Stabilized explicit multirate methods for stiff differential equations

Assyr Abdulle Marcus J. Grote Giacomo Rosilho de Souza

June 1, 2020

Abstract

Stabilized Runge–Kutta (aka Chebyshev) methods are especially efficient for the numerical solution of large systems of stiff differential equations because they are fully explicit; hence, they are inherently parallel and easily accommodate nonlinearity. For semi-discrete parabolic (or diffusion dominated) problems, for instance, stabilized Runge–Kutta methods overcome the stringent stability condition of standard methods without sacrificing explicitness. However, when much of the stiffness is only induced by a few components, as in the presence of spatially local mesh refinement, their efficiency deteriorates. To remove the crippling effect of a few severely stiff components on the entire system of differential equations, we derive a modified equation, whose stiffness solely depend on the remaining mildly stiff components. By applying stabilized Runge–Kutta methods to this modified equation, we then devise an explicit multirate Runge–Kutta–Chebyshev (mRKC) method whose stability conditions are independent of a few severely stiff components. Stability of the mRKC method is proved for a model problem, whereas its efficiency and usefulness are demonstrated through a series of numerical experiments.

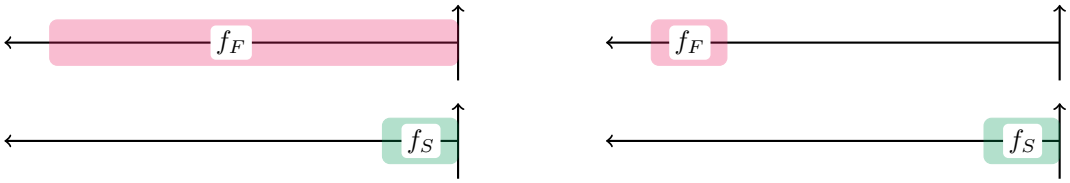
Key words. stabilized Runge–Kutta methods, explicit time integrators, stiff equations, multirate methods, local time-stepping, parabolic problems, Chebyshev methods.
AMS subject classifications. 65L04, 65L06, 65L20.

1 Introduction

We consider the system of stiff (nonlinear) differential equations,

$$y' = f(y) := f_F(y) + f_S(y), \qquad y(0) = y_0, \tag{1.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}^n$ splits into an expensive but only mildly stiff part, f_S, associated with relatively slow (S) time-scales, and a cheap but severely stiff part, f_F, associated with fast (F) time-scales. Typical applications include chemical reactions and electrical circuits with disparate time-scales, but also spatial discretizations of diffusion dominated (parabolic) partial differential equations (PDEs) with local mesh refinement. Semi-discrete parabolic PDEs, in particular, lead to large systems of stiff ordinary differential equations, where the eigenvalues of the Jacobian matrix, $\partial f/\partial y$, lie in a narrow strip along the negative real axis whose extent scales as H^{-2} for a mesh size H. In the presence of local mesh refinement, f_S, f_F in (1.1) then correspond to discrete diffusion operators in the coarse and locally refined region of the mesh, respectively. Although f_F involves only a small number of degrees of freedom, the extreme eigenvalue of its Jacobian will determine the spectral radius ρ of $\partial f/\partial y$.



(a) *Spectrum without scale separation.* (b) *Spectrum with scale separation.*

Figure 1. *Stiff problems with identical spectral radii.*

In contrast to multiscale methods [3, 9, 10, 46], *we do not assume any scale separation* in (1.1); hence, f_F may contain *both fast and slow* scales. In a situation of local mesh refinement, for instance, when f_S and f_F each represent the discrete Laplacian in the coarse and refined regions, both Jacobians in fact contain small eigenvalues in magnitude; hence, the spectrum of $\partial f/\partial y$ cannot simply be split into fast and slow modes, as in Figure 1(a). This stands in sharp contrast to the underlying assumption of recently introduced multiscale methods for stiff (dissipative) ODEs, such as the heterogeneous multiscale method (HMM) [9, 10] or the projective method [16] which all require scale separation, as in Figure 1(b).

Standard explicit methods are notoriously inefficient for stiff differential equations due to their stringent stability constraint on the step size, τ, which for parabolic problems must be proportional to H^2. Implicit methods, on the other hand, are unconditionally stable but require at every time step the solution of an $n \times n$ linear (or possibly nonlinear) system of equations, a high price to pay when n is large. Moreover, when sheer size calls for using iterative methods, the overall performance heavily relies on the availability of efficient preconditioners while the convergence of Newton-like nonlinear iterations is not even guaranteed for larger step sizes.

Stabilized Runge–Kutta (RK) (or Chebyshev) methods fall somewhere between explicit and implicit methods: they are explicit and thus avoid the solution of large systems of equations, while their stability interval on the negative real axis is proportional to s^2 for an s-stage method. Thanks to this remarkable quadratic dependency, the work load (number of stages s) per time step only needs to scale linearly with H^{-1} for parabolic PDEs, in contrast to the quadratic increase in the number of time steps required by standard explicit integrators. Stabilized RK methods are thus particularly efficient for the time integration of large-scale, possibly nonlinear, parabolic PDEs. Several stabilized RK methods have been proposed in the literature, such as DUMKA methods, based on the composition of Euler steps [28, 29, 31], Runge–Kutta–Chebyshev (RKC) methods, based on the linear combination of Chebyshev polynomials [40, 45, 49] and orthogonal Runge–Kutta–Chebyshev methods (ROCK), based on optimal orthogonal stabilized functions [1, 2]; note that ROCK and RKC methods differ only beyond order one. Still, when applied to (1.1), the number of stages s of any standard stabilized RK method will, yet again, be determined by the stiffest part f_F and the method eventually become inefficient.

To overcome the stringent step size restriction due to the cheap but stiffer part, f_F, while retaining the efficiency of explicit time integration for f_S, *multirate methods* use a smaller step size, or even an entirely different scheme, for integrating f_F. Since the early work of Rice [34] and the work of Gear and Wells [17] who proposed a number of multirate strategies for the interlaced time integration of the "fast" and "slow" components using classical multistep schemes, various explicit, implicit or hybrid multirate schemes have been developed based on Runge–Kutta methods using splitting techniques for "fast" and "slow" components or extrapolation techniques [5, 11, 20, 21, 25, 27, 36, 37, 39]. All these methods require a predictor step and either interpolate or extrapolate between "fast" and "slow"

state variables, which is prone to instability. Although some of the implicit-explicit (IMEX) methods are provably stable, they are more cumbersome to implement and rapidly become too expensive as the number of "fast" unknowns increases.

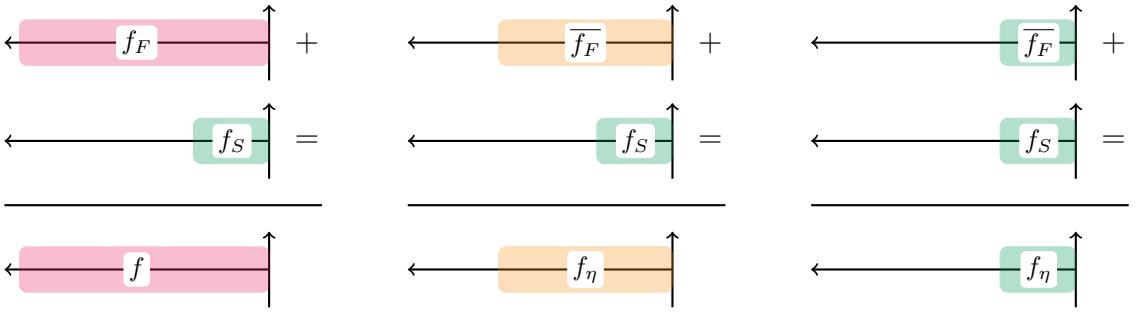
Local adaptivity and mesh refinement are certainly key to the efficient numerical solution of PDEs with heterogeneous media or complex geometry. Locally refined meshes, however, also cause a severe bottleneck for any standard explicit time integration, as the maximal time-step is dictated by maybe a few small elements in the mesh. To overcome the crippling effect of local mesh refinement, various multirate (or local time-stepping) methods [18] were proposed following the original local adaptive mesh refinement (AMR) strategy for first-order hyperbolic conservation laws by Berger and Oliger [6] — see [15] for a review. For parabolic problems, Ewing et al. [13, 14] derived and analyzed implicit finite difference schemes when local refinement is utilized in space and time. Dawson, Du and Dupont [7] combined implicit time integration in subdomains with an explicit treatment of the interfaces, which leads to a decoupled but conditionally stable system. In [32, 38], various predictor-corrector and domain decomposition methods were combined to iteratively correct the solution or its boundary values at artificial interfaces. By using static-regridding, Trompert and Verwer [41, 42, 43, 44] developed a number of multirate time-stepping strategies for local uniform grid refinement (LUGR), where a first integration is performed on a global coarse grid and the accuracy is iteratively improved locally on nested and increasingly finer subgrids.

In contrast to the above implicit, or locally implicit, multirate strategies, fully explicit RKC time integration was recently combined with the AMR approach [4, 33] to tackle diffusion dominated problems. Again, the mesh is divided into two distinct regions, the "coarse region," which contains the larger elements and corresponds to the mildly stiff part f_S, and the "fine region," which contains the smallest elements and thus corresponds to the severly stiff component f_F. In either subregion, the number of stages s in chosen according to the local mesh size, while "ghost cell" values at the coarse-to-fine interface are obtained by interpolating in time between stage values. For certain problems, however, time interpolation of missing stage values from the other RKC method can cause numerical instabilities [4].

To overcome the stringent stability condition due to a few severely stiff degrees of freedom, we first introduce in Section 2 a *modified equation*, where the spectral radius of its right-hand side, or *averaged force*, is bounded by that of the slower term f_S yet still remains a good approximation of (1.1). Evaluation of this averaged force requires the solution to a stiff, but cheap, auxiliary problem over short time and forms the basis of our multirate strategy. In fact, the numerical integration of the modified equation by any explicit method via such a multirate approach will be more efficient than integrating (1.1) directly with the same explicit method. In Section 3, we devise an explicit multirate Runge–Kutta–Chebyshev (RKC) method by utilizing two different RKC methods to integrate the modified equation and evaluate the averaged force. The resulting multirate RKC (mRKC) method assumes no scale separation, requires no interpolation between stages, and remains accurate even if the roles of f_F and f_S change in time. Stability and accuracy of the mRKC scheme are analyzed in Section 4. Finally, in Section 5, we apply our mRKC method to a series of test problems from both stiff ordinary and partial differential equations to demonstrate its usefulness and efficiency.

2 Averaged force, modified equation and multirate algorithm

First, we introduce the *modified equation* where f in (1.1) is replaced by an *averaged force*, f_η, which depends on a free parameter $\eta \geq 0$. For $\eta = 0$ it holds $f_\eta = f$ whereas for $\eta > 0$, the spectrum of f_η is compressed and thus f_η is less stiff than f, see Figure 2. In fact for $\eta > 0$ sufficiently large, the spectral radius ρ_η of the Jacobian of f_η is bounded by the spectral radius ρ_S of the Jacobian of f_S,



(a) *For* $\eta = 0$, $f = f_\eta$.

(b) *For* $\eta > 0$, *the fast term is replaced by a mean* $\overline{f_F}$ *and* f_η *becomes less stiff than* f.

(c) *For* $\eta > 0$ *sufficiently large, the spectrum of* f_η *and* f_S *are comparable.*

Figure 2. *Spectrum of* f *and of the averaged force* f_η *for varying* η.

i.e. $\rho_\eta \leq \rho_S$; then, the stiffness of the modified equation depends solely on f_S and its integration by any explicit method is cheaper than (1.1) integrated with the same method. Since the condition $\rho_\eta \leq \rho_S$ is already satisfied for η relatively small, f_η actually remains a good approximation of f. Next, we devise a multirate strategy based on the modified equation, which is implemented in Section 3 using two separate RKC methods. Finally, we analyze the properties of f_η, derive a priori error bounds for the solution of the modified equation and perform a stability analysis.

2.1 Averaged force and modified equation

We now define an average f_η of f such that the solution y_η of *the modified equation*,

$$y'_\eta = f_\eta(y_\eta), \qquad y_\eta(0) = y_0 \tag{2.1}$$

is a good approximation of the exact solution y of (1.1), yet the stiffness of (2.1) only depends on f_S.

Definition 2.1. *Let* $\eta \geq 0$, $u_0 \in \mathbb{R}^n$ *and* $u : [0, \eta] \to \mathbb{R}^n$ *be defined by the auxiliary equation,*

$$u' = f_F(u) + f_S(u_0) \qquad t \in [0, \eta], \qquad u(0) = u_0. \tag{2.2}$$

For $\eta > 0$, *define the averaged force* f_η *as*

$$f_\eta(u_0) = \frac{1}{\eta}(u(\eta) - u_0) \tag{2.3}$$

and for $\eta = 0$, *let* $f_0 = f$.

From (2.2) and (2.3) with $u_0 = y_\eta$, it follows that

$$f_\eta(y_\eta) = \frac{1}{\eta}\int_0^\eta u'(s)\,\mathrm{d}s = f_S(y_\eta) + \frac{1}{\eta}\int_0^\eta f_F(u(s))\,\mathrm{d}s.$$

Hence, f_η is an average of f over the time interval $[0, \eta]$ along the *auxiliary solution* u. For η sufficiently large, we show in Section 2.3 that $\rho_\eta \leq \rho_S$.

Multirate strategy Starting from the modified equation (2.1), we propose the following explicit multirate strategy for the numerical approximation of (1.1): Solve (2.1) with an explicit numerical method using a large time step determined by ρ_S, while solving (2.2) with a (possibly different) explicit method using a smaller time step determined by f_F, whenever an evaluation of f_η is needed. Since f_F is cheap, evaluating f_η in (2.1) by solving (2.2) carries about the same computational cost as evaluating f in (1.1). On the other hand, since the stiffness, and hence the step size, needed for (2.1) no longer depends on the fastest scales in the problem, the number of expensive f_S evaluations will be greatly reduced.

Although the above multirate strategy might at first resemble recently introduced multiscale methods, such as HMM for dissipative ODEs, it is fundamentally different: First, the averaged equation (2.1) requires no scale separation, as both f_S and f_F may contain slow scales, in contrast to the effective equations derived in [9], for instance. Second, the fastest scales present in (2.1) are comparable to those of f_S in (1.1).

2.2 A priori error analysis for the solution of the modified equation

Here we analyze the effect of the parameter η on f_η and show that f_η satisfies a one-sided Lipschitz condition, which is fundamental for proving convergence and contractivity for general nonlinear problems [23, IV.12]. Then, we derive bounds on the error introduced by solving (2.1) instead of (1.1), which are independent of the problem's stiffness.

Let $\langle\cdot,\cdot\rangle$ and $\|\cdot\|$ denote the standard Euclidean scalar product and norm in $\mathbb{R}^n$, respectively. To begin, we prove that f_F has a smoothing effect on f_η, if it satisfies a one-sided Lipschitz condition.

Lemma 2.2. *Let $\mu_F \in \mathbb{R}$ and f_F satisfy*

$$\langle f_F(z) - f_F(y), z - y\rangle \le \mu_F \|z - y\|^2 \quad \forall z, y \in \mathbb{R}^n. \tag{2.4}$$

Then

$$\|f_\eta(u_0)\| \le \varphi(\eta\mu_F)\|f(u_0)\|, \quad \text{where} \quad \varphi(z) = \frac{e^z - 1}{z} \ \text{ for } \ z \neq 0 \tag{2.5}$$

and $\varphi(0) = 1$ is defined by continuous extension. Moreover, if $f_F(y) = A_F\, y$ with $A_F \in \mathbb{R}^{n\times n}$, then

$$f_\eta(u_0) = \varphi(\eta A_F) f(u_0). \tag{2.6}$$

Proof. Let $v : [0, \eta] \to \mathbb{R}^n$ be defined by $v(s) = u_0$ for all s. We set

$$\delta := \|v'(s) - f_F(v(s)) - f_S(u_0)\| = \|f(u_0)\|.$$

Since the logarithmic norm of the Jacobian of f_F is bounded by μ_F, we obtain from a classical result on differential inequalities (see [22, Chapter I.10, Theorem 10.6])

$$\|u(\eta) - u_0\| = \|u(\eta) - v(\eta)\| \le e^{\eta\mu_F} \int_0^\eta e^{-s\mu_F}\, \delta \,\mathrm{d}s = \eta\varphi(\eta\mu_F)\|f(u_0)\|,$$

which yields (2.5) by (2.3).

Now, let $f_F(u) = A_F\, u$ with $A_F \in \mathbb{R}^{n\times n}$ nonsingular. Then, the variation-of-constants formula with $f_F(u) = A_F\, u$ in (2.2) yields

$$u(\eta) = e^{A_F\eta}\left(u_0 + \int_0^\eta e^{-A_F s} f_S(u_0)\,\mathrm{d}s\right) = e^{A_F\eta}u_0 + A_F^{-1}(e^{A_F\eta} - I) f_S(u_0),$$

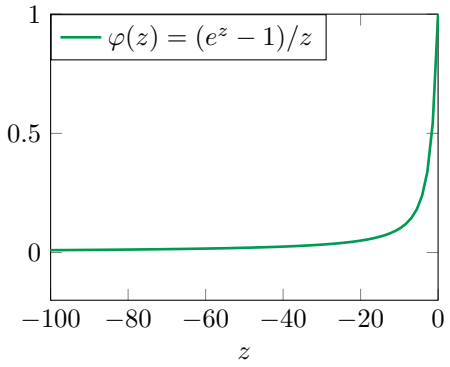


Figure 3. *The entire function* $\varphi(z)$.

with I the identity matrix. Hence,

$$u(\eta) = e^{A_F \eta} u_0 + \eta \varphi(\eta A_F) f_S(u_0). \tag{2.7}$$

Since φ has no poles, $u(\eta)$ as in (2.7) is well-defined and satisfies (2.2) for all matrices A_F. By using (2.3), we thus obtain

$$f_\eta(u_0) = \frac{1}{\eta}(e^{A_F \eta} - I)u_0 + \varphi(\eta A_F) f_S(u_0) = \varphi(\eta A_F)(A_F u_0 + f_S(u_0)) = \varphi(\eta A_F) f(u_0). \qquad \square$$

Remark 2.3. *The entire function* $\varphi(z)$ *is quite common in the theory of exponential integrators [24]. Indeed, when* $f_F(y) = A_F\, y$*, the solution* (2.7) *to the auxiliary problem* (2.2) *corresponds to a single step of the exponential Euler method applied to* (1.1)*. Our multirate approach, however, differs from exponential integrators: First,* $u(\eta)$ *is just an auxiliary solution used to compute* f_η*, which is distinct from the solution* y_η *of the modified equation* (2.1)*. Second, we do not use an exponential integrator but an RKC method to obtain* $u(\eta)$*; thus,* $\varphi(\eta A_F)$ *is never computed explicitly. Third,* η *is not the step size here but a free parameter indicating the length of the integration interval in* (2.2)*.*

The function $\varphi(z)$, shown in Figure 3, satisfies

$$\varphi(0) = 1, \qquad \varphi'(0) = \frac{1}{2}, \qquad \lim_{z \to -\infty} \varphi(z) = 0, \qquad 0 < \varphi(z) < 1, \quad \forall z < 0.$$

Hence, if A_F is negative definite, multiplication of f by $\varphi(\eta A_F)$ in (2.6) has a smoothing effect, which can be tuned by varying $\eta \geq 0$ — see Theorem 2.7 below. A similar property holds for any nonlinear f_F that is contractive, i.e. with $\mu_F \leq 0$ in (2.4), because of (2.5). Next, we prove under certain assumptions on the Jacobians of f_F and f_S that the averaged force f_η satisfies a one-sided Lipschitz condition.

Theorem 2.4. *Let* $A_F \in \mathbb{R}^{n \times n}$ *be symmetric and* $f_F(y) = A_F\, y$*. Suppose*

$$\langle \frac{\partial f}{\partial y}(w)(z - y), z - y \rangle \leq \mu \|z - y\|^2 \quad \forall w, y, z \in \mathbb{R}^n \tag{2.8}$$

with $\mu \leq 0$ *and that* $A_F \frac{\partial f_S}{\partial y}(w) = \frac{\partial f_S}{\partial y}(w) A_F$ *for all* $w \in \mathbb{R}^n$*. Then,*

$$\langle f_\eta(z) - f_\eta(y), z - y \rangle \leq \mu_\eta \|z - y\|^2,$$

where $\mu_\eta = \mu \min_{\lambda \in \lambda(A_F)} \{\varphi(\eta \lambda)\} \leq 0$ *and* $\lambda(A_F)$ *is the spectrum of* A_F*.*

Proof. Let $y, z \in \mathbb{R}^n$ and $w(r) = rz + (1-r)y$ for $r \in [0,1]$. We have

$$\begin{aligned}\langle f_\eta(z) - f_\eta(y), z - y\rangle &= \langle \varphi(\eta A_F)(f(z) - f(y)), z - y\rangle \\ &= \langle \varphi(\eta A_F)^{1/2} \int_0^1 \frac{\partial f}{\partial y}(w(r))(z-y)\,\mathrm{d}r, \varphi(\eta A_F)^{1/2}(z-y)\rangle.\end{aligned}$$

Since $\varphi(z) > 0$ for all z, $\varphi(\eta A_F)$ is symmetric positive definite and $\varphi(\eta A_F)^{1/2}$ exists. By hypothesis, $\varphi(\eta A_F)^{1/2}$ and $\frac{\partial f}{\partial y}(w(r))$ commute. Therefore

$$\begin{aligned}\langle f_\eta(z) - f_\eta(y), z - y\rangle &= \int_0^1 \langle \frac{\partial f}{\partial y}(w(r))\varphi(\eta A_F)^{1/2}(z-y), \varphi(\eta A_F)^{1/2}(z-y)\rangle\,\mathrm{d}r \\ &\le \mu \|\varphi(\eta A_F)^{1/2}(z-y)\|^2 \le \mu \min_{\lambda\in\lambda(A)}\{\varphi(\eta\lambda)\}\|z-y\|^2.\end{aligned}$$

□

Theorem 2.4 shows that f_η indeed satisfies a one-sided Lipschitz condition, if the Jacobians of f_F, f_S commute and (2.8) holds, which is slightly stronger than requiring that f satisfies a one-sided Lipschitz condition. Indeed, f is one-sided Lipschitz if, and only if, (2.8) holds for all z, y and $w \in [z, y]$, see [22, I.10]. Next, we bound the error between the solutions of (1.1) and (2.1).

Theorem 2.5. *Under the assumptions of Theorem 2.4, it holds*

$$\|y(t) - y_\eta(t)\| \le \max_{\lambda\in\lambda(A_F)} |1 - \varphi(\eta\lambda)| \int_0^t e^{\mu_\eta(t-s)}\|f(y(s))\|\,\mathrm{d}s, \tag{2.9}$$

with $\mu_\eta = \mu \min_{\lambda\in\lambda(A_F)}\{\varphi(\eta\lambda)\} \le 0$.

Proof. Let y be the solution to (1.1). From Theorem 2.4, we have

$$\begin{aligned}\|y'(t) - f_\eta(y(t))\| &= \|f(y(t)) - f_\eta(y(t))\| = \|(I - \varphi(\eta A_F))f(y(t))\| \\ &\le \max_{\lambda\in\lambda(A_F)} |1 - \varphi(\eta\lambda)|\|f(y(t))\| =: \delta(t).\end{aligned}$$

Since the logarithmic norm of the Jacobian of f_η is bounded by $\mu_\eta = \mu \min_{\lambda\in\lambda(A_F)}\{\varphi(\eta\lambda)\}$, as implied by Theorem 2.4, the estimate (2.9) follows from classical results on differential inequalities (see [22, Chapter I.10, Theorem 10.6]). □

Note that the error bound (2.9) is independent of the stiffness present in f_F and that $\|y(t) - y_\eta(t)\| \le C\eta$ as $\eta \to 0$ because $\varphi(\eta\lambda) = 1 + \mathcal{O}(\eta)$.

2.3 Stability analysis of the modified equation

We now study the stiffness of the modified equation (2.1) given by the spectral radius ρ_η of the Jacobian of f_η. In particular, we determine conditions which guarantee that $\rho_\eta \le \rho_S$, with ρ_S the spectral radius of the Jacobian of f_S, and hence that the stiffness of the modified equation only depends on the slow components. In doing so, we assume as in Theorem 2.4 that the Jacobians of f_F and f_S commute and are therefore simultaneously triangularizable. Then, the stability analysis of (1.1) and (2.1) reduces to the scalar *multirate test equation*

$$y' = \lambda y + \zeta y, \qquad y(0) = y_0, \tag{2.10}$$

with $\lambda, \zeta \le 0$ *and* $y_0 \in \mathbb{R}$, *which corresponds to setting* $f_F(y) = \lambda y$ *and* $f_S(y) = \zeta y$; *thus,* $\rho_F = |\lambda|$ *and* $\rho_S = |\zeta|$. Since we do not assume any scale separation, λ can take any nonpositive value.

Since (2.10) satisfies the hypotheses of Lemma 2.2 with $\mu_F = A_F = \lambda$, we have

$$u(\eta) = [e^{\eta\lambda} + \varphi(\eta\lambda)\eta\zeta]u_0, \tag{2.11}$$

$$f_\eta(u_0) = \varphi(\eta\lambda)(\lambda+\zeta)u_0 \tag{2.12}$$

and (2.1) reduces to

$$y'_\eta = \varphi(\eta\lambda)(\lambda+\zeta)y_\eta, \qquad y_\eta(0) = y_0. \tag{2.13}$$

Next, we detemine conditions on η, λ and ζ which guarantee that

$$|\varphi(\eta\lambda)(\lambda+\zeta)| \le |\zeta|,$$

and hence that the stiffness of (2.13) exclusively depends on $\rho_S = |\zeta|$. The following technical lemma is used to prove Theorem 2.7 below.

Lemma 2.6. *Let $w \le 0$ and $\varphi(z)$ be given by (2.5). Then, $\varphi(z)(z+w) \in [w, 0]$ for all $z \le 0$ if, and only if, $\varphi'(0)|w| \ge 1$, i.e. $|w| \ge 2$ since $\varphi'(0) = 1/2$.*

Proof. For $z, w \le 0$, the upper bound $\varphi(z)(z+w) \le 0$ always holds. Hence, we only need to consider the lower bound,

$$\varphi(z)(z+w) \ge w. \tag{2.14}$$

Suppose that (2.14) holds for all $z \le 0$. In a neighborhood of $z = 0$, this yields

$$\begin{aligned} 0 \le \varphi(z)(z+w) - w &= (\varphi(0) + \varphi'(0)z + \mathcal{O}(z^2))(z+w) - w \\ &= z(1 + \varphi'(0)(z+w)) + \mathcal{O}(z^2(z+w)), \end{aligned} \tag{2.15}$$

where we have used that $\varphi(0) = 1$. Dividing (2.15) by $z < 0$ and letting $z \to 0$ yields $\varphi'(0)w \le -1$, and thus $\varphi'(0)|w| \ge 1$.

Now, let $\varphi'(0)|w| \ge 1$ and hence $w \le -1/\varphi'(0) = -2$. For $z = 0$, (2.14) trivially holds. For $z < 0$, we multiply (2.14) by z and prove the resulting equivalent condition:

$$\alpha(z) = zw + (1 - e^z)(z+w) \ge 0 \quad \forall z < 0.$$

Since

$$\alpha'(z) = 1 + w - (1 + w + z)e^z, \qquad \alpha''(z) = -(2 + w + z)e^z,$$

we have $\alpha(0) = \alpha'(0) = 0$. Since $w \le -2$, we have

$$\alpha(z) = \int_0^z \int_0^s \alpha''(r)\,\mathrm{d}r\,\mathrm{d}s = -\int_z^0 \int_s^0 (2 + w + r)e^r\,\mathrm{d}r\,\mathrm{d}s \ge 0,$$

which concludes the proof. □

Theorem 2.7. *Let $\zeta < 0$. Then, $\varphi(\eta\lambda)(\lambda+\zeta) \in [\zeta, 0]$ for all $\lambda \le 0$ if, and only if, $\eta \ge 2/|\zeta|$.*

Proof. Setting $z = \eta\lambda$ and $w = \eta\zeta$, we have that

$$\varphi(\eta\lambda)(\lambda+\zeta) \in [\zeta, 0] \qquad \text{is equivalent to} \qquad \varphi(z)(z+w) \in [w, 0].$$

In view of Lemma 2.6, this holds for all $\lambda \le 0$, if and only if $\eta|\zeta| = |w| \ge 2$. □

Theorem 2.7 implies that for $\eta \ge 2/\rho_S$ the stiffness of (2.13) depends only on the slow term f_S. Since η does not depend on λ and the result holds for all $\lambda \le 0$, there is no need for any assumption on scale separation.

3 A stabilized method based on the modified equation: the multirate Runge–Kutta–Chebyshev method

Although the modified equation (2.1) has reduced stiffness, implementing a multirate strategy based on (2.1) and (2.2) with classical explicit methods remains inefficient, as it will lead to step size restrictions due to their inherent stiffness. Instead, we introduce here the mRKC method, which is based on two RKC methods and thus has no step size restrictions. Moreover, thanks to the multirate strategy, the number of f_S (expensive) evaluations is independent of the stiffness of f_F and thus no longer suffers from the efficiency loss of any classical stabilized scheme. Furthermore, the mRKC method is explicit, requires no interpolation or extrapolation, and makes no assumption about scale separation. Lastly, the role of f_F and f_S can even change in time without compromising the accuracy of the solution. In Section 3.1 we briefly recall some basic definitions and properties of the RKC scheme before introducing the mRKC method in Section 3.2

3.1 Stabilized Runge–Kutta methods

In this section we recall stabilized Runge–Kutta methods, which are explicit one-step Runge–Kutta (RK) methods with an extended stability domain along the negative real axis.

The simplest method for the numerical solution of (1.1) is the explicit Euler method, $y_{n+1} = y_n + \tau f(y_n)$, where τ is the step size, $t_n = t_{n-1} + \tau$ and y_n an approximation of $y(t_n)$. When applied to the test equation

$$y' = \lambda y, \qquad y(0) = y_0, \tag{3.1}$$

with $\lambda \in \mathbb{C}^- := \{z \in \mathbb{C} : \mathrm{Re}(z) \leq 0\}$, it yields $y_{n+1} = R(z)y_n$, where $z = \tau\lambda$ and $R(z) = 1 + z$. Recursive application then yields $y_n = R(z)^n y_0$. Since $y(t_n) = e^{t_n\lambda}y_0$, $|y(t_n)| \leq |y_0|$ if $\mathrm{Re}(\lambda) \leq 0$. A similar property is desirable for the numerical solution and it is clear that $|y_n| \leq |y_0|$ if, and only if, $|R(z)| \leq 1$. A similar derivation holds for any explicit RK method and we call

$$\mathcal{S} = \{z \in \mathbb{C} : |R(z)| \leq 1\}$$

the stability domain and $R(z)$ the stability polynomial of the method. For the stable numerical integration of (3.1), the step size τ has to satisfy $z = \tau\lambda \in \mathcal{S}$. For the explicit Euler method, for instance, and $\lambda \in \mathbb{R}^-$, it holds $z \in \mathcal{S}$ if and only if $z \in [-2, 0]$. Thus, for $\lambda < 0$ and $|\lambda|$ large, the stability condition $\tau \leq 2/|\lambda|$ imposes a very stringent restriction on the step size.

Stabilized Runge–Kutta methods [1, 2, 28, 29, 31, 40, 45, 49] use an increased number of stages to improve stability properties along the negative real axis and thereby relax the stringent constraint of standard explicit RK methods on the step size. Given $s \in \mathbb{N}$, the stability polynomial $R_s(z)$ of a first-order s-stage stabilized scheme has degree s and solves the optimization problem:

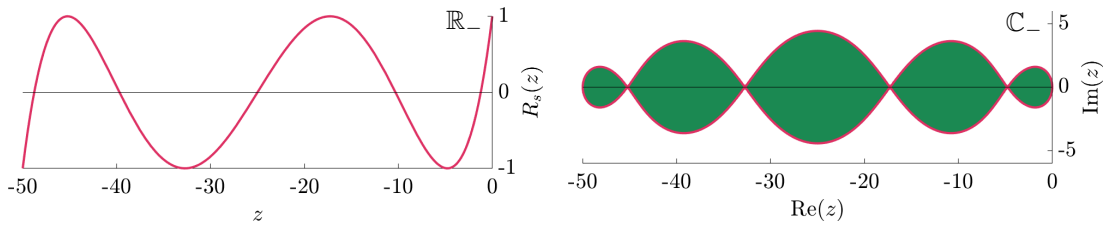
$$R_s(0) = R_s'(0) = 1 \qquad \text{and} \qquad |R_s(z)| \leq 1 \text{ for } z \in [-\ell_s, 0] \text{ with } \ell_s \text{ maximal.} \tag{3.2}$$

The first condition in (3.2) is necessary to guarantee first-order accuracy, as an RK scheme has first order if, and only if, its stability polynomial $R(z)$ satisfies $R(z) - e^z = \mathcal{O}(z^2)$. The second condition yields the longest stability domain along the negative real axis.

For first-order methods, the unique solution to (3.2) is $R_s(z) = T_s(1 + z/s^2)$, where $T_s(x)$ is the Chebyshev polynomial of the first kind [19, 48]. It is defined recursively by

$$T_0(x) = 1, \qquad T_1(x) = x, \qquad T_j(x) = 2xT_{j-1}(x) - T_{j-2}(x).$$

Since $T_s(x) = \cos(s \arccos(x))$ for $x \in [-1, 1]$, $R_s(0) = R_s'(0) = 1$ is satisfied. Moreover, if $z \in [-2s^2, 0]$ then $1 + z/s^2 \in [-1, 1]$ and thus $|R_s(z)| \leq 1$.



(a) *Stability polynomial* $R_s(z)$. (b) *Stability domain* $\mathcal{S}$.

Figure 4. *Stability polynomial and domain of the undamped* $R_s(z) = T_s(1 + z/s^2)$ *for* $s = 5$.

Hence, the size of the stability domain $\ell_s = 2s^2$ increases *quadratically* with the number of stages along the negative real axis, as shown in Figure 4 for $s = 5$. However, we also observe in Figure 4(b) distinct points with no stability in the imaginary direction, precisely where $R_s(x) = \pm 1$. Those intersection points can cause instabilities in the presence of nonlinearity or convection, for instance.

To avoid those discrete but critical points of potential instability, Guillou and Lago [19] (see also [48]) introduce a damping parameter $\varepsilon \geq 0$ and replace the second condition in (3.2) by

$$|R_s(z)| \leq 1 - \varepsilon \text{ for } z \in [-\ell_s^\varepsilon, -\delta] \text{ with } \ell_s^\varepsilon \text{ maximal and } \delta > 0 \text{ small.}$$

Approximate solutions to this problem are the scaled and shifted Chebyshev polynomials

$$R_s(z) = T_s(\omega_0)^{-1} T_s(\omega_0 + \omega_1 z), \qquad \text{with} \qquad \omega_0 = 1 + \varepsilon/s^2, \qquad \omega_1 = T_s(\omega_0)/T_s'(\omega_0). \tag{3.3}$$

In Figure 5, we display the stability polynomial and domain of a damped stabilized scheme with $s = 5$ and $\varepsilon = 0.05$; we note the improved stability at those intersection points.

The resulting stability domain now extends along the negative real axis with $\ell_s^\varepsilon = (1 + \omega_0)/\omega_1$. In fact, the size of the stability domain scales like s^2, as it can be shown that $\ell_s^\varepsilon \geq \beta s^2$ for all s, with $\beta = 2 - 4\varepsilon/3$ [48]. The definitions of ω_0, ω_1 in (3.3) ensure that $R_s(0) = R_s'(0) = 1$ and hence that the method is first-order accurate. Moreover, since $T_s(\omega_0)^{-1} < 1$, we have $|R_s(z)| < 1$ for $z \in [-\ell_s^\varepsilon, -\delta]$, or more precisely:

$$|R_s(z)| \leq 1 - \varepsilon + \mathcal{O}(\varepsilon^2) \text{ for } z \in [-\ell_s^\varepsilon, -\delta].$$

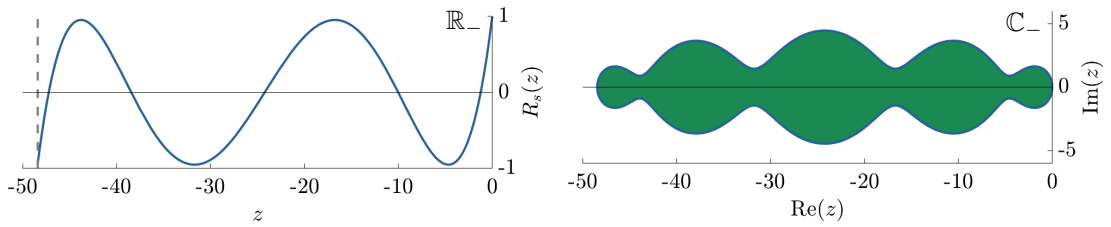
For $\varepsilon = 0$, we have $\omega_0 = 1$, $\omega_1 = 1/s^2$ and thus we recover the original (undamped) stability polynomial $T_s(1 + z/s^2)$.

Following [45, 49], we can define a RK scheme having $R_s(z)$ as stability polynomial by using the recursive definition of $T_s(x)$. Let $s \in \mathbb{N}$, $\varepsilon \geq 0$, ω_0, ω_1 as in (3.3), $b_j = 1/T_j(\omega_0)$ for $j = 0, \ldots, s$ and

$$\begin{aligned} &\mu_1 = \omega_1/\omega_0, && \mu_j = 2\omega_1 b_j/b_{j-1}, && \\ &\nu_j = 2\omega_0 b_j/b_{j-1}, && \kappa_j = -b_j/b_{j-2}, && \text{for } j = 2, \ldots, s. \end{aligned} \tag{3.4}$$

Then, the RKC method from [45, 49], is given by

$$\begin{aligned} k_0 &= y_n, \\ k_1 &= k_0 + \mu_1 \tau f(k_0), \\ k_j &= \nu_j k_{j-1} + \kappa_j k_{j-2} + \mu_j \tau f(k_{j-1}) \quad j = 2, \ldots, s, \\ y_{n+1} &= k_s. \end{aligned} \tag{3.5}$$



(a) *Stability polynomial* $R_s(z)$. (b) *Stability domain* $\mathcal{S}$.

Figure 5. *Stability polynomial and domain of the damped* $R_s(z) = T_s(\omega_0)^{-1}T_s(\omega_0 + \omega_1 z)$ *for* $s = 5$ *and* $\varepsilon = 0.05$.

Hence, if we let $f(y) = \lambda y$ it holds $k_j = R_j(z)y_n$, where $R_j(z) = b_j T_j(\omega_0 + \omega_1 z)y_0$. Therefore, $y_{n+1} = R_s(z)y_n$ and $R_s(z)$ indeed is the stability polynomial of the RKC scheme (3.5). For a general right-hand side $f : \mathbb{R}^n \to \mathbb{R}^n$ as in (1.1), the number of stages s is typically chosen such that $\tau\rho \leq \beta s^2$, where ρ is the spectral radius of the Jacobian of f. In (3.5) and below, we consider autonomous problems for convenience only and refer to [49] for the RKC method in nonautonomous form.

When integrating (1.1), the step size τ should be set to meet the desired accuracy. For the explicit Euler scheme, however, τ is bounded by $2/\rho$, which scales as H^2 for diffusion dominated problems. Hence, the step size must be smaller than necessary, purely for stability reasons, thereby incurring a high computational cost. On the other hand, the stability condition of RKC is determined by the number of stages; hence, the step size can be truly chosen according to the desired accuracy.

Let us compare the computational costs of the explicit Euler and RKC method. For a given step size τ needed to attain a desired accuracy, the number of function evaluations per time-step required by RKC is $s \approx \sqrt{\tau\rho/2}$. In contrast, the explicit Euler scheme has a stronger stability constraint and must take much smaller steps of size $\tilde{\tau} = 2/\rho$; thus, it requires $\tau/\tilde{\tau} = \tau\rho/2$ function evaluations to advance the solution by τ. Consequently, RKC is approximately $\sqrt{\tau\rho/2}$ times cheaper than explicit Euler. In particular, when f contains a discretized Laplacian and ρ scales as C/H^2, where H is the smallest mesh size, the efficiency gain in using RKC over the simple Euler method scales as $\sqrt{\tau}/H$. The same arguments apply to classic higher order explicit schemes such as the midpoint and RK4 methods. For higher order stabilized explicit methods we refer to [1, 2, 31, 40].

3.2 The multirate RKC method

The multirate RKC scheme is obtained by discretizing (2.1) with an s-stage RKC method, where f_η, given by Definition 2.1, is approximated by solving problem (2.2) with one step of an m-stage RKC method. In this section, we first define the mRKC algorithm and then compare its efficiency to that of the standard RKC method (3.5).

The mRKC Algorithm

Let $\tau > 0$ be the step size and ρ_F, ρ_S the spectral radii of the Jacobians of f_F, f_S, respectively (they can be cheaply estimated employing nonlinear power methods [30, 47]). Now, let the number of stages s, m be the smallest integers satisfying

$$\tau\rho_S \leq \beta s^2, \qquad \eta\rho_F \leq \beta m^2, \qquad \text{with} \qquad \eta = \frac{6\tau}{\beta s^2}\frac{m^2}{m^2 - 1}, \tag{3.6}$$

for the standard RKC parameter settings $\beta = 2 - 4\varepsilon/3$ and $\varepsilon = 0.05$ – see Section 3.1.

One step of the mRKC scheme is then given by

$$\begin{aligned} k_0 &= y_n, \\ k_1 &= k_0 + \mu_1 \tau \overline{f}_\eta(k_0), \\ k_j &= \nu_j k_{j-1} + \kappa_j k_{j-2} + \mu_j \tau \overline{f}_\eta(k_{j-1}) \quad j = 2, \ldots, s, \\ y_{n+1} &= k_s, \end{aligned} \tag{3.7}$$

where the parameters μ_j, ν_j, κ_j are defined in (3.4) and

$$\overline{f}_\eta(u_0) = \frac{1}{\eta}(u_\eta - u_0) \tag{3.8}$$

corresponds to the numerical counterpart of $f_\eta(u_0)$ in (2.3). The approximation u_η of $u(\eta)$ is computed at each evaluation of $\overline{f}_\eta$ by applying one step, of size η, of the m-stage RKC scheme to (2.2). Hence, u_η is given by

$$\begin{aligned} u_1 &= u_0 + \alpha_1 \eta (f_F(u_0) + f_S(u_0)), \\ u_j &= \beta_j u_{j-1} + \gamma_j u_{j-2} + \alpha_j \eta (f_F(u_{j-1}) + f_S(u_0)) \quad j = 2, \ldots, m, \\ u_\eta &= u_m. \end{aligned} \tag{3.9}$$

Here, the parameters $\alpha_j, \beta_j, \gamma_j$ of the m-stage RKC scheme (3.9) are given by

$$\upsilon_0 = 1 + \varepsilon/m^2, \qquad \upsilon_1 = T_m(\upsilon_0)/T_m'(\upsilon_0), \qquad a_j = 1/T_j(\upsilon_0) \qquad \text{for } j = 0, \ldots, m \tag{3.10}$$

and

$$\begin{aligned} \alpha_1 &= \upsilon_1/\upsilon_0, & \alpha_j &= 2\upsilon_1 a_j/a_{j-1}, \\ \beta_j &= 2\upsilon_0 a_j/a_{j-1}, & \gamma_j &= -a_j/a_{j-2}, \qquad \text{for } j = 2, \ldots, m. \end{aligned} \tag{3.11}$$

To compute m, η in (3.6), we let $\eta = 6\tau m^2/(\beta s^2(m^2-1))$ in $\eta\rho_F \le \beta m^2$, which implies

$$6\tau\rho_F \le \beta^2 s^2 (m^2 - 1). \tag{3.12}$$

Thus, we use (3.12) to compute m and then (3.6) to determine η.

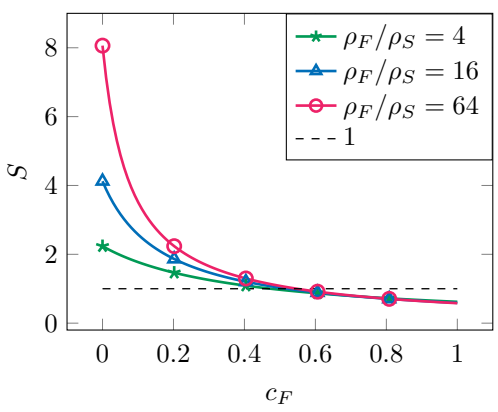
The mRKC method is given by (3.6)–(3.9). Its stability and first-order accuracy are proved in Theorems 4.4 and 4.5 in Section 4 below.

Efficiency of the multirate RKC method

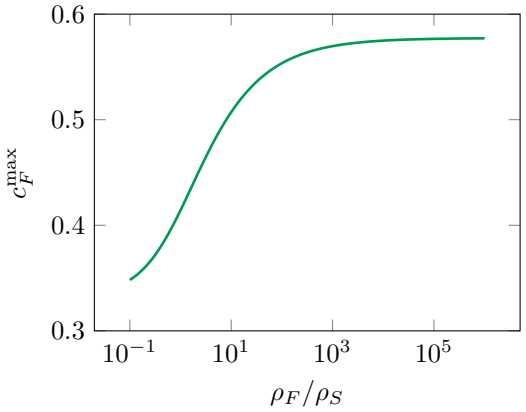
Given the spectral radii ρ_F and ρ_S of the Jacobians of f_F and f_S, respectively, we now evaluate the theoretical speed-up in using the mRKC method (3.6) to (3.9) over the standard RKC method (3.5). In doing so, we set $\varepsilon = 0$ and let s, m vary in $\mathbb{R}$. Now, we let c_F and c_S denote the cost of evaluating f_F and f_S, relatively to the cost of evaluating f itself, with $c_F, c_S \in [0, 1]$ and $c_F + c_S = 1$. Here, we suppose that the spectral radius ρ of the Jacobian of f is $\rho = \rho_F + \rho_S$, instead of setting $\rho = \rho_F$, to allow for a wide range of possible values for ρ_F even down to zero.

Since the RKC scheme requires $s = \sqrt{\tau\rho/2}$ evaluations of f per time step, its cost per time step is

$$C_{\text{RKC}} = s(c_F + c_S) = \sqrt{\frac{\tau(\rho_F + \rho_S)}{2}}. \tag{3.13}$$



(a) *Theoretical speed-up S of the mRKC method over the standard RKC scheme, with respect to c_F and ρ_F/ρ_S.*



(b) *Maximal c_F which still yields speed-up $S > 1$, w.r.t. ρ_F/ρ_S.*

Figure 6. *The relative speed-up S of the mRKC method over the RKC scheme with respect to c_F and the maximal value for c_F which still leads to an efficiency gain.*

For the mRKC method, on the other hand, we infer from (3.6) with $\beta = 2$ that it needs $s = \sqrt{\tau\rho_S/2}$ external stages and from (3.12) that it needs $m = \sqrt{3\rho_F/\rho_S + 1}$ internal stages. Since mRKC needs s evaluations of f_S and $s\,m$ evaluations of f_F, its cost per time step is

$$C_{\text{mRKC}} = s\,c_S + s\,m\,c_F = (1 - c_F)\sqrt{\frac{\tau\rho_S}{2}} + c_F\sqrt{\frac{3\tau\rho_F}{2} + \frac{\tau\rho_S}{2}}. \tag{3.14}$$

The ratio between (3.13) and (3.14) yields the *relative speed-up*

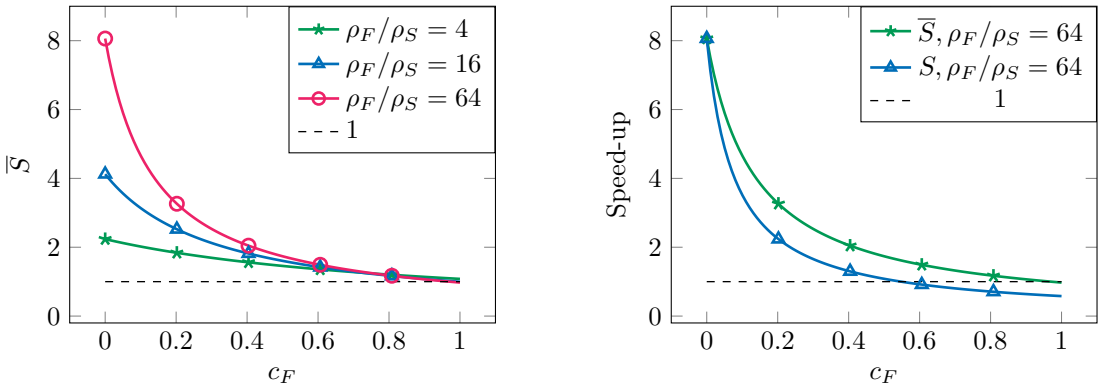
$$S = \frac{C_{\text{RKC}}}{C_{\text{mRKC}}} = \frac{\sqrt{\rho_F + \rho_S}}{(1 - c_F)\sqrt{\rho_S} + c_F\sqrt{\rho_S + 3\rho_F}} = \frac{\sqrt{1 + r_\rho}}{1 + c_F\left(\sqrt{1 + 3r_\rho} - 1\right)}, \tag{3.15}$$

with *stiffness ratio* $r_\rho = \rho_F/\rho_S \in [0, \infty)$.

In Figure 6(a), we show the speed-up S as a function of c_F for different values of $r_\rho = \rho_F/\rho_S$. For c_F sufficiently small, we observe that the mRKC scheme is always faster than RKC ($S > 1$). When $c_F \approx 1$, however, the mRKC scheme is slightly slower than RKC ($S < 1$), though this case is somewhat irrelevant since by assumption f_F is cheap to evaluate. Nevertheless, we solve the inequality $S > 1$, with S as in (3.15), for varying c_F to determine the maximal value of c_F that still leads to a reduced cost in using mRKC. We find that the speed-up $S > 1$ if, and only if,

$$c_F < c_F^{\max} = \frac{\sqrt{1 + r_\rho} - 1}{\sqrt{1 + 3r_\rho} - 1}.$$

In Figure 6(b), we monitor $c_F^{\max}$ as a function of the stiffness ratio r_ρ. For small $r_\rho = \rho_F/\rho_S$, we observe that the evaluation of f_F must be quite cheap. As ρ_F/ρ_S increases, however, the mRKC method is faster than RKC, even if f_F is relatively expensive to evaluate ($c_F^{\max} > 0.5$ for $\rho_F/\rho_S > 8$).



(a) *Theoretical speed-up $\overline{S}$ of the mRKC method over the standard RKC scheme, with respect to c_F and ρ_F/ρ_S.*

(b) *Comparison of $\overline{S}$ and S.*

Figure 7. *The relative speed-up $\overline{S}$ obtained using* (3.16) *compared to S, obtained with* (3.6).

Relaxed stability conditions

The stability conditions (3.6) are necessary when solving a general problem (1.1) without any scale separation. However, in case of scale separation ($\lambda \ll \zeta$), conditions (3.6) can in fact be replaced by

$$\tau\rho_S \leq \beta s^2, \qquad \eta\rho_F \leq \overline{\beta} m^2 \qquad \text{with} \qquad \eta = \frac{2\tau}{\beta s^2}, \tag{3.16}$$

$\overline{\beta} = 2 - 4\overline{\varepsilon}/3 \approx 1.86$ and $\overline{\varepsilon} = 0.1$ [35]. Since the value for η in (3.16) is smaller than that in (3.6), m can also be smaller which results in fewer evaluations of f_F in (3.9) and improved efficiency. Let $\overline{S}$ be the relative speed-up in using (3.16) instead of (3.6). In Figure 7(a), we plot $\overline{S}$ as a function of c_F for different values of ρ_F/ρ_S, as in Figure 6(a) for S. We observe that $\overline{S} > 1$ for all $c_F \in [0, 1-\epsilon]$, for $\epsilon > 0$ very small. In Figure 7(b), we compare S and $\overline{S}$ and observe that $\overline{S} > S$ for all values of c_F.

Even when the underlying problem is not scale separable, conditions (3.16) may in fact be sufficient to guarantee the stability of the mRKC scheme. For instance, if (1.1) stems from the spatial discretization of a parabolic problem on a locally refined mesh, where f_S and f_F correspond to the discrete Laplacians in the coarse and locally refined region, and the problem thus is not scale separable, (3.16) nonetheless suffices to guarantee stability — see Section 5.4.

4 Stability and convergence analysis

In this section, we perform a stability and convergence analysis of the multirate RKC method introduced in Section 3. We will show stability of the scheme on the multirate test equation (2.10) and on a 2×2 model problem. Then we prove its first-order accuracy.

4.1 Stability analysis

First, we prove that the mRKC method is stable when it is applied to the multirate test equation (2.10), which is sufficient when the Jacobians of f_F and f_S are simultaneously triangularizable.

Then, we also show stability for a 2×2 model problem where the Jacobians of f_F and f_S are not simultaneously triangularizable, and hence the stability analysis cannot be reduced to (2.10).

Stability analysis for the multirate test equation

Since (2.2) is approximated numerically, the stability analysis performed in Section 2.3 is no longer valid; indeed, $\varphi(z)$ is now replaced by a numerical approximation with different stability properties. Hence, we now compute a closed expression for u_η given u_0, as in (2.11) for $u(\eta)$. We denote by

$$P_m(z) = a_m T_m(\upsilon_0 + \upsilon_1 z) \tag{4.1}$$

the stability polynomial of the m-stage RKC scheme, with $a_m, \upsilon_0, \upsilon_1$ from (3.10). The next lemma computes the solution u_η of (3.9) in the case of the multirate test equation (2.10).

Lemma 4.1. *Let $\lambda, \zeta \leq 0$, $f_S(y) = \zeta y$, $f_F(y) = \lambda y$, $\eta > 0$, $m \in \mathbb{N}$ and $u_0 \in \mathbb{R}$. Then, the solution u_η of (3.9), is given by*

$$u_\eta = (P_m(\eta\lambda) + \Phi_m(\eta\lambda)\eta\zeta)u_0, \tag{4.2}$$

where $P_m(z)$ is given in (4.1),

$$\Phi_m(z) = \frac{P_m(z) - 1}{z} \quad \text{for } z \neq 0 \tag{4.3}$$

and $\Phi_m(0) = 1$ is defined by continuous extension.

Proof. We set $f_S(y) = \zeta y$ and $f_F(y) = \lambda y$ in (3.9) and let $r_j = \alpha_j \eta\zeta u_0$ to obtain

$$\begin{aligned} u_1 &= u_0 + \alpha_1 \eta\lambda u_0 + r_1, \\ u_j &= \beta_j u_{j-1} + \gamma_j u_{j-2} + \alpha_j \eta\lambda u_{j-1} + r_j, \quad j = 2, \ldots, m. \end{aligned} \tag{4.4}$$

We note that (4.4) corresponds to an RKC scheme where each stage is perturbed by r_j. Perturbed RKC schemes are studied in [49] and lead to Chebyshev polynomials of the second kind.

Let $U_j(x)$ be the Chebyshev polynomial of second kind of degree j. Then, one can show by induction [49] that

$$u_j = a_j T_j(\upsilon_0 + \upsilon_1 \eta\lambda)u_0 + \sum_{k=1}^{j} \frac{a_j}{a_k} U_{j-k}(\upsilon_0 + \upsilon_1 \eta\lambda) r_k,$$

with $a_j, \upsilon_0, \upsilon_1$ as in (3.10). Thus, as $u_\eta = u_m$ and $r_k = \alpha_k \eta\zeta u_0$,

$$u_\eta = P_m(\eta\lambda)u_0 + \sum_{k=1}^{m} \frac{a_m}{a_k} \alpha_k U_{m-k}(\upsilon_0 + \upsilon_1 \eta\lambda)\eta\zeta u_0. \tag{4.5}$$

The identity

$$\Phi_m(\eta\lambda) = \sum_{k=1}^{m} \frac{a_m}{a_k} \alpha_k U_{m-k}(\upsilon_0 + \upsilon_1 \eta\lambda),$$

from Lemma A.1 below together with (4.5) implies (4.2). □

Note the similarity between (2.11) and (4.2), with $e^z, \varphi(z)$ replaced by $P_m(z), \Phi_m(z)$, respectively. In Figure 8, we also observe that $\Phi_m(z)$ and $\varphi(z)$ share similar stability properties. Indeed, $\Phi_m(z)$ is the numerical counterpart of $\varphi(z)$, yet with the exponential replaced by the stability polynomial – compare (2.5) and (4.3).

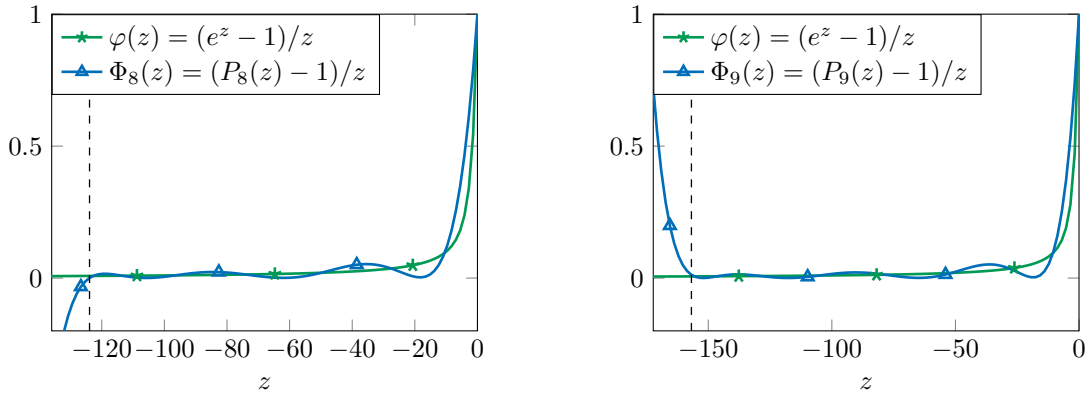


Figure 8. *Illustration of $\varphi(z)$ and $\Phi_m(z)$ for $m = 8$ (left) and $m = 9$ (right). The dashed line indicates the end of the stability domain.*

We can now compute the stability polynomial of the mRKC scheme. From (3.8), (4.2) and (4.3), we get

$$\overline{f}_\eta(u_0) = \frac{1}{\eta}(P_m(\eta\lambda) + \Phi_m(\eta\lambda)\eta\zeta - 1)u_0 = \Phi_m(\eta\lambda)(\lambda + \zeta)u_0, \tag{4.6}$$

which is the numerical counterpart of f_η in (2.12). Now, we introduce (4.6) into (3.7), which leads to

$$y_{n+1} = R_s(\tau\Phi_m(\eta\lambda)(\lambda + \zeta))y_n$$

and hence motivates the following definition.

Definition 4.2. *Let $s, m \in \mathbb{N}$, $\tau > 0$ be a step size, $\eta > 0$ and $\lambda, \zeta \le 0$. The stability polynomial of the (s, m)-stage mRKC scheme* (3.7)–(3.9) *is defined as*

$$R_{s,m}(\lambda, \zeta, \tau, \eta) = R_s(\tau\Phi_m(\eta\lambda)(\lambda + \zeta)).$$

The following lemma is the discrete version of Lemma 2.6 and is needed to prove stability of the mRKC scheme in Theorem 4.4 below. Its proof is purely technical and postponed to Appendix A.

Lemma 4.3. *Let $m \in \mathbb{N}$ and $w \le 0$. There exists $\bar{\varepsilon}_m > 0$ such that for $\varepsilon \le \bar{\varepsilon}_m$, $\Phi_m(z)(z+w) \in [w, 0]$ for all $z \in [-\ell_m^\varepsilon, 0]$ if, and only if, $\Phi_m'(0)|w| \ge 1$, i.e. $|w| \ge 2/P_m''(0)$ since $\Phi_m'(0) = P_m''(0)/2$.*

For $\varepsilon = 0$, it holds $2/P_m''(0) = 6m^2/(m^2 - 1) > 6$. In the continuous setting, the condition on w in Lemma 2.6 was $|w| \ge 2$. For the discrete mRKC scheme, however, $|w| > 6$ is necessary because of the milder slope of $\Phi_m(z)$ at the origin, see Figure 8.

Theorem 4.4. *Let $\bar{\varepsilon}_m$ be as in Lemma 4.3 and, for $\varepsilon \ge 0$, let $\varepsilon_m = \min\{\varepsilon, \bar{\varepsilon}_m\}$. Let $\lambda \le 0$ and $\zeta < 0$. Then, for all $\tau > 0, s, m$ and η such that*

$$\tau|\zeta| \le \ell_s^\varepsilon, \qquad \eta|\lambda| \le \ell_m^{\varepsilon_m} \qquad \text{with} \qquad \eta \ge \frac{6\tau}{\ell_s^\varepsilon}\frac{m^2}{m^2 - 1}, \tag{4.7}$$

$|R_{s,m}(\lambda, \zeta, \tau, \eta)| \le 1$, *i.e. the mRKC scheme is stable.*

Proof. If $\tau\Phi_m(\eta\lambda)(\lambda+\zeta) \in [-\ell_s^\varepsilon, 0]$ then $|R_{s,m}(\lambda,\zeta,\tau,\eta)| = |R_s(\tau\Phi_m(\eta\lambda)(\lambda+\zeta))| \le 1$. Hence, it is sufficient to prove the equivalent condition:

$$\Phi_m(\eta\lambda)(\eta\lambda+\eta\zeta) \in [w(\eta), 0], \qquad \text{with} \qquad w(\eta) = -\frac{\eta}{\tau}\ell_s^\varepsilon.$$

Since $\eta\lambda \in [-\ell_m^{\varepsilon_m}, 0]$, it holds $|P_m(\eta\lambda)| \le 1$ and from (4.3) we thus deduce that $\Phi_m(\eta\lambda) \ge 0$. Furthermore, (4.7) yields $\eta\zeta \ge w(\eta)$ which implies

$$0 \ge \Phi_m(\eta\lambda)(\eta\lambda+\eta\zeta) \ge \Phi_m(z(\eta))(z(\eta)+w(\eta)),$$

with $z(\eta) = \eta\lambda$. Hence, it is sufficient to show that $\Phi_m(z(\eta))(z(\eta)+w(\eta)) \in [w(\eta), 0]$ for all $z(\eta) \in [-\ell_m^{\varepsilon_m}, 0]$. From Lemma 4.3, we know that

$$|w(\eta)| \ge \frac{2}{P_m''(0)}$$

is necessary and sufficient. Since $T_m'(v_0)^2/(T_m(v_0)T_m''(v_0))$ is decreasing for $v_0 \ge 1$ (see Lemma A.2), we infer from the definition of η in (4.7) that

$$|w(\eta)| \ge 6\frac{m^2}{m^2-1} = \frac{2\,T_m'(1)^2}{T_m(1)T_m''(1)} \ge \frac{2\,T_m'(v_0)^2}{T_m(v_0)T_m''(v_0)} = \frac{2}{P_m''(0)}. \qquad \square$$

In the continuous setting in Section 2.3, η directly depends on f_S; indeed, the condition $|\varphi(\eta\lambda)(\lambda+\zeta)| \le |\zeta|$ implies $\eta \ge 2/|\zeta|$ (see Theorem 2.7). Therefore, η could rapidly grow as $\zeta \to 0$. In contrast, for the mRKC method, η depends only indirectly on f_S: η depends on the s-stage RKC method, which in turn depends on f_S. This indirect dependence of η on f_S creates a "protective buffer", which prevents the explosion of η as $\zeta \to 0$; indeed, $\ell_s^\varepsilon \ge 2$ for all $s \in \mathbb{N}$.

The restriction $\varepsilon \le \bar{\varepsilon}_m$ is necessary for proving Lemma 4.3, but probably not needed in practice. Indeed, we have verified numerically that for any $\varepsilon \ge 0$, $\Phi_m(z)(z+w) \in [w, 0]$ for all $z \in [-\ell_m^\varepsilon, 0]$ if, and only if, $|w| \ge 2/P_m''(0)$. Hence, we can suppose $\varepsilon_m = \varepsilon$ in (4.7) and replace $\ell_s^\varepsilon, \ell_m^{\varepsilon_m}$ by $\beta s^2, \beta m^2$, respectively, which yields (3.6). In Figure 9, we display the stability polynomial $R_{s,m}(\lambda,\zeta,\tau,\eta)$ for $s = 5$ and $m = 3$ as a function of λ for $\varepsilon = 0.05$ or $\varepsilon = 1$. Here, we set $\tau = 1$, η to its lower bound in (4.7), and $\zeta = -\ell_s^\varepsilon, -\ell_s^\varepsilon/2$ or 0. Since $|R_{s,m}(\lambda,\zeta,\tau,\eta)| \le 1$, the mRKC method is always stable.

Stability analysis for a 2×2 model problem

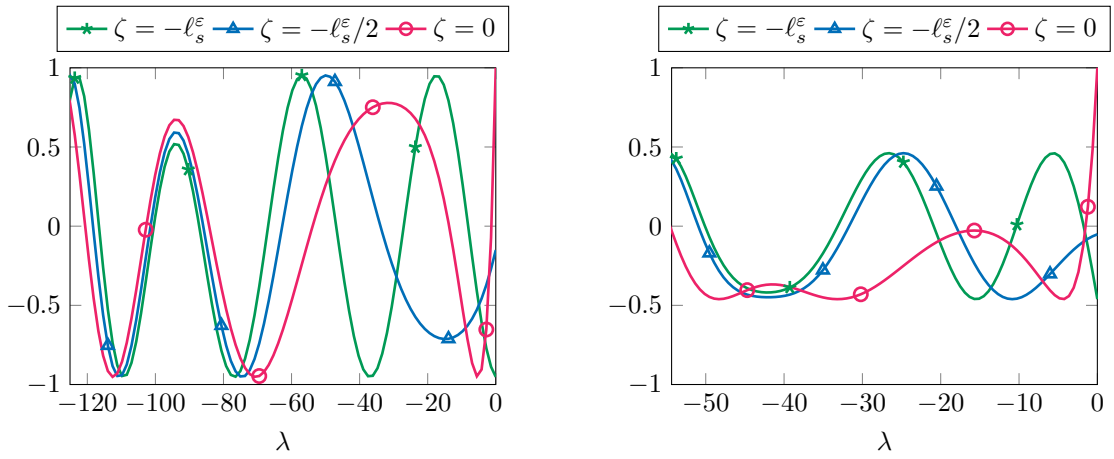
Here, we consider a 2×2 linear model problem where the Jacobians of f_F and f_S are not simultaneously triangularizable. Then, the stability analysis cannot be reduced to the scalar multirate test equation (2.10), yet we shall show that the same stability conditions still hold. Moreover, we introduce a coupling term between the fast and slow variables and show that the same stability conditions are necessary even when the coupling is weak.

Thus, we consider the system of differential equations

$$y' = Ay, \qquad \text{with} \qquad A = \begin{pmatrix} \zeta & \sigma \\ \sigma & \lambda \end{pmatrix} \tag{4.8}$$

and $y(0) = y_0 \in \mathbb{R}^2$. We let $\lambda, \zeta < 0$, $\sigma \in \mathbb{R}$ the coupling term, and assume that $\sigma^2 \le \lambda\zeta$ to ensure that both eigenvalues of A are negative or zero. We note $D \in \mathbb{R}^{2\times 2}$ the diagonal matrix satisfying $D_{11} = 0$ and $D_{22} = 1$ and consider the splitting defined by $f_F(y) = A_F\,y$ and $f_S(y) = A_S\,y$, where

$$A_F := DA = \begin{pmatrix} 0 & 0 \\ \sigma & \lambda \end{pmatrix}, \qquad A_S := (I-D)A = \begin{pmatrix} \zeta & \sigma \\ 0 & 0 \end{pmatrix}. \tag{4.9}$$



(a) *Small damping* $\varepsilon = 0.05$.

(b) *High damping* $\varepsilon = 1$.

Figure 9. *Stability polynomial* $R_{s,m}(\lambda, \zeta, \tau, \eta)$ *of the mRKC method vs.* λ *for* $\zeta = -\ell_s^\varepsilon, -\ell_s^\varepsilon/2$ *or* 0 *and* $s = 5$, $m = 3$, $\tau = 1$, η *as in* (4.7) *and damping* $\varepsilon = 0.05$ *(left) or* $\varepsilon = 1$ *(right).*

Observe that $\rho_F = |\lambda|$ and $\rho_S = |\zeta|$. The matrices A_F, A_S are simultaneously triangularizable if, and only if, they have a common eigenvector, which occurs only for $\sigma = 0$ or $\sigma^2 = \lambda\zeta$. We set $\sigma = 0.1\sqrt{\lambda\zeta}$, so that the present stability analysis cannot be reduced to the previous multirate test equation. Furthermore, as the eigenvalues of A are negative or zero for all $|\sigma| \le \sqrt{\lambda\zeta}$, the current coupling $\sigma = 0.1\sqrt{\lambda\zeta}$ can be considered to be weak when compared to the maximal coupling $\sqrt{\lambda\zeta}$.

Given $u_0 \in \mathbb{R}^2$, we obtain $\overline{f}_\eta(u_0)$ by replacing λ, ζ in (4.6) by A_F, A_S, respectively. This yields

$$\overline{f}_\eta(u_0) = A_\eta u_0, \qquad \text{with} \qquad A_\eta = \Phi_m(\eta A_F) A u_0, \tag{4.10}$$

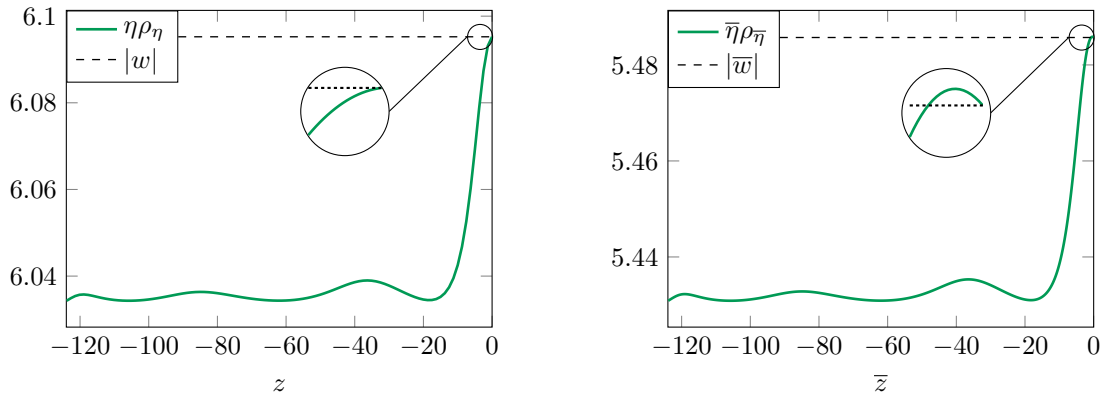
and since Φ_m is a polynomial, A_η is well-defined. From (3.7) it follows $y_{n+1} = R_s(\tau A_\eta) y_n$. If the eigenvalues of τA_η are in the interval $[-\ell_s^\varepsilon, 0]$, the mRKC method is stable. For convenience, we set $\tau = 1$, $|\zeta| = \ell_s^\varepsilon$ with $s = 10$, and also fix $m = 8$ and $\eta = \frac{6\tau}{\ell_s^\varepsilon}\frac{m^2}{m^2-1}$ (as in (4.7)). Then, the mRKC method is stable if the spectral radius ρ_η of A_η satisfies $\rho_\eta \le |\zeta|$ for all $\eta\lambda \in [-\ell_m^\varepsilon, 0]$, or equivalently $\eta\rho_\eta \le \eta|\zeta| = |w|$.

In Figure 10(a), we display $\eta\rho_\eta$ as a function of $z = \eta\lambda \in [-\ell_m^\varepsilon, 0]$ and observe that $\eta\rho_\eta \le |w|$; thus, the mRKC scheme is stable. Hence, the stability conditions (3.6) guarantee stability of the scheme even though the Jacobians of f_F, f_S are not simultaneously triangularizable.

Next, in Figure 10(b), we consider a value of η smaller than that dictated by (4.7). For $\overline{\eta} = 0.9\eta$, we again display $\overline{\eta}\rho_{\overline{\eta}}$ as a function of $\overline{z} = \overline{\eta}\lambda \in [-\ell_m^\varepsilon, 0]$. Then, a small region of instability appears for $\overline{z}$ close to zero, where $\overline{\eta}\rho_{\overline{\eta}} > |\overline{w}|$. Hence, the stability conditions (4.7) are necessary even for systems of equations with a weak coupling $\sigma = 0.1\sqrt{\lambda\zeta}$, where $\sqrt{\lambda\zeta}$ corresponds to the maximal coupling strength.

4.2 Convergence analysis

We end this section by proving that the mRKC scheme is first-order accurate. Although we only consider the test equation (2.10) here, this is sufficient to prove first-order accuracy for general nonlinear problems [22].



(a) $\eta\rho_\eta$ *vs.* z*, for* η *as in* (4.7).

(b) $\overline{\eta}\rho_{\overline{\eta}}$ *vs.* z*, for* $\overline{\eta} = 0.9\eta$.

Figure 10. *Verification that conditions* (4.7) *are sufficient and necessary for the stability of mRKC applied to the* 2×2 *test problem* (4.8).

Theorem 4.5. *The mRKC scheme is first-order accurate.*

Proof. Let $y_1(\tau, \eta) = R_{s,m}(\lambda, \zeta, \tau, \eta)y_0 = R_s(\tau\Phi_m(\eta\lambda)(\lambda + \zeta))y_0$, hence

$$\nabla y_1(\tau, \eta) = \begin{pmatrix} R_s'(\tau\Phi_m(\eta\lambda)(\lambda+\zeta))\Phi_m(\eta\lambda)(\lambda+\zeta) \\ R_s'(\tau\Phi_m(\eta\lambda)(\lambda+\zeta))\tau\lambda\Phi_m'(\eta\lambda)(\lambda+\zeta) \end{pmatrix} y_0 \qquad \text{and} \qquad \nabla y_1(0,0) = \begin{pmatrix} \lambda+\zeta \\ 0 \end{pmatrix} y_0.$$

By Taylor expansion of $y_1(\tau, \eta)$ in $(\tau, \eta) = (0, 0)$, we obtain

$$y_1(\tau, \eta) = y_1(0, 0) + \nabla y_1(0, 0) \cdot (\tau, \eta) + \mathcal{O}(\tau^2 + \eta^2) = y_0 + \tau(\lambda + \zeta)y_0 + \mathcal{O}(\tau^2 + \eta^2).$$

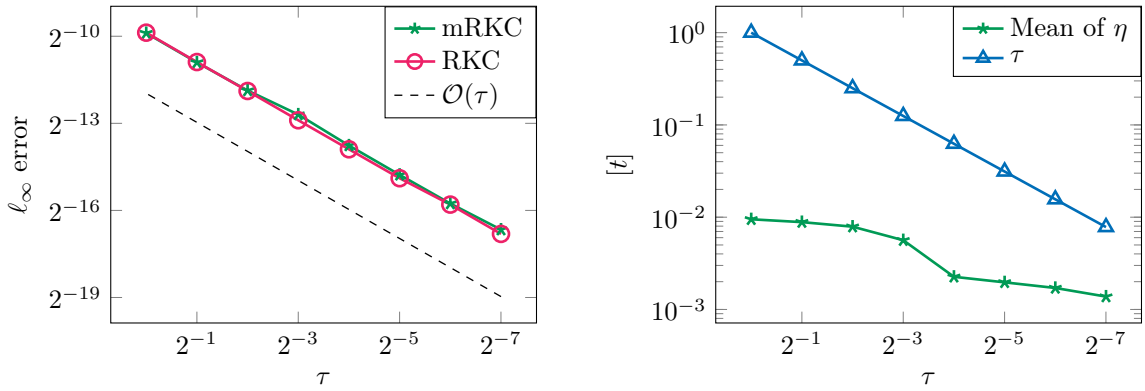
From the definition of m and η, we deduce that $m \geq 2$ (see (3.12)) and hence $\eta \leq 8\tau/(\beta s^2)$. It follows $y_1(\tau, \eta) - y(\tau) = \mathcal{O}(\tau^2)$, which implies first-order accuracy. □

Typically $s \gg 1$, i.e. $\eta \ll \tau$, and the error made when approximating f by the averaged force f_η is negligible. In fact, we observe that the difference between the RKC and the mRKC solutions in our numerical experiments in Section 5 is always very small.

5 Numerical Experiments

In this section we compare the mRKC scheme from Section 3.2 against the classical RKC method of Section 3.1 through a series of experiments. First, we apply mRKC to a stiff nonlinear dynamical system to verify convergence in the standard "ODE sense" and do a first efficiency comparison in the ODE context. Then, we apply mRKC to the heat equation to verify convergence in the "PDE sense", i.e. when both the mesh size H and the time step τ decrease simultaneously. In the third experiment, we compare the performance and efficiency of the mRKC and RKC schemes when applied to a PDE. In the last experiment, we study numerically the stability of mRKC when it is applied to various advection-diffusion-reaction problems.

Both the RKC and mRKC methods need bounds on the spectral radii of the Jacobians of f_F and f_S to determine the number of stages s, m needed for stability. In our experiments, we estimate them with a cheap nonlinear power method [30, 47]. The numerical experiments in Sections 5.2 to 5.4 were performed using the C++ library `libMesh` [26].



(a) ℓ_∞-*error vs. the step size* τ. (b) *Comparing the size of* η *and* τ.

Figure 11. *Robertson's stiff test problem. Convergence and comparison of* η *against* τ.

5.1 Robertson's stiff test problem

First, we study the convergence of the mRKC scheme on a popular stiff test problem, Robertson's nonlinear chemical reaction model [12, 23]:

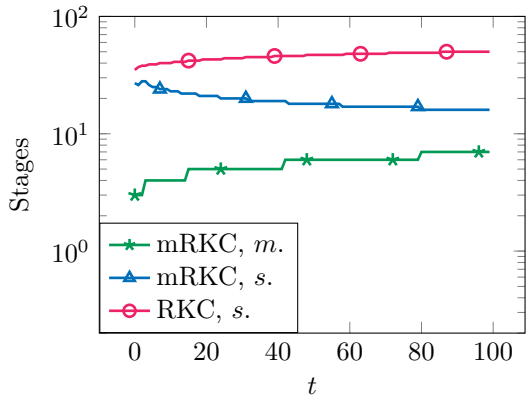
$$\begin{aligned} y_1' &= -0.04\, y_1 + 10^4\, y_2\, y_3, & y_1(0) &= 1, \\ y_2' &= 0.04\, y_1 - 10^4\, y_2\, y_3 - 3 \cdot 10^7\, y_2^2, & y_2(0) &= 2 \cdot 10^{-5}, \\ y_3' &= 3 \cdot 10^7\, y_2^2, & y_3(0) &= 10^{-1}, \end{aligned} \tag{5.1}$$

where $t \in [0, 100]$. With this set of parameters and initial conditions, the only term inducing severe stiffness is $-10^4\, y_2\, y_3$. Thus, we let

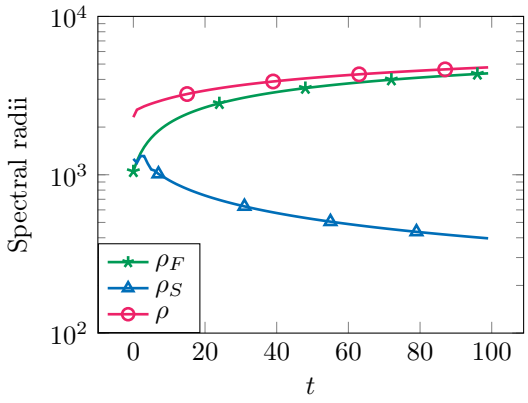
$$f_F(y) = \begin{pmatrix} 0 \\ -10^4\, y_2\, y_3 \\ 0 \end{pmatrix}, \qquad f_S(y) = \begin{pmatrix} -0.04\, y_1 + 10^4\, y_2\, y_3 \\ 0.04\, y_1 - 3 \cdot 10^7\, y_2^2 \\ 3 \cdot 10^7\, y_2^2 \end{pmatrix}, \qquad f(y) = f_F(y) + f_S(y).$$

Now, we solve (5.1) either with the RKC or the mRKC scheme using step sizes $\tau = 1/2^k$, $k = 0, \dots, 7$. For comparison, we use a reference solution obtained with the standard fourth-order Runge–Kutta scheme using $\tau = 10^{-4}$. In Figure 11(a), we observe that both the RKC and the mRKC method achieve first-order convergence. In fact, both errors are hardly distinguishable, indicating that the error introduced by the approximation of f by f_η is negligible. We observe in Figure 11(b) that the mean value of η during integration is indeed considerably smaller than τ.

Next, we compare the two schemes for a fixed step size $\tau = 1$. In Figure 12(a), we display the number of stages taken by the mRKC and the RKC method at each time step with respect to $t \in [0, 100]$. Moreover, Figure 12(b) depicts the evolution of the spectral radii ρ, ρ_F, ρ_S of the Jacobians of f, f_F, f_S, respectively. We observe that ρ_S decreases with time and consequently the mRKC scheme decreases the number s of expensive function evaluations f_S per step. In contrast, ρ increases and thus the RKC scheme must increase the number s of f_S function evaluations, although this term does not introduce any stiffness; indeed, ρ increases only because of the term contained in f_F. Finally, we notice in Figure 12(a) that the mRKC scheme increases the number m of (cheap) function evaluations f_F because of the increase in ρ_F and η; indeed, η also increases due to the



(a) *Stages needed by RKC and mRKC vs. time t, for a fixed step size $\tau = 1$.*



(b) *Evolution of the spectral radii ρ, ρ_F, ρ_S vs. time t.*

Figure 12. *Robertson's stiff test problem. Comparison of spectral radii and number of stages taken by the mRKC and the RKC scheme.*

decrease in s and (3.6). This added cost, however, is much smaller than that from the many additional (expensive) evaluations of f_S required by the RKC method.

5.2 Heat equation in the unit square

Next, we verify the space-time convergence properties of the mRKC method. To do so, we consider the heat equation in the unit square $\Omega = [0,1] \times [0,1]$,

$$\begin{aligned} \partial_t u - \Delta u &= g && \text{in } \Omega \times [0,T], \\ u &= 0 && \text{in } \partial\Omega \times [0,T], \\ u &= 0 && \text{in } \Omega \times \{0\}, \end{aligned} \tag{5.2}$$

where $T = 1/2$ and g is chosen such that $u(\boldsymbol{x},t) = \sin(\pi x_1)^2 \sin(\pi x_2)^2 \sin(\pi t)^2$ is the exact solution.

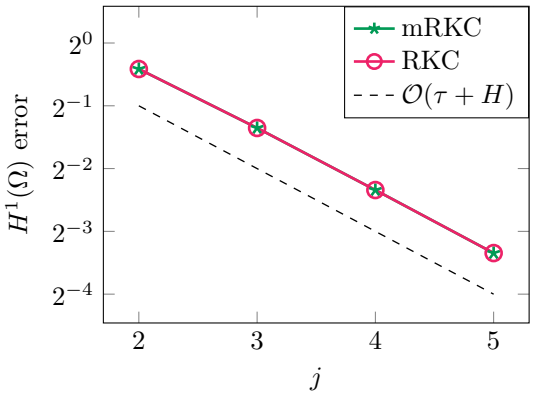
Starting from a mesh of $2^j \times 2^j$ simplicial elements with $j = 2,\dots,5$, we locally refine twice all the elements inside the square $\Omega_F = (1/4, 3/4) \times (1/4, 3/4)$. Each refinement step is performed by splitting all edges of any simplex, i.e. every triangle is split into four self-similar children. Let $\mathcal{M}$ be the set of elements in the mesh and $\mathcal{M}_F = \{T \in \mathcal{M} : \overline{T} \cap \overline{\Omega}_F \neq \emptyset\}$ the set of refined elements or their direct neighbors. Then $h = H/4$ is the diameter of the elements inside of Ω_F, with H the diameter of the elements outside of Ω_F.

Next, we discretize (5.2) in space with first-order DG-FE [8] on the mesh $\mathcal{M}$. After inverting the block-diagonal mass matrix, the resulting system is

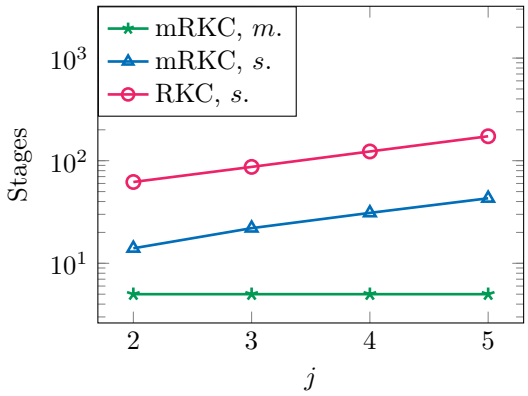
$$y' = A\,y + G, \qquad y(0) = y_0,$$

where $A \in \mathbb{R}^{N\times N}$ and $G \in C([0,T],\mathbb{R}^N)$ corresponds to the spatial discretization of $g(\cdot,t)$. Let $D \in \mathbb{R}^{N\times N}$ be a diagonal matrix with $D_{ii} = 1$ if the ith degree of freedom belongs to an element in $\mathcal{M}_F$ and $D_{ii} = 0$ otherwise. We also introduce

$$A_F = DA, \qquad A_S = (I - D)A \qquad \text{and} \qquad f_F(y) = A_F\,y, \qquad f_S(t,y) = A_S\,y + G(t), \tag{5.3}$$



(a) *Convergence of RKC and mRKC.*



(b) *Number of stages needed by RKC and mRKC vs. refinement level j.*

Figure 13. *Heat equation in the unit square. Space-time convergence and number of stages.*

with I the identity. It is well-known that the spectral radii ρ_S and ρ_F of A_S and A_F behave as $\mathcal{O}(1/H^2)$ and $\mathcal{O}(1/h^2) = \mathcal{O}(16/H^2)$, respectively.

We now consider a sequence of meshes with $j = 2, \dots, 5$ and solve (5.2) either with the mRKC or the RKC scheme using the same step size $\tau = 1/2^j$. The parameters s and m for mRKC are chosen according to (3.16). In Figure 13(a), we display the $H^1(\Omega)$ errors at final time for mRKC and RKC. Both methods yield space-time first-order convergence and result in similar errors. In Figure 13(b), we show the number of stages needed by RKC and mRKC. For both schemes, s increases as the mesh size H decreases, but for mRKC, s is much smaller, since it only depends on the coarse elements, while m remains constant due to the constant ratio between ρ_F and ρ_S.

5.3 Diffusion across a narrow channel

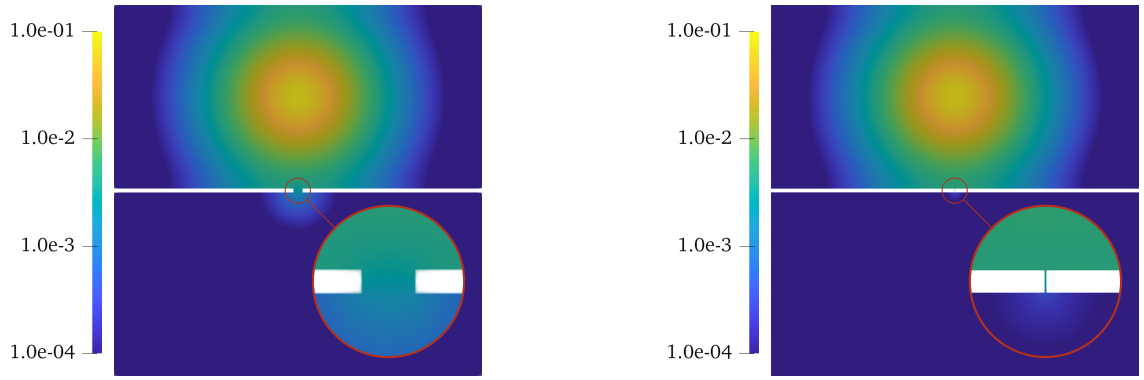
To illustrate the efficiency of the mRKC method in a situation where geometry constraints require local mesh refinement, we consider the heat equation

$$\begin{aligned} \partial_t u - \Delta u &= g && \text{in } \Omega_\delta \times [0,T], \\ \nabla u \cdot \boldsymbol{n} &= 0 && \text{in } \partial\Omega_\delta \times [0,T], \\ u &= 0 && \text{in } \Omega_\delta \times \{0\}, \end{aligned} \tag{5.4}$$

with $T = 0.1$ inside Ω_δ, which consists of two 10×5 rectangles linked by a narrow $\delta \times 0.05$ channel of width $\delta > 0$, see Figure 14. The right-hand side $g(\boldsymbol{x}, t) = \sin(10\pi t)^2 e^{-5\|\boldsymbol{x}-\boldsymbol{c}\|^2}$ corresponds to a smoothed Gaussian point source centered at $\boldsymbol{c}$ in the middle of the upper rectangle.

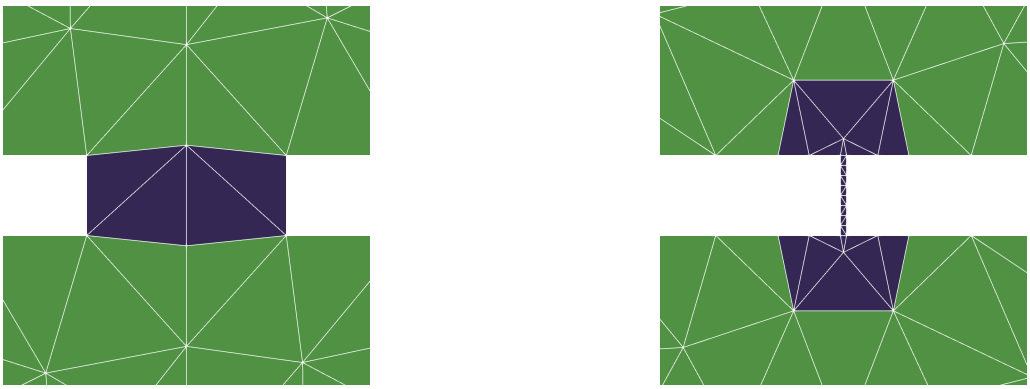
Inside Ω_δ, we use a Delaunay triangulation with maximal element size $H \approx 0.015$. As δ approaches zero, the elements inside the channel become increasingly smaller and the system stiffer. For each $\delta > 0$, we define a neighborhood $\Omega_{F,\delta} \subset \Omega_\delta$ of the channel and $\mathcal{M}_F$, A, A_F, A_S, f_F, f_S as in Section 5.2. Here, $\Omega_{F,\delta}$ is chosen such that the spectral radius of A_S is almost independent of δ and only that of A_F increases with decreasing δ. Hence, $\Omega_{F,\delta}$ contains the channel together with all neighboring elements of mesh size smaller than H, see Figures 15(a) and 15(b).

For varying channel width $\delta = 1/2^k$, $k = 0, \dots, 15$, we now solve (5.4) with the RKC and mRKC method using the choice of parameters (3.16) with $\tau = 0.01$. In Figure 16(a), the relative speed-up



(a) *Solution for* $\delta = 1/2^2$. (b) *Solution for* $\delta = 1/2^7$.

Figure 14. *Narrow channel. Numerical solutions of* (5.4) *at* $t = 1$ *using mRKC for a channel width* $\delta = 1/2^2$ *or* $\delta = 1/2^7$.



(a) *Zoom over the channel with* $\delta = 1/2^2$. (b) *Zoom over the channel with* $\delta = 1/2^7$.

Figure 15. *Narrow channel. Zoom of the FE mesh for a channel width* $\delta = 1/2^2$ *or* $1/2^7$*, with the subdomain* $\Omega_{F,\delta}$ *(in blue).*

defined as the ratio between the computational times of RKC and mRKC always exceeds one and reaches a value as high as 60. Note that the relative error between the two solutions in $H^1(\Omega_\delta)$ norm is at most $3 \cdot 10^{-4}$, as shown in Figure 16(c).

In Figure 16(d), we display for varying δ also the spectral radii ρ, ρ_F, ρ_S of A, A_F, A_S, respectively; note that ρ_F and ρ essentially coincide. For large δ, we also have $\rho_F \approx \rho_S$ since the typical element size is sufficiently small to resolve the channel (Figure 15(a)). For δ small, we observe that ρ, ρ_F increase as $1/\delta^2$ while ρ_S remains almost constant. Figure 16(e) shows that the number s of stages in the mRKC scheme remains constant, as does ρ_S in Figure 16(d), while m increases (as ρ_F). For large δ, we have $\rho_F \approx \rho_S$ and thus $m = 1$; then, the RKC and mRKC schemes coincide. Indeed, as is shown in Figure 16(c), for $m = 1$ then the relative error between the RKC and mRKC solutions in $H^1(\Omega_\delta)$ norm is of the order of machine precision.

In Figure 16(b), we observe that for δ large the CPU times of the two methods are similar; thus, despite $\rho_F \approx \rho_S$, there is no loss in efficiency and the speed-up is at least one (Figure 16(a)). For moderate values of δ, the cost of RKC increases proportionally to $1/\delta$, while the cost of mRKC is hardly affected. For even smaller δ, the number of evaluations of f_F increases and so does its cost with respect to f_S (see Figure 16(f)), since the number of elements in $\mathcal{M}_F$ increases (Figure 15). In this regime, evaluation of f_F dominates the computational cost of mRKC, which increases linearly in $1/\delta$, too. Still, the mRKC method remains about sixty times faster than the classical RKC method for this particular discretization inside Ω_δ, see Figure 16(a).

5.4 Reaction-convection-diffusion problem

In Section 4.1 we proved that the stability conditions of the mRKC method are the same for the 2×2 model problem (4.8) and for the scalar multirate test equation (2.10). The splitting of the discrete Laplace operator in (5.3) in fact is similar to that in (4.9) for the 2×2 model problem. Thus, one could expect that the stability conditions (3.6) are also necessary for more general parabolic problems. However, spatial discretizations of parabolic problems are much more complex than (4.8). Here we shall demonstrate via numerical experiment that the weaker stability conditions (3.16) in fact are also necessary and sufficient for general parabolic reaction-convection-diffusion problems, such as

$$\begin{aligned} \partial_t u - \nabla \cdot (K \nabla u) + \boldsymbol{\beta} \cdot \nabla u + \mu u &= g && \text{in } \Omega \times [0,T], \\ u &= 0 && \text{in } \partial\Omega \times [0,T], \\ u &= u_0 && \text{in } \Omega \times \{0\}. \end{aligned}$$

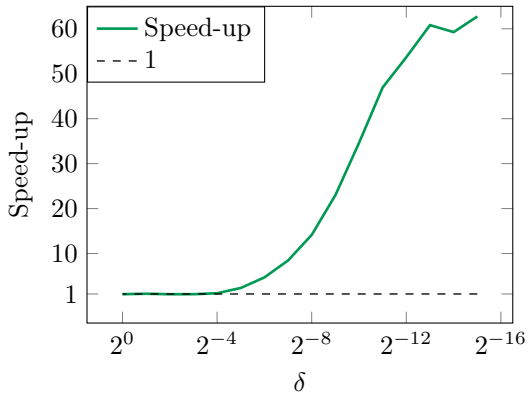
These experiments also illustrate that the mRKC method indeed requires no scale separation.

We now consider three distinct parameter regimes. First, we let $\Omega = [0,2] \times [0,1]$, $K = I_{2\times 2}$, $\boldsymbol{\beta} = \mathbf{0}$ and $\mu = 0$. Inside Ω, we build a 16×8 uniform mesh and refine twice the elements inside of $\Omega_F = (1,2) \times (0,1)$ (see Figure 17(a)). Again, we use DG-FE for the spatial discretization, which yields the two matrices A_F and A_S, as described in Section 5.2. Next, we set $\tau = 1$, s, m, η as in (3.6) and A_η as in (4.10). One step of the mRKC scheme is given by $y_1 = R_s(\tau A_\eta) y_0$. We recall that a necessary condition for stability of the scheme (at least for linear problems) is $\tau \rho_\eta \leq \beta s^2$, where ρ_η is the spectral radius of A_η.

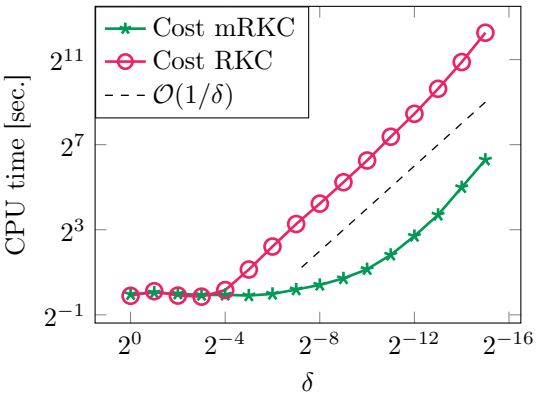
Let $\overline{\beta}$ be as in (3.16), $\overline{\eta} \in [0, \eta]$, $\overline{m}$ such that $\overline{\eta}\rho_F \leq \overline{\beta}\overline{m}^2$,

$$\overline{A}_\eta = \Phi_{\overline{m}}(\overline{\eta} A_F) A$$

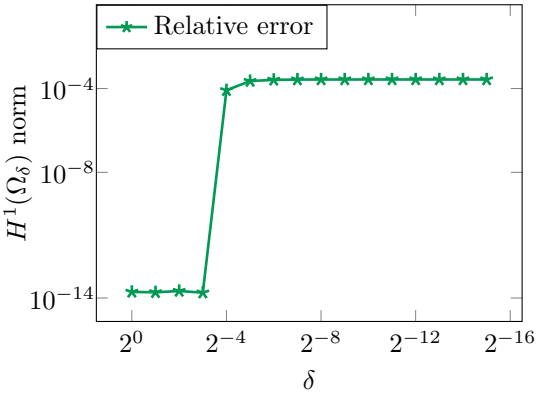
and $\overline{\rho}_\eta$ be the spectral radius of $\overline{A}_\eta$. We wish to study for which $\overline{\eta}$ it holds $\tau\overline{\rho}_\eta \leq \beta s^2$. In Figure 17(b), we display $\tau\overline{\rho}_\eta$ for $\overline{\eta} \in (0, \eta)$ with respect to $w(\overline{\eta}) = -\overline{\eta}\beta s^2/\tau$: for $|w(\overline{\eta})| \geq 2$, it holds $\tau\overline{\rho}_\eta \leq \beta s^2$ and thus the scheme is stable. Observe that $|w(\overline{\eta})| \geq 2$ is equivalent to $\overline{\eta} \geq 2\tau/(\beta s^2)$, as



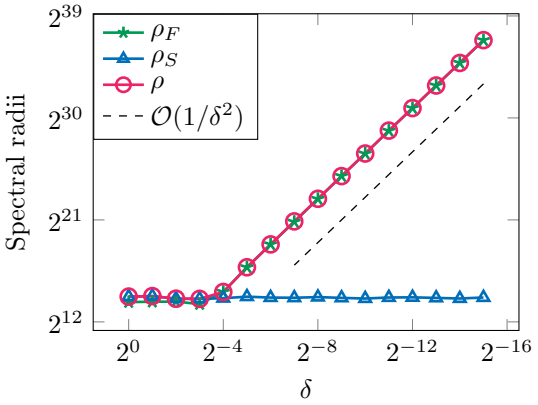
(a) *Relative speed-up of mRKC over RKC.*



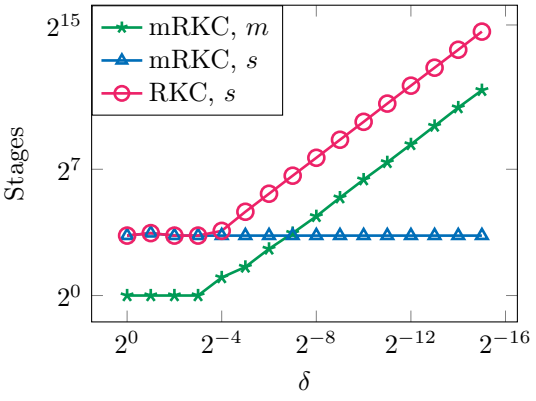
(b) *Total CPU time w.r.t. δ.*



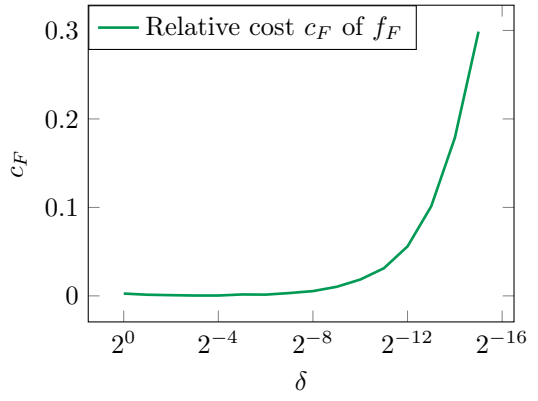
(c) *RKC and mRKC solutions' relative error $\|u^{\mathrm{mRKC}} - u^{\mathrm{RKC}}\|/\|u^{\mathrm{RKC}}\|$ in $H^1(\Omega_\delta)$ norm.*



(d) *Spectral radii w.r.t. δ.*

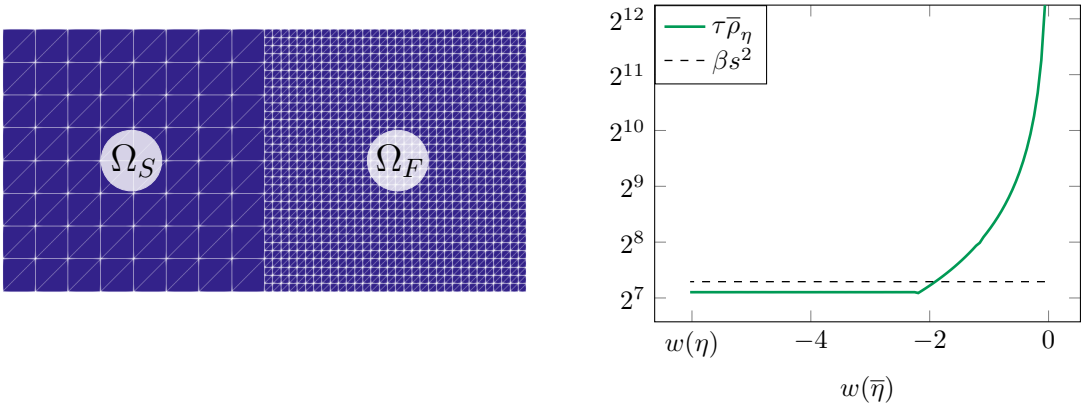


(e) *Number of stages needed by RKC and mRKC.*



(f) *Relative evaluation cost of $f_F(y)$ w.r.t. $f_F + f_S$ as a function of δ.*

Figure 16. *Narrow channel. Speed-up, error, spectral radii and stages number w.r.t. channel width δ.*



(a) *Domains Ω_S and Ω_F*

(b) *$\tau\overline{\rho}_\eta$ vs. $w(\overline{\eta})$, first problem.*

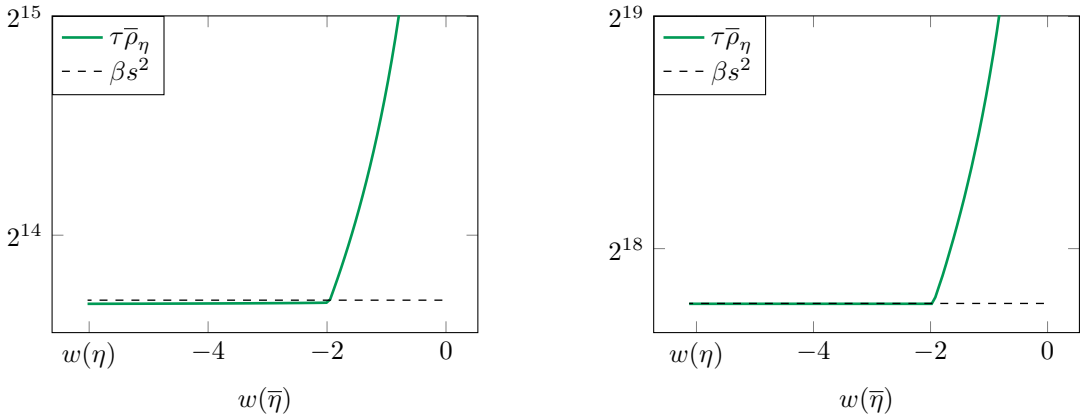
Figure 17. *Reaction-convection-diffusion problem. Mesh and spectral radius $\tau\overline{\rho}_\eta$ of the first problem setting.*

in (3.16). Since the smallest (in magnitude) nonzero eigenvalues of the discrete Laplacians, A_F and A_S, do not depend on the mesh size, but only depends on the size of the domain, they essentially coincide; hence, this problem exhibits no scale separation assumption. Nevertheless the mRKC scheme remains stable, as expected from theory.

Finally, we consider two additional cases that further corroborate the previous findings. First, we set $\Omega = [0,1] \times [0,1]$, $K = I_{2\times 2}$, $\boldsymbol{\beta} = (1,1)^\top$ and $\mu = 1$. In Ω, we build a 8×8 uniform mesh and refine three times the elements inside the small inner square $\Omega_F = (1/4, 3/4) \times (1/4, 3/4)$. In Figure 18(a), we show again $\tau\overline{\rho}_\eta$ for $\overline{\eta} \in (0, \eta)$ with respect to $w(\overline{\eta})$: for $|w(\overline{\eta})| \geq 2$, $\tau\overline{\rho}_\eta \leq \beta s^2$ holds. Next, we use a uniform 32×32 mesh in $\Omega = [0,1] \times [0,1]$ which is refined twice in the lower left corner $\Omega_F = (0, 1/32) \times (0, 1/32)$. We also set $\boldsymbol{\beta} = 0$, $\mu = 0$, $K(\boldsymbol{x}) = 1$ for $x_1 \geq x_2$ and $K(\boldsymbol{x}) = 0.1$ elsewhere. The results, shown in Figure 18(b), again confirm the stability of the mRKC with parameters chosen according to (3.16).

6 Conclusion

Starting from the stiff differential equation $y' = f_S(y) + f_F(y)$, where f_F represents a few severely stiff, but cheap, "fast" components, we have proposed a modified equation (2.1) whose stiffness no longer depends on f_F. It involves an averaged force f_η of $f = f_S + f_F$, evaluated by solving the stiff, but cheap, auxiliary problem (2.2) over short time and thus forms the basis of the following multirate strategy: Solve (2.1) with an explicit numerical method using a large step size determined by the mildly stiff, but expensive, "slow" components f_S, while solving (2.2) with a separate explicit method using a smaller step size determined by f_F, whenever an evaluation of f_η is needed. In Theorems 2.4 and 2.5, we have proved that the modified equation (2.1) approximates the original problem to first-order accuracy while preserving its contractivity properties. The stability analysis of the multirate test equation (2.13) underpins the reduced stiffness of the modified equation, which no longer depends on the fastest components f_F for η sufficiently large – see Theorem 2.7.



(a) $\tau\overline{\rho}_\eta$ *vs.* $w(\overline{\eta})$, *second problem.*

(b) $\tau\overline{\rho}_\eta$ *vs.* $w(\overline{\eta})$, *third problem.*

Figure 18. *Stability experiment. Illustration of* $\tau\overline{\rho}_\eta$ *versus* $w(\overline{\eta})$, *second and third problem setting.*

By discretizing (2.1) with an s-stage (explicit) Runge-Kutta-Chebyshev (RKC) method while evaluating f_η with one step of a separate m-stage RKC method, we have devised a new multirate RKC method. The resulting mRKC method, given by (3.6)–(3.9), is fully explicit, stable, and first-order accurate, as proved in Theorems 4.4 and 4.5, without the need for interpolation or extrapolation of missing stage values. Thanks to the reduced stiffness in (2.1), the number of expensive f_S evaluations is greatly reduced and independent of the severe stiffness induced by just a few degrees of freedom in f_F, without any assumption about scale separation.

For semi-discrete parabolic problems, where f_S and f_F correspond to discretized diffusion operators in the coarse and locally refined regions of the mesh, respectively, the mRKC method permits to overcome the crippling effect on explicit time integrators due to a few tiny elements or grid cells. Thus, the mRKC method recovers the well-known efficiency of RKC methods for large-scale, possibly nonlinear, parabolic problems without sacrificing explicitness, even in the presence of local mesh refinement.

A Proofs of lemmas

In this section we prove Lemmas 4.3, A.1 and A.2, needed in the proof of Theorem 4.4.

Lemma A.1. *Let* $\Phi_m(z)$ *be as in* (4.3), *then*

$$\Phi_m(z) = \sum_{k=1}^{m} \frac{a_m}{a_k} \alpha_k U_{m-k}(\upsilon_0 + \upsilon_1 z), \tag{A.1}$$

with $\upsilon_0, \upsilon_1, a_k, \alpha_k$ *as in* (3.10) *and* (3.11).

Proof. We start observing that $\Phi_m(z)$ defined in (4.3) is a polynomial. Indeed, since $P_m(0) = 1$ then $P_m(z) - 1$ is a polynomial with no zero order term. Hence, both sides of (A.1) are polynomials and if we can show that (A.1) holds in an interval then it holds everywhere. We will consider the values of z such that $\upsilon_0 + \upsilon_1 z \in]-1, 1[$, hence there exists $\theta \in]0, \pi[$ such that $\upsilon_0 + \upsilon_1 z = \cos(\theta)$. For

such z, the right-hand side of (A.1) is

$$\sum_{j=1}^{m} \frac{a_m}{a_j} \alpha_j U_{m-j}(\cos(\theta)) = \frac{v_1}{T_m(v_0)} \left(U_{m-1}(\cos(\theta)) + 2 \sum_{j=1}^{m-1} T_{m-j}(v_0) U_{j-1}(\cos(\theta)) \right),$$

where we used $\alpha_1 = v_1/v_0$ and $\alpha_j = 2v_1 a_j / a_{j-1}$ for $j = 2, \ldots, m$. Hence, (A.1) is equivalent to

$$\Phi_m(z) = \frac{v_1}{T_m(v_0)} \overline{\Phi}_m(\theta), \tag{A.2}$$

with

$$\overline{\Phi}_m(\theta) := U_{m-1}(\cos(\theta)) + 2 \sum_{j=1}^{m-1} T_{m-j}(v_0) U_{j-1}(\cos(\theta)).$$

Since

$$U_{j-1}(\cos(\theta)) = \frac{\sin(j\theta)}{\sin(\theta)}$$

we have

$$\overline{\Phi}_m(\theta) = \frac{\sin(m\theta)}{\sin(\theta)} + 2 \sum_{j=1}^{m-1} T_{m-j}(v_0) \frac{\sin(j\theta)}{\sin(\theta)}.$$

For $x \geq 1$ we have the identity $T_n(x) = \cosh(n \operatorname{acosh}(x))$, letting $\sigma = \operatorname{acosh}(v_0)$ it holds

$$T_{m-j}(v_0) = \cosh((m-j)\sigma) = \frac{1}{2} \left(e^{(m-j)\sigma} + e^{-(m-j)\sigma} \right). \tag{A.3}$$

Let $u = e^{-\sigma + i\theta}$ and $v = e^{\sigma + i\theta}$, from (A.2) and (A.3) we deduce

$$\begin{aligned} \overline{\Phi}_m(\theta) &= \frac{1}{\sin(\theta)} \left(\sin(m\theta) + \sum_{j=1}^{m-1} \left(e^{(m-j)\sigma} + e^{-(m-j)\sigma} \right) \sin(j\theta) \right) \\ &= \frac{1}{\sin(\theta)} \left(\sin(m\theta) + e^{m\sigma} \sum_{j=1}^{m-1} \operatorname{Im}(u^j) + e^{-m\sigma} \sum_{j=1}^{m-1} \operatorname{Im}(v^j) \right). \end{aligned}$$

We have

$$\begin{aligned} e^{m\sigma} \sum_{j=1}^{m-1} \operatorname{Im}(u^j) = e^{m\sigma} \operatorname{Im} \left(\sum_{j=0}^{m-1} u^j \right) &= e^{m\sigma} \operatorname{Im} \left(\frac{u^m - 1}{u - 1} \right) \\ &= \frac{\sin((m-1)\theta) - e^{\sigma} \sin(m\theta) + e^{m\sigma} \sin(\theta)}{2(\cosh(\sigma) - \cos(\theta))} \end{aligned}$$

and similarly

$$e^{-m\sigma} \sum_{j=1}^{m-1} \operatorname{Im}(v^j) = \frac{\sin((m-1)\theta) - e^{-\sigma} \sin(m\theta) + e^{-m\sigma} \sin(\theta)}{2(\cosh(\sigma) - \cos(\theta))}.$$

Using $\cosh(\sigma) = v_0$ and (A.3) for $k = 0$ we obtain

$$\overline{\Phi}_m(\theta) = \frac{\sin((m-1)\theta) - \sin(m\theta)\cos(\theta) + T_m(v_0)\sin(\theta)}{\sin(\theta)(v_0 - \cos(\theta))} = \frac{T_m(v_0) - \cos(m\theta)}{v_0 - \cos(\theta)},$$

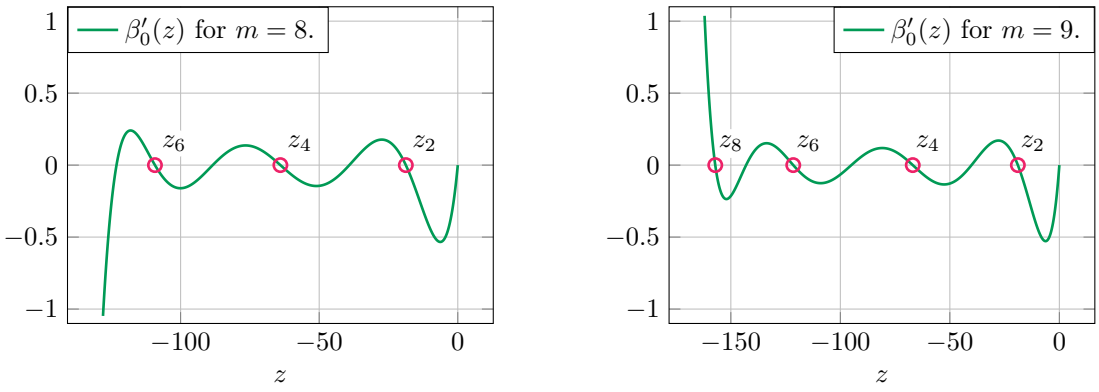


Figure 19. *Plot of $\beta_0'(z)$ for $m = 8, 9$ and $|w| = 2/P_m''(0)$.*

where we used $\sin((m-1)\theta) = \sin(m\theta)\cos(\theta) - \sin(\theta)\cos(m\theta)$. Using the identity $T_m(\cos(\theta)) = \cos(m\theta)$, $\cos(\theta) = v_0 + v_1 z$ we get

$$\frac{v_1}{T_m(v_0)}\overline{\Phi}_m(\theta) = \frac{v_1}{T_m(v_0)}\frac{T_m(v_0 + v_1 z) - T_m(v_0)}{v_1 z} = \frac{P_m(z) - 1}{z}$$

and thus (A.2) holds. □

Proof of Lemma 4.3. For the only if part we follow the lines of the proof of Lemma 2.6 and find that $\Phi_m'(0)|w| \geq 1$ is a necessary condition. The identity $\Phi_m'(0) = P_m''(0)/2$ follows from the definition of $\Phi_m(z)$ in (4.3).

Now, let us suppose $2/P_m''(0) \leq |w|$ and show $\beta_\varepsilon(z) = \Phi_m(z)(z+w) \geq w$, where ε is the damping. For $z = 0$ it is clear, independently of ε. We will show $\beta_0(z) > w$ for all $z < 0$, since $\beta_\varepsilon(z)$ depends continuously on ε there exists $\varepsilon_m > 0$ such that $\beta_\varepsilon(z) \geq w$ for all $\varepsilon \leq \varepsilon_m$. We have

$$\begin{aligned}
\beta_0'(z) &= \frac{P_m'(z)}{z}(z+w) - \frac{P_m(z)-1}{z^2}w, \\
\beta_0''(z) &= \frac{P_m''(z)}{z}(z+w) - 2\frac{P_m'(z)}{z^2}w + 2\frac{P_m(z)-1}{z^3}w
\end{aligned}$$

and since $\beta_0(z)$ is a polynomial of degree m then $\beta_0'(z)$ has at most $m-1$ zeros. We are going to locate the zeros $z_{m-1} < \cdots < z_3 < z_2$ of $\beta_0'(z)$. Then we will use the fact that $\beta_0'(z)$ has at most one zero on the right of z_2. In order to help the understanding of the proof we plot $\beta_0'(z)$ in Figure 19 for two values of m.

Since $P_m(z) = T_m(1 + z/m^2)$ and $T_m(\cos(\theta)) = \cos(m\theta)$, choosing z_{2k} such that

$$1 + \frac{z_{2k}}{m^2} = \cos(\theta_{2k}) \qquad \text{with} \qquad \theta_{2k} = \frac{2k\pi}{m}$$

it yields

$$\beta_0'(z_{2k}) = 0 \quad \text{for} \quad 2k = 2, 4, \ldots, 2\lfloor\frac{m-1}{2}\rfloor.$$

Since z_{2k} is a local maximum of $P_m(z)$ then $P_m''(z_{2k}) < 0$ and $\beta_0''(z_{2k}) < 0$. In a neighborhood of z_{2k} we have

$$\beta_0'(z) = \beta_0''(z_{2k})(z - z_{2k}) + \mathcal{O}((z - z_{2k})^2),$$

hence for $\delta > 0$ small and $2k = 2, 4, \ldots, 2(\lfloor \frac{m-1}{2} \rfloor - 1)$ we have $\beta_0'(z_{2k+2} + \delta) < 0$ and $\beta_0'(z_{2k} - \delta) > 0$, implying that there exists $z_{2k+1} \in [z_{2k+2}, z_{2k}]$ such that $\beta_0'(z_{2k+1}) = 0$. If m is odd then $2\lfloor \frac{m-1}{2} \rfloor = m - 1$ and we located the zeros z_j for $j = 2, 3, \ldots, m-1$. If m is even then $2\lfloor \frac{m-1}{2} \rfloor = m - 2$, but $P_m(-2m^2) = 1$ and $P_m'(-2m^2) = -1$ and hence $\beta_0'(-2m^2) < 0$. Thus, since $\beta_0'(z_{m-2} - \delta) > 0$ there exists $z_{m-1} \in]-2m^2, z_{m-2}[$ such that $\beta_0'(z_{m-1}) = 0$. Finally, we located z_j for $j = 2, 3, \ldots, m-1$ for m even and odd. We will show $\beta_0(z) > w$ for $z \in [z_2, 0[$ and then for $z \in [-2m^2, z_2]$.

Let $z \in [z_2, 0[$, if $z = z_1$ then $\beta_0'(z) = 0$ and else $\beta_0'(z) < 0$. Indeed, for z close to zero we have

$$\beta_0'(z) = 1 + \frac{1}{2}P_m''(0)w + (P_m''(0) + \frac{1}{3}P_m'''(0)w)z + \mathcal{O}(z^2).$$

If $2/P_m''(0) < |w|$ then $1 + \frac{1}{2}P_m''(0)w < 0$ and $\beta_0'(z) < 0$ in the neighborhood of zero. If $2/P_m''(0) = |w|$ then

$$\beta_0'(z) = \left(P_m''(0) - \frac{2}{3}\frac{P_m'''(0)}{P_m''(0)}\right)z + \mathcal{O}(z^2) = \frac{m^2+1}{5m^2}z + \mathcal{O}(z^2),$$

and $\beta_0'(z) < 0$ in the neighborhood of zero as well. If there exists $\bar{z} \in]z_2, 0[$ such that $\beta_0'(\bar{z}) > 0$ we can take $\delta > 0$ small enough to have $\bar{z} \in]z_2 + \delta, -\delta[$ and $\beta_0'(z_2 + \delta) < 0$ and $\beta_0'(-\delta) < 0$. Hence, β_0' would change sign twice in the interval $]z_2 + \delta, -\delta[$, which is impossible since β_0' has at most one zero on the right of z_2. Hence, $\beta_0'(z) < 0$ for all $z \in [z_2, 0[$ except at most one point, since $\beta_0(0) = w$ it follows $\beta_0(z) > w$ for all $z \in [z_2, 0[$.

We consider now $z \leq z_2$. Using $1 - \cos(\theta) = 2\sin(\theta/2)^2$ it holds $z_2 = -2m^2\sin(\pi/m)$ and

$$\begin{aligned}\beta_0(z) = \Phi_m(z)(z+w) = \frac{P_m(z)-1}{z}(z+w) \geq -2\frac{z+w}{z} \geq -2 - 2\frac{w}{z} \geq wP_m''(0) - 2\frac{w}{z_2} \\ = \left(\frac{m^2-1}{3m^2} + \frac{1}{m^2\sin(\pi/m)^2}\right)w > \left(\frac{1}{3} + \frac{1}{m^2\sin(\pi/m)^2}\right)w.\end{aligned}$$

Thus, if $m^2\sin(\pi/m)^2 \geq 3/2$ then $\beta_0(z) > w$. For $m = 2$ it is clearly true. We let $g(x) = x^2\sin(\pi/x)^2$ and show that $g(x)$ is strictly increasing in $x \in [2, \infty[$, which implies $m^2\sin(\pi/m)^2 \geq 3/2$ for all $m \geq 2$. We have

$$g'(x) = 2\sin(\pi/x)(x\sin(\pi/x) - \pi\cos(\pi/x)) \geq 0$$

if and only if $x\sin(\pi/x) - \pi\cos(\pi/x) \geq 0$, which is equivalent to $\tan(\pi/x) \geq \pi/x$. The latter holds true since $\tan(\theta) \geq \theta$ for $\theta \in [0, \pi/2]$. □

The next lemma has been used in the proof of Theorem 4.4.

Lemma A.2. *$P_m''(0) = T_m(v_0)T_m''(v_0)/T_m'(v_0)^2$ is increasing for $v_0 \geq 1$.*

Proof. For $v_0 \geq 1$ we have the relation $T_m(v_0) = \cosh(m\,\mathrm{arccosh}(v_0))$. Let $\theta(v_0) = \mathrm{arccosh}(v_0)$, it holds

$$\frac{T_m(v_0)T_m''(v_0)}{T_m'(v_0)^2} = \coth(m\theta)(\coth(m\theta) - \frac{1}{m}\coth(\theta)) = c(\theta).$$

Since $\theta(v_0)$ is increasing for $v_0 \geq 1$, if $c(\theta)$ is increasing for $\theta \geq 0$ then $T_m(v_0)T_m''(v_0)/T_m'(v_0)^2$ is increasing for $v_0 \geq 1$. We have

$$c'(\theta) \geq 0 \qquad \Longleftrightarrow \qquad \tanh(\theta)\tanh(m\theta) - 2m\tanh(\theta)^2 + \frac{1}{m}\frac{\sinh(m\theta)^2}{\cosh(\theta)^2} \geq 0.$$

From the convexity of $\sinh(\theta)$ we deduce $\sinh(m\theta)^2/m \geq m\sinh(\theta)^2$ and thus

$$c'(\theta) \geq 0 \qquad \Longleftarrow \qquad \tanh(\theta)\tanh(m\theta) - m\tanh(\theta)^2 \geq 0.$$

The latter holds true since $\tanh(\theta)$ is convex and hence $\tanh(m\theta) \geq m\tanh(\theta)$. □

Acknowledgments

This research is partially supported by the Swiss National Science Foundation, grant no. 20020_172710.

References

[1] A. Abdulle. Fourth order Chebyshev methods with recurrence relation. *SIAM Journal on Scientific Computing*, 23(6):2041–2054, 2002.

[2] A. Abdulle and A. A. Medovikov. Second order Chebyshev methods based on orthogonal polynomials. *Numerische Mathematik*, 18:1–18, 2001.

[3] A. Abdulle and G. A. Pavliotis. Numerical methods for stochastic partial differential equations with multiple scales. *Journal of Computational Physics*, 231(6):2482–2497, 2012.

[4] A. Abdulle and G. Rosilho de Souza. Instabilities and order reduction phenomenon of an interpolation based multirate Runge–Kutta–Chebyshev method. *Technical Report, EPFL*, 2020, arXiv:2003.03154 [math.NA].

[5] J. F. Andrus. Numerical solution of systems of ordinary differential equations into subsytems. *SIAM Journal on Numerical Analysis*, 16(4):605–611, 1979.

[6] M. J. Berger and J. Oliger. Adaptive mesh refinement for hyperbolic partial differential equations. *Journal of Computational Physics*, 53(3):484–512, 1984.

[7] C. N. Dawson, D. Qiang, and T. F. Dupont. A finite difference domain decomposition algorithm for numerical solution of the heat equation. *Mathematics of computation*, 57(195):63–71, 1991.

[8] D. A. Di Pietro and A. Ern. *Mathematical aspects of discontinuous Galerkin methods*, volume 69 of *Mathématiques et Applications*. Springer, Berlin and Heidelberg, 2012.

[9] W. E. Analysis of the heterogeneous multiscale method for ordinary differential equations. *Communications in Mathematical Sciences*, 1(3):423–436, 2003.

[10] B. Engquist and Y. Tsai. Heterogeneous multiscale methods for stiff ordinary differential equations. *Mathematics of Computation*, 74(252):1707–1743, 2005.

[11] C. Engstler and C. Lubich. Multirate extrapolation methods for differential equations with different time scales. *Computing*, 58(2):173–185, 1997.

[12] W. H. Enright, T. E. Hull, and B. Lindberg. Comparing numerical methods for stiff systems of O.D.E:s. *BIT Numerical Mathematics*, 15(1):10–48, 1975.

[13] R. E. Ewing, R. D. Lazarov, and A. Vassilev. Finite difference scheme for parabolic problems on composite grids with refinement in time and space. *SIAM Journal on Numerical Analysis*, 31(6):1605–1622, 1994.

[14] R. E. Ewing, R. D. Lazarov, and P. S. Vassilevski. Finite difference schemes on grids with local refinement in time and space for parabolic problems I. Derivation, stability, and error analysis. *Computing*, 45(3):193–215, 1990.

[15] M. J. Gander and L. Halpern. Techniques for locally adaptive time stepping developed over the last two decades. *Lecture Notes in Computational Science and Engineering*, 91(1):377–385, 2013.

[16] C. W. Gear, G. Ioannis, and G. Kevrekidis. Projective methods for stiff differential equations: problems with gaps in their eigenvalue spectrum. *SIAM Journal on Scientific Computing*, 24(4):1091–1106, 2003.

[17] C. W. Gear and D. R. Wells. Multirate linear multistep methods. *BIT Numerical Mathematics*, 24(4):484–502, 1984.

[18] M. J. Grote, M. Mehlin, and T. Mitkova. Runge–Kutta-based explicit local time-stepping methods for wave propagation. *SIAM Journal on Scientific Computing*, 37(2):A747–A775, 2015.

[19] A. Guillou and B. Lago. Domaine de stabilité associé aux formules d'intégration numérique d'équations différentielles, à pas séparés et à pas liés. Recherche de formules à grand rayon de stabilité. In *1er Congr. Ass. Fran. Calcul., AFCAL*, pages 43–56, Grenoble, 1960.

[20] M. Günther, A. Kværnø, and P. Rentrop. Multirate partitioned Runge–Kutta methods. *BIT Numerical Mathematics*, 41(3):504–514, 2001.

[21] M. Günther and P. Rentrop. Multirate ROW methods and latency of electric circuits. *Applied Numerical Mathematics*, 13(1-3):83–102, 1993.

[22] E. Hairer, S. P. Nörsett, and G. Wanner. *Solving ordinary differential equations I*, volume 8 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 2008.

[23] E. Hairer and G. Wanner. *Solving ordinary differential equations II*, volume 14 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 2002.

[24] M. Hochbruck and A. Ostermann. Exponential integrators. *Acta Numerica*, 19:209–286, 2010.

[25] E. Hofer. A partially implicit method for large stiff systems of ODEs with only few equations introducing small time-constants. *SIAM Journal on Numerical Analysis*, 13(5):645–663, 1976.

[26] B. S. Kirk, J. W. Peterson, R. H. Stogner, and G. F. Carey. libMesh : a C++ library for parallel adaptive mesh refinement/coarsening simulations. *Engineering with Computers*, 22(3-4):237–254, 2006.

[27] A. Kværnø. Stability of multirate Runge–Kutta schemes. In *Proc. of the 10th Coll. on Differential Equations*, volume 1A, pages 97–105, 1999.

[28] V. I. Lebedev. How to solve stiff systems of differential equations by explicit methods. In *Numerical methods and applications*, pages 45–80. CRC, Boca Raton, FL, 1994.

[29] V. I. Lebedev and A. A. Medovikov. Explicit methods of second order for the solution of stiff systems of ODEs. *Russian Academy of Science*, 1994.

[30] B. Lindberg. IMPEX: a program package for solution of systems of stiff differential equations. Technical report, Dept. of Information Processing, Royal Inst. of Tech., Stockholm, 1972.

[31] A. A. Medovikov. High order explicit methods for parabolic equations. *BIT Numerical Mathematics*, 38(2):372–390, 1998.

[32] R. Minero, M. J. H. Anthonissen, and R. M. M. Mattheij. A local defect correction technique for time-dependent problems. *Numerical Methods for Partial Differential Equations*, 22(1):128–144, 2006.

[33] T. Mirzakhanian. *Multi-rate Runge–Kutta–Chebyshev time stepping for parabolic equations on adaptively refined meshes.* Master thesis, Boise State University, 2017. doi:10.18122/B2V715.

[34] J. R. Rice. Split Runge–Kutta method for simultaneous equations. *Journal of research, National Bureau of Standards. Section B, Mathematics and mathematical physics*, 64B(3):151–170, 1960.

[35] G. Rosilho de Souza. *Numerical methods for deterministic and stochastic differential equations with high contrast and multiple scales.* PhD thesis, EPFL, 2020.

[36] V. Savcenco, W. Hundsdorfer, and J. Verwer. A multirate time stepping strategy for stiff ordinary differential equations. *BIT Numerical Mathematics*, 47(1):137–155, 2007.

[37] V. Savcenco and R. M. M. Mattheij. Multirate numerical integration for stiff ODEs. In *Progress in industrial mathematics at ECMI 2008*, volume 15, pages 327–332. Springer, Heidelberg, 2010.

[38] G. I. Shishkin and P. N. Vabishchevich. Interpolation finite difference schemes on grids locally refined in time. *Computer Methods in Applied Mechanics and Engineering*, 190(8-10):889–901, 2000.

[39] S. Skelboe and P. U. Andersen. Stability properties of backward Euler multirate formulas. *SIAM Journal on Scientific and Statistical Computing*, 10(5):1000–1009, 1989.

[40] B. P. Sommeijer, L. Shampine, and J. G. Verwer. RKC: An explicit solver for parabolic PDEs. *Journal of Computational and Applied Mathematics*, 88(2):315–326, 1998.

[41] R. Trompert and J. Verwer. A static-regridding method for two-dimensional parabolic partial differential equations. *Applied Numerical Mathematics*, 8(1):65–90, 1991.

[42] R. Trompert and J. Verwer. Analysis of local uniform grid refinement. *Applied Numerical Mathematics*, 13(1-3):251–270, 1993.

[43] R. Trompert and J. Verwer. Analysis of the implicit Euler local uniform grid refinement method. *SIAM Journal on Scientific Computing*, 14(2):259–278, 1993.

[44] R. Trompert and J. Verwer. Runge–Kutta methods and local uniform grid refinement. *Mathematics of Computation*, 60(202):591–616, 1993.

[45] P. J. Van der Houwen and B. P. Sommeijer. On the internal stability of explicit, m-stage Runge–Kutta methods for large m-values. *Z. Angew. Math. Mech.*, 60(10):479–485, 1980.

[46] E. Vanden-Eijnden. Numerical techniques for multi-scale dynamical systems with stochastic effects. *Communications in Mathematical Sciences*, 1(2):385–391, 2003.

[47] J. G. Verwer. An implementation of a class of stabilized explicit methods for the time integration of parabolic equations. *ACM Transactions on Mathematical Software (TOMS)*, 6(2):188–205, 1980.

[48] J. G. Verwer. Explicit Runge–Kutta methods for parabolic partial differential equations. *Applied Numerical Mathematics*, 22(1-3):359–379, 1996.

[49] J. G. Verwer, W. Hundsdorfer, and B. P. Sommeijer. Convergence properties of the Runge–Kutta–Chebyshev method. *Numerische Mathematik*, 57(1):157–178, 1990.