Second order explicit stabilized multirate method for stiff differential equations with error control

Mathieu Benninghoff[a] and Gilles Vilmart[b]

October 17, 2025

Abstract

Explicit stabilized methods are highly efficient time integrators for large and stiff systems of ordinary differential equations especially when applied to semi-discrete parabolic problems. However, when local spatial mesh refinement is introduced, their efficiency decreases, since the stiffness is driven by only the smallest mesh element. A natural approach is to split the system into fast stiff and slower mildly stiff components. In this context, [A. Abdulle, M.J. Grote and G. Rosilho de Souza 2022] proposed the order one multirate explicit stabilized method (mRKC). We extend their approach to second order and introduce the new multirate ROCK2 method (mROCK2), which achieves high precision and allows a step-size strategy with error control. Numerical methods including the heat equation with local spatial mesh refinements confirm the accuracy and efficiency of the scheme.

Key words. stabilized second-order Runge–Kutta methods, explicit time integrators, stiff equations, multirate methods, local time-stepping, parabolic problems, Chebyshev methods.

AMS subject classifications (2020). 65L04, 65L06, 65L20.

1 Introduction

Semi-discrete parabolic problems yield large systems of ordinary differential equations (ODEs) where the stiffness is strongly influenced by the spatial mesh refinement. In fact, considering a finite element discretization, the stiffness depends only on the diameter of the smallest element. Consequently, when the spatial discretization involves local refinements, the stiffness of the system and the stability of numerical integrators is determined by only a few elements. A natural strategy is to split the system into two parts: an expensive but only mildly stiff component, associated with relatively slow (S) time scales, and a cheap but severely stiff component, associated with fast (F) time scales. This leads us to consider the system:

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

where $\dot{y}(t) = \frac{dy}{dt}$ and the vector field $f : \mathbb{R}^n \to \mathbb{R}^n$ can be spitted into f_S, the expensive but only mildly stiff part and f_F, the cheap but severely stiff part. This decomposition also arises naturally in other contexts, such as chemical reaction kinetics and the modeling of electrical circuits.

For stiff problems on high-dimensional systems, standard explicit Runge-Kutta methods are limited by severe time-step restrictions, while implicit methods are computationally costly for large systems. Explicit stabilized Runge–Kutta schemes, such as RKC based on Chebyshev polynomials [17, 18, 20, 15] (see also the presentation in [10, 2]), and ROCK methods which rely on orthogonal polynomials with quasi-optimal stability [1, 4], overcome this limitation by enlarging

[1] Université de Genève, Section de mathématiques, CP 64, 1211 Genève 4, Switzerland, Mathieu.Benninghoff@unige.ch

[2] Université de Genève, Section de mathématiques, CP 64, 1211 Genève 4, Switzerland, Gilles.Vilmart@unige.ch

stability regions along the negative real axis, making them efficient for uniformly refined meshes, see for instance the explicit stabilized method PIROCK [6] buit upon ROCK2 and applied recently in astrophysics in the context of solar atmosphere simulations [21]. However, their performance deteriorates under local mesh refinement, where stiffness depends only on a few fine elements.

For such problems, it is desirable to construct methods that combine a multirate strategy with the favorable stability properties of explicit stabilized schemes. A partitioned approach has been proposed in the literature [7, 23] and more recently in [16], but these methods are adapted to problems where the slow component is assumed non-stiff, which is not the case in the context of local mesh refinement. To overcome this limitation, a method combining multirate strategy and stabilized integrators was introduced by A. Abdulle, G.R. de Souza and M. J. Grote [3]: the multirate RKC scheme (mRKC)[1]. This scheme applies explicit stabilized Runge–Kutta techniques to an averaged force f_η, whose stiffness is independent of f_F. A micro-step procedure with step size η is then employed to approximate the averaged force while stabilizing the fast component f_F, whereas the macro-step τ is used to integrate the full system and stabilize the slow component f_S. These techniques have been developed in the context of first-order methods. Extending such stabilized multirate methods to high order is however challenging. Indeed, an earlier attempt in [5] is an interpolation-based approach that employs two different explicit stabilized methods, but which suffers from undesired order reduction phenomena and instabilities.

In this paper, we extend this approach [5, 3] by introducing mROCK2, a second-order explicit stabilized multirate method based on the same averaged-force concept. Analogously to the first order mRKC, mROCK2 does not require any scale separation or interpolation between stages, but its improved order two of convergence yields not only a better accuracy, but also it allows for an adaptive step-size strategy with error control which is essential of an efficient time integration of large dimensional dissipative systems.

To construct this method, this paper is organized as follows. Section 2 builds on the approach introduced in [3], focusing on a modified equation whose spectrum is bounded by the slow component of (1.1). While the previous work relies on a modified equation providing a first-order approximation of the original problem, we extend this approach by introducing a new modified equation that achieves a second-order approximation. In Section 3, we present the stabilized methods used to approximate the fast and slow components of the system based on the modified equation. Section 4 is the presentation and the convergence and stability analysis of the new second-order multirate stabilized scheme, the mROCK2 method. Finally, in Section 5, we demonstrate the effectiveness and efficiency of our approach through applications to stiff ordinary and partial differential equations (PDEs).

2 Averaged force and modified equation

The goal of this section is to construct a multirate strategy based on explicit stabilized methods. Inspired by [3], we replace the force function f with an averaged force whose stiffness depends only on the spectral radius of the slow component. The micro-macro strategy consists of first approximating with a micro method with step size η this averaged force, and then applying a macro method to the corresponding modified equation with a macro-step size τ. We now explain how the averaged force proposed in [3] can be modified to to attain second-order accuracy and we analyze its stability properties.

2.1 Averaged force and limitations for acheiving second-order

The main idea in [3] is to introduce a modified equation

$$\dot{y}_{\eta,1} = f_{\eta,1}(y_\eta), \quad y_{\eta,1}(0) = y_0, \tag{2.1}$$

[1] In the literature RKC usually refers to the second order Runge-Kutta Chebyshev methods [18, 20, 15]. In this paper, it shall refer to the order one version with optimal stability domain size used for the design of mRKC [3].

where the original force function f in (1.1) is replaced by an averaged force $f_{\eta,1}$ that depends on a parameter $\eta > 0$ and defined by

$$f_{\eta,1}(y) = \frac{1}{\eta}\big(u(\eta) - y\big) \tag{2.2}$$

where the auxiliary solution $u : [0, \eta] \to \mathbb{R}^n$ is given by

$$\dot{u} = f_F(u) + f_S(y), \quad u(0) = y. \tag{2.3}$$

This modification results in a reduced spectral radius. Specifically, the spectrum of the Jacobian $\partial f_{\eta,1}/\partial y$, denoted by $\rho_{\eta,1}$, is smaller than that of the Jacobian $\partial f_S/\partial y$, denoted by ρ_S, for $\eta > \frac{2}{\rho_S}$. Thus, the stiffness is governed only by the slow component f_S, which makes the integration of the modified system significantly cheaper. However, the averaged vector field $f_{\eta,1}$ does not yield straightforwardly a second-order method. Indeed, if we perform a Taylor expansion of $y_{\eta,1}$ around $y_0 = y(0)$, we obtain $y_{\eta,1} - y_0 = \tau f_{\eta,1}(y_0) + \frac{\tau^2}{2} f'_{\eta,1}(y_0) f(y) + \mathcal{O}(\tau^3)$ and $f_{\eta,1}(y_0) = f(y) + \frac{\eta}{2} f'_F(y_0) f(y_0) + \mathcal{O}(\eta^2)$. Then, it follows that $y_{\eta,1} - y(\eta) = \frac{\tau\eta}{2} f'_F(y_0) f(y_0) + \frac{\tau^2}{2} f'_{\eta,1}(y_0) f(y_0) - \frac{\tau^2}{2} f'(y_0) f(y_0) + \mathcal{O}(\tau\eta^2 + \tau^2\eta + \tau^3)$. Hence, when using this modified equation, we obtain only a first-order approximation. To construct a second-order scheme, we need to modifiy the definition of of the averaged force (2.2) and introduce a new modification $f_{\eta,2}$, which satisfies

$$f_{\eta,2}(y) = f(y) + \mathcal{O}(\eta^2). \tag{2.4}$$

In the next section, we detail the construction of such a *second-order averaged force.*

2.2 The second order averaged force

Analogously to the first-order case, the original force function f from equation (1.1) is replaced by an *averaged force* $f_{\eta,2}$. The resulting equation becomes

$$\dot{y}_{\eta,2} = f_{\eta,2}(y_\eta), \qquad y_\eta(0) = y_0, \tag{2.5}$$

Definition 2.1. *For $\eta > 0$, the second-order averaged force $f_{\eta,2} : \mathbb{R}^n \to \mathbb{R}^n$ is defined as*

$$f_{\eta,2}(y) = \frac{1}{\eta}(v(\eta) - y), \tag{2.6}$$

where the auxiliary solution $v : [0, \eta] \to \mathbb{R}^n$ is defined by solving successively two fast ODE systems. Considering first $u : [0, \eta] \to \mathbb{R}^n$ given by:

$$\dot{u} = f_F(u) + f_S(y), \qquad u(0) = y, \tag{2.7}$$

we obtain as an intermediate step $f_{\eta,1}(y) = \frac{1}{\eta}(u(\eta) - y)$, the classical averaged-force as defined in (2.3)*. Then we consider the differential equation for $v(t)$ given by:*

$$\dot{v} = f_F\left(v - \frac{\eta}{2} f_{\eta,1}(y)\right) + f_S(y), \qquad v(0) = y. \tag{2.8}$$

For $\eta = 0$, let $f_{0,2} = f$.

This second-order averaged force also provides a spectral compression, Theorem 2.2 (Section 2.3) establishes that the spectral radius satisfies $\rho_{\eta,2} < 1.5\,\rho_s$. In exchange for increased accuracy, we incur an additional computational cost: the evaluation of f_F twice per time step, which is acceptable since f_F is the cheaper part of the force. Furthermore, although spectral reduction is slightly less effective than with $f_{\eta,1}$, the numerical method will later improve this bound to $\bar{\rho}_{\eta,2} < 1.35\,\rho_s$ (Theorem 4.5).

2.3 Stability analysis of the modified equation

We now study the stiffness of the modified equation (2.5). We determine necessary conditions to have robust bounds on $\rho_{\eta,2}$ of the system. It will only depend on ρ_S the spectral radius of the Jacobian of f_S, and hence the stiffness of the modified equation only depends on the slow components.

Let the Jacobians of f_F and f_S be simultaneously diagonalizable in the same basis. Then, after linearization of the dynamics, the stability analysis of equations (1.1) and (2.5) reduces to the scalar *multirate test equation*:

$$\dot{y} = \lambda y + \zeta y, \qquad\qquad y(0) = y_0, \tag{2.9}$$

with $\lambda, \zeta \leq 0$ 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, we have $\lambda \in [-\infty, 0]$ where $|\lambda|$ is possibly very large. Using the variation of constant formula with $\varphi(z) = \frac{e^z - 1}{z}$, we first calculate $f_{\eta,1}$ obtained from (2.7):

$$u(\eta) = (e^{\eta\lambda} + \varphi(\eta\lambda)\eta\zeta)y, \tag{2.10}$$

$$f_{\eta,1}(y) = \varphi(\eta\lambda)(\lambda + \zeta)y \tag{2.11}$$

Then, we evaluate $v(\eta)$ (2.8) and $f_{\eta,2}$ defined in (2.6):

$$v(\eta) = e^{\eta\lambda}y + \varphi(\eta\lambda)\eta\zeta y - \varphi(\eta\lambda)\frac{\eta^2\lambda}{2}f_{\eta,1}(y), \tag{2.12}$$

$$f_{\eta,2}(y) = \varphi(\eta\lambda)\left(\lambda + \zeta - \frac{\eta\lambda}{2}\varphi(\eta\lambda)(\lambda + \zeta)\right)y. \tag{2.13}$$

Thus, we can rewrite the modified equation (2.5):

$$\dot{y}_{\eta,2} = \varphi(\eta\lambda)(\lambda + \zeta)\left(1 - \frac{\eta\lambda}{2}\varphi(\eta\lambda)\right)y.$$

Now we can establish the stability equation for the modified equation.

Theorem 2.2. *Let $\zeta < 0$, then $\varphi(\eta\lambda)(\lambda + \zeta)(1 - \frac{\eta\lambda}{2}\varphi(\eta\lambda)) \in [\frac{3}{2}\zeta, 0]$ for all $\lambda \leq 0$, if $\eta \geq 2/|\zeta|$.*

Proof. By Theorem 2.7 in [3],

$$\varphi(\eta\lambda)(\lambda + \zeta) \in [-\zeta, 0], \quad \text{for all}\, \zeta < 0,\ \lambda < 0,\ \eta \geq 2/|\zeta|. \tag{2.14}$$

Moreover,

$$1 - \tfrac{\eta\lambda}{2}\varphi(\eta\lambda) = 1 - \tfrac{1}{2}(e^{\eta\lambda} - 1) \in [0, \tfrac{3}{2}], \quad \text{since } e^{\eta\lambda} \in [0, 1].$$

The product of these two intervals gives the result. □

Thus, η does not depend on λ and the result holds for all $\lambda < 0$.

3 Construction of explicit stabilized multirate methods

In the previous section, we introduce a new second-order averaged force $f_{\eta,2}$, extending the first-order averaged force $f_{\eta,1}$. This construction allows the use of a multirate strategy inspired by the mRKC method [3], now adapted to a second-order context by employing $f_{\eta,2}$. The key advantage of this multirate approach lies in its efficiency: the number of evaluations of the expensive component f_S remains independent of the stiffness of the fast component f_F, leading to significant gains over classical explicit stabilized methods.

This section sets the foundation for the new multirate method by first reviewing the explicit stabilized methods RKC [17] and ROCK2 methods [4] that we use for the design of mROCK2. In the new method, RKC is employed as the micro-method to approximate the modified equation $f_{\eta,2}$ (2.6), while the ROCK2 method serves as the macro-method for integration the modified equation (2.5).

In contrast, we recall the previous multirate RKC (mRKC) method [3], which already uses this averaging-based multirate strategy in only first-order accurate. Specifically, it uses the RKC method both to approximate the averaged force as a micro method and to solve the modified equation as a macro method.

3.1 The explicit stabilized Runge-Kutta methods

Stability is a fundamental property of numerical method for stiff differential equations. A method is stable if the numerical solution is bounded throughout the integration process. To study this, we use the scalar test function:

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

with $\lambda \in \mathbb{C}$. When applying a Runge–Kutta method (RK) with step size τ to this problem, the iteration takes the form $y_{k+1} = R(\lambda\tau)y_k$, which by induction yields $y_k = R(\lambda\tau)^k y_0$. The function $R(z)$ is called the stability function. The stability region is then defined as $\mathcal{S} = \{z \in \mathbb{C} : |R(z)| \leq 1\}$. Hence, y_k remains bounded for all k if and only if $\lambda h \in \mathcal{S}$. For implicit RK methods $R(z)$ is a rational function, while for explicit ones, $R(z)$ is a polynomial.

The explicit stabilized Runge methods [1, 4, 12, 17] (see also the survey [2]) are are families of explicit Runge-Kutta methods, with a large number of internal stages built to extend the stability domain along the negative real axis. By increasing the number of internal stages s, these methods enlarge the portion of the negative real axis contained in the stability region ($[-\ell_s, 0] \subset \mathcal{S}$) with size that grows quadratically w.r.t. the number of internal stages s as $\ell_s \sim \beta s^2$ with a constant β independent of $s \to +\infty$. We present here the methods that we use for the design of mROCK2, the first-order RKC method [17] and the second-order ROCK2 method [4].

3.1.1 The RKC method

The first-order RKC method [10, 17] is not much used in practice for deterministic problems due to its low (first) order of accuracy. However, the method possesses an optimally large stability domain. It is defined by the following m-stage Runge-Kutta method that can be implemented as a two-term reccurence as:

$$\begin{aligned} k_0 &= y_n, \\ k_1 &= k_0 + \mu_1 \tau f(k_0), \\ k_j &= \nu_j k_{j-1} + \kappa_j k_{j-2} + \mu_j \tau f(k_{j-1}) \quad j = 2, \ldots, m, \\ y_{n+1} &= k_m, \end{aligned} \tag{3.2}$$

where τ is the step size, $\mu_1 = \omega_1/\omega_0$ and

$$\mu_j = 2\omega_1 b_j / b_{j-1}, \qquad \nu_j = 2\omega_0 b_j / b_{j-1}, \qquad \kappa_j = -b_j / b_{j-2} \qquad \text{for } j = 2, \ldots, m,$$

with $\varepsilon \geq 0$, $\omega_0 = 1 + \varepsilon/m^2$, $\omega_1 = T_m(\omega_0)/T_m'(\omega_0)$ and $b_j = 1/T_j(\omega_0)$ for $j = 0, \ldots, s$.

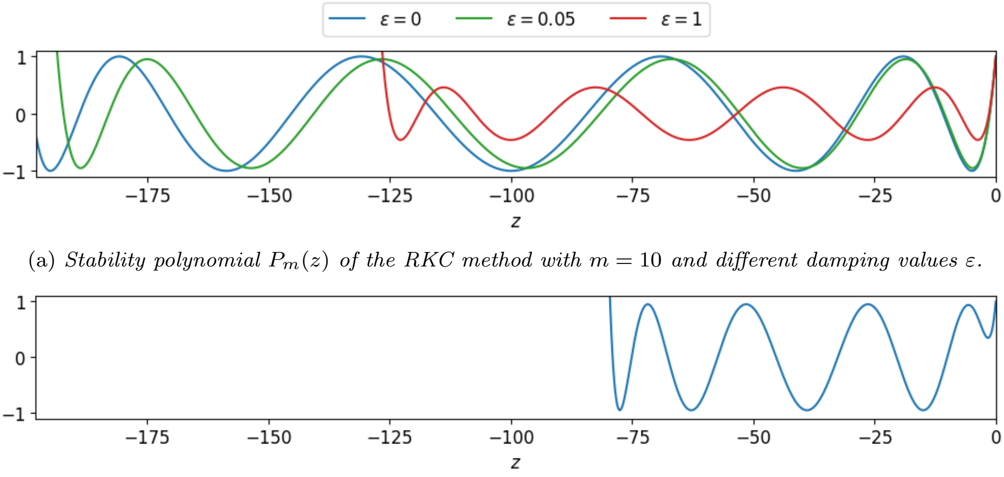
When applied to the test equation (3.1) and using

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

equation (3.2) yields $y_{n+1} = P_m(z)y_n$, where $z = \tau\lambda$ and

$$P_m(z) = b_m T_m(\omega_0 + \omega_1 z) \tag{3.3}$$

is the stability polynomial of the method. As $|T_m(x)| \leq 1$ for $x \in [-1, 1]$ then $|P_m(z)| \leq 1$ for $z \in [-\ell_s^\varepsilon, 0]$ and $\ell_s^\varepsilon = 2\omega_0/\omega_1$. As $\beta m^2 \leq \ell_m^\varepsilon$, for $\beta = 2 - 4\varepsilon/3$, then $|z| \leq \beta m^2$ is a sufficient



(a) *Stability polynomial $P_m(z)$ of the RKC method with $m = 10$ and different damping values ε.*

(b) *Stability polynomial $R_s(z)$ of the ROCK2 method with $s = 10$.*

Figure 1. *The stability function for the RKC and the ROCK2 method.*

condition for stability (see [19]) and the stability domain $\mathcal{S}$ increases quadratically, with respect to the number stage m on the real negative axis. The parameter ε is the damping parameter which avoids for $\varepsilon > 0$ points z in the domain such that $P_m(z) = 1$. The stability domain becomes a bit shorter with size $\ell_s^\varepsilon = \beta m^2$ where $\beta = \big(2 - 4/3\varepsilon\big)$ instead of the optimal length $\ell_s^0 = 2m^2$ achieved without damping $\varepsilon = 0$. In Figure 1(a), we represent the stability function $P_m(z)$ for $m = 10$ and different damping values ε. The number of stages s is chosen large enough such that $\tau\rho \leq \beta s^2$ where ρ is the spectral radius of the Jacobian of f for the equation $\dot{y} = f(y)$. In practice, we choose $s = \left\lfloor \sqrt{\frac{\tau\rho}{\beta}} \right\rfloor + 1$ where $\lfloor\cdot\rfloor$ denotes the integer part function.

3.1.2 The ROCK2 method

Constructing a second-order explicit stabilized is not a trivial task and various methods exist. There exist several methods as the second order RKC [10] or the DUMKA method [11]. We describe here the ROCK2 method [4]. The central idea behind this method was to achieve the second-order accuracy [4] while preserving a three-term recurrence relation and a nearly optimal stability domain with size $0.81\, s^2$. The scheme is defined as follows:

$$\begin{aligned}
k_0 &= y_n \\
k_1 &= k_0 + \tau\mu_1 f(k_0) \\
k_j &= h\mu_j f(k_{j-1}) - \nu_j k_{j-1} - \kappa_j k_{j-2}, \quad j = 2, \dots, s-2 \\
k_{s-1} &= k_{s-2} + \tau\sigma_1 f(k_{s-2}) \\
k_s^\star &= k_{s-1} + \tau\sigma_1 f(k_{s-1}) \\
k_s &= k_s^\star - \tau\sigma_1(1 - \tfrac{\sigma_1}{\sigma_2^2})(f(k_{s-1}) - f(k_{s-2})),
\end{aligned} \tag{3.4}$$

where μ_j, ν_j, and κ_j for $j = 1...s-2$ are the three-term relation of the orthogonal polynomial and $\hat{\omega}(x) = 1 + 2\sigma_1 z + \sigma_2 z^2$. Using the same test function as the first one, we have $R_s(z) = \hat{\omega}(z)\hat{P}_{s-2}(z)$, where $\hat{P}_{s-2}(z)$ is a member of the orthogonal polynomial family $\{\hat{P}_j(z)\}_{j\geq 0}$ with respect to the weight function $\omega(x)^2/(1-x^2)$. Moreover, we note that $\hat{\omega}(z) = \omega(1 + 2z/\ell_s)$, where ℓ_s denotes the length of the stability domain. The polynomial P_{s-2} has degree $s-2$, while ω is a positive polynomial of degree 2 (depending on s). One constructs the polynomials ω such that R_s satisfies

$$R_s(z) = 1 + z + \frac{z^2}{2} + O(z^3),$$

together with a large domain of stability, increasing approximately as $0.81\,s^2$ along the negative real axis. Figure 1(b) represents the stability function of the ROCK2 method for $s = 10$. Thus the length of the negative real axis included in the stability domain is approximately 81.

3.2 Explicit stabilized multirate Runge-Kutta method of order one (mRKC)

The multirate Runge–Kutta–Chebyshev method (mRKC) [3] is a first-order multirate explicit stabilized method. It was the first method that approximates the averaged force (2.1) in order to reduce its spectral radius. This scheme [3] is obtained by discretizing the modified equation with an s-stage RKC method, where $\bar{f}_{\eta,1}$, the approximation of $f_{\eta,1}$, is constructed by solving problem (2.3) with one step of an m-stage RKC method. Let $\tau > 0$ be the step size and ρ_F, ρ_S the spectral radii of the Jacobians of f_F, f_S. Now, let the number of stages s, m be the smallest integers satisfying

$$\tau\rho_S \leq \beta s^2, \qquad \eta\rho_F \leq \beta m^2, \qquad \text{with} \qquad \eta = \frac{6\tau}{\beta s^2}\frac{m^2}{m^2-1}, \tag{3.5}$$

The function Get_Stages$(\Delta t, \rho_S, \rho_F, \epsilon)$ in Algorithm 1 returns the smallest values s, m and η that solve 3.5.

Algorithm 1 One step of the mRKC method [3] with damping $\epsilon = 0.05$ (as presented in [14]).

```
function mRKC_Step($t_n, y_n, \Delta t, f_F, f_S, \rho_F, \rho_S$)
    $s, m, \eta \leftarrow$ Get_Stages($\Delta t, \rho_S, \epsilon$)                ▷ Solve (3.5)
    $\bar{f}_{\eta,1}(t, y) \leftarrow$ Averaged_Force($t, y, \eta, f_F, f_S, m$)
    $y_{n+1} \leftarrow$ RKC($t_n, y_n, \Delta t, \bar{f}_\eta, s, \epsilon$)      ▷ Compute $y_\eta$ from (2.1) using RKC (3.2)
    return $y_{n+1}$
end function
function Averaged_Force($t, y, \eta, f_F, f_S, m$)
    $f_u(r, u) \leftarrow f_F(r, u) + f_S(t, y)$                  ▷ Iterate on $f_F$ with $f_S$ frozen
    $u_\eta \leftarrow$ RKC($t, y, \eta, f_u, m, \epsilon$)                 ▷ Approximate $u(\eta)$ (2.3) using RKC
    return $\bar{f}_{\eta,1} = \frac{1}{\eta}(u_\eta - y)$            ▷ Compute the numerical averaged force $\bar{f}_{\eta,1}$
end function
```

In what follows, we analyze the stability of the scheme using the multirate test equation (2.9). We use the stability analysis of the averaged force $f_{\eta,1}$ (2.10) to understand the stability analysis of the multirate method. The stability analysis from the modified equation is no longer applies once is approximated numerically, since $\varphi(z)$ is replaced by a discrete stability function. For the multirate test equation (2.9), the RKC approximation of u_η is given by

$$u_\eta = \big(P_m(\eta\lambda) + \Phi_m(\eta\lambda)\eta\zeta\big)\, y,$$

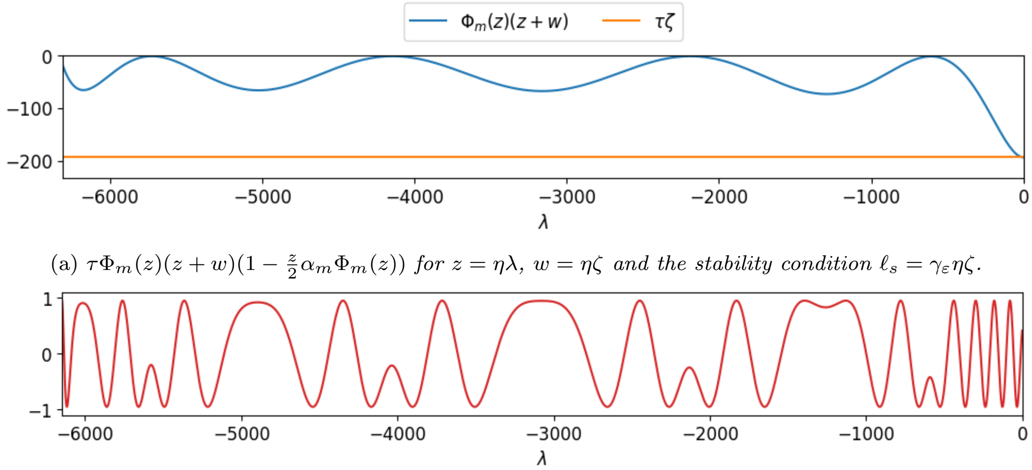
where $P_m(z) = a_m T_m(v_0 + v_1 z)$ is the m-stage RKC stability polynomial, and $\Phi_m(z) = \frac{P_m(z)-1}{z}$ for $z \neq 0$ with $\Phi_m(0) = 1$. For the multirate test equation, the first order averaged force equation (2.11) becomes

$$\overline{f}_{\eta,1}(y) = \Phi_m(\eta\lambda)(\lambda + \zeta)y, \tag{3.6}$$

and thus the full stability function of the mRKC scheme reads

$$y_{n+1} = R_s\big(\tau\Phi_m(\eta\lambda)(\lambda + \zeta)\big)y_n, \tag{3.7}$$

where $R_s(z)$ is the s-stage RKC stability polynomial. By Theorem 4.5 of [3], the method is stable for the multirate test equation (2.9), namely $|R_s\big(\tau\Phi_m(\eta\lambda)(\lambda + \zeta)\big)| \leq 1$ under the condition (3.5).



(a) $\tau\Phi_m(z)(z+w)(1-\frac{z}{2}\alpha_m\Phi_m(z))$ *for* $z=\eta\lambda$, $w=\eta\zeta$ *and the stability condition* $\ell_s=\gamma_\varepsilon\eta\zeta$.

(b) *Stability polynomial* $R_s\big(\tau\Phi_m(\eta\lambda)(\lambda+\zeta)\big)$.

Figure 2. *mRKC on the multirate test equation with* $\tau=1$, $\eta=\frac{6\tau}{\beta s^2}\frac{m^2}{m^2-1}$, $\varepsilon=0.05$, $s=10$, $m=10$, *and* $\zeta=\ell_s$.

In Figure 2, we display the inner (micro) (3.6) and outer (macro) (3.7) stability polynomials for $s=m=10$ as a function of λ. We set $\tau=1$, $\eta=\frac{6\tau}{\beta s^2}\frac{m^2}{m^2-1}$, and $\zeta=\ell_s$. The first plot shows the stability region of the micro-step; the orange line indicates the boundary condition: $\tau\zeta\leq\ell_S$. The second plot show the stability of the RKC method, thus $|R_s\big(\tau\Phi_m(\eta\lambda)(\lambda+\zeta)\big)|\leq 1$ under the condition (3.5).

4 New mROCK2 method: second-order multirate explicit stabilized explicit method

This section introduces the new multirate ROCK2 method (mROCK2), which is constructed by applying an s-stage ROCK2 scheme to discretize the modified equation (2.5), where the second-order averaged force $f_{\eta,2}$ (2.7) is approximated using two successive steps of an m-stage RKC method as introduced in Section 2.2.

We first introduce the mROCK2 algorithm, detailing also a variable step size selection with error control. Then we compare its efficiency gains with the ROCK2 method, and finally analyze its stability and convergence properties.

4.1 New second-order mROCK2 method

Let $\tau>0$ be the step size and ρ_F,ρ_S the spectral radii of the Jacobians of f_F,f_S, the fast and slow component as (1.1). Now, let the number of stages s,m be the smallest integers satisfying

$$\gamma_m\,\tau\rho_S\leq\tilde{\beta}_s s^2,\qquad\qquad \eta\,\rho_F\leq\beta_m m^2,\qquad\qquad \text{with}\quad \eta=\frac{6\tau}{\tilde{\beta}_s s^2}\cdot\frac{m^2}{m^2-1}.$$

Here, $\tilde{\beta}_s$ denotes the stability constant of the ROCK2 method with stability domain size βs^2 ($\tilde{\beta}_s\approx 0.81$ for large s), while β_m is the stability constant of the RKC method ($\beta_m\approx 1.93$ for $\varepsilon=0.05$). In addition, γ_m is a constant that depends on the second derivatives of the RKC method: $\alpha_m=P_m''(0)$ and $\gamma_m=1+\alpha_m$. The function Get_Stages($\Delta t,\rho_S,\rho_F,\epsilon$) in Algorithm 2 returns the smallest values s,m and η that solve the stability constraints (4.5) defined below. Algorithm 2 gives one step of the mROCK2 scheme.

To better understand how the method works, we provide in Table 1 a comparison of the number of evaluations of the slow and fast components of the vector field for one (macro) step

Algorithm 2 One step of the new mROCK2 method

```
function mROCK2_Step($t_n, y_n, \Delta t, f_F, f_S, \rho_F, \rho_S$)
    $s, m, \eta \leftarrow$ Get_Stages($\Delta t, \rho_S, \rho_F, \epsilon$)    ▷ Solve (4.5)
    $\bar{f}_{\eta,2}(t, y) \leftarrow$ Averaged_Force($t, y, \eta, f_F, f_S, m$)
    $y_{n+1} \leftarrow$ ROCK2($t_n, y_n, \Delta t, \bar{f}_{\eta,2}, s$)    ▷ Approximate $y_\eta$ from (2.5) using ROCK2 (3.2)
    return $y_{n+1}$
end function

function Averaged_Force($t, y, \eta, f_F, f_S, m$)
    $f_u(u) \leftarrow f_F(u) + f_S(y)$    ▷ Iterate on $f_F$ with $f_S$ frozen
    $u_\eta \leftarrow$ RKC($t, y, \eta, f_u, m, \epsilon$)    ▷ Approximate $u(\eta)$ from (2.7) using RKC (3.2)
    $\bar{f}_{\eta,1} = \frac{1}{\eta}(u_\eta - y)$
    $f_v(v) \leftarrow f_F(v - \frac{\alpha_m \eta}{2}\bar{f}_{\eta,1}) + f_S(y)$    ▷ As $\alpha_m = P_m''(0)$, we impose $\bar{f}_{\eta,2}(y) = f(y) + \mathcal{O}(\eta^2)$
    $v_\eta \leftarrow$ RKC($t, y, \eta, f_v, m, \epsilon$)    ▷ Approximate $v(\eta)$ from (2.8) using RKC a second time
    return $\bar{f}_{\eta,2} = \frac{1}{\eta}(v_\eta - y)$    ▷ Compute the numerical averaged force $\bar{f}_{\eta,2}$
end function
```

of the methods, as well as the size of the stability domain, with those the mRKC and ROCK2 methods. Analogously to RKC and ROCK2, we observe that increasing the order of convergence from mRKC to mROCK2 reduces the size of the stability domain. However, this trade-off is clearly acceptable, as the improved accuracy of the second-order method compensates for the reduction in the stability domain (see Section 5). We also included RKC and ROCK2 in the comparison, as it requires a large number of f_S evaluations when f_F is very stiff. In Section 4.4, we discuss in more detail the situations where the multirate method is more efficient.

	Order	$\#f_S$	$\#f_F$	l_s	l_m
RKC	1	m	m	$\approx 2m^2$	$\approx 2m^2$
mRKC	1	s	ms	$\approx 2s^2$	$\approx 0.66s^2m^2$
ROCK2	2	m	m	$\approx 0.81m^2$	$\approx 0.81m^2$
mROCK2	2	s	$2ms$	$\approx 0.6s^2$	$\approx 0.27m^2s^2$

Table 1. *Comparison of the number of function evaluation per time step and the sizes of the stability domain for the RKC, mRKC, ROCK2, and mROCK2 methods.*

4.2 Variable time step with error control

A key advantage of second-order methods over first-order ones is their compatibility with variable time stepping and error control. To implement variable time steps, the error of the mROCK2 method must first be estimated. The microstep can be neglected in this estimate, while the overall error is dominated by the macrostep. Indeed, if $s \gg 1$, then $\eta \ll \tau$ and the errors made when approximating f by the averaged force $f_{\eta,2}$ is negligible. We observe numerically in Section 5 that the numerical solutions of mROCK2 and ROCK2 are very close. Therefore, it is natural to adopt the error estimator originally developed for ROCK2 ([4], Section 6.3), that we recall here for completeness:

$$\mathrm{err} = \|k_s^\star - k_s\|$$

where $k_s^\star$ and k_s denote, respectively, the penultimate and the final stages of the ROCK2 algorithm as defined in (3.4). Finally, compared to the usual stepsize selection for non-stiff problems

$$\tau_{\text{new}} = \text{fac} \cdot \tau_n \left(\frac{1}{\text{err}_{n+1}} \right)^{1/2}, \tag{4.1}$$

we shall consider the same step size selection strategy with memory, originally introduced by Watts [22] and Gustafsson [9] to tackle the integration of stiff dynamics:

$$\tau_{\text{new}} = \text{fac} \cdot \tau_n \left(\frac{1}{\text{err}_{n+1}} \right)^{1/2} \frac{\tau_n}{\tau_{n-1}} \left(\frac{\text{err}_n}{\text{err}_{n+1}} \right)^{1/2}. \tag{4.2}$$

Here we use the value fac $= 0.8$ commended for stiff problems (instead of fac $= 0.9$ for non-stiff Runge-Kutta methods). As recommended in [10, p. 125], we take for τ_{new} the minimum of the step sizes proposed by (4.2) and the conventional strategy (4.1). This conventional strategy is also applied after a step rejection which occurs if err $>$ tol where tol is the brecribed accuracy. In Section 5, we observe the efficiency of such a variable time step strategy notably for the stiff Robertson problem (see Figure 6(b)).

4.3 Computation of the number of stages

To simplify the calculation of s, m, and η, we fix the damping parameter to $\varepsilon = 0.05$, although any value $\varepsilon > 0$ could be used to increase the damping if needed. The general system of constraints obtained from the stability analysis in Section 4.5 reads:

$$(1 + \alpha_m)\, \tau \rho_S \leq \tilde{\beta}_s s^2, \qquad \eta \rho_F \leq \beta_m m^2, \qquad \text{with} \qquad \eta \geq \frac{2\tau}{\alpha_m \tilde{\beta}_s s^2}. \tag{4.3}$$

We note that obtaining a computable system requires an estimate of α_m. Hence, we give the following bounds.

Lemma 4.1. *Let $m \in \mathbb{N}$ and $\varepsilon > 0$ and define $\alpha_m(\varepsilon) := P_m''(0)$ where $P_m(z)$ is the stability polynomial given in* (3.3) *of the RKC method. Then,*

$$\alpha_m(0) < \alpha_m(\varepsilon) < \alpha(\varepsilon), \tag{4.4}$$

where $\alpha_m(0) = \frac{1}{3} \cdot \frac{m^2-1}{m^2}$ and $\alpha(\varepsilon) := \lim_{m\to\infty} \alpha_m(\varepsilon) = \frac{1}{\tanh(\sqrt{2\varepsilon})} \left(\frac{1}{\tanh(\sqrt{2\varepsilon})} - \frac{1}{\sqrt{2\varepsilon}} \right)$.

The first inequality in (4.4) follows directly from Lemma 4.2 proved in [3], while the second inequality in (4.4) is proved in Appendix.

Lemma 4.2. *[3, Lemma A.1] The function $\alpha_m(\varepsilon)$ is an increasing function of $\varepsilon > 0$.*

Using (4.4), we obtain:

$$1 + \alpha_m < 1 + \alpha(\varepsilon), \quad \text{and} \quad \frac{2\tau}{\alpha_m \tilde{\beta}_s s^2} \geq \frac{6\tau}{\tilde{\beta}_s s^2} \cdot \frac{m^2}{m^2 - 1}.$$

For the typical value $\varepsilon = 0.05$, we estimate $\alpha(\varepsilon) \approx 0.34 < 0.35$. Based on these bounds, we can reformulate the stability constraints as follows:

$$1.35\, \tau \rho_S \leq \tilde{\beta}_s s^2, \qquad \eta\, \rho_F \leq \beta_m m^2, \qquad \text{with} \quad \eta = \frac{6\tau}{\tilde{\beta}_s s^2} \cdot \frac{m^2}{m^2 - 1}. \tag{4.5}$$

Remark 4.3 (Relaxed stability condition). In the presence of scale separation, when $\lambda \ll \zeta$, the stability requirements can be relaxed, as observed also for mRKC in [3]. In this regime of scale separation, it becomes advantageous to adjust the micro–step size η to a modified value $\bar{\eta}$, which leads to the following stability constraints for mROCK2:

$$1.35\,\tau\rho_S \leq \tilde{\beta}_s s^2, \qquad \bar{\eta}\,\rho_F \leq \beta_m m^2, \qquad \text{with} \quad \bar{\eta} = \frac{2.8}{\tilde{\beta}_s s^2}. \tag{4.6}$$

For more details, see Remark 4.8. The value of $\bar{\eta}$ is clearly smaller than (4.5). Thus, we need fewer evaluations of f_F and the computational cost is reduced.

4.4 Computational cost of the multirate ROCK2 method

Given the spectral radii ρ_f and ρ_s of the Jacobians of f_F and f_S, respectively, we now analyze the theoretical speed-up of the mROCK2 method compared to the standard ROCK2 method. For this purpose, we assume $\varepsilon = 0$ and allow the parameters s and m to vary continuously in $\mathbb{R}$.

We introduce c_F and c_S as relative costs of evaluating f_F and f_S, normalized with respect to the cost of evaluating the function f. These coefficients satisfy $c_F, c_S \in [0,1]$ and $c_F + c_S = 1$. We also suppose that the spectral radius of the Jacobian $\rho = \rho_F + \rho_S$, to allow $\rho_F < \rho_S$.

Since the ROCK2 scheme requires $s = \sqrt{5\tau\rho/4}$ evaluations of f per time step, its cost per time step is

$$C_{ROCK2} = s(c_F + c_S) = \sqrt{\frac{5\tau(\rho_F + \rho_S)}{4}}. \tag{4.7}$$

For the mROCK2 method, on the other hand, we infer from (4.3) with the approximation $\tilde{\beta}_s = \frac{4}{5}$,$\beta_m = 2$ and $\gamma_\epsilon = \frac{4}{3}$ and $\frac{m-1}{m^2} \approx 1$. Thus, we have the system of equations:

$$\frac{4}{3}\,\tau\rho_S = \frac{4}{5}s^2, \qquad \eta\,\rho_F = 2m^2, \qquad \text{with} \quad \eta = \frac{15\tau}{2s^2}.$$

Its gives $s = \sqrt{\frac{5}{3}\tau\rho_S}$ external stages and $\eta = \frac{9}{2}\frac{1}{\rho_S}$that it needs $m = \sqrt{\eta\rho_F/2} = \sqrt{\frac{9}{4}\rho_F/\rho_S}$ internal stages. Since mROCK2 needs s evaluations of f_S and $2\,s\,m$ evaluations of f_F, its cost per time step is

$$C_{mROCK2} = s\,c_S + 2s\,m\,c_F \approx (1 - c_F)\sqrt{\frac{5\tau\rho_S}{3}} + c_F\left(\sqrt{15\tau\rho_F}\right) \tag{4.8}$$

The ratio between equations (4.7) and (4.8) yields the *relative speed-up*

$$S = \frac{C_{ROCK2}}{C_{mROCK2}} = \frac{\sqrt{\rho_F + \rho_S}}{(1 - c_F)\sqrt{\frac{4\tau\rho_S}{3}} + c_F\left(\sqrt{12\tau\rho_F}\right)} = \sqrt{\frac{3}{4}}\frac{\sqrt{1 + r_\rho}}{1 + c_F\left(\sqrt{9r_\rho} - 1\right)},$$

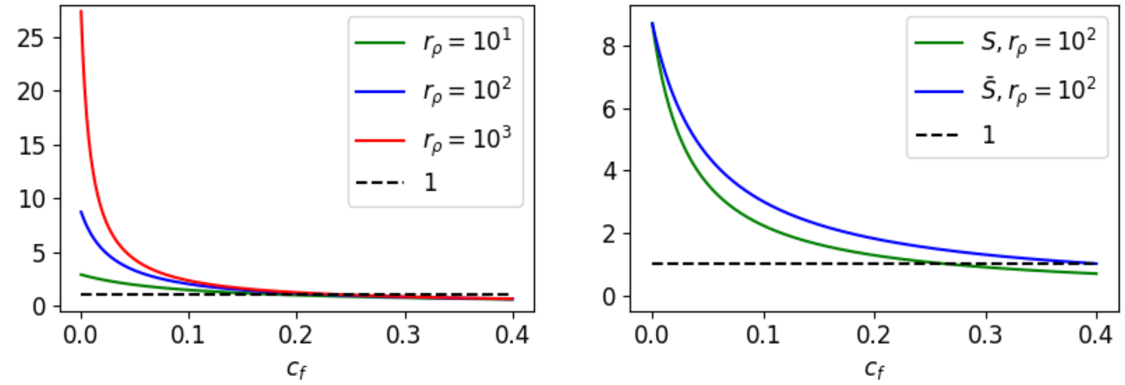
with *stiffness ratio* $r_\rho = \rho_F/\rho_S \in [0, \infty)$.

In Figure 3(a), we display the speed-up S as a function of c_F for various values of the ratio $r_\rho = \rho_f/\rho_s$. When c_F is sufficiently small, the mROCK2 scheme consistently outperforms the standard ROCK2 method ($S > 1$). However, for values around $c_F \approx 0.3$, the mROCK2 scheme becomes slightly less efficient ($S < 1$). This case is less important, since we assume that f_F is relatively inexpensive to evaluate.

Remark 4.4. (Relaxed stability condition) In the case of scale separation (4.6), where the stability analysis is provided in Remark 4.8, the value of η is smaller than in (4.5). This reduction leads to fewer evaluations of f_F. We denote $\bar{S}$ the relative speed-up of mROCK2 compared to the ROCK2 scheme under condition (4.6). The corresponding speed-up is then given by:

$$\bar{S} = \sqrt{\frac{3}{4}}\frac{\sqrt{1 + r_\rho}}{1 + c_F\left(\sqrt{4r_\rho} - 1\right)}. \tag{4.9}$$

In Figure 3(b), we compare it with the speed-up obtained without scale separation, S. We have always $\bar{S} > S$.



(a) *Theoretical speed-up S of the mROCK2 over ROCK2 scheme scheme with respect to c_F and $r_\rho = \rho_F/\rho_S$*

(b) *Comparison between S (the relative speed-up without scale separation) and $\bar{S}$ (the relative speed-up with scale separation) for $r_\rho = 100$*

Figure 3. *Computational cost of the mROCK2 compared to the ROCK2 method*

4.5 Stability analysis of the mROCK2 method

We consider the multirate test equation (2.9) and examine both the discrete analogue of the second-order force (2.6) and the modified equation (2.5). The objective of this section is to establish the stability of the mROCK2 method under the condition (4.3). Let $R_{s,m}(\lambda,\zeta,\tau,\eta)$, denote the stability function of the mROCK2 method, defined by

$$y_{k+1} = R_{s,m}(\lambda,\zeta,\tau,\eta)y_k, \tag{4.10}$$

when we apply the mROCK2 method on the multirate test equation: $\dot{y} = (\lambda+\zeta)y$, $y(0) = y_k$ With this formulation, we are now in a position to state the following theorem.

Theorem 4.5. *Let $\lambda \le 0$ and $\zeta < 0$, $\alpha_m = P''_m(0)$ and $\tilde{\beta}_s$ the stability coefficient of the ROCK2 method. Then, for all $\tau > 0, s, m$ and η such that*

$$(1+\alpha_m)\tau|\zeta| \le \tilde{\beta}_s s^2, \qquad \eta|\lambda| \le \ell_m^{\varepsilon_m} \qquad \text{with} \qquad \eta \ge \frac{2\tau(1+\alpha_m)}{\alpha_m \ell_s}, \tag{4.11}$$

$|R_{s,m}(\lambda,\zeta,\tau,\eta)| \le 1$, i.e. the mROCK2 scheme is stable.

First we have to find the value of $R_{s,m}(\lambda,\zeta,\tau,\eta)$:

Lemma 4.6. *Let $\lambda,\zeta \le 0$, $f_S(y) = \zeta y$, $f_F(y) = \lambda y$, $\eta > 0$, $m,s \in \mathbb{N}$ and $y \in \mathbb{R}$. Then, the stability function of the mROCK2 method is given by*

$$R_{s,m}(\lambda,\zeta,\tau,\eta) = R_s\left(\tau\Phi_m(\eta\lambda)(\lambda+\zeta)\left(1-\frac{\eta\lambda\alpha_m}{2}\Phi_m(\eta\lambda)\right)\right)$$

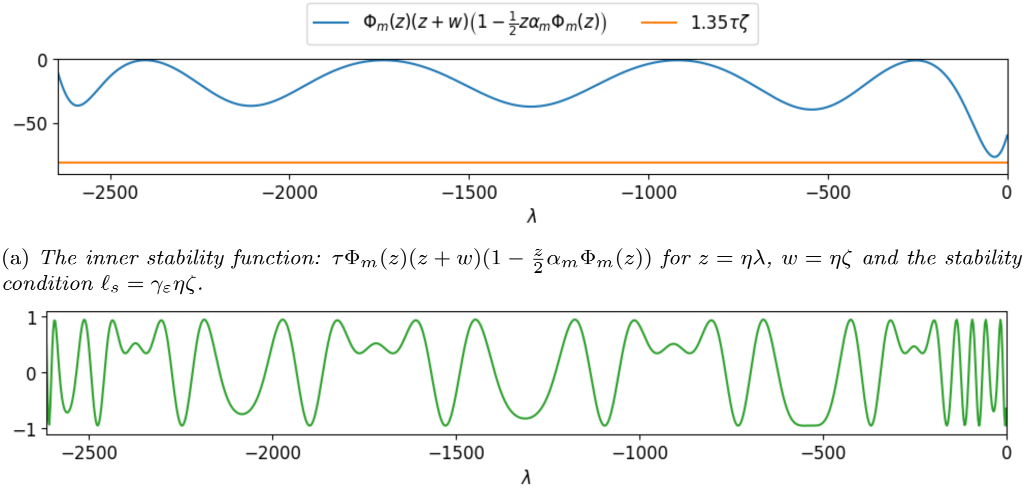
where $\Phi_m(z) = \frac{P_m(z)-1}{z}$ and $\alpha_m = P''_m(0)$ and $R_s(z)$ is the stability polynomial of the s-stage ROCK2 scheme.

Proof. We use the fact $\bar{f}_{\eta,1} = \Phi_m(\eta\lambda)(\lambda+\zeta)y$ from (3.6). Then, by definition of v_η,

$$v_\eta = P_m(\eta\lambda)y + \eta\Phi_m(\eta\lambda)(\zeta y - \frac{\eta\lambda\alpha_m}{2}\bar{f}_{\eta,1})$$

Then, we calculate $\bar{f}_{\eta,2} = \frac{1}{\eta}(v_\eta - y) = \Phi_m(\eta\lambda)(\lambda+\zeta)\left(1-\frac{\eta\lambda\alpha_m}{2}\Phi_m(\eta\lambda)\right)$.
Finally, we compute the internal stage into the external stage, which leads to

$$y_{n+1} = R_s\left(\tau\Phi_m(\eta\lambda)(\lambda+\zeta)\left(1-\frac{\eta\lambda\alpha_m}{2}\Phi_m(\eta\lambda)\right)\right)y_n,$$



(a) *The inner stability function:* $\tau\Phi_m(z)(z+w)(1-\frac{z}{2}\alpha_m\Phi_m(z))$ *for* $z=\eta\lambda$*,* $w=\eta\zeta$ *and the stability condition* $\ell_s=\gamma_\varepsilon\eta\zeta$*.*

(b) *Stability function* $R_{s,m}(\lambda,\zeta,\tau,\eta)$

Figure 4. *The inner and outer stability function of the mROCK2 for* $w=\eta\zeta$*,* $z=\eta\lambda$*,* $\tau=1$*,* $\eta=\frac{2\tau(1+\alpha_m)}{\alpha_m\ell_s}$*,* $s=m=10$ *and* $w\approx 60$ *chosen such that the condition* $\gamma_\varepsilon\zeta\tau=0.81\ell_s$ *for* $\varepsilon=0.05$*.*

with $R_s(z)$ the stability polynomial of the s-stage ROCK2 scheme. □

To prove the stability of the mROCK2 scheme, we need the following lemma on the stability property of $\Phi_m(z)$ used and proved in [3]:

Lemma 4.7. *(see [3, Lemma 4.3]) Let* $m\in\mathbb{N}$ *and* $w\le 0$*. There exists* $\overline{\varepsilon}_m>0$ *such that for* $\varepsilon\le\overline{\varepsilon}_m$*,* $\Phi_m(z)(z+w)\in[w,0]$ *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* $\Phi_m'(0)=P_m''(0)/2$*.*

Proof of Theorem 4.5 . $|R_{s,m}(\lambda,\zeta,\tau,\eta)| = |R_s(\tau\Phi_m(\eta\lambda)(\lambda+\zeta))| \le 1$, if $\tau\Phi_m(\eta\lambda)(\lambda+\zeta)(1-\frac{\alpha_m\eta\lambda}{2}\Phi(\eta\lambda))\in[-\ell_s^\varepsilon,0]$. So we have to prove this condition. First, we can remark

$$1-\frac{\alpha_m\eta\lambda}{2}\Phi(\eta\lambda)=1-\frac{\alpha_m}{2}(P_m(\eta\lambda)-1)\in[1,1+\alpha_m]$$

because $P_m(\eta\lambda)\in[-1,1]$. Hence, it is sufficient to prove the equivalent condition:

$$\Phi_m(\eta\lambda)(\eta\lambda+\eta\zeta)\in[w(\eta),0],\qquad \text{with}\qquad w(\eta)=-\frac{\eta}{(1+\alpha_m)\tau}\ell_s^\varepsilon.$$

Since $\eta\lambda\in[-\ell_m^{\varepsilon_m},0]$, it holds $|P_m(\eta\lambda)|\le 1$ and we deduce that $\Phi_m(\eta\lambda)\ge 0$. Furthermore, (4.11) 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 4.7, we know that

$$|w(\eta)|\ge\frac{2}{\alpha_m}$$

is a necessary and sufficient condition which is verified when $\eta\ge\frac{2\tau(1+\alpha_m)}{\alpha_m\ell_s}$. □

Figure 4(a) represents the inner stability polynomial: $\tau\Phi_m(z)(z+w)(1-\frac{1}{2}z\alpha_m\Phi_m(z))$ for $w=\eta\zeta$, $z=\eta\lambda$, $\tau=1$, $\eta=\frac{2\tau(1+\alpha_m)}{\alpha_m\ell_s}$, $s=m=10$ and $w\approx 60$ chosen such that the condition

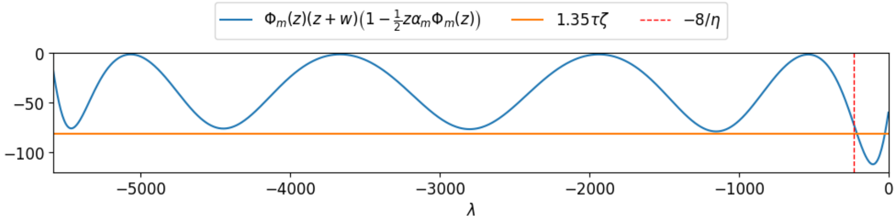


Figure 5. *The inner stability function in the case of scale separation:* $\tau\Phi_m(z)(z+w)(1-\frac{1}{2}z\alpha_m\Phi_m(z))$ *for* $w=\eta\zeta$, $z=\eta\lambda$, $\tau=1$, $\frac{2.8\tau}{\ell_s}$, $s=m=10$ *and* $w\approx 60$ *chosen such that the condition* $\gamma_\varepsilon\zeta\tau=0.81\ell_s$

$\gamma_\varepsilon\zeta\tau=0.81\ell_s$. The orange line represents the stability condition for the macro-step $\ell_s\geq 1.35\tau\rho_s$. We observe that the method remains stable with a certain margin. Thus, the stability function $R_{s,m}(\lambda,\zeta,\tau,\eta)$ shown in Figure 4(b) remains stable for the same parameter values.

As emphasized in [3] for mRKC, the restriction $\varepsilon\leq\bar{\varepsilon}_m$ is necessary to prove Lemma 4.7, but probably not necessary in practice also for mROCK2. Indeed, we can numerically verify that for any $\varepsilon\geq 0$, $\Phi_m(z)(z+w)\in[w,0]$ for all $z\in[-\ell_m^\varepsilon,0]$, if and only if, $|w|\geq 2/P_m''(0)$. Hence, we can suppose $\varepsilon_m=\varepsilon$ in (4.11) and replace $\ell_m^{\varepsilon_m}$ by $\beta_m m^2$ where β_m is the stability coefficient of the RKC method.

Similarly, the condition $\eta\geq\frac{2\tau(1+\alpha_m)}{\alpha_m\ell_s}$ is not restrictive in practice and smaller values of η may be chosen. Indeed, the minimum of $\Phi_m(z)(z+w)\left(1-\frac{1}{2}z\alpha_m\Phi_m(z)\right)$ does not correspond to the product of the minimum of $\Phi_m(z)(z+w)$ and the maximum of $\Phi_m(z)(z+w)\left(1-\frac{1}{2}z\alpha_m\Phi_m(z)\right)$. Based on numerical stability experiments for $m\in[1,200]$, we observe that the micro-step size of the mRKC method remains applicable, namely $\eta=\frac{6\tau}{\ell_s}\frac{m^2-1}{m^2}$.

Remark 4.8 (Stability in the scale separation case). From [13], it is shown that $\Phi_m(z)(z+w)\in[0,w]$, for $w>2$ whenever $z<-8\quad(\lambda<-\frac{8}{\eta})$. This improves upon the condition $w>\frac{2}{P_m''(0)}\approx 6$, which holds under a larger damping parameter $\bar{\varepsilon}=0.1$. In our setting, we keep the original damping $\bar{\varepsilon}=0.05$, but adjust $\bar{\eta}$ in order to increase the stability margin, choosing $\bar{\eta}=\frac{2.8\tau}{\ell_s}>\frac{2\gamma_\varepsilon\tau}{\ell_s}$. Figure 5 illustrates this stability: the blue curve shows the quantity $\tau\Phi_m(z)(z+w)\left(1-\frac{1}{2}z\alpha_m\Phi_m(z)\right)$, the orange line corresponds to the stability condition needed to apply the macro-step mROCK2, and the red vertical line marks the threshold value $\lambda<-\frac{8}{\eta}$ for scale separation.

4.6 Convergence analysis of the mROCK2 method

This section presents the proof of second-order accuracy for mROCK2, the first high-order stabilized multirate method.

Theorem 4.9. *The mROCK2 scheme is second-order accurate*

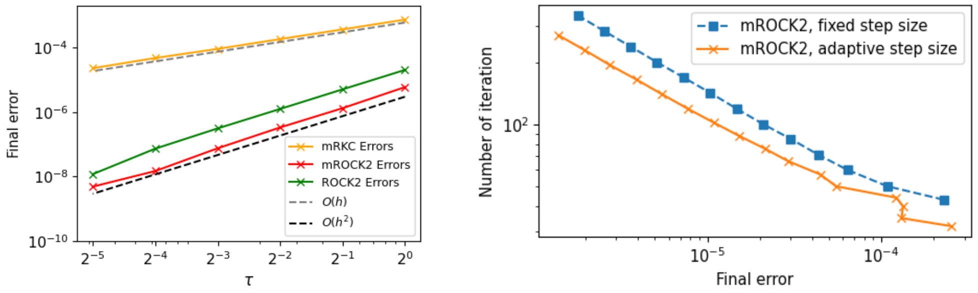
Proof. We estimate the local error after one step. From equation (2.1) with y replaced by y_0 in equation (2.8).

Let $\bar{y}_\eta(\tau)$ be the solution of $\bar{y}_\eta'=\bar{f}_{\eta,2}(\bar{y}_\eta)$ with $\bar{y}_\eta(0)=y_0$ and $\bar{f}_\eta$ as in Algorithm 2. Since the RKC scheme is first-order accurate then $u_\eta=y_0+\eta f(y_0)+\mathcal{O}(\eta^2)$ and $\bar{f}_{\eta,1}(y_0)=f(y_0)+\mathcal{O}(\eta)$. Then,

$$v_\eta=y_0+\eta\left(f(y_0)-\frac{\eta\,\alpha_m}{2}f'(y_0)\bar{f}_{\eta,1}(y_0)\right)+\frac{\eta^2\alpha_m}{2}f'(y_0)f(y_0)+\mathcal{O}(\eta^3)=y_0+\eta f(y_0)+\mathcal{O}(\eta^3)-$$

Thus $\bar{f}_{\eta,2}(y_0)=f(y_0)+\mathcal{O}(\eta^2)$ and $f_{\eta,2}'(y_0)=f'(y_0)+\mathcal{O}(\eta)$, which yields with $y(\tau)$ the exact solution at time τ:

$$\begin{aligned}y(\tau)-\bar{y}_\eta(\tau)&=\tau(f(y_0)-\bar{f}_{\eta,2}(y_0))+\frac{\tau^2}{2}(f'(y_0)f(y_0)-\bar{f}_{\eta,2}'(y_0)\bar{f}_{\eta,2}(y_0))+\mathcal{O}(\tau^3)\\&=\mathcal{O}(\eta^2\tau+\eta\tau^2+\tau^3).\end{aligned}$$



(a) *Convergence of mROCK2, mRKC and ROCK2 on Robertson's stiff problem for* $T = 100$

(b) *Error versus number of iterations for fixed and adaptive step size strategies at* $T = 100$

Figure 6. *Test of the mROCK2 method on the Robertson's stiff problem*

Finally, let y_1 be the solution after one step of the mROCK2 scheme Algorithm 2, which can also be seen as the solution after one step of the ROCK2 scheme applied to $\bar{y}'_\eta = \overline{f}_\eta(\bar{y}_\eta)$. Using the fact that the ROCK2 scheme has second-order accuracy then

$$\bar{y}_\eta(\tau) - y_1 = \mathcal{O}(\tau^3).$$

By triangular inequality we obtain $|y(\tau) - y_1| = \mathcal{O}(\eta^2\tau + \eta\tau^2 + \tau^3)$ and from equation (4.3) follows $\eta \leq \tau$, thus $|y(\tau) - y_1| = \mathcal{O}(\tau^3)$ and the scheme is second-order accurate. □

5 Numerical experiments

In this section, we compare the performance of the mROCK2 method with the classical ROCK2 method through a series of numerical experiments. We first apply both methods to a stiff nonlinear ODE. Additionally, we include a comparison with the order one mRKC to illustrate the advantage of the higher order two of accuracy.

For the second and third experiments, involving PDEs, we study the performance of the methods as both the mesh size H and the time step τ are refined. The efficiency of mROCK2 is evaluated relatively to ROCK2 in each case.

All simulations are implemented in `Python` using the `NumPy` and `Matplotlib` libraries.

5.1 Robertson's stiff test problem

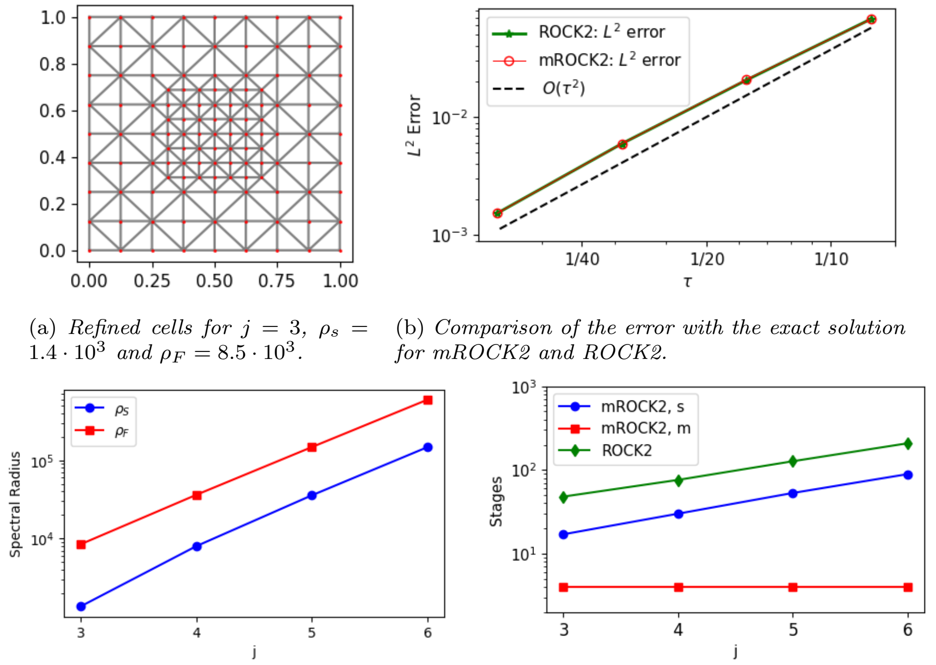
First, we study the convergence of the mROCK2 scheme on a popular stiff test problem, Robertson's nonlinear chemical reaction model [10]:

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

where $t \in [0, 100]$. With this set of parameters and initial conditions, following [3], the term inducing the most severe stiffness is $-10^4\, y_2\, y_3$. Thus, we let

$$f_F(y) = \begin{pmatrix} 0 \\ -10^4\, y_2\, y_3 \\ 0 \end{pmatrix}, \qquad f_S(y) = \begin{pmatrix} -0.04\, y_1 + 10^4\, y_2\, y_3 \\ 0.04\, y_1 - 3\cdot 10^7\, y_2^2 \\ 3\cdot 10^7\, y_2^2 \end{pmatrix}, \qquad f(y) = f_F(y) + f_S(y). \tag{5.2}$$

Now, we solve (5.1) either with the ROCK2, mROCK2 or mRKC scheme using step sizes $\tau = 1/2^k$, $k = 0, \ldots, 5$. For comparison, we use a reference solution obtained with the standard fourth-order



(a) *Refined cells for $j = 3$, $\rho_s = 1.4 \cdot 10^3$ and $\rho_F = 8.5 \cdot 10^3$.*

(b) *Comparison of the error with the exact solution for mROCK2 and ROCK2.*

(c) *Spectral radius of A_F and A_S as the spatial mesh is refined.*

(d) *Theoretical numbers of stages m and s for mROCK2, and n for ROCK2.*

Figure 7. *Heat equation on the unit square with local mesh refinement.*

Runge–Kutta scheme using $\tau = 10^{-4}$. In Figure 6(a), we observe that both the mRKC method achieves first-order convergence while mROCK2 and ROCK2 are two to four orders of magnitude more precise thanks to their second-order convergence. Figure 6(b) illustrates the efficiency of the adaptive step size strategy for the mROCK2 method by comparing the computational cost versus the error. We contrast the performance of mROCK2 with and without adaptivity, using the same adaptive time step strategy (see Section 4.2).

5.2 Heat equation on the unit square

Next, we verify the space-time convergence properties of the considered multirate methods. To do so, we consider the heat equation in the unit square $\Omega = (0, 1) \times (0, 1)$,

$$\begin{aligned} \partial_t u - \Delta u &= g && \text{in } \Omega \times (0, T) \\ u &= 0 && \text{on } \partial\Omega \times (0, T), \\ u &= 0 && \text{on } \Omega \times \{0\}, \end{aligned} \tag{5.3}$$

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 uniform mesh of $2^j \times 2^j$ simplicial elements with $j = 2, \ldots, 5$, we perform two successive local refinements of all elements located within the square domain $\Omega_F = (1/4, 3/4) \times (1/4, 3/4)$. Each refinement step is carried out by subdividing interior triangles into four smaller triangles and boundary triangles into two, by inserting a point at the midpoint of the hypotenuse, see top left picture in Figure 8.

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 (5.3) in space with first-order FEM on the mesh $\mathcal{M}$. After inverting the block-diagonal mass matrix ($A = M^{-1}K$ where M is the mass matrix and K is the stiffness matrix), the resulting system is

$$\dot{y} = A\,y + G, \qquad y(0) = y_0, \tag{5.4}$$

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

$$A_F = DA, \qquad A_S = (I-D)A \qquad \text{and} \qquad f_F(y) = A_F\,y, \qquad f_S(t,y) = A_S\,y + G(t), \tag{5.5}$$

with I the identity. We recall that the spectral radii ρ_S and ρ_F of A_S and A_F behave as $\mathcal{O}(1/H^2)$ and $\mathcal{O}(1/h^2) = \mathcal{O}(4/H^2)$, respectively.

We now consider a sequence of meshes with $j = 2,\dots,5$ and solve (5.3) the mRKC using the same step size $\tau = 1/2^{j/2}$ to have $H = O(\tau^2)$. The parameters s and m for mRKC are chosen according to (4.3).

In Figure 7(b), we display the error at the final time with respect to the time step τ for both mROCK2 and ROCK2. The results show that the two methods exhibit identical convergence behavior. We also plot the spectral radius for each value of j in Figure 7(a), and compare the number of iterations required by each method in Figure 7(d). The number of evaluations of the slow function f_S is observed to be approximately three times higher for the ROCK2 method compared to mROCK2.

5.3 The L-shape diffusion problem

Next, we verify the space-time convergence properties on a non-convex surface, which naturally gives rise to singularities near the corners, as discussed in [8]. To do so, we consider a diffusion in the L-shape includes unit square $\Omega = [0,1]\times[0,1]$ without the square $\Omega = [0.5,1]\times[0,0.5]$,

$$\begin{aligned} \partial_t u - \Delta u &= g && \text{in } \Omega\times[0,T], \\ u &= 0 && \text{in } \partial\Omega\times[0,T], \\ u &= 0 && \text{in } \Omega\times\{0\}, \end{aligned} \tag{5.6}$$

We consider the problem with final time $T = 1$ and a Gaussian source term given by

$$g(x,t) = 10\exp\left(-\frac{(x_1-x_0)^2 + (x_2-x_0)^2}{\sigma^2}\right).$$

As a reference solution, we first perform a standard finite element discretization using a uniform mesh composed of small triangles with diameter 0.005. The classical finite element method (FEM) is used for the spatial discretization, and the ROCK2 method is applied for time integration with a time step $\tau = 0.005$ to obtain a high-quality approximation of the exact solution. Next, we consider a multiresolution approach: we construct a coarser mesh with larger triangles of diameter 0.1, except in the region surrounding the singularity, where we refine the mesh with smaller triangles of diameter 0.01. We then use the matrices A_F and A_S with the same construction as in the previous experiment (but this time without the square $\Omega = [0.5,1]\times[0,0.5]$), and apply the mROCK2 method for time integration on this refined mesh. Figure 8 shows the approximate solution computed on the coarser mesh with local refinement using the mROCK2 method, alongside a comparison with the solution obtained on a uniformly coarse mesh without refinement. We observe that the locally refined mesh yields better accuracy and efficiency. In cases like this, the mROCK2 method proves to be particularly effective: for example, the number of evaluations at each step of the slow component is reduced to 28 with mROCK2, compared to 96 with the ROCK2 method. We add the plot of the number of iteration of f_S (Figure 9(b)) and f_F (Figure 9(c)) and the relative number of iteration for f (Figure 9(d)) and compared it with the cost for the ROCK2 method for $H = 1/5, 1/10, 1/20, 1/40, 1/80$ and $h = H/10$. The relative cost of evaluating f is modeled as $1 = c_F + c_S$, and the relative number of function evaluations per step in the mROCK2 method is given by $(1-c_F)s + 2c_F sm$. In our experiments, we estimate $c_F = 0.05$.

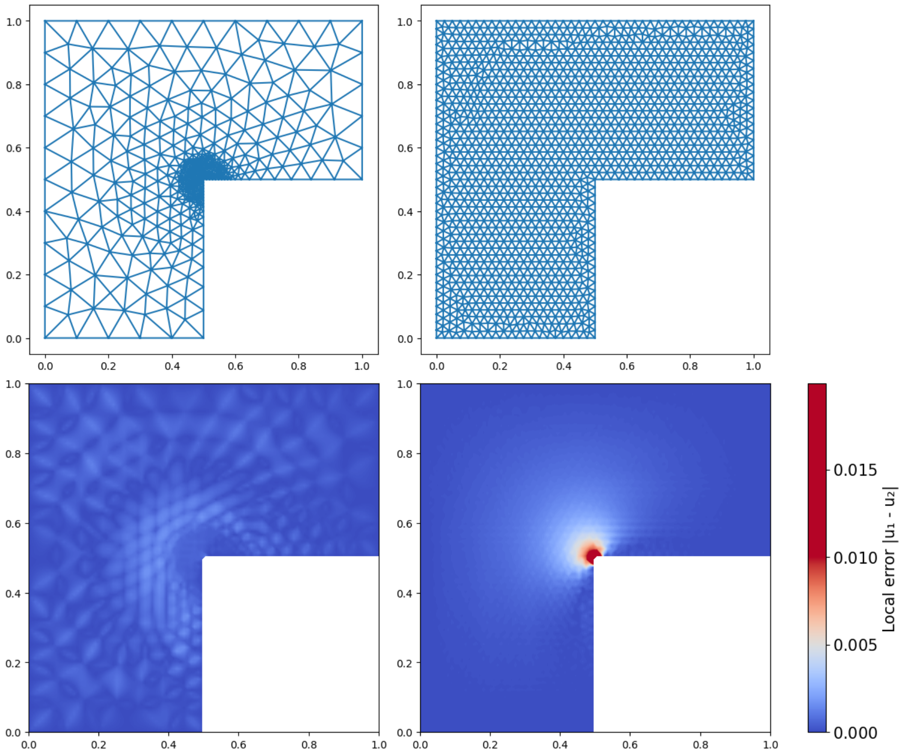
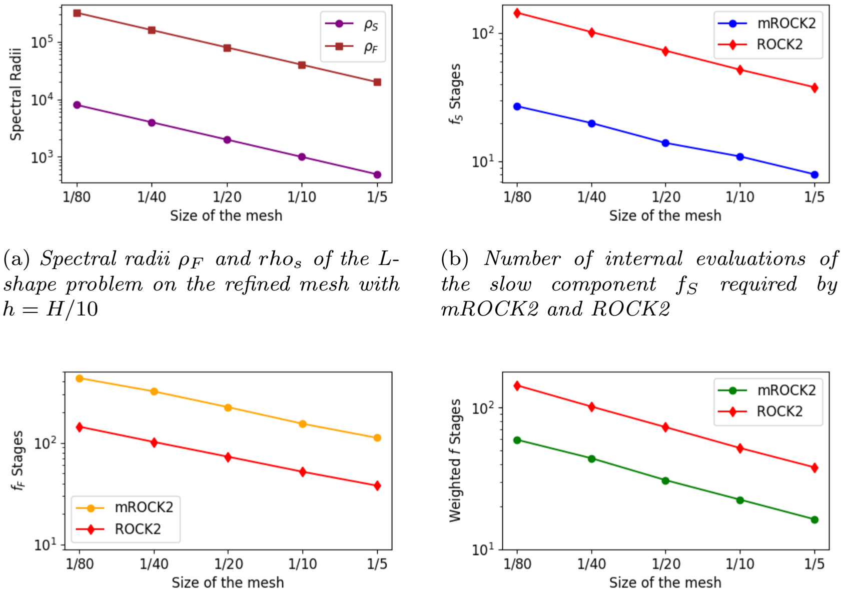


Figure 8. *Local numerical error on the L-shaped domain with finite element discretization with a local mesh refinement (left pictures) and a uniform mesh (right pictures).*



(a) *Spectral radii ρ_F and rho_s of the L-shape problem on the refined mesh with $h = H/10$*

(b) *Number of internal evaluations of the slow component f_S required by mROCK2 and ROCK2*

(c) *Number of internal evaluations of the fast component f_F required by mROCK2 and ROCK2*

(d) *Theoretical number of evaluations of the function f for mROCK2 compared with ROCK2*

Figure 9. *Spectral properties and stage requirements of the mROCK2 and ROCK2 methods for the L-shaped problem*

Acknowledgments

The authors thank Giacomo Rosilho de Souza for providing his code and for his interest in the further development of multirate explicit stabilized methods. This work was partially supported by the Swiss National Science Foundation, projects No 200020_214819 and No. 200020_192129.

6 Appendix

In this appendix, we prove the second inequality in Lemma 4.1. We first establish a few preliminary lemmas:

Lemma 6.1. *For any $\omega_0 > 1$, the following inequality holds:*

$$\frac{\omega_0}{\sqrt{\omega_0^2 - 1}} \operatorname{arccosh}(\omega_0) > 1.$$

Proof. Let $g(\omega_0) := \omega_0 \operatorname{arccosh}(\omega_0) - \sqrt{\omega_0^2 - 1}, \quad \text{for } \omega_0 > 1.$

First, we observe that $g(1) = 0$. Next,we compute the derivative $g'(\omega_0) = \operatorname{arccosh}(\omega_0) > 1$. Thus, g is strictly increasing on $[1, \infty)$. Since $g(1) = 0$ and g increases thereafter, it follows that $g(\omega_0) > 0$ for all $\omega_0 > 1$. This completes the proof. □

Lemma 6.2. *For any integer m and $\varepsilon > 0$,*

$$m \operatorname{arccosh}\left(1 + \frac{\varepsilon}{m^2}\right) < \sqrt{2\varepsilon}$$

Proof. We rewrite the inequality by setting $\delta = \frac{\varepsilon}{m^2}$. Hence, it remains to prove that $\operatorname{arccosh}(1 + \delta) < \sqrt{2\delta}$ for $\delta > 0$.

We define the function $h(\delta) = \sqrt{2\delta} - \operatorname{arccosh}(1 + \delta)$. We want to show that $h(\delta) > 0$ for all $\delta > 0$. $h(0) = 0$ and $h'(\delta) = \frac{1}{\sqrt{2\delta}} - \frac{1}{\sqrt{2\delta + \delta^2}} > 0$ for $\delta > 0$. Therefore, h is strictly increasing on $(0, \infty)$, and since $h(0) = 0$, it follows that $h(\delta) > 0$ for all $\delta > 0$, which completes the proof. □

Lemma 6.3. *For any $x > 0$, the function $\frac{1}{\tanh^2(x)} - \frac{1}{x \tanh(x)}$ is an increasing function*

Proof. Let $\phi(x) = \frac{1}{\tanh^2 x} - \frac{1}{x \tanh(x)}$. We will show that $\phi'(x) > 0$ for $x > 0$.

$$\phi'(x) = \frac{\cosh(x)\sinh^2(x) + x\sinh(x) - 2x^2\cosh(x)}{x^2 \sinh^3(x)}.$$

Since the denominator $x^2 \sinh^3(x) > 0$ for all $x > 0$, the sign of $\phi'(x)$ depends on the numerator. Thus, showing that $\phi'(x) > 0$ is the same as:

$$\psi(x) = \cosh(x)\sinh^2(x) + x\sinh(x) - 2x^2\cosh(x) > 0 \quad \text{for} \quad x > 0.$$

We consider the Taylor series expansions of hyperbolic functions. In particular, from the series for $\sinh(x)$, it follows that $\sinh^2(x) > x^2 + \frac{x^4}{3}$ for all $x > 0$. Exploiting both the Taylor series and this inequality, we bound $\psi(x)$ from below:

$$\begin{aligned}
\psi(x) &> \frac{x^4}{3}\cosh(x) + x\sinh(x) - x^2\cosh(x) \\
&= \sum_{n=0}^{\infty} \frac{1}{3}\frac{x^{2n+4}}{(2n)!} + \frac{x^{2n+2}}{(2n+1)!} - \frac{x^{2n+2}}{(2n)!} \\
&= x^2 - x^2 + \sum_{n=1}^{\infty} \left(\frac{1}{3}\frac{1}{(2n-2)!} + \frac{1}{(2n+1)!} - \frac{1}{(2n)!} \right) x^{2n+2} \\
&= \frac{1}{3} \sum_{n=1}^{\infty} \left(\frac{8n^3 - 6n - 2}{(2n+1)!} \right) y^{2n+2}.
\end{aligned}$$

If $n = 1$, $8n^3 - 6n - 2 = 0$ and it's easy to prove that for all $n > 1$, $8n^3 - 6n - 2 > 0$, so every term in the sum with $n > 1$ is positive. Hence, $\psi(x) > 0$ for all $x > 0$, and consequently $\phi'(x) > 0$ for all $x > 0$. This proves that ϕ is strictly increasing on $(0, \infty)$. □

Proof of the second bound in Lemma 4.1. By definition, $\alpha_m(\varepsilon)$ is given by

$$\alpha_m(\varepsilon) = \frac{T_m(\omega_0)\, T_m''(\omega_0)}{T_m'(\omega_0)^2}, \qquad \text{where } \omega_0 = 1 + \frac{\varepsilon}{m^2}.$$

The Chebyshev polynomials of the first kind satisfy for $\omega_0 > 1$, that $T_m(\omega_0) = \cosh(m \ \mathrm{arccosh}(\omega_0))$ and their derivatives can be expressed as:

$$T_m'(\omega_0) = \frac{m \ \sinh(m \ \mathrm{arccosh}(\omega_0))}{\sqrt{\omega_0^2 - 1}}, \qquad T_m''(\omega_0) = \frac{m^2 \ \cosh(m \ \mathrm{arccosh}(\omega_0))}{\omega_0^2 - 1} - \frac{m\, \omega_0 \ \sinh(m \ \mathrm{arccosh}(\omega_0))}{(\omega_0^2 - 1)^{3/2}}.$$

Using these expressions, we obtain the formula:

$$\alpha_m(\varepsilon) = \frac{1}{\tanh^2(x)} - \frac{1}{m} \cdot \frac{\omega_0}{\sqrt{\omega_0^2 - 1}} \cdot \frac{1}{\tanh(x)} = \frac{1}{\tanh(x)} \left(\frac{1}{\tanh(x)} - \frac{1}{m} \cdot \frac{\omega_0}{\sqrt{\omega_0^2 - 1}} \right),$$

where we set $x := m \ \mathrm{arccosh}(\omega_0)$. Thus, $\frac{1}{m} \cdot \frac{\omega_0}{\sqrt{\omega_0^2 - 1}} = \frac{1}{x} \cdot \frac{\omega_0}{\sqrt{\omega_0^2 - 1}} \cdot \mathrm{arccosh}(\omega_0)$. Finally we obtain:

$$\alpha_m(\varepsilon) = \frac{1}{\tanh(x)} \left(\frac{1}{\tanh(x)} - \frac{1}{x} \cdot \frac{\omega_0}{\sqrt{\omega_0^2 - 1}} \cdot \mathrm{arccosh}(\omega_0) \right).$$

From Lemma 6.1, we deduce that $\frac{1}{x} \cdot \frac{\omega_0}{\sqrt{\omega_0^2 - 1}} \cdot \mathrm{arccosh}(\omega_0) > \frac{1}{x}$. Therefore,

$$\alpha_m(\varepsilon) < \frac{1}{\tanh(x)} \left(\frac{1}{\tanh(x)} - \frac{1}{x} \right).$$

Next, since $x < \sqrt{2\varepsilon}$ by Lemma 6.2, and since the function $x \mapsto \frac{1}{\tanh(x)} \left(\frac{1}{\tanh(x)} - \frac{1}{x} \right)$ is increasing by Lemma 6.3, we conclude that:

$$\alpha_m(\varepsilon) < \frac{1}{\tanh(\sqrt{2\varepsilon})} \left(\frac{1}{\tanh(\sqrt{2\varepsilon})} - \frac{1}{\sqrt{2\varepsilon}} \right)$$

This completes the proof of the first part of the theorem.

We now prove $lim_{m \to \infty} \alpha_m(\varepsilon) = \frac{1}{\tanh(\sqrt{2\varepsilon})} \left(\frac{1}{\tanh(\sqrt{2\varepsilon})} - \frac{1}{\sqrt{2\varepsilon}} \right)$. We recall the definition of α_m:

$$\alpha_m(\varepsilon) = \frac{1}{\tanh^2(m \ \mathrm{arccosh}(\omega_0))} - \frac{1}{m} \cdot \frac{\omega_0}{\sqrt{\omega_0^2 - 1}} \cdot \frac{1}{\tanh(m \ \mathrm{arccosh}(\omega_0))},$$

with $\omega_0 = 1 + \frac{\varepsilon}{m^2}$. As $m \to \infty$, we expand $\mathrm{arccosh}(\omega_0)$ using $\mathrm{arccosh}(1 + \delta) = \sqrt{2\delta} \left(1 + \mathcal{O}(\delta)\right)$, as $\delta \to 0^+$. Applying this with $\delta = \frac{\varepsilon}{m^2}$, we get

$$m \ \mathrm{arccosh} \left(1 + \frac{\varepsilon}{m^2} \right) = \sqrt{2\varepsilon} \left(1 + \mathcal{O}(\frac{\varepsilon}{m}) \right).$$

Hence, as $m \to \infty$,

$$\tanh(m \ \mathrm{arccosh}(\omega_0)) \to \tanh(\sqrt{2\varepsilon}) \quad \text{and} \quad \frac{1}{m\sqrt{\omega_0^2 - 1}} = \frac{1}{m\sqrt{\frac{2\varepsilon}{m^2} + \frac{\varepsilon^2}{m^4}}} \longrightarrow \frac{1}{\sqrt{2\varepsilon}}.$$

Taking the limit in the original expression, we conclude:

$$\lim_{m \to \infty} \alpha_m(\varepsilon) = \frac{1}{\tanh^2(\sqrt{2\varepsilon})} - \frac{1}{\sqrt{2\varepsilon}\tanh(\sqrt{2\varepsilon})} = \frac{1}{\tanh(\sqrt{2\varepsilon})} \left(\frac{1}{\tanh(\sqrt{2\varepsilon})} - \frac{1}{\sqrt{2\varepsilon}} \right).$$

This completes the proof of the second inequality in (4.4).

□

References

[1] A. Abdulle. Fourth order Chebyshev methods with recurrence relation. *SIAM J. Sci. Comput.*, 23(6):2041–2054, 2002.

[2] A. Abdulle. Explicit stabilized runge–kutta methods. In *Encyclopedia of Applied and Computational Mathematics*, pages 460–468. Springer Berlin Heidelberg, 2015.

[3] A. Abdulle, M. J. Grote, and G. Rosilho de Souza. Explicit stabilized multirate method for stiff differential equations. *Math. Comp.*, 91(338):2681–2714, 2022.

[4] A. Abdulle and A. A. Medovikov. Second order Chebyshev methods based on orthogonal polynomials. *Numer. Math.*, 90(1):1–18, 2001.

[5] A. Abdulle and G. Rosilho de Souza. Instabilities and order reduction phenomenon of an interpolation based multirate Runge-Kutta-Chebyshev method, 2020. arXiv:2003.03154 (47 pages).

[6] A. Abdulle and G. Vilmart. PIROCK: a swiss-knife partitioned implicit-explicit orthogonal Runge-Kutta Chebyshev integrator for stiff diffusion-advection-reaction problems with or without noise. *J. Comput. Phys.*, 242:869–888, 2013.

[7] I. Almuslimani. A fully adaptive explicit stabilized integrator for advection-diffusion-reaction problems. *BIT*, 63:3, 2023.

[8] M. J. Grote, S. Michel, and S. A. Sauter. Stabilized leapfrog based local time-stepping method for the wave equation. *Math. Comp.*, 90(332):2603–2643, 2021.

[9] K. Gustafsson. Control-theoretic techniques for stepsize selection in implicit runge-kutta methods. *ACM Trans. Math. Softw.*, 20:496–517, 1994.

[10] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems*, volume 14 of *Springer Series in Comput. Math.* Springer, 2nd edition, 1996.

[11] V. Lebedev. Explicit difference schemes with time-variable steps for solving stiff systems of equations. *Russ. J. Numer. Anal. Math. Model.*, 4(2):111–135, 1989.

[12] V. Lebedev and A. Medovikov. An explicit method of the second order of accuracy for solving stiff systems of ordinary differential equations. *Izv. Vyssh. Uchebn. Zaved. Mat. (in Russian), translation in Russian Math. (Iz. VUZ)*, 42(9):52–60, 1995.

[13] G. Rosilho De Souza. *Numerical methods for deterministic and stochastic differential equations with multiple scales and high contrasts.* PhD thesis, EPFL, Lausanne, 2020.

[14] G. Rosilho de Souza, M. J. Grote, S. Pezzuto, and R. Krause. Explicit stabilized multirate methods for the monodomain model in cardiac electrophysiology. *ESAIM Math. Model. Numer. Anal.*, 58(6):2225–2254, 2024.

[15] B. P. Sommeijer, L. F. Shampine, and J. G. Verwer. RKC: an explicit solver for parabolic PDEs. *J. Comput. Appl. Math.*, 88(2):315–326, 1998.

[16] X. Tang and H. J. A class of flexible and efficient partitioned Runge-Kutta-Chebyshev methods for some time-dependent partial differential equations, 2025. arXiv:2509.14847 (25 pages).

[17] P. J. Van der Houwen and B. P. Sommeijer. On the internal stability of explicit, m-stage runge–kutta methods for large m-values. *Zeitschrift für Angewandte Mathematik und Mechanik (Z. Angew. Math. Mech.)*, 60(10):479–485, 1980.

[18] 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.

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

[20] J. G. Verwer, W. H. Hundsdorfer, and B. P. Sommeijer. Convergence properties of the Runge-Kutta-Chebyshev method. *Numer. Math.*, 57(2):157–178, 1990.

[21] Q. M. Wargnier, G. Vilmart, J. Martínez-Sykora, V. H. Hansteen, and B. De Pontieu. Time-adaptive PIROCK method with error control for multi-fluid and single-fluid MHD systems. *A&A*, 695:A262, 2025.

[22] H. A. Watts. Step size control in ordinary differential equation solvers. *Transactions of the Society for Computer Simulation*, 1:15–25, 1984.

[23] C. J. Zbinden. Partitioned Runge-Kutta-Chebyshev methods for diffusion-advection-reaction problems. *SIAM J. Sci. Comput.*, 33(4):1707–1725, 2011.