multistep methods. For comparison, the optimal multistep method with 22 steps and order 10 has $\mathcal{C}_{\text{eff}} = 0.10$ [12].

4.2.9. *Recommended methods.* In this section we collect the methods that we believe to be the most useful. In general, we think SSP Runge–Kutta methods should still be preferred up to fourth order since they allow the step size to easily be changed and they are self-starting. Nevertheless, we select some third- and fourth-order methods for testing purposes. Among the third-order methods, there is no need to go past three steps, so that the $(s,k,p)=(2,3,3)$ method with $\mathcal{C}_{\text{eff}}=0.56$ and the (7,3,3) method with $\mathcal{C}_{\text{eff}}=0.64$ are both excellent choices. For fourth-order methods, the one-step SSPRK(10,4) method with $\mathcal{C}_{\text{eff}}=0.6$ is still one of the most efficient methods, but the $(s,k,p)=(3,4,4)$ method is not too far behind with $\mathcal{C}_{\text{eff}}=0.455$.

The true benefit of additional steps begins past fourth order, where SSP Runge–Kutta methods are not available. The two best options for fifth order are the $(s,k,p)=(3,4,5)$ method with $\mathcal{C}_{\text{eff}}=0.33$ and the $(s,k,p)=(6,3,5)$ method with $\mathcal{C}_{\text{eff}}=0.404$. For sixth order, we recommend the $(s,k,p)=(5,3,6)$ method with $\mathcal{C}_{\text{eff}}=0.272$, or if one is willing to incur the additional storage cost of five steps, the $(s,k,p)=(6,5,6)$ method with $\mathcal{C}_{\text{eff}}=0.345$. The recommended seventh-order methods are $(s,k,p)=(7,3,7)$ with $\mathcal{C}_{\text{eff}}=0.243$ or, for the cost of an additional step, the $(s,k,p)=(7,4,7)$ method with $\mathcal{C}_{\text{eff}}=0.286$. The eighth-order method $(s,k,p)=(8,3,8)$ is a good method, with $\mathcal{C}_{\text{eff}}=0.1$, but increasing the number of stages by one and the number of steps by two yields an $(s,k,p)=(9,5,8)$ with more than double allowable time step, a $\mathcal{C}_{\text{eff}}=0.229$. Finally, among the ninth- and tenth-order methods there are fewer to choose, with two good options being $(s,k,p)=(9,4,9)$ with $\mathcal{C}_{\text{eff}}=0.1766$ and $(s,k,p)=(20,3,10)$ with $\mathcal{C}_{\text{eff}}=0.0917$.

5. Numerical tests

In this section we present numerical tests of the optimized MSRK methods identified above. The numerical tests have three purposes: (1) to verify that the methods have the designed order of accuracy; (2) to demonstrate the value of high-order time-stepping methods when using high-order spatial discretizations; and (3) to study the strong stability properties of the newly designed MSRK methods in practice on both linear and nonlinear test cases for which the forward Euler method is known to be total variation diminishing or positivity preserving.

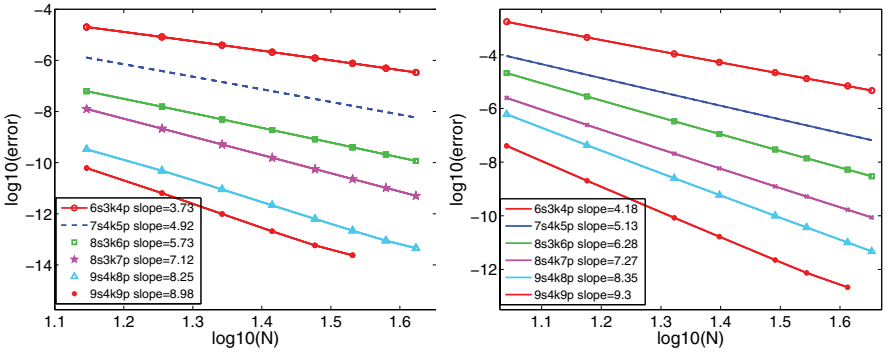
5.1. **Order verification.** Convergence studies for ordinary differential equations were performed using the van der Pol oscillator, a nonlinear system, to confirm the design orders of the methods. As these methods were designed for use as time integrators for partial differential equations, we include a convergence study for a PDE with high-order spatial discretization.

The van der Pol oscillator problem. The van der Pol problem is:

$$u_1' = u_2, \tag{33}$$

$$u_2' = \tfrac{1}{\epsilon}(-u_1 + (1-u_1^2)u_2). \tag{34}$$

We use $\epsilon = 10$ and initial conditions $u_0 = (0.5; 0)$. This was run to final time $T_{final} = 4.0$, with $\Delta t = \frac{T_{final}}{N-1}$ where $N = 15, 19, 23, 27, 31, 35, 39, 43$. Starting values and exact solution (for error calculation) were calculated using MATLAB's



(a) van der Pol (b) linear advection

FIGURE 2. Order verification of multistep Runge–Kutta methods on ordinary differential equations (left) and partial differential equations (right).

ODE45 with all tolerances set to 10^{-14}. In Figure 2(a) we show the convergence of the error in $u_1(4)$. Due to space limitations, we present only the results for a few methods. All methods were tested and exhibited the expected rate of convergence.

Linear advection with a Fourier spectral method. For the PDE convergence test, we chose the Fourier spectral method on the advection equation with sine wave initial conditions and periodic boundaries:

$$\begin{aligned} u_t &= -u_x, \quad x \in [0,1], \\ u(0,x) &= \sin(4\pi x), \quad u(t,0) = u(t,1). \end{aligned} \tag{35}$$

The exact solution to this problem is a sine wave with period 4 that travels in time. Due to the periodicity of the exact solution, the Fourier spectral method gives us an exact solution in space [14] once we have two points per wavelength, allowing us to isolate the effect of the temporal discretization on the error. We ran this problem with $N = (11, 15, 21, 25, 31, 35, 41, 45)$ to $T_{final} = 1$ with $\Delta t = 0.4\Delta x$, where $\Delta x = \frac{1}{N-1}$. For each multistep Runge–Kutta method of order p we generated the $k-1$ initial values using the third-order Shu-Osher SSP Runge–Kutta method with time-step $\Delta t^{p/3}$. Errors are computed at the final time, compared to the exact solution. Figure 2(b) contains the l_2 norm of the errors and demonstrates that the methods achieve the expected convergence rates.

5.2. **Benefits of high-order time discretizations.** High-order spatial discretizations for hyperbolic PDEs have usually been paired with lower-order time discretizations, e.g. [2–4, 6, 7, 18, 22, 29, 34]. Although spatial truncation errors are often observed to be larger than temporal errors in practice, this discrepancy can lead to loss of accuracy unless the time-step is significantly reduced.

If the order of the time-stepping method is p_1 and the order of the spatial method is p_2, then asymptotic convergence at rate p_2 is assured only if $\Delta t = \mathcal{O}(\Delta x^{p_2/p_1})$. For hyperbolic PDEs, one typically wishes to take $\Delta t = \mathcal{O}(\Delta x)$ for accuracy reasons.

In the following example we solve the two-dimensional advection equation $u_t + u_x + u_y = 0$ over the unit square with periodic boundary conditions in each direction and initial data $u(0,x) = \sin(2\pi(x+y))$. We take $\Delta x = \Delta y = \frac{1}{N-1}$. We solve for $0 \le t \le \frac{1}{8}$ with $\Delta t = \frac{1}{4}\Delta x$. We use ninth-order WENO finite differences in space. For each multistep Runge–Kutta method of order p we generated the $k-1$ initial values using the third-order Shu-Osher SSP Runge–Kutta method with time-step $\Delta t^{p/3}$. Figure 3 shows the accuracy of several of our high-order multistep Runge–Kutta methods applied to this problem. Observe that while methods of order $p \le 6$ exhibit an asymptotic convergence rate of less than ninth-order, our newly found methods of order $p \ge 7$ allow the high-order behavior of the WENO to become apparent.

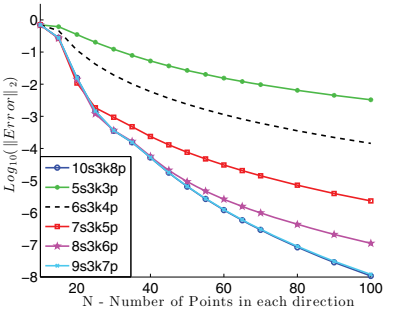


FIGURE 3. Convergence of a 2D advection equation with ninth-order WENO in space and MSRK in time.

5.3. **Strong stability performance of the new MSRK methods.** In this section we discuss the strong stability performance of the new methods in practice. The SSP condition is a very general condition: it holds for any convex functional and any starting value for arbitrary nonlinear non-autonomous equations, assuming only that the forward Euler method satisfies the corresponding monotonicity condition. In other words, it is a bound based on the worst-case behavior. Hence it should not be surprising that larger step sizes are possible when one considers a particular problem and a particular convex functional.

Here we explore the behavior of these methods in practice on the linear advection and nonlinear Buckley-Leverett equations, looking only at the total variation and positivity properties.

Example 1 (Advection)**.** Our first example is the advection equation

$$u_t + u_x = 0$$

with a step function initial condition:

$$u(0,x) = \begin{cases} 1, & \text{if } 0 \le x \le 1/2, \\ 0, & \text{if } x > 1/2, \end{cases}$$

on the domain $[0,1)$ with periodic boundary conditions. The problem was semi-discretized using a first-order forward difference on a grid with $N = 101$ points and evolved to a final time of $t = \frac{1}{8}$. We used the exact solution for the $k-1$ initial values. Euler's method is TVD and positive for step sizes up to $\Delta t_{\text{FE}} = \Delta x$. Table 7 shows the normalized observed effective time-step $\frac{1}{s}\frac{\Delta t_{TVD}}{\Delta x}$ for which each method maintains the total variation diminishing property and the normalized observed effective time-step $\frac{1}{s}\frac{\Delta t^+}{\Delta x}$ for which each method maintains positivity. We

TABLE 7. Observed effective TVD time-step compared with the theoretical values for Example 1.

method	$\frac{1}{s}\frac{\Delta t_{TVD}}{\Delta x}$	$\mathcal{C}_{\text{eff}}\frac{\Delta t_{FE}}{\Delta x}$
(2,3,3)	0.556	0.556
(7,3,3)	0.900	0.641
(3,4,4)	0.455	0.455
SSP RK10,4	0.600	0.600
(3,4,5)	0.334	0.334
(6,3,5)	0.404	0.404
(5,3,6)	0.272	0.272
(6,5,6)	0.345	0.345
(7,3,7)	0.243	0.243
(7,4,7)	0.286	0.286
(8,3,8)	0.112	0.100
(9,5,8)	0.229	0.229
(9,4,9)	0.182	0.177
(20,3,10)	0.107	0.092

compare these values to the normalized effective time-step guaranteed by the theory, $\mathcal{C}_{\text{eff}}\frac{\Delta t_{FE}}{\Delta x}$. These examples confirm that the observed positivity preserving time-step correlates well with the size of the SSP coefficient, and these methods compare favorably with the baseline methods. Also, the methods perform in practice as well or better than the lower bound guaranteed by the theory.

Example 2 (Buckley-Leverett problem)**.** We solve the Buckley-Leverett equation, a nonlinear PDE used to model two-phase flow through porous media:

$$u_t + f(u)_x = 0, \qquad \text{where } f(u) = \frac{u^2}{u^2 + a(1-u)^2},$$

on $x \in [0,1)$, with periodic boundary conditions. We take $a = \frac{1}{3}$ and initial condition

$$u(x,0) = \begin{cases} 1/2, & \text{if } x \geq 1/2, \\ 0, & \text{otherwise.} \end{cases} \tag{36}$$

The problem is semi-discretized using a conservative scheme with a Koren Limiter as in [20] with $\Delta x = \frac{1}{100}$, and run to $t_f = \frac{1}{8}$. For this problem the theoretical TVD time-step is $\Delta t_{FE} = \frac{1}{4}\Delta x = 0.0025$. For each multistep Runge–Kutta method of order p we generated the $k-1$ initial values using the third-order Shu-Osher SSP Runge–Kutta method with time-step $\Delta t^{p/3}$.

In Table 8 we list the ratio of the observed TVD time-step to the theoretical value. We see that the observed values are significantly higher than the theoretical values. This has been typical of our experience with SSP methods: the linear advection problem with step function initial conditions is one of the more stringent tests and usually matches well with the theoretical value. However, with nonlinear problems we often see that the allowable time-step in practice is larger than the theory requires.

TABLE 8. The ratio of the observed TVD time-steps and the theoretical time-step for Example 2.

method	$\frac{\Delta t^{TVD}}{C\Delta t_{FE}}$
(3,4,4)	1.25
(6,3,5)	1.26
(3,4,5)	1.37
(5,3,6)	1.54
(6,5,6)	1.24
(7,3,7)	1.40
(7,4,7)	1.34
(8,3,8)	3.36
(9,5,8)	1.33
(9,4,9)	1.68
(20,3,10)	3.10

ACKNOWLEDGMENTS

This publication is based on work supported by Award No. FIC/2010/05 - 2000000231, made by King Abdullah University of Science and Technology (KAUST), and by AFOSR grant FA-9550-12-1-0224.

REFERENCES

[1] P. Albrecht, *The Runge-Kutta theory in a nutshell*, SIAM J. Numer. Anal. **33** (1996), no. 5, 1712–1735, DOI 10.1137/S0036142994260872. MR1411846 (97j:65105)

[2] J. A. Carrillo, I. M. Gamba, A. Majorana, and C.-W. Shu, *A WENO-solver for the transients of Boltzmann-Poisson system for semiconductor devices: performance and comparisons with Monte Carlo methods*, J. Comput. Phys. **184** (2003), no. 2, 498–525, DOI 10.1016/S0021-9991(02)00032-3. MR1959405 (2003m:82087)

[3] L.-T. Cheng, H. Liu, and S. Osher, *Computational high-frequency wave propagation using the level set method, with applications to the semi-classical limit of Schrödinger equations*, Commun. Math. Sci. **1** (2003), no. 3, 593–621. MR2069945 (2006b:35275)

[4] V. Cheruvu, R. D. Nair, and H. M. Tufo, *A spectral finite volume transport scheme on the cubed-sphere*, Appl. Numer. Math. **57** (2007), no. 9, 1021–1032, DOI 10.1016/j.apnum.2006.09.008. MR2335233 (2008c:86002)

[5] E. M. Constantinescu and A. Sandu, *Optimal explicit strong-stability-preserving general linear methods*, SIAM J. Sci. Comput. **32** (2010), no. 5, 3130–3150, DOI 10.1137/090766206. MR2729454 (2011m:65209)

[6] D. Enright, R. Fedkiw, J. Ferziger, and I. Mitchell, *A hybrid particle level set method for improved interface capturing*, J. Comput. Phys. **183** (2002), no. 1, 83–116, DOI 10.1006/jcph.2002.7166. MR1944529 (2003j:76084)

[7] L. Feng, C. Shu, and M. Zhang, *A hybrid cosmological hydrodynamic/N-body code based on a weighted essentially nonoscillatory scheme*, The Astrophysical Journal **612** (2004), 1–13.

[8] L. Ferracina and M. N. Spijker, *Stepsize restrictions for the total-variation-diminishing property in general Runge-Kutta methods*, SIAM J. Numer. Anal. **42** (2004), no. 3, 1073–1093 (electronic), DOI 10.1137/S0036142902415584. MR2113676 (2005k:65126)

[9] L. Ferracina and M. N. Spijker, *An extension and analysis of the Shu-Osher representation of Runge-Kutta methods*, Math. Comp. **74** (2005), no. 249, 201–219, DOI 10.1090/S0025-5718-04-01664-3. MR2085408

[10] S. Gottlieb, D. Higgs, and D. I. Ketcheson, *Strong stability preserving site*, `http:www.sspsite.org/msrk.html`.

[11] S. Gottlieb, D. I. Ketcheson, and C.-W. Shu, *High order strong stability preserving time discretizations*, J. Sci. Comput. **38** (2009), no. 3, 251–289, DOI 10.1007/s10915-008-9239-z. MR2475652 (2010b:65161)

[12] S. Gottlieb, D. Ketcheson, and C.-W. Shu, *Strong Stability Preserving Runge-Kutta and Multistep Time Discretizations*, World Scientific Publishing Co. Pte. Ltd., Hackensack, NJ, 2011. MR2789749

[13] S. Gottlieb, C.-W. Shu, and E. Tadmor, *Strong stability-preserving high-order time discretization methods*, SIAM Rev. **43** (2001), no. 1, 89–112 (electronic), DOI 10.1137/S003614450036757X. MR1854647 (2002f:65132)

[14] J. S. Hesthaven, S. Gottlieb, and D. Gottlieb, *Spectral Methods for Time-Dependent Problems*, Cambridge Monographs on Applied and Computational Mathematics, vol. 21, Cambridge University Press, Cambridge, 2007. MR2333926 (2008i:65223)

[15] I. Higueras, *On strong stability preserving time discretization methods*, J. Sci. Comput. **21** (2004), no. 2, 193–223, DOI 10.1023/B:JOMP.0000030075.59237.61. MR2069949 (2005d:65112)

[16] I. Higueras, *Representations of Runge-Kutta methods and strong stability preserving methods*, SIAM J. Numer. Anal. **43** (2005), no. 3, 924–948, DOI 10.1137/S0036142903427068. MR2177549

[17] C. Huang, *Strong stability preserving hybrid methods*, Appl. Numer. Math. **59** (2009), no. 5, 891–904, DOI 10.1016/j.apnum.2008.03.030. MR2495128 (2009m:65100)

[18] S. Jin, H. Liu, S. Osher, and Y.-H. R. Tsai, *Computing multivalued physical observables for the semiclassical limit of the Schrödinger equation*, J. Comput. Phys. **205** (2005), no. 1, 222–241, DOI 10.1016/j.jcp.2004.11.008. MR2132308 (2005m:81106)

[19] D. I. Ketcheson, *Computation of optimal monotonicity preserving general linear methods*, Math. Comp. **78** (2009), no. 267, 1497–1513, DOI 10.1090/S0025-5718-09-02209-1. MR2501060 (2010a:65114)

[20] D. I. Ketcheson, S. Gottlieb, and C. B. Macdonald, *Strong stability preserving two-step Runge-Kutta methods*, SIAM J. Numer. Anal. **49** (2011), no. 6, 2618–2639, DOI 10.1137/10080960X. MR2873250

[21] J. F. B. M. Kraaijevanger, *Contractivity of Runge-Kutta methods*, BIT **31** (1991), no. 3, 482–528, DOI 10.1007/BF01933264. MR1127488 (92i:65120)

[22] S. Labrunie, J. A. Carrillo, and P. Bertrand, *Numerical study on hydrodynamic and quasi-neutral approximations for collisionless two-species plasmas*, J. Comput. Phys. **200** (2004), no. 1, 267–298, DOI 10.1016/j.jcp.2004.04.020. MR2086195 (2005d:76046)

[23] H. W. J. Lenferink, *Contractivity preserving explicit linear multistep methods*, Numer. Math. **55** (1989), no. 2, 213–223, DOI 10.1007/BF01406515. MR987386 (90f:65058)

[24] T. Nguyen-Ba, H. Nguyen-Thu, T. Giordano, and R. Vaillancourt, *Strong-stability-preserving 3-stage Hermite-Birkhoff time-discretization methods*, Appl. Numer. Math. **61** (2011), no. 4, 487–500, DOI 10.1016/j.apnum.2010.11.013. MR2754573 (2012g:65116)

[25] T. Nguyen-Ba, H. Nguyen-Thu, T. Giordano, and R. Vaillancourt, *Strong-stability-preserving 7-stage Hermite-Birkhoff time-discretization methods*, J. Sci. Comput. **50** (2012), no. 1, 63–90, DOI 10.1007/s10915-011-9473-7. MR2886319

[26] T. Nguyen-Ba, H. Nguyen-Thu, and R. Vaillancourt, *Strong-stability-preserving, k-step, 5- to 10-stage, Hermite-Birkhoff time-discretizations of order 12*, American J. Computational Mathematics **1** (2011), 72–82.

[27] H. Nguyen-Thu, *Strong-stability-preserving Hermite-Birkhoff time-discretization methods*, Dissertation, University of Ottawa, Canada, (2012).

[28] H. Nguyen-Thu, T. Nguyen-Ba, and R. Vaillancourt, *Strong-stability-preserving, Hermite-Birkhoff time-discretization based on k step methods and 8-stage explicit Runge-Kutta methods of order 5 and 4*, J. Comput. Appl. Math. **263** (2014), 45–58, DOI 10.1016/j.cam.2013.11.013. MR3162335

[29] D. Peng, B. Merriman, S. Osher, H. Zhao, and M. Kang, *A PDE-based fast local level set method*, J. Comput. Phys. **155** (1999), no. 2, 410–438, DOI 10.1006/jcph.1999.6345. MR1723321 (2000j:65104)

[30] S. J. Ruuth and R. J. Spiteri, *Two barriers on strong-stability-preserving time discretization methods*, Proceedings of the Fifth International Conference on Spectral and High Order Methods (ICOSAHOM-01) (Uppsala), J. Sci. Comput. **17** (2002), no. 1-4, 211–220, DOI 10.1023/A:1015156832269. MR1910562

[31] C.-W. Shu, *Total-variation-diminishing time discretizations*, SIAM J. Sci. Statist. Comput. **9** (1988), no. 6, 1073–1084, DOI 10.1137/0909073. MR963855 (90a:65196)
[32] M. N. Spijker, *Stepsize conditions for general monotonicity in numerical initial value problems*, SIAM J. Numer. Anal. **45** (2007), no. 3, 1226–1245 (electronic), DOI 10.1137/060661739. MR2318810 (2008e:65199)
[33] M. N. Spijker, *Contractivity in the numerical solution of initial value problems*, Numer. Math. **42** (1983), no. 3, 271–290, DOI 10.1007/BF01389573. MR723625 (85b:65067)
[34] M. Tanguay and T. Colonius, *Progress in modeling and simulation of shock wave lithotripsy (SWL)*, in Fifth International Symposium on cavitation (CAV2003), 2003.

Department of Mathematics, University of Massachusetts, Dartmouth, 285 Old Westport Road, North Dartmouth, Massachusetts 02747
E-mail address: cbresten@umassd.edu

Department of Mathematics, University of Massachusetts, Dartmouth, 285 Old Westport Road, North Dartmouth, Massachusetts 02747
E-mail address: sgottlieb@umassd.edu

Department of Mathematics, University of Massachusetts, Dartmouth, 285 Old Westport Road, North Dartmouth Massachusetts 02747
E-mail address: zgrant@umassd.edu

Department of Mathematics, University of Massachusetts, Dartmouth, 285 Old Westport Road, North Dartmouth, Massachusetts 02747

King Abdullah University of Science & Technology (KAUST), Thuwal, Saudi Arabia

Department of Mathematics and Computational Sciences, Széchenyi István University, Győr, Hungary