A numerical study of diagonally split Runge–Kutta methods for PDEs with discontinuities

Colin B. Macdonald[*], Sigal Gottlieb[†], and Steven J. Ruuth[‡]

December 17, 2007

Abstract

Diagonally split Runge–Kutta (DSRK) time discretization methods are a class of implicit time-stepping schemes which offer both high-order convergence and a form of nonlinear stability known as unconditional contractivity. This combination is not possible within the classes of Runge–Kutta or linear multistep methods and therefore appears promising for the strong stability preserving (SSP) time-stepping community which is generally concerned with computing oscillation-free numerical solutions of PDEs. Using a variety of numerical test problems, we show that although second- and third-order unconditionally contractive DSRK methods do preserve the strong stability property for all time step-sizes, they suffer from order reduction at large step-sizes. Indeed, for time-steps larger than those typically chosen for explicit methods, these DSRK methods behave like first-order implicit methods. This is unfortunate, because it is precisely to

[*] Department of Mathematics, Simon Fraser University, Burnaby, British Columbia, V5A 1S6 Canada (cbm@sfu.ca). The work of this author was partially supported by an NSERC Canada PGS-D scholarship, a grant from NSERC Canada, and a scholarship from the Pacific Institute for the Mathematical Sciences (PIMS).

[†] Department of Mathematics, University of Massachusetts Dartmouth, North Dartmouth MA 02747 USA (sgottlieb@umassd.edu). This work was supported by AFOSR grant number FA9550-06-1-0255.

[‡] Department of Mathematics, Simon Fraser University, Burnaby, British Columbia, V5A 1S6 Canada (sruuth@sfu.ca). The work of this author was partially supported by a grant from NSERC Canada.

allow a large time-step that we choose to use implicit methods. These results suggest that unconditionally contractive DSRK methods are limited in usefulness as they are unable to compete with either the first-order backward Euler method for large step-sizes or with Crank-Nicolson or high-order explicit SSP Runge–Kutta methods for smaller step-sizes.

We also present stage order conditions for DSRK methods and show that the observed order reduction is associated with the necessarily low stage order of the unconditionally contractive DSRK methods.

1 Introduction

Strong stability preserving (SSP) high-order time discretizations [32, 33, 13] were developed for the solution of semi-discrete method-of-lines approximations of hyperbolic partial differential equations (PDEs) with discontinuous solutions. In such cases, carefully constructed spatial discretization methods guarantee a desired nonlinear or *strong* stability property (for example, that the solution be free of oscillations) when coupled with first-order forward Euler (FE) time-stepping. However, for practical computation, higher-order time discretizations are usually needed, and there is no guarantee that the nonlinearly stable spatial discretization will produce stable results when coupled with an only linearly stable higher-order time discretization. In fact, numerical evidence [12] shows that oscillations may occur when using a linearly stable, high-order time discretization which does not preserve the stability properties of forward Euler, even if the same spatial discretization is total variation diminishing (TVD) when combined with the first-order forward Euler time-discretization. SSP methods are high-order time discretization methods that preserve the strong stability properties—in any norm or semi-norm—of the spatial discretization coupled with forward Euler time-stepping.

The idea behind SSP methods is to assume that the spatial discretization is strongly stable under a certain semi-norm when coupled with the forward Euler time discretization, for a suitably restricted time-step, and then find a higher-order time discretization that maintains strong stability for the same semi-norm, perhaps under a different time-step restriction. The class of high-order SSP time discretization methods for the semi-discrete method-of-lines approximations of PDEs was developed in [33, 32] and was at that time known as TVD time discretizations. This class of methods was further studied by Gottlieb and Shu, Spiteri and Ruuth, Higueras, Ferracina and Spijker and others (e.g., [12, 36, 16, 8]). The methods preserve the stability properties of forward Euler in any norm or semi-norm. In fact, because the stability arguments are based on convex decompositions of high-order methods in terms of the forward Euler method, any convex function will be preserved by SSP high-order time discretizations. SSP time discretizations can then be safely used with any spatial discretization which has the required stability properties when coupled with forward Euler.

The drawback of explicit SSP methods is that they suffer from restrictive time-step conditions. To obviate these difficulties we turn to implicit time-stepping methods with SSP properties. It was shown in [19] and [15], that any spatial discretization which is strongly stable in some semi-norm for the explicit forward Euler method under a certain time restriction will also be strongly stable, in the same semi-norm, with the implicit backward Euler (BE) method, *without* a time-step restriction. In previous work [13], efforts have been made to design higher-order implicit methods which share the strong stability properties of backward Euler, without any restriction on the time-step. Unfortunately, this goal cannot be realized for methods within the class of Runge–Kutta or linear multistep methods. For both implicit Runge–Kutta and multistep methods it has been proved that any higher-order SSP method, even for linear constant coefficient problems, will have some time-step restriction [13, 34]. This step-size restriction becomes apparent even in the simplest computations. An example of this is seen in Section 2.1, Figure 1 where the solution to a linear advection equation is discretized using a TVD forward difference spatial discretization and the implicit Crank–Nicolson (CN) time discretization. The numerical solution develops oscillations when the time-step restriction is exceeded. However, when the first-order, unconditionally SSP backward Euler method is used with this spatial discretization, the numerical solution remains TVD even for large step sizes.

To identify methods with no step-size restriction, we must extend our search beyond the standard Runge–Kutta and linear multistep methods. One such class, in particular, is the family of diagonally split Runge–Kutta methods (DSRK) [1, 2, 17, 20], which have been shown to allow a form of nonlinear stability known as unconditional contractivity. In this paper, we study unconditionally contractive DSRK methods and examine numerically the nonlinear stability properties exhibited by these methods. We then compare their performance in terms of nonlinear stability and accuracy to standard implicit and explicit SSP time-stepping methods. The paper is structured as follows: in Section 2 we describe the construction of SSP Runge–Kutta methods and review the results for explicit and implicit SSP Runge–Kutta methods. In Section 3 we introduce the DSRK methods and their properties. In Section 4 we present numerical studies comparing DSRK with implicit and explicit Runge–Kutta methods, in terms of both accuracy and efficiency. In Section 5, we discuss order reduction of DSRK and present stage order conditions to avoid it. In Section 6, we draw conclusions about the use of unconditionally contractive DSRK methods and future research directions.

2 SSP Runge–Kutta Methods

We wish to approximate the solution of the ODE system

$$\boldsymbol{u}_t = L(\boldsymbol{u}), \tag{1}$$

with initial conditions $\boldsymbol{u}(t_0) = \boldsymbol{u}_0$, typically arising from the spatial discretization of the PDE

$$u_t + f(u)_x = 0,$$

in which case $\boldsymbol{u} = (u_j)$ is a vector which gives the numerical solution of the PDE at spatial points x_j, $j = 1, \ldots, m$. The spatial discretization $L(\boldsymbol{u})$ is often chosen so that forward Euler

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^n + \Delta t L(\boldsymbol{u}^n),$$

is *strong stability preserving* (SSP)

$$||\boldsymbol{u}^{n+1}|| \leq ||\boldsymbol{u}^n||, \tag{2}$$

in some norm, semi-norm or convex functional $||\cdot||$, under the restricted time-step

$$\Delta t \leq \Delta t_{\mathrm{FE}}.$$

The original choice for $||\cdot||$ was the *total variation semi-norm*

$$||\boldsymbol{u}||_{\mathrm{TV}} = \sum_j |u_{j+1} - u_j|,$$

and a spatial discretization which, when combined with forward Euler, results in a method which is SSP in this semi-norm is said to be *total variation diminishing* (TVD).

A general explicit m-stage Runge–Kutta method for (1) is written in Shu–Osher form [33]

$$\begin{aligned}\boldsymbol{u}^{(0)} &= \boldsymbol{u}^n,\\ \boldsymbol{u}^{(i)} &= \sum_{k=0}^{i-1} \left(\alpha_{ik}\boldsymbol{u}^{(k)} + \Delta t \beta_{ik} L(\boldsymbol{u}^{(k)})\right), \quad \alpha_{ik} \geq 0, \quad i = 1, \ldots, m, \\ \boldsymbol{u}^{n+1} &= \boldsymbol{u}^{(m)}.\end{aligned} \tag{3}$$

Consistency requires that $\sum_{k=0}^{i-1} \alpha_{ik} = 1$ and if $\alpha_{ik} \geq 0$ and $\beta_{ik} \geq 0$, all the intermediate stages in (3), $\boldsymbol{u}^{(i)}$, are simply convex combinations of forward Euler operators, with Δt replaced by $\frac{\beta_{ik}}{\alpha_{ik}}\Delta t$. Therefore—as originally shown in [33]—any norm, semi-norm or convex function property satisfied by the forward Euler method will be preserved by the Runge–Kutta method, under the time-step restriction

$$\Delta t \leq \min_{i<k} \frac{\alpha_{ik}}{\beta_{ik}} \Delta t_{\mathrm{FE}}, \tag{4}$$

where $\frac{\alpha_{ik}}{\beta_{ik}} = \infty$ if $\beta_{ik} = 0$.

Much of the research in the field of SSP methods centers around the search for high-order SSP methods where the allowable time-step is as large as possible. If a method has a SSP time-step restriction $\Delta t \leq \mathcal{C}\Delta t_{\mathrm{FE}}$, then we will often use $\mathcal{C}$, the *SSP coefficient* or *CFL coefficient*, to measure the allowable time-step of a method relative to that of forward Euler. Many optimal methods with the largest possible SSP coefficients are listed in [30, 37, 11] and some popular explicit SSP Runge–Kutta methods are given below.

Two-stage, second-order SSP Runge–Kutta (SSP22) An optimal second-order SSP Runge–Kutta method is given by

$$\begin{aligned}\boldsymbol{u}^{(1)} &= \boldsymbol{u}^n + \Delta t L(\boldsymbol{u}^n),\\ \boldsymbol{u}^{n+1} &= \frac{1}{2}\boldsymbol{u}^n + \frac{1}{2}\boldsymbol{u}^{(1)} + \frac{1}{2}\Delta t L(\boldsymbol{u}^{(1)}).\end{aligned}$$

The step-size restriction for this method is $\Delta t \leq \Delta t_{\mathrm{FE}}$, which means that it has a SSP coefficient of $\mathcal{C} = 1$. However, note that the computational work required is doubled compared to forward Euler.

Three-stage, third-order SSP Runge–Kutta (SSP33) An optimal third-order SSP Runge–Kutta method is given by

$$\begin{aligned}
\boldsymbol{u}^{(1)} &= \boldsymbol{u}^n + \Delta t L(\boldsymbol{u}^n), \\
\boldsymbol{u}^{(2)} &= \frac{3}{4}\boldsymbol{u}^n + \frac{1}{4}\boldsymbol{u}^{(1)} + \frac{1}{4}\Delta t L(\boldsymbol{u}^{(1)}), \\
\boldsymbol{u}^{n+1} &= \frac{1}{3}\boldsymbol{u}^n + \frac{2}{3}\boldsymbol{u}^{(2)} + \frac{2}{3}\Delta t L(\boldsymbol{u}^{(2)}).
\end{aligned}$$

The step-size restriction for this method is $\Delta t \leq \Delta t_{\mathrm{FE}}$, so it has a value of $\mathcal{C} = 1$. However, the computational work in this method is three times that of forward Euler. This method is very commonly used and is often referred to as the third-order TVD Runge-Kutta scheme or the Shu–Osher method.

Five-stage, fourth-order SSP Runge–Kutta (SSP54) An optimal method developed in [36, 29, 22] with coefficients expressed to 15 digits is

$$\begin{aligned}
\boldsymbol{u}^{(1)} &= \boldsymbol{u}^n + 0.391752226571890\Delta t L(\boldsymbol{u}^n), \\
\boldsymbol{u}^{(2)} &= 0.444370493651235\boldsymbol{u}^n + 0.555629506348765\boldsymbol{u}^{(1)} \\
&\quad + 0.368410593050371\Delta t L(\boldsymbol{u}^{(1)}), \\
\boldsymbol{u}^{(3)} &= 0.620101851488403\boldsymbol{u}^n + 0.379898148511597\boldsymbol{u}^{(2)} \\
&\quad + 0.251891774271694\Delta t L(\boldsymbol{u}^{(2)}), \\
\boldsymbol{u}^{(4)} &= 0.178079954393132\boldsymbol{u}^n + 0.821920045606868\boldsymbol{u}^{(3)} \\
&\quad + 0.544974750228521\Delta t L(\boldsymbol{u}^{(3)}), \\
\boldsymbol{u}^{n+1} &= 0.517231671970585\boldsymbol{u}^{(2)} \\
&\quad + 0.096059710526146\boldsymbol{u}^{(3)} + 0.063692468666290\Delta t L(\boldsymbol{u}^{(3)}) \\
&\quad + 0.386708617503269\boldsymbol{u}^{(4)} + 0.226007483236906\Delta t L(\boldsymbol{u}^{(4)}).
\end{aligned}$$

The step-size restriction for this method is approximately $\Delta t \leq 1.508\Delta t_{\mathrm{FE}}$, which means that it has a value of $\mathcal{C} \approx 1.508$. The computational work in this method is five times that of forward Euler, but the allowable time-step makes this method almost as efficient as the SSP33 method, yet higher order.

In the development of new methods and in the numerical tests below, these explicit methods will serve as the gold standard, to be compared to implicit methods in terms of the time-step allowed and the computational cost required.

2.1 Implicit SSP methods

Historically, total variation diminishing (TVD) spatial discretizations are constructed in conjunction with the forward Euler method. The implicit backward Euler method will then preserve this property for all step-sizes [19, 15]. However, a higher-order time-discretization, such as the second-order Crank–Nicolson (CN) method, may only preserve the TVD property for a limited range of step-sizes. For example, consider the case of the linear wave equation

$$u_t + au_x = 0,$$

with $a = -2\pi$, a step-function initial condition

$$u(x,0) = \begin{cases} 1 & \text{if } \frac{\pi}{2} \leq x \leq \frac{3\pi}{2}, \\ 0 & \text{otherwise,} \end{cases}$$

and periodic boundary conditions on the domain $(0, 2\pi]$. The solution is a step function convected around the domain. For a simple first-order forward-difference TVD spatial discretization $L(\boldsymbol{u})$ of $-au_x$, the result will be TVD for all sizes of Δt when using the implicit backward Euler method. If we use the forward Euler time-stepping, the result is TVD for $\Delta t \leq \Delta t_{\text{FE}} = \frac{\Delta x}{|a|}$. On the other hand, consider the Crank–Nicolson method

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^n + \frac{1}{2}\Delta t L(\boldsymbol{u}^n) + \frac{1}{2}\Delta t L(\boldsymbol{u}^{n+1}). \tag{5}$$

Using the Shu–Osher theory, CN can be shown to be SSP only for values $\Delta t \leq 2\Delta t_{\text{FE}}$ [11]. This restriction is illustrated in Figure 1 where an excessively large Δt leads to oscillations and a clear violation of the TVD property.

Crank–Nicolson requires extra computational cost due to the solution of an implicit system, but with respect to strong stability only allows a doubling of the step-size compared to forward Euler or the second-order SSP22. This means that, in general, it will not be efficient to use this method.

The Shu–Osher form (3) has been generalized for implicit Runge–Kutta methods [11, 8, 16], and the search for implicit methods which are SSP without a time-step restriction has generated much interest. The first-order backward Euler method is one such method. Unfortunately, there are no Runge–Kutta or linear multistep methods of order greater than one which will satisfy this property [34, 18]. The search for implicit SSP Runge–Kutta methods with optimal SSP coefficients has been documented in [6, 21]. As

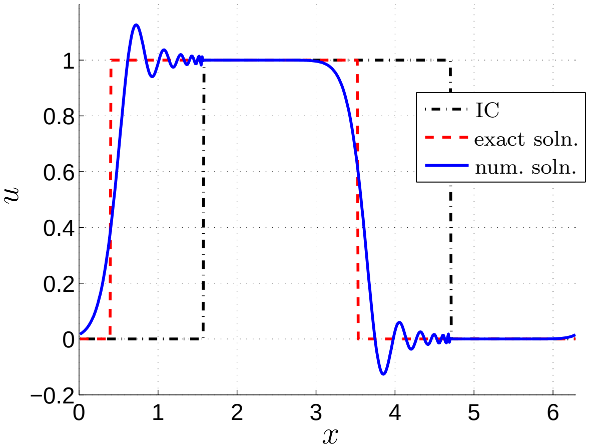


Figure 1: Oscillations from Crank–Nicolson time-stepping in the advection of a square wave with $\Delta t = 8\Delta t_{\text{FE}} = 8\Delta x$ and $\Delta x = \frac{2\pi}{512}$.

discussed further in Section 3.1, strong stability and contractivity are closely related for the class of implicit Runge–Kutta methods. This motivates us to search outside the class of Runge–Kutta methods for methods which are unconditionally contractive and high-order in the hope that they have good SSP properties as well. One class of high-order unconditionally contractive methods is the family of diagonally split Runge–Kutta (DSRK) methods.

3 Diagonally Split Runge–Kutta Methods

DSRK methods [1, 2, 20, 17] are one-step methods which are based on a Runge–Kutta formulation, but where the ODE operator L in (1) has different inputs used for the diagonal and off-diagonal components. We define the *diagonal splitting function* of L as

$$\mathfrak{L}_j(\boldsymbol{u}, \boldsymbol{z}) = L(z_1, z_2, \ldots, z_{j-1}, u_j, z_{j+1}, \ldots, z_m), \quad j = 1, \ldots, m, \tag{6}$$

that is, the j^{th} component of $\mathfrak{L}(\boldsymbol{u}, \boldsymbol{z})$ is computed using the j^{th} component of $\boldsymbol{u}$ for the j^{th} input of L and components of $\boldsymbol{z}$ for the other inputs of L.

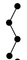
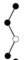
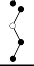
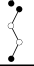
order 1		$\boldsymbol{b}^{\mathrm{T}}\boldsymbol{e} = 1$		
order 2		$\boldsymbol{b}^{\mathrm{T}}\mathbf{C}\boldsymbol{e} = \frac{1}{2}$		
order 3		$\boldsymbol{b}^{\mathrm{T}}\mathbf{C}^2\boldsymbol{e} = \frac{1}{3}$		
		$\boldsymbol{b}^{\mathrm{T}}\mathbf{W}\mathbf{C}\boldsymbol{e} = \frac{1}{6}$		$\boldsymbol{b}^{\mathrm{T}}\mathbf{A}\mathbf{C}\boldsymbol{e} = \frac{1}{6}$
order 4		$\boldsymbol{b}^{\mathrm{T}}\mathbf{C}^3\boldsymbol{e} = \frac{1}{4}$		
		$\boldsymbol{b}^{\mathrm{T}}\mathbf{C}\mathbf{W}\mathbf{C}\boldsymbol{e} = \frac{1}{8}$		$\boldsymbol{b}^{\mathrm{T}}\mathbf{C}\mathbf{A}\mathbf{C}\boldsymbol{e} = \frac{1}{8}$
		$\boldsymbol{b}^{\mathrm{T}}\mathbf{W}\mathbf{C}^2\boldsymbol{e} = \frac{1}{12}$		$\boldsymbol{b}^{\mathrm{T}}\mathbf{A}\mathbf{C}^2\boldsymbol{e} = \frac{1}{12}$
		$\boldsymbol{b}^{\mathrm{T}}\mathbf{W}^2\mathbf{C}\boldsymbol{e} = \frac{1}{24}$		$\boldsymbol{b}^{\mathrm{T}}\mathbf{A}\mathbf{W}\mathbf{C}\boldsymbol{e} = \frac{1}{24}$
		$\boldsymbol{b}^{\mathrm{T}}\mathbf{W}\mathbf{A}\mathbf{C}\boldsymbol{e} = \frac{1}{24}$		$\boldsymbol{b}^{\mathrm{T}}\mathbf{A}^2\mathbf{C}\boldsymbol{e} = \frac{1}{24}$

Table 1: The 14 order conditions for fourth-order DSRK schemes written in matrix form where $\mathbf{C} = \mathrm{diag}(\boldsymbol{c})$. See [2] for an explanation of the trees.

The general DSRK method is

$$\boldsymbol{U}^i = \boldsymbol{u}^n + \Delta t \sum_{j=1}^{m} a_{ij} \mathfrak{L}(\boldsymbol{U}^j, \boldsymbol{Z}^j), \tag{7a}$$

$$\boldsymbol{Z}^i = \boldsymbol{u}^n + \Delta t \sum_{j=1}^{m} w_{ij} \mathfrak{L}(\boldsymbol{U}^j, \boldsymbol{Z}^j), \tag{7b}$$

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^n + \Delta t \sum_{j=1}^{m} b_j \mathfrak{L}(\boldsymbol{U}^j, \boldsymbol{Z}^j). \tag{7c}$$

The schemes are consistent [1] and the coefficients $(\mathbf{A}, \boldsymbol{b}^{\mathrm{T}}, \boldsymbol{c}, \mathbf{W})$ must satisfy the order conditions [2] in Table 1. We note that these include the order conditions of the so-called *underlying Runge–Kutta method* (i.e., conditions only on $\mathbf{A} = (a_{ij})$, $\boldsymbol{b}$, and $\boldsymbol{c}$) and are augmented by additional order conditions on the coefficients $\mathbf{W} = (w_{ij})$.

3.1 Dissipative systems and contractivity

Bellen et al. [1] introduced the class of DSRK methods for *dissipative systems* $\boldsymbol{u}_t = L(t, \boldsymbol{u})$. In the special case of the maximum norm $\|\cdot\|_\infty$, a dissipative system is characterized (see e.g., [2]) by the condition

$$\sum_{j=1,j\neq i}^{m} \left| \frac{\partial L_i(t,\boldsymbol{u})}{\partial u_j} \right| \leq -\frac{\partial L_i(t,\boldsymbol{u})}{\partial y_i}, \qquad i = 1,\ldots,m,$$

for all $t \leq t_0$ and $\boldsymbol{u} \in \mathbb{R}^m$. We note in particular that the ODEs resulting from the spatial discretizations of our linear PDE test problems in Sections 4.1, 4.2, and 4.3 satisfy this condition. The ODE system resulting from the nonlinear problem in Section 4.4 can be shown to be dissipative in $\|\cdot\|_1$.

If the ODE system is dissipative, then solutions satisfy a *contractivity* property [34, 22, 38]. Specifically, if $\boldsymbol{u}(t)$ and $\boldsymbol{v}(t)$ are two solutions corresponding to initial conditions $\boldsymbol{u}(t_0)$ and $\boldsymbol{v}(t_0)$ then

$$\|\boldsymbol{u}(t) - \boldsymbol{v}(t)\| \leq \|\boldsymbol{u}(t_0) - \boldsymbol{v}(t_0)\|,$$

in some norm of interest. Naturally, if solutions to the ODE system obey a contractivity property then it is desirable that a numerical method for solving the problem be contractive as well, i.e., that given numerical solutions $\boldsymbol{u}_n$ and $\tilde{\boldsymbol{u}}_n$, $||\tilde{\boldsymbol{u}}_{n+1} - \boldsymbol{u}_{n+1}|| \leq ||\tilde{\boldsymbol{u}}_n - \boldsymbol{u}_n||$ (possibly subject to a time-step restriction).

In [20], in 't Hout showed that if a DSRK method is unconditionally contractive in the maximum norm, the underlying Runge–Kutta method is of classical order $p \leq 4$, and has stage order $\tilde{p} \leq 1$. In [17], Horváth studied the positivity of Runge–Kutta and DSRK methods, and showed that DSRK schemes can be unconditionally positive.

The results on DSRK methods in terms of positivity and contractivity appear promising when searching for implicit SSP schemes, because positivity, contractivity, and the SSP condition are all very closely related for Runge–Kutta and multistep methods [15, 16, 7, 8, 22]. For example, a loss of positivity implies the loss of the max-norm SSP property. For Runge–Kutta methods a link has also been established between time-step restrictions under the SSP condition and contractivity, namely that the time-step restrictions under either property agree [7], thereby enabling the possibility of transferring results established for the contractive case to the SSP case [15], and vice versa. For multistep methods, the time-step restrictions coming from either an SSP or contractivity analysis are the same, as can be seen by examining the proofs appearing in [25, 24, 32]. If we include the starting procedure into the analysis, or if we consider boundedness (a related nonlinear stability property) rather than the SSP property, significantly milder time-step restrictions may arise [19]. However, even with this less restrictive boundedness property, we find that unconditional strong stability is impossible for schemes that are more than first order [18]. The promise of DSRK method is that there exist higher-order implicit unconditionally contractive methods, and therefore possibly DSRK methods which are unconditionally SSP, in this class.

3.2 DSRK schemes

It is illustrative to examine (7) when the ODE operator L is linear. In this case, with matrix $\mathbf{L}$ decomposed into $\mathbf{L} = \mathbf{L}_\mathrm{D} + \mathbf{L}_\mathrm{N}$ where $\mathbf{L}_\mathrm{D} = \operatorname{diag}(\mathbf{L})$, we have $\mathfrak{L}(\boldsymbol{u}, \boldsymbol{z}) = \mathbf{L}_\mathrm{D}\boldsymbol{u} + \mathbf{L}_\mathrm{N}\boldsymbol{z}$ and (7) becomes

$$\boldsymbol{U}^i = \boldsymbol{u}^n + \Delta t \sum_{j=1}^{m} a_{ij} \left(\mathbf{L}_\mathrm{D}\boldsymbol{U}^j + \mathbf{L}_\mathrm{N}\boldsymbol{Z}^j\right), \tag{8a}$$

$$\boldsymbol{Z}^i = \boldsymbol{u}^n + \Delta t \sum_{j=1}^{m} w_{ij} \left(\mathbf{L}_\mathrm{D}\boldsymbol{U}^j + \mathbf{L}_\mathrm{N}\boldsymbol{Z}^j\right), \tag{8b}$$

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^n + \Delta t \sum_{j=1}^{m} b_j \left(\mathbf{L}_\mathrm{D}\boldsymbol{U}^j + \mathbf{L}_\mathrm{N}\boldsymbol{Z}^j\right), \tag{8c}$$

and thus we see that for a linear ODE system, DSRK methods decompose the system into diagonal and off-diagonal components and treat each differently.

We now list the DSRK schemes which are used in Section 4 for our numerical tests.

Second-order DSRK ("DSRK2") This second-order DSRK from [1] is based on the underlying two-stage, second-order implicit Runge–Kutta method specified by the Butcher tableau

$$\begin{array}{c|c} \boldsymbol{c} & \mathbf{A} \\ \hline & \boldsymbol{b}^\mathrm{T} \end{array} = \begin{array}{c|cc} 0 & \frac{1}{2} & -\frac{1}{2} \\ 1 & \frac{1}{2} & \frac{1}{2} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}, \qquad \text{combined with } \mathbf{W} = \begin{bmatrix} 0 & 0 \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix}. \tag{9a}$$

Thus the DSRK2 scheme is

$$\boldsymbol{U}^1 = \boldsymbol{u}^n + \frac{1}{2}\Delta t \mathfrak{L}(\boldsymbol{u}^n, \boldsymbol{U}^1) - \frac{1}{2}\Delta t L(\boldsymbol{u}^{n+1}), \tag{9b}$$

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^n + \frac{1}{2}\Delta t \mathfrak{L}(\boldsymbol{u}^n, \boldsymbol{U}^1) + \frac{1}{2}\Delta t L(\boldsymbol{u}^{n+1}). \tag{9c}$$

Note that the $\boldsymbol{u}^{n+1}$ terms are not split. For linear problems, (9) becomes

$$\boldsymbol{U}^1 = \boldsymbol{u}^n + \frac{1}{2}\Delta t \left[\mathbf{L}_{\mathrm{N}}\boldsymbol{u}^n + \mathbf{L}_{\mathrm{D}}\boldsymbol{U}^1\right] - \frac{1}{2}\Delta t \left[\mathbf{L}\boldsymbol{u}^{n+1}\right], \tag{10a}$$

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^n + \frac{1}{2}\Delta t \left[\mathbf{L}_{\mathrm{N}}\boldsymbol{u}^n + \mathbf{L}_{\mathrm{D}}\boldsymbol{U}^1\right] + \frac{1}{2}\Delta t \left[\mathbf{L}\boldsymbol{u}^{n+1}\right]. \tag{10b}$$

Note also in the special case when $\mathbf{L}_{\mathrm{D}} = \mathbf{0}$, (10) decouples and (10b) is exactly the Crank–Nicolson method.

Third-order DSRK ("DSRK3") This formally third-order DSRK scheme [1, 2, 20] is based on the underlying Runge–Kutta method:

$$\begin{array}{c|c} \boldsymbol{c} & \mathbf{A} \\ \hline & \boldsymbol{b}^{\mathrm{T}} \end{array} = \begin{array}{c|ccc} 0 & \frac{5}{2} & -2 & -\frac{1}{2} \\ \frac{1}{2} & -1 & 2 & -\frac{1}{2} \\ 1 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}, \qquad \text{combined with } \mathbf{W} = \begin{bmatrix} 0 & 0 & 0 \\ \frac{7}{24} & \frac{1}{6} & \frac{1}{24} \\ \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{bmatrix}.$$

Higher-order DSRK schemes Thus far, no unconditionally contractive fourth-order DSRK methods have been found. We begin searching for fourth-order DSRK using necessary conditions for maximum norm unconditionally contractivity found in the proof of [20, Theorem 2.4]; specifically,

$$\text{all principal minors of } \mathbf{A} - \boldsymbol{e}\boldsymbol{b}^{\mathrm{T}} \text{ are nonnegative,} \tag{11a}$$

$$\begin{aligned} &\text{for each } i \in \{1, 2, \ldots, s\},\ \det[(\mathbf{A} \leftarrow_i \boldsymbol{b}^{\mathrm{T}})(\mathcal{I})] \geq 0 \\ &\qquad\qquad \text{for every } \mathcal{I} \subset \{1, 2, \ldots, s\} \text{ with } i \in \mathcal{I}, \end{aligned} \tag{11b}$$

where the notation $\mathbf{M}(\mathcal{I})$ indicates the principal submatrix formed by selecting from $\mathbf{M}$ only those rows and columns indexed by $\mathcal{I}$.

In [26, 29, 21] the proprietary Branch and Reduce Optimization Navigator (Baron) software [31] was used to find optimal SSP Runge–Kutta schemes. Here we begin by searching for *any* feasible DSRK methods by imposing the 14 order conditions in Table 1 and the 48 necessary conditions (11) as constraints and minimizing the sum of the squares of the $\boldsymbol{b}$ coefficients. Baron was interrupted after 30 days of calculation (on an Athlon MP 2800+ with 1 GiB of RAM) and was unable to find any feasible solutions. Constrained only by the order conditions, Baron was able to quickly find DSRK44 schemes; it was also able to quickly find five-stage fourth-order

DSRK54 methods satisfying the order conditions and necessary conditions (11).

Altogether, this is a strong indication that unconditionally contractive DSRK44 methods do not exist. We leave open the question of the existence of unconditionally contractive DSRK54 schemes, noting however that such schemes are still likely to suffer from the order reduction noted in Section 4.

3.3 Numerical implementation of DSRK

For linear problems, we implement DSRK using (8) by re-arranging all the unknowns into a larger linear system, in general $(2sm)\times(2sm)$ where m is the size of the linear system (1) and s is the number of stages in the underlying Runge–Kutta scheme. However particular choices of methods may result in smaller systems; for example, the two-stage DSRK2 (10) can be written as the $2m \times 2m$ system

$$\left[\begin{array}{c|c} \mathbf{I} - \frac{1}{2}\Delta t\mathbf{L}_{\mathrm{D}} & \frac{1}{2}\Delta t\mathbf{L} \\ \hline -\frac{1}{2}\Delta t\mathbf{L}_{\mathrm{D}} & \mathbf{I} - \frac{1}{2}\Delta t\mathbf{L} \end{array}\right] \begin{pmatrix} \boldsymbol{U}^1 \\ \boldsymbol{u}^{n+1} \end{pmatrix} = \begin{pmatrix} \boldsymbol{u}^n + \frac{1}{2}\Delta t\mathbf{L}_{\mathrm{N}}\boldsymbol{u}^n \\ \boldsymbol{u}^n + \frac{1}{2}\Delta t\mathbf{L}_{\mathrm{N}}\boldsymbol{u}^n \end{pmatrix},$$

where $\mathbf{I}$ represents the $m \times m$ identity. We then simply solve this linear system to advance one time-step. As is usually the case, nonlinear systems are considerably more difficult. For the non-linear problems, we use a numerical zero-finding method to solve the nonlinear equations.

All numerical computations are performed with Matlab versions 7.0 and 7.3 using double precision on x86 and x86-64 architectures. Linear systems were solved using Matlab's `backslash` operator, whereas for the nonlinear problems in Sections 4.4 and 5.1, we implement the diagonal splitting function (6), and use a black-box equation solver (Matlab's `fsolve`) directly on (7).

4 Numerical Results

Our primary aim is to show that unconditionally contractive DSRK methods preserve the desired strong stability properties when applied to a variety of test cases. We focus our numerical experiments on three types of problems: convection, diffusion, and convection-diffusion. The SSP property is perhaps most important for convection driven problems, such as hyperbolic problems with discontinuous solutions. The methods have also been used to treat problems where the slope or some derivative of the solution is discontinuous and, for this reason, SSP schemes have been used widely to treat Hamilton–Jacobi equations (see, e.g., [27]). Many other problems of reaction-advection-diffusion type also can benefit from nonlinearly stable time-stepping. For example time-stepping a spatially discretized Black–Scholes equation (an equation we consider in Section 4.3) can lead to spurious oscillations in the solution. These oscillations are particularly undesirable in option-pricing problems because they can lead to highly oscillatory results in the first and second spatial derivatives—known respectively as γ and δ ("the Greeks") in computational finance.

4.1 Convection driven problems

An important prototype problem for SSP methods is the linear wave equation, or *advection equation*

$$u_t + au_x = 0, \qquad 0 \le x \le 2\pi. \tag{12}$$

We consider (12) with $a = -2\pi$, periodic boundary conditions and various initial conditions. We use a method-of-lines approach, discretizing the interval $(0, 2\pi]$ into m points $x_j = j\Delta x$, $j = 1, \ldots, m$, and then discretizing $-au_x$ with first-order upwind finite differences. We solve the resulting linear system using the time-stepping schemes described in Sections 2, and 3.

4.1.1 Smooth initial conditions

To study the order of accuracy of the methods, we consider (12) with smooth initial conditions

$$u(0, x) = \sin(x).$$

Table 2 shows a convergence study with fixed Δx. The implicit time-discretization methods used are backward Euler (BE), Crank–Nicolson (CN), DSRK2 and DSRK3. We also evolve the system with several explicit methods: forward Euler (FE), SSP22, SSP33, and SSP54. To isolate the effect of the time-discretization error, we exclude the effect of the error associated with the spatial discretization by comparing the numerical solution to the exact solution of the ODE system (1), rather than to the exact solution of the underlying PDE. In lieu of the exact solution we use a very accurate numerical solution obtained using MATLAB's `ode45` with minimal tolerances

		discrete error, l_∞-norm							
c	N	BE	order	CN	order	DSRK2	order	DSRK3	order
4	16	0.518		0.0582		0.408		0.395	
2	32	0.336	0.62	0.0147	1.98	0.194	1.08	0.178	1.15
1	64	0.194	0.79	3.70e-3	2.00	0.0714	1.44	0.0590	1.59
$\frac{1}{2}$	128	0.105	0.89	9.25e-4	2.00	0.0223	1.68	0.0152	1.95
…		…		…		…		…	
$\frac{1}{32}$	2048	7.04e-3		3.61e-6		1.09e-4		1.21e-5	
$\frac{1}{64}$	4096	3.53e-3	1.00	9.04e-7	2.00	2.74e-5	1.99	1.61e-6	2.91
$\frac{1}{128}$	8192	1.77e-3	1.00	2.26e-7	2.00	6.87e-6	1.99	2.09e-7	2.95
c	N	FE	order	SSP22	order	SSP33	order	SSP54	order
2	32	unstable		unstable		unstable		2.66e-5	
1	64	0.265		7.43e-3		1.82e-4		1.66e-6	4.00
$\frac{1}{2}$	128	0.122	1.12	1.85e-3	2.01	2.27e-5	3.00	1.03e-7	4.01

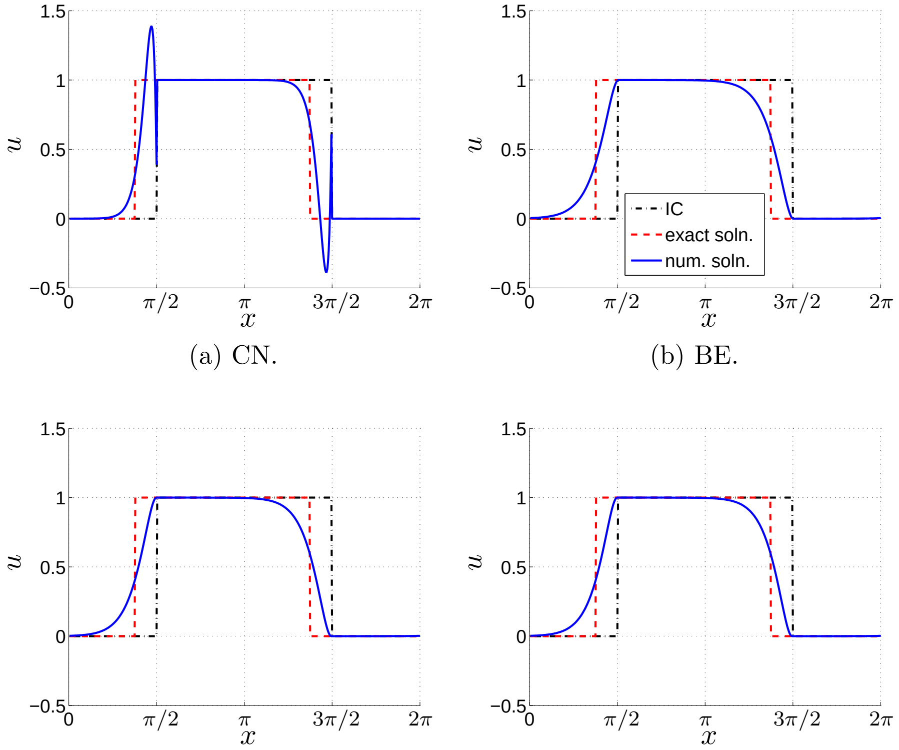
Table 2: Convergence study for the linear advection of a sine wave to $t_f = 1$ using N time-steps, $m = 64$ points and a first-order upwinding spatial discretization. Here c measures the size of the step relative to Δt_{FE}.

($\texttt{AbsTol} = 1 \times 10^{-14}$, $\texttt{RelTol} = 1 \times 10^{-13}$). Table 2 shows that all the methods achieve their design order when Δt is sufficiently small. However, the errors from CN are typically smaller than the errors produced by the other implicit methods. For large Δt, the second- and third-order DSRK schemes are far worse than CN. If we broaden our experiments to include explicit schemes, and take time-steps which are within the stability time-step restriction, we obtain smaller errors still. Given the relatively inexpensive cost of explicit time-stepping, it would appear that high-order explicit schemes (e.g., SSP54) are preferred for this smooth problem, unless, perhaps, very large time-steps are preferred over accuracy considerations.

4.1.2 Discontinuous initial conditions

To study the nonlinear stability properties of the methods, we consider the case of advection of discontinuous data

$$u(x,0) = \begin{cases} 1 & \text{if } \frac{\pi}{2} \le x \le \frac{3\pi}{2}, \\ 0 & \text{otherwise.} \end{cases} \tag{13}$$



(a) CN.

(b) BE.

(c) DSRK2.

(d) DSRK3.

Figure 2: Advection of a square wave after two time-steps, showing oscillations from Crank–Nicolson and none in the other methods. Here $c = 16$ and we take a first-order upwinding spatial discretization with $m = 512$ points in space.

Figure 2 shows typical results. Note that oscillations are observed in the Crank–Nicolson results, while the DSRK schemes are free of such oscillations. In fact, Table 3 shows that for any time-step size BE, DSRK2 and DSRK3 preserve the TVD property of the spatial discretization coupled with forward Euler. In contrast, Crank–Nicolson exhibits oscillations for time-steps larger than $\Delta t = \frac{2}{|a|}\Delta x$ (i.e., $c > 2$). These results suggest that the unconditionally contractive DSRK schemes do preserve the strong stability properties of the ODE system.

		$\max_t TV(\boldsymbol{u})$				
c	N	exact	CN	BE	DSRK2	DSRK3
32	16	2	8.78	2	2	2
16	32	2	6.64	2	2	2
8	64	2	4.73	2	2	2
4	128	2	3.33	2	2	2
2	256	2	2	2	2	2
1	512	2	2	2	2	2

Table 3: Total variation of the solution for the advection of a square wave (N time-steps, $t_f = 1$). The spatial discretization uses $m = 512$ points, first-order upwinding, and periodic BCs.

4.1.3 Order reduction and scheme selection

We now delve deeper into the observed convergence rates of our smooth and nonsmooth problems.

Figures 3 and 4 show that for large time-steps, the DSRK methods exhibit behavior similar to backward Euler in that they exhibit large errors and as we decrease size of the time-steps, the error decreases at a rate which appears only first order. As the time-steps are taken smaller still, the convergence rate increases to the design order of the DSRK schemes. In contrast, we note that Crank–Nicolson shows consistent second-order convergence over a wide range of time-steps.

On the discontinuous problem (Figure 4) we note the DSRK schemes do not produce significantly improved errors over backward Euler until the time-step sizes are small enough that Crank–Nicolson no longer exhibits spurious oscillations ($c = 2$ in Figure 4). In fact, once the time-steps are small enough that DSRK are competitive, we are almost within the nonlinear stability constraint of explicit methods such as SSP22 ($c = 1$ in Figure 4) .

We note that neither Figure 3 nor Figure 4 takes into account the differences in computational work required by the various methods. The costs for DSRK2 and DSRK3 are significantly larger than BE and CN, because the underlying systems are larger. In the linear case, the size of the DSRK2 system is $2m \times 2m$ and the DSRK3 system is $5m \times 5m$ whereas the BE and CN systems are only $m \times m$. Even if the cost of solving the system rose only linearly with the size of the system, the cost is doubled for DSRK2 and increased five-fold for DSRK3. In reality, the cost may increase more rapidly,

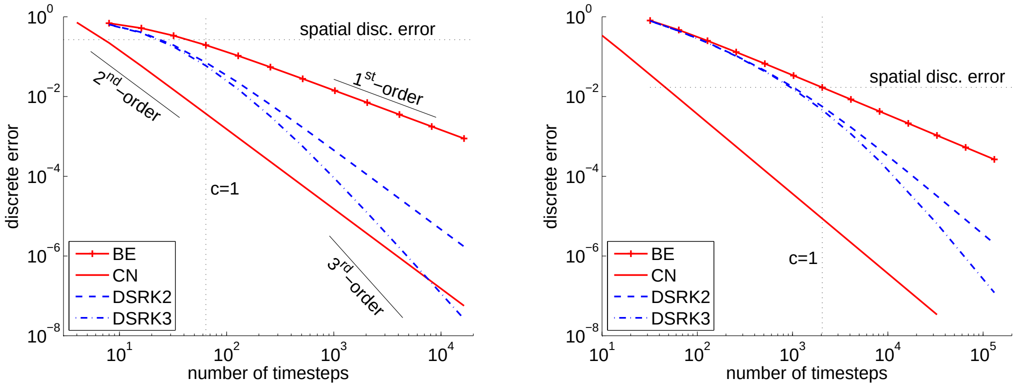


Figure 3: Convergence study for linear advection of a sine wave to $t_f = 1$. The spatial discretization here is first-order upwinding with 64 points (left) and 2048 points (right). We indicate the spatial discretization error with a dotted horizontal line.

depending on the structure of the implicit system and the method used to solve the implicit equations. Furthermore, if a nonlinear system is solved, this cost may increase even further. It is even more difficult to quantify the increased cost of an implicit method over that of an explicit method. However, it is clear that implicit methods in general and DSRK methods in particular are significantly more costly than explicit methods.

We note that phase errors were also investigated to see if the DSRK schemes had improved phase error properties compared to BE but they do not. In general, for large Δt, DSRK methods behave similarly in many aspects to backward Euler.

In summary, our results on the advection equation show that although the unconditionally contractive DSRK method are formally high order, in practice we encounter a reduction of order for large time-steps. If one requires large time-steps and no oscillations, backward Euler is a good choice. If on the other hand, one requires accuracy, an explicit high-order SSP method is probably better suited. We will see that these results are typical for unconditionally contractive DSRK schemes.

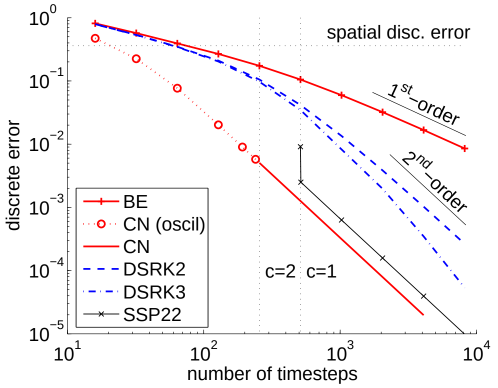


Figure 4: Linear advection of a square wave to time $t_f = 1$ using first-order upwinding and 512 points in space. Note that Crank–Nicolson produces oscillations during the computation for $c > 2$. We indicate the spatial discretization error with a dotted horizontal line.

4.2 Diffusion driven problems

Consider the diffusion or heat equation

$$u_t = \nu u_{xx}, \tag{14}$$

with heat coefficient ν on a periodic domain $(0, 2\pi]$. We begin by discretizing the u_{xx} term with second-order centered finite differences to obtain ODE system (1).

In Figure 5 and Table 4, we consider (14) with smooth initial conditions

$$u(0, x) = \sin(x) + \cos(2x).$$

Once again, we note that the DSRK schemes achieve their design order as Δt gets smaller, but for large time-steps they exhibit large errors and reduced convergence rates.

Figure 6 shows that Crank–Nicolson produces spurious oscillations in the solution to the heat equation with discontinuous initial conditions (13). Also, Figure 6 shows that the DSRK schemes are not competitive with backward Euler until the time-steps are smaller than the explicit stability limit (in this case, the restrictive $\Delta t \leq \frac{\Delta x^2}{2\nu}$ shown by the dotted vertical line). Clearly, the unconditionally contractive DSRK methods exhibit order reduction for this parabolic problem as well.

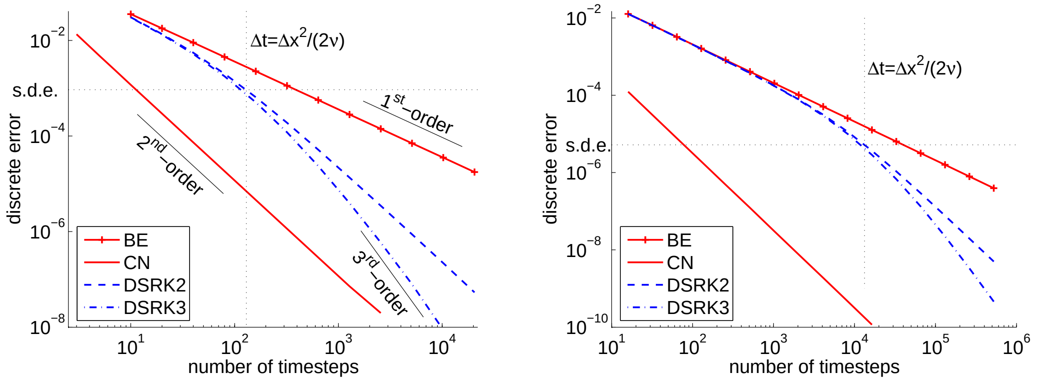


Figure 5: Convergence studies for the heat equation with smooth initial conditions. Left: $m = 64$, $t_f = 10$, $\nu = \frac{1}{16}$. Right: $m = 1024$, $t_f = 1$, $\nu = \frac{1}{4}$. The spatial discretization uses second-order centered differences and the level of spatial discretization error is indicated by the horizontal dotted line labeled “s.d.e.”

		discrete error l_∞-norm			
c	N	BE	CN	DSRK2	DSRK3
830	16	0.0127	1.24e-4	0.0127	0.0127
415	32	0.00643	3.09e-5	0.00640	0.00640
. . .		. . .	. . .	. . .	. . .
12.97	1024	2.03e-4	3.02e-8	1.76e-4	1.74e-4
6.48	2048	1.02e-4	7.55e-9	7.77e-5	7.55e-5
1	13280	1.57e-5	1.80e-10	5.23e-6	4.28e-6
		FE	SSP22	SSP33	SSP54
2	6640	unstable	unstable	unstable	8.74e-13
1	13280	1.57e-5	3.59e-10	4.42e-13	1.32e-12

Table 4: Convergence study for the heat equation with smooth initial conditions. Here $\nu = 1/4$, $m = 1024$, $t_f = 1$. The discrete error is computed against the ODE solution calculated with MATLAB’s `ode15s`. For comparison explicit methods are shown near their stability limits around $c = 1$.

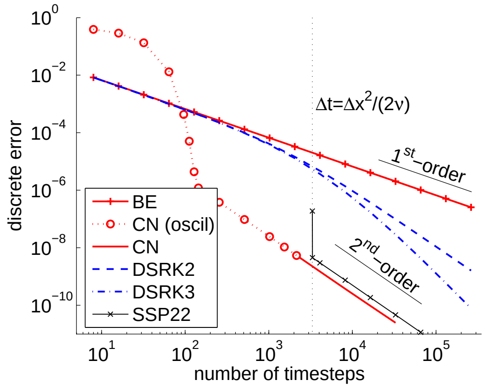


Figure 6: Heat equation with discontinuous initial conditions using $m = 512$, $t_f = 1$, and $\nu = \frac{1}{4}$. The spatial discretization in this example is second-order centered differences.

4.3 The Black–Scholes equation

The Black–Scholes equation [3]

$$V_\tau = \frac{\sigma}{2} S^2 V_{SS} + rSV_S - rV, \tag{15}$$

is a PDE used in computational finance [9] for determining the fair price V of an option at stock price S, where σ is the volatility and r is the risk-free interest rate. Note S is the independent (we can think "spatial") variable on the positive half-line and τ is a rescaled time (the actual time runs backwards from "final conditions"). We consider the initial conditions shown in Figure 7 which have a discontinuity in the first derivative at $S = 100$ (these initial conditions are known as a "put option" with a "strike price" of $S = 100$).

We note that for our purposes, (15) is a linear non-constant coefficient advection-reaction-diffusion equation and we treat it as the ODE system (1) by approximating the V_S term with first-order upwind finite differences and the V_{SS} term with second-order centered finite differences. We use $\sigma = 0.8$, $r = 0.1$ and for this choice we did not notice any significant difference between upwind and centered differences for the advection term. The right-hand boundary condition is an approximation to $\lim_{S\to\infty} V(S) = 0$, specifically $V(S_{\max}) = 0$. At the left-hand end of the domain, we note that (15) reduces to

$$\dot{V}_0 = -rV_0,$$

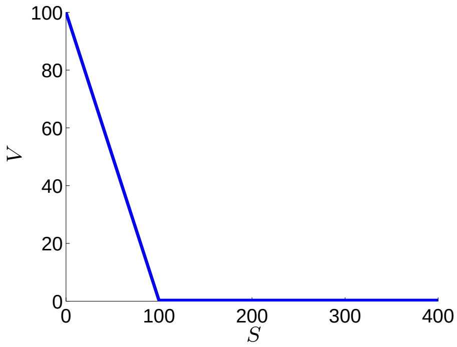


Figure 7: Computational domain and initial conditions for the Black–Scholes problem.

and thus it is both natural and convenient to simply solve this ODE coupled with the other components V_j as part of our method-of-lines computation.

Figure 8 shows the problem of oscillations which show up in a Crank–Nicolson calculation of the Black–Scholes problem. The oscillations are amplified in "the Greeks" i.e., the first and second spatial derivatives. We note this is a well-known phenomenon [5] associated with the CN numerical solution of (15); in practice, Rannacher time-stepping consisting of several initial steps of BE followed by CN steps [10] is often used to avoid these oscillations. DSRK schemes also avoid oscillations but are not likely competitive with Rannacher time-stepping in terms of efficiency due to the order reduction illustrated in Table 5. A great number of time-steps ($N = 17800$ in the case considered in Table 5) are required before the Crank–Nicolson calculation is completely oscillation-free in "the Greeks".

We note that explicit methods are not practical for this problem because of the severe linear stability restriction imposed by the diffusion term in (15). If an oscillation-free calculation is desired, then backward Euler is preferred over DSRK methods because DSRK methods cost more and offer essentially the same first-order convergence rates for step-sizes of practical interest. Moreover, DSRK schemes can offer little practical advantage over current Rannacher time-stepping techniques which attempt to combine the best aspects of backward Euler and Crank–Nicolson.

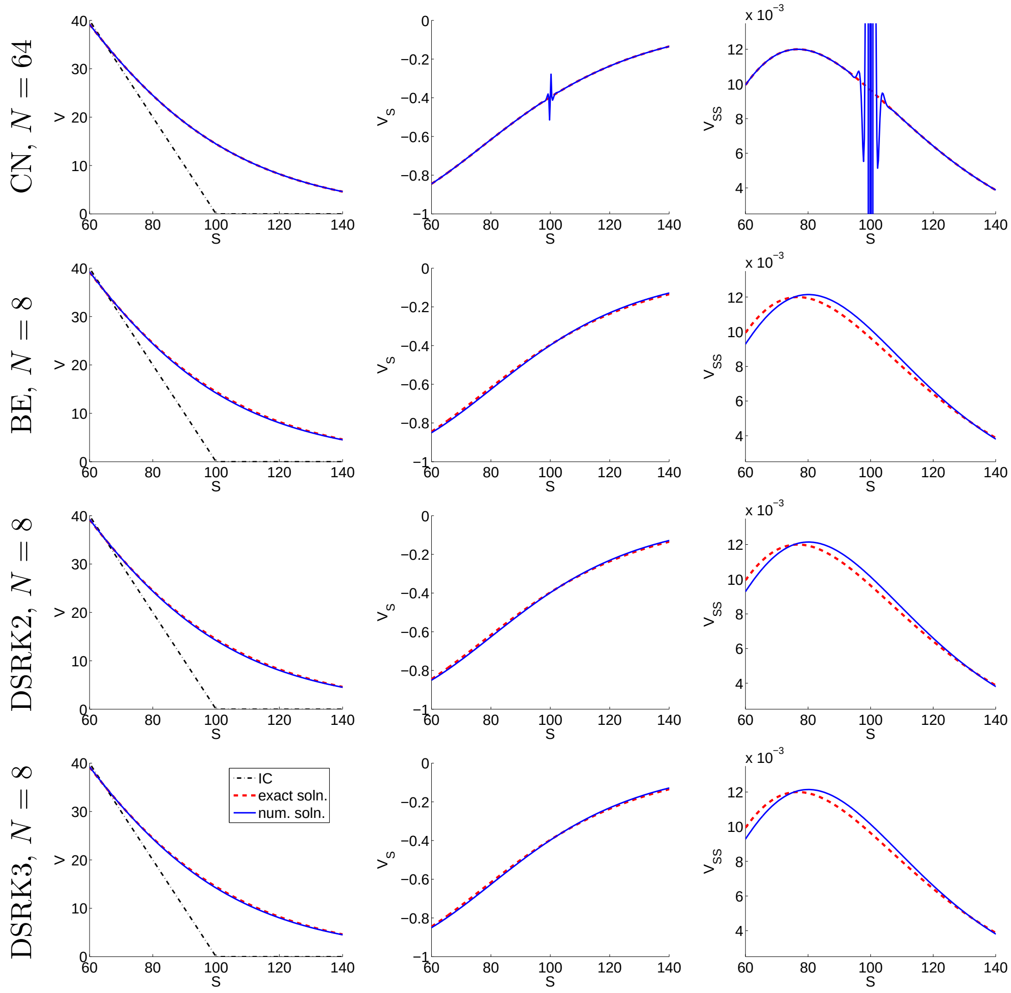


Figure 8: Numerical solutions of the Black–Scholes problem magnified near $S = 100$ with $m = 1600$, $t_f = \frac{1}{4}$, $\sigma = 0.8$, $r = 0.1$, and $S_{\max} = 400$ using N time-steps. From left-to-right: V, V_S and V_{SS}. Note that Crank–Nicolson exhibits oscillations with $N = 64$ whereas BE and the DSRK schemes appear free of oscillation even with the larger time-steps corresponding to $N = 8$.

	discrete error l_∞-norm							
N	BE	order	CN	order	DSRK2	order	DSRK3	order
32	0.0655		0.115 *		0.0654		0.0654	
64	0.0328	1.00	0.0452 *	1.35	0.0327	1.00	0.0326	1.00
128	0.0164	1.00	8.64e-3 *	2.39	0.0163	1.00	0.0163	1.00
256	8.21e-3	1.00	8.76e-5 *	6.62	8.07e-3	1.01	8.06e-3	1.02
512	4.10e-3	1.00	1.95e-6 *	5.49	3.97e-3	1.02	3.96e-3	1.03
1024	2.05e-3	1.00	4.88e-7 *	2.00	1.92e-3	1.05	1.91e-3	1.05
...	...		...		...		...	
8192	2.57e-4		7.62e-9 *		1.60e-4		1.51e-4	
16384	1.28e-4	1.00	1.90e-9 *	2.00	5.67e-5	1.50	4.98e-5	1.60
32768	6.41e-5	1.00	4.75e-10	2.00	1.78e-5	1.67	1.67e-5	1.58

Table 5: Black–Scholes convergence study. * indicates oscillations in V, V_S or V_{SS}. Here, $m = 1600$, $S_{\max} = 400$, $\Delta x = \frac{1}{4}$, $t_f = \frac{1}{4}$, $\sigma = 0.8$, and $r = 0.1$. The discrete error is calculated against a numerical solution from MATLAB's `ode15s` with `AbsTol` $= 1 \times 10^{-14}$, `RelTol` $= 1 \times 10^{-13}$.

4.4 Hyperbolic conservation laws: Burgers' equation

Up to now we have dealt exclusively with linear problems. In this Section we consider Burgers' equation

$$u_t = -f(u)_x = -\left(\frac{1}{2}u^2\right)_x,$$

with initial condition $u(0,x) = \frac{1}{2} - \frac{1}{4}\sin(\pi x)$ on the periodic domain $x \in [0,2)$. The solution is right-travelling and over time steepens into a shock. We discretize $-f(u)_x$ using a conservative simple upwind approximation

$$-f(u)_x \approx -\frac{1}{\Delta x}\left(\tilde{f}_{i+\frac{1}{2}} - \tilde{f}_{i-\frac{1}{2}}\right) = -\frac{1}{\Delta x}\left(f(u_i) - f(u_{i-1})\right).$$

Figure 9 shows that Crank–Nicolson produces spurious oscillations in the wake of the shock, for $c = 8$ (in fact, we observe oscillations from CN for $c \geq 4$ as noted in Table 6). As expected, BE, DSRK2 and DSRK3 produce a non-oscillatory TVD solution. Table 6 shows a convergence study for this problem which illustrates the familiar pattern of order reduction.

Notice, in particular, that for any time-step size considered, one of BE or CN gives non-oscillatory results with smaller errors than the DSRK schemes

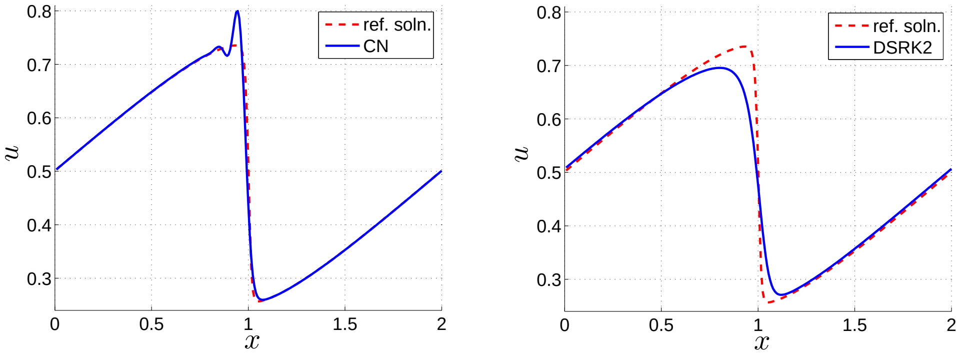


Figure 9: Burgers' equation with Crank–Nicolson (left) and DSRK2 (right) with $m = 256$ spatial points and $t_f = 2$, $N = 32$ ($c = 8$). For CN, the solution appears smooth until the shock develops, then an oscillation develops at the trailing edge of the shock. Note that DSRK2 appears overly dissipative. The reference solution is calculated with CN and $N = 8192$.

considered here. However, for small time-steps, the explicit methods clearly outperform the other choices.

5 Stage Order and Order Reduction

In our numerical experiments in Section 4, we have shown that the unconditionally contractive DSRK2 and DSRK3 methods preserve nonlinear stability properties when applied to our test cases in Section 4. Unfortunately, however, these methods suffer from order reduction. This implies that the unconditionally contractive DSRK methods are not likely a appropriate choice for a time-stepping scheme, because they cannot compete with BE for large time-steps or with SSP explicit methods for smaller time-steps.

5.1 The van der Pol equation

To further investigate the order reduction observed in the previous numerical tests, we apply the DSRK methods to the van der Pol equation, a problem often used for testing for reduction of order (see, e.g., [23] and references therein). The problem can be written as an ODE initial value problem con-

		error (l_∞-norm against ref. soln.)							
c	N	BE	order	CN	order	DSRK2	order	DSRK3	order
16	16	0.192		0.193 *		0.195		0.195	
8	32	0.173	0.15	0.109 *	0.82	0.153	0.35	0.154	0.34
4	64	0.140	0.31	0.0399 *	1.45	0.110	0.47	0.114	0.43
2	128	0.0964	0.54	0.0124	1.68	0.0644	0.78	0.0673	0.76
1	256	0.0589	0.71	3.11e-3	2.00	0.0273	1.24	0.0249	1.43
0.5	512	0.0320	0.88	7.72e-4	2.01	8.72e-3	1.65	6.79e-3	1.87
0.25	1024	0.0165	0.96	1.90e-4	2.02	2.45e-3	1.83	1.39e-3	2.29
		FE	order	SSP22	order	SSP33	order	SSP54	order
4	64	unstable		unstable		unstable		unstable	
2	128	unstable		unstable		unstable		2.50e-4	
1	256	0.0880		5.98e-3		3.54e-4		1.36e-5	4.20
0.5	512	0.0377	1.22	1.45e-3	2.04	4.32e-5	3.03	7.63e-7	2.88
0.25	1024	0.0172	1.13	3.63e-4	2.00	5.34e-6	3.02	4.46e-8	4.10
0.125	2048	8.43e-3	1.03	9.08e-5	2.00	6.61e-7	3.01	2.68e-9	4.06

Table 6: Burgers' equation convergence study. Values for which oscillations appear are indicated with *. The setup here is the same as in Figure 9 except the reference solution is calculated with SSP54 and $N = 8192$.

sisting of two components

$$y_1' = y_2, \tag{16a}$$

$$y_2' = \frac{1}{\epsilon}\left(-y_1 + (1 - y_1^2)y_2\right), \tag{16b}$$

with ϵ-dependent initial conditions [23, Table 5.1] and becomes increasingly stiff as ϵ is decreased. We solve until $t_f = \frac{1}{2}$.

Figure 10 shows the distinctive "flattening" [23] that occurs during the convergence studies whereby the error exhibits a region (depending on ϵ) of first-order behaviour as the step-size decreases before eventually approaching the design order of the method. This suggests that DSRK schemes suffer from order reduction whereas Crank–Nicolson does not. Before the flattened region, all the high-order methods produce similar errors. In particular DSRK3 does no better than the second-order Crank–Nicolson until after the flattening region. We note that this order reduction is noticeable despite the fact that our choices of ϵ do not correspond to particularly stiff systems.

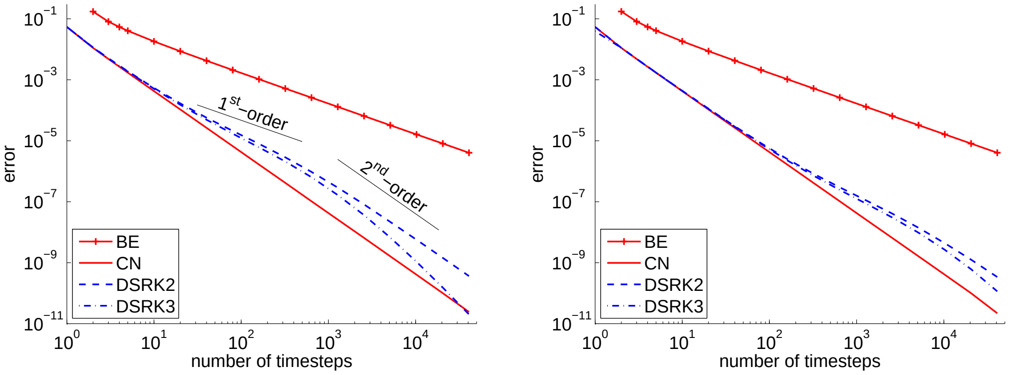


Figure 10: A convergence study on the van der Pol equation. Error shown is in the second component, where we have taken $\epsilon = 1 \times 10^{-3}$ (left) and $\epsilon = 1 \times 10^{-4}$ (right).

5.2 DSRK schemes with higher underlying stage order

The order reduction is not completely unexpected, as [20] showed that the underlying Runge–Kutta methods must have stage order at most one, and low stage order—at least in Runge–Kutta schemes—is known to lead to order reduction [14]. For comparison, we consider a DSRK method which is based on the two-stage, second-order, stage order two implicit Runge–Kutta method

$$\begin{array}{c|c} \boldsymbol{c} & \mathbf{A} \\ \hline & \boldsymbol{b}^{\mathrm{T}} \end{array} = \begin{array}{c|cc} \frac{1}{2} & \frac{3}{4} & -\frac{1}{4} \\ 1 & 1 & 0 \\ \hline & 1 & 0 \end{array}, \qquad \text{combined with } \mathbf{W} = \begin{bmatrix} \frac{1}{2} & 0 \\ 1 & 0 \end{bmatrix}. \tag{17}$$

We call this method DSRK2uso2. Because the underlying method has stage order larger than one (i.e., two), the DSRK2uso2 method cannot be unconditionally contractive [20].

5.3 DSRK schemes with higher stage order

Figures 11 and 12 indicate that DSRK2uso2 also suffers from order reduction. Thus it appears that higher stage order of the underlying Runge–Kutta scheme is not sufficient to avoid order reduction. We thus investigate stage order properties of the DSRK scheme itself by considering the test problem of [28]. However, because DSRK schemes reduce to Runge–Kutta schemes on scalar problems, we use a modified vector version

$$\boldsymbol{u}' = \boldsymbol{\Lambda}\left(\boldsymbol{u} - \boldsymbol{\phi}(t)\right) + \boldsymbol{\phi}'(t), \tag{18}$$

where $\boldsymbol{u}(t_0) = \boldsymbol{\phi}(t_0)$ and $\boldsymbol{\Lambda}$ is negative semidefinite, where the exact solution is $\boldsymbol{u}(t) = \boldsymbol{\phi}(t)$.

We apply a general DSRK scheme (8) to this test problem and, following Section IV.15 of [14], we use Taylor series expansions to determine the defect of each stage $\boldsymbol{U}^i$ and $\boldsymbol{Z}^i$ and the final $\boldsymbol{u}^{n+1}$. The order of each defect is determined by the relations

$$\boldsymbol{b}^{\mathrm{T}}\boldsymbol{c}^{k-1} = \frac{1}{k}, \qquad \text{for } k = 1, \ldots, q_0, \tag{19a}$$

$$\mathbf{A}\boldsymbol{c}^{k-1} = \frac{\boldsymbol{c}^k}{k}, \qquad \text{for } k = 1, \ldots, q_1, \tag{19b}$$

$$\mathbf{W}\boldsymbol{c}^{k-1} = \frac{\boldsymbol{c}^k}{k}, \qquad \text{for } k = 1, \ldots, q_2, \tag{19c}$$

where the $\boldsymbol{c}^k$ indicates component-wise exponentiation. We define the *stage order* of the DSRK method as $\min(q_0, q_1, q_2)$. Note that $\min(q_0, q_1)$ is the stage order of the underlying Runge–Kutta scheme and that $q_0 \geq p$, where p is the order of the DSRK scheme.

Our scheme DSRK2uso2 has $q_1 = 2$ and $q_2 = 1$. The DSRK2 scheme as $q_1 = 1$ and $q_2 = 2$. It does not appear possible to create two-stage second-order DSRK scheme with $q_1 = q_2 = 2$. However, we can find many three-stage second-order DSRK schemes with $q_1 = q_2 = 2$; a particular example is the method we call DSRK32so2 with

$$\begin{array}{c|c} \boldsymbol{c} & \mathbf{A} \\ \hline & \boldsymbol{b}^{\mathrm{T}} \end{array} = \begin{array}{c|ccc} 0 & \frac{1}{4} & -\frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{4} & 0 \\ 1 & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \hline & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \end{array}, \qquad \mathbf{W} = \begin{bmatrix} 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{12} & \frac{1}{12} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \end{bmatrix}.$$

We can also find third-order, three-stage DSRK methods with $q_1 = q_2 = 2$, for example, DSRK33so2 with

$$\begin{array}{c|c} \boldsymbol{c} & \mathbf{A} \\ \hline & \boldsymbol{b}^{\mathrm{T}} \end{array} = \begin{array}{c|ccc} 0 & \frac{1}{4} & -\frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & \frac{1}{2} & -\frac{1}{4} & \frac{1}{4} \\ 1 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}, \qquad \mathbf{W} = \begin{bmatrix} \frac{1}{3} & -\frac{2}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{12} & \frac{1}{12} \\ \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{bmatrix}.$$

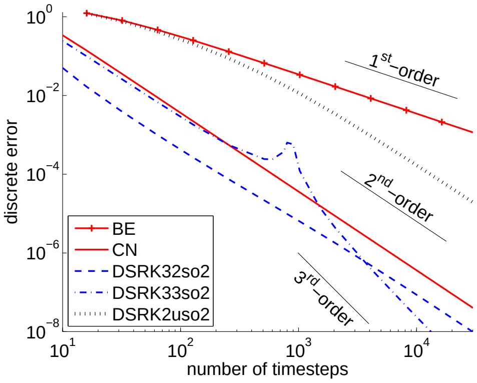


Figure 11: Stage Order convergence study for linear advection of a sine wave to $t_f = 1$. The spatial discretization here is first-order upwinding with $m = 2048$ points.

We reiterate that none of these higher stage order schemes can be unconditionally contractive and in numerical tests (not included) we observed that indeed, DSRK2uso2, DSRK32so2 and DSRK33so2 violated the strong stability property for large enough Δt.

Figures 11 and 12 show that the DSRK32so2 scheme is free from order reduction. However, we note that DSRK33so2 still exhibits order reduction as its stage order is one less than its design order.

The apparent importance of high stage order for DSRK schemes is intriguing especially because we do not observe order reduction when using implicit SSP schemes (which necessarily have stage order at most two) even when tested [21] on some of the same test problems used here.

6 Conclusions and Future Directions

We studied the performance of unconditionally contractive diagonally split Runge–Kutta (DSRK) schemes of orders two and three on a variety of archetypal test cases. The numerical tests verified the asymptotic order of the schemes as well as the unconditional contractivity property. However, in every numerical experiment, the unconditionally contractive DSRK methods were out-performed by the first-order backward Euler (BE) scheme when $\Delta t > 2\Delta t_{\mathrm{FE}}$, and by explicit Runge–Kutta methods or Crank–Nicolson (CN)

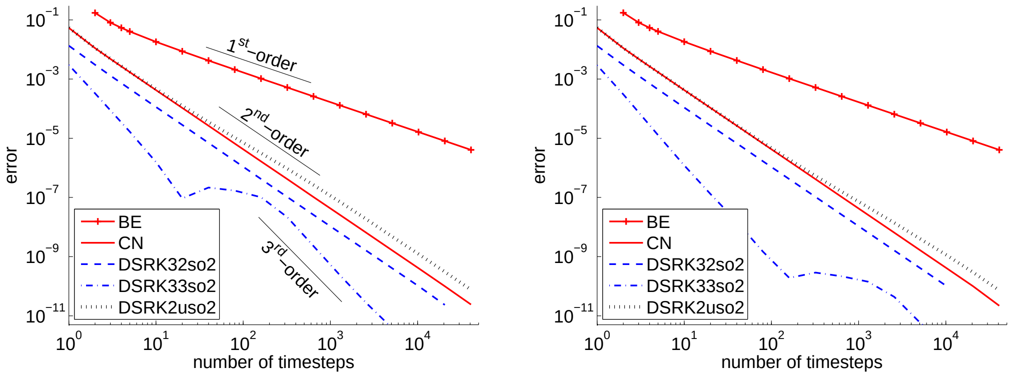


Figure 12: Stage order convergence study on the van der Pol equation. Error shown is in the second component and we have taken $\epsilon = 1 \times 10^{-3}$ (left) and $\epsilon = 1 \times 10^{-4}$ (right).

when $\Delta t \leq 2\Delta t_{\mathrm{FE}}$. At larger time-steps, the unconditionally contractive DSRK schemes are strong stability preserving (SSP) but suffer from order reduction, making BE a better choice. At small step-sizes, CN and explicit SSP Runge–Kutta methods are SSP, and produce far more accurate results at a smaller computational cost.

We showed that higher stage order of the underlying Runge–Kutta schemes was insufficient to avoid order reduction. We then derived DSRK stage order conditions and constructed DSRK schemes with stage order two which do not suffer from order reduction. However, because of the high stage order, these schemes cannot be unconditionally contractive.

The class of unconditionally contractive DSRK methods does not produce viable alternatives to well-established conditionally SSP Runge–Kutta and linear multistep methods. Recent research has focused on implicit and diagonally implicit Runge–Kutta [21, 6] as well as on General Linear Methods [4, 35]. This work is ongoing. Future research will focus on high stage order DSRK methods which are not unconditionally contractive, but which may have a large allowable step-size while not suffering from order reduction.

Acknowledgements

The authors thank Z. Jackiewicz for suggesting we look at stage order of the underlying Runge–Kutta schemes, and suggesting the method (17). We are deeply grateful to the anonymous referees for their suggestions and in particular for encouraging us to investigate stage order further.

References

[1] A. Bellen, Z. Jackiewicz, and M. Zennaro. Contractivity of waveform relaxation Runge–Kutta iterations and related limit methods for dissipative systems in the maximum norm. *SIAM J. Numer. Anal.*, 31(2):499–523, 1994.

[2] A. Bellen and L. Torelli. Unconditional contractivity in the maximum norm of diagonally split Runge–Kutta methods. *SIAM J. Numer. Anal.*, 34(2):528–543, 1997.

[3] F. Black and M. Scholes. The Pricing of Options and Corporate Liabilities. *The Journal of Political Economy*, 81(3):637–654, 1973.

[4] J. C. Butcher. General linear methods. *Acta Numer.*, 15:157–256, 2006.

[5] Thomas Coleman. Option pricing: The hazards of computing delta and gamma. website, 2006. `http://www.fenews.com/fen49/where_num_matters/numerics.htm`.

[6] L. Ferracina and M. N. Spijker. Strong stability of singly-diagonally-implicit Runge–Kutta methods. *Appl. Numer. Math.* to appear, doi:10.1016/j.apnum.2007.10.004.

[7] L. Ferracina and M. N. Spijker. Stepsize restrictions for the total-variation-diminishing property in general Runge–Kutta methods. *SIAM J. Numer. Anal.*, 42(3):1073–1093 (electronic), 2004.

[8] L. Ferracina and M. N. Spijker. An extension and analysis of the Shu–Osher representation of Runge–Kutta methods. *Math. Comp.*, 74(249):201–219 (electronic), 2005.

[9] P.A. Forsyth. An introduction to computational finance without agonizing pain. Available on author's website, `http://www.cs.uwaterloo.ca/~paforsyt/agon.pdf`, February 2005.

[10] Michael B. Giles and Rebecca Carter. Convergence analysis of Crank–Nicolson and Rannacher time-marching. *Journal of Computational Finance*, 9(4):89–112, 2006.

[11] Sigal Gottlieb. On high order strong stability preserving Runge–Kutta and multi step time discretizations. *J. Sci. Comput.*, 25(1-2):105–128, 2005.

[12] Sigal Gottlieb and Chi-Wang Shu. Total variation diminishing Runge–Kutta schemes. *Math. Comp.*, 67(221):73–85, 1998.

[13] Sigal Gottlieb, Chi-Wang Shu, and Eitan Tadmor. Strong stability-preserving high-order time discretization methods. *SIAM Rev.*, 43(1):89–112 (electronic), 2001.

[14] E. Hairer and G. Wanner. *Solving ordinary differential equations II: Stiff and differential-algebraic problems*, volume 14 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 1991.

[15] Inmaculada Higueras. On strong stability preserving time discretization methods. *J. Sci. Comput.*, 21(2):193–223, 2004.

[16] Inmaculada Higueras. Representations of Runge–Kutta methods and strong stability preserving methods. *SIAM J. Numer. Anal.*, 43(3):924–948 (electronic), 2005.

[17] Zoltán Horváth. Positivity of Runge–Kutta and diagonally split Runge–Kutta methods. *Appl. Numer. Math.*, 28(2-4):309–326, 1998. Eighth Conference on the Numerical Treatment of Differential Equations (Alexisbad, 1997).

[18] Willem Hundsdorfer and Steven J. Ruuth. On monotonicity and boundedness properties of linear multistep methods. *Math. Comp.*, 75(254):655–672 (electronic), 2006.

[19] Willem Hundsdorfer, Steven J. Ruuth, and Raymond J. Spiteri. Monotonicity-preserving linear multistep methods. *SIAM J. Numer. Anal.*, 41(2):605–623 (electronic), 2003.

[20] K. J. in 't Hout. A note on unconditional maximum norm contractivity of diagonally split Runge–Kutta methods. *SIAM J. Numer. Anal.*, 33(3):1125–1134, 1996.

[21] David I. Ketcheson, Colin B. Macdonald, and Sigal Gottlieb. Optimal implicit strong stability preserving Runge–Kutta methods. submitted.

[22] J. F. B. M. Kraaijevanger. Contractivity of Runge–Kutta methods. *BIT*, 31(3):482–528, 1991.

[23] Anita T. Layton and Michael L. Minion. Implications of the choice of quadrature nodes for Picard integral deferred corrections methods for ordinary differential equations. *BIT*, 45(2):341–373, 2005.

[24] H. W. J. Lenferink. Contractivity preserving explicit linear multistep methods. *Numer. Math.*, 55(2):213–223, 1989.

[25] H. W. J. Lenferink. Contractivity-preserving implicit linear multistep methods. *Math. Comp.*, 56(193):177–199, 1991.

[26] Colin B. Macdonald. Constructing high-order Runge–Kutta methods with embedded strong-stability-preserving pairs. Master's thesis, Simon Fraser University, August 2003.

[27] Stanley Osher and Ronald Fedkiw. *Level set methods and dynamic implicit surfaces*, volume 153 of *Applied Mathematical Sciences*. Springer-Verlag, New York, 2003.

[28] A. Prothero and A. Robinson. On the stability and accuracy of one-step methods for solving stiff systems of ordinary differential equations. *Math. Comp.*, 28:145–162, 1974.

[29] Steven J. Ruuth. Global optimization of explicit strong-stability-preserving Runge–Kutta methods. *Math. Comp.*, 75(253):183–207 (electronic), 2006.

[30] Steven J. Ruuth and Raymond J. Spiteri. Two barriers on strong-stability-preserving time discretization methods. *J. Sci. Comput.*, 17(1-4):211–220, 2002. Proceedings of the Fifth International Conference on Spectral and High Order Methods (ICOSAHOM-01) (Uppsala).

[31] N. V. Sahinidis and M. Tawarmalani. *BARON 7.2: Global Optimization of Mixed-Integer Nonlinear Programs,* User's Manual, 2004. Available at `http://www.gams.com/dd/docs/solvers/baron.pdf`.

[32] Chi-Wang Shu. Total-variation-diminishing time discretizations. *SIAM J. Sci. Statist. Comput.*, 9(6):1073–1084, 1988.

[33] Chi-Wang Shu and Stanley Osher. Efficient implementation of essentially nonoscillatory shock-capturing schemes. *J. Comput. Phys.*, 77(2):439–471, 1988.

[34] M. N. Spijker. Contractivity in the numerical solution of initial value problems. *Numer. Math.*, 42(3):271–290, 1983.

[35] M. N. Spijker. Stepsize conditions for general monotonicity in numerical initial value problems. *SIAM J. Numer. Anal.*, 45(3):1226–1245 (electronic), 2007.

[36] Raymond J. Spiteri and Steven J. Ruuth. A new class of optimal high-order strong-stability-preserving time discretization methods. *SIAM J. Numer. Anal.*, 40(2):469–491, 2002.

[37] Raymond J. Spiteri and Steven J. Ruuth. Non-linear evolution using optimal fourth-order strong-stability-preserving Runge–Kutta methods. *Math. Comput. Simulation*, 62(1-2):125–135, 2003. Nonlinear waves: computation and theory, II (Athens, GA, 2001).

[38] M. Zennaro. Contractivity of Runge–Kutta methods with respect to forcing terms. *Appl. Numer. Math.*, 11(4):321–345, 1993.