

Computational Science Laboratory
Technical Report CSL-TR-21 -4
December 22, 2021

Arash Sarshar, Steven Roberts and Adrian Sandu

“Design of High-Order Decoupled Multirate GARK Schemes”

Computational Science Laboratory
“Compute the Future!”

Computer Science Department
Virginia Polytechnic Institute and State University
Blacksburg, VA 24060
Phone: (540)-231-2193
Fax: (540)-231-6075
Email: sarshar@vt.edu
Web: <http://csl.cs.vt.edu>



DESIGN OF HIGH-ORDER DECOUPLED MULTIRATE GARK SCHEMES*

ARASH SARSHAR[†], STEVEN ROBERTS[†], AND ADRIAN SANDU[†]

Abstract. Multirate time integration methods apply different step sizes to resolve different components of the system based on the local activity levels. This local selection of step sizes allows increased computational efficiency while achieving the desired solution accuracy. While the multirate idea is elegant and has been around for decades, multirate methods are not yet widely used in applications. This is due, in part, to the difficulties raised by the construction of high order multirate schemes.

Seeking to overcome these challenges, this work focuses on the design of practical high-order multirate methods using the theoretical framework of generalized additive Runge–Kutta (MrGARK) methods [19], which provides the generic order conditions and the linear and nonlinear stability analyses. A set of design criteria for practical multirate methods is defined herein: method coefficients should be generic in the step size ratio, but should not depend strongly on this ratio; unnecessary coupling between the fast and the slow components should be avoided; and the step size controllers should adjust both the micro- and the macro-steps. Using these criteria, we develop MrGARK schemes of up to order four that are explicit-explicit (both the fast and slow component are treated explicitly), implicit-explicit (implicit in the fast component and explicit in the slow one), and explicit-implicit (explicit in the fast component and implicit in the slow one). Numerical experiments illustrate the performance of these new schemes.

Key words. Multirate integration, generalized additive Runge–Kutta schemes

AMS subject classifications. 65L05, 65L06, 65L07, 65L020

*

Funding: The work of A. Sandu has been supported in part by NSF through awards NSF OCI-8670904397, NSF CCF-0916493, NSF DMS-0915047, NSF CMMI-1130667, NSF CCF-1218454, AFOSR FA9550-12-1-0293-DEF, AFOSR 12-2640-06, and by the Computational Science Laboratory at Virginia Tech.

[†] Virginia Polytechnic Institute and State University, Computational Science Laboratory, Department of Computer Science, 2202 Kraft Drive, Blacksburg, VA 24060, USA (sarshar@vt.edu, steven94@vt.edu, sandu@cs.vt.edu)

1. Introduction. Many applications in science and engineering require the simulation of dynamical systems where different components evolve at different characteristic time scales. Clearly, these systems challenge traditional time discretizations that use a single time step for the entire system: either the fast components are resolved inaccurately, or the slow components are resolved with more accuracy than required, therefore increasing computational costs. Multirate methods exploit the partitioning of a system into components with different time scales, and use small step sizes to discretize the fast components, and large step sizes to discretize the slow components.

A first approach to constructing multirate methods is to employ traditional integrators with different time steps for different components, and to carefully orchestrate the coupling between these components. Early efforts to develop multirate Runge-Kutta methods are due to Rice [29] and Andrus [3, 4]. In the first discussion of “multirate methods” Gear and Wells [14] propose pairing various linear multistep methods. This fundamental contribution already points to a number of challenges facing multirate methods such as coupling, automatic step size selection, and efficiency of the overall computational process. Other work to construct multirate linear multistep schemes includes [22]. Günther et al. [17, 18] developed multirate methods for partitioned Runge-Kutta schemes, as well as Rosenbrock-W methods [16] of order three that are well-suited for treatment of systems with both stiff and non-stiff variables. Similarly, Kværnø and Rentrop [24, 25] constructed explicit multirate Runge-Kutta methods of order three. Bartel et al. [5] propose one-step methods where internal stages are used to provide the coupling between the fast and slow components. Constantinescu and Sandu developed strong stability preserving (SSP) multirate methods of Runge-Kutta [7] and linear multistep [30] type that are suited for solving hyperbolic partial differential equations (PDEs).

Another approach, tracing back to Engstler et al. [12], derives multirate methods using Richardson extrapolation. Here, the solution is recursively improved on partitions of the system until required tolerances are satisfied. An important advantage of such schemes is the naturally available dynamic partitioning of the system. Constantinescu and Sandu [8, 9, 31] considered explicit and implicit base methods for multirate extrapolation methods and study their stability properties. Other multirate approaches include Galerkin [27], and combined multiscale [11] methodologies.

Günther and Sandu [19] built a class of multirate methods in based on the General Additive Runge-Kutta framework (GARK) [32]. Bremicker-Trübelhorn and Ortleb [6] developed third order multirate GARK (MrGARK) methods for fluid-structure interaction, and allowed for non-uniform fast steps in the order conditions of the methods.

This study develops a systematic design approach for constructing multirate methods in the MrGARK framework of Günther and Sandu [19, 32]. Several high-order schemes are constructed that combine implicit and explicit components for the fast and slow subsystems.

The paper is organized as follows. [Section 2](#) reviews the GARK framework and multirate GARK family. We study order conditions for high order MrGARK methods in [section 3](#), and their scalar linear stability in [section 4](#). [Section 5](#) provides insight into fast-slow coupling requirements. [Section 6](#) discusses the design criteria for practical methods, and [section 7](#) studies error estimation and the adaptivity of both micro- and macro-steps. Newly constructed methods are listed in [section 8](#), and method coefficients are detailed in [Appendix A](#). Numerical tests are performed on different test problems and the results are reported in [section 9](#).

2. Multirate generalized additive Runge–Kutta schemes (MrGARK).

In this section we review some background on MrGARK methods.

2.1. GARK methods. The generalized additive Runge–Kutta (GARK) methods introduced in [15, 32] allow derivation of advanced multi-methods for solving *additively* partitioned systems of ordinary differential equations:

$$(1) \quad y' = f(y) = \sum_{m=1}^N f^{\{m\}}(y), \quad y(t_0) = y_0,$$

where the right-hand side function $f : \mathbb{R}^d \rightarrow \mathbb{R}^d$ is split into N different partitions based on properties such as stiffness, non-linearity, dynamical behavior, and evaluation cost. We note that additive partitioning includes the case of *component partitioning* in which the state vector is split into a number subsets.

A GARK method advances the numerical solution as follows [32]:

$$(2a) \quad Y_i^{\{q\}} = y_n + h \sum_{m=1}^N \sum_{j=1}^{s^{\{m\}}} a_{i,j}^{\{q,m\}} f^{\{m\}} \left(Y_j^{\{m\}} \right) \quad \begin{array}{l} q=1,\dots,N, \\ i=1,\dots,s^{\{q\}}, \end{array}$$

$$(2b) \quad y_{n+1} = y_n + h \sum_{q=1}^N \sum_{i=1}^{s^{\{q\}}} b_i^{\{q\}} f^{\{q\}} \left(Y_i^{\{q\}} \right).$$

2.2. Multirate GARK methods. In the case of a two-way partitioned system

(1) with slow component $\{s\}$, and fast component $\{f\}$ we have:

$$(3) \quad y' = f(y) = f^{\{s\}}(y) + f^{\{f\}}(y), \quad y(t_0) = y_0.$$

A multirate GARK method [32] integrates the slow component with a Runge–Kutta method $(A^{\{s,s\}}, b^{\{s\}})$ and a large step size H , and the fast component with another Runge–Kutta method $(A^{\{f,f\}}, b^{\{f\}})$ and a small step size $h = H/M$. Here $M \geq 1$ represents the (integer) number of fast steps that are executed for each of the slow steps. MrGARK methods are formally derived in [19]. One step of the method utilizes $s^{\{s\}}$ slow stages, denoted by $Y_i^{\{s\}}$, and $M s^{\{f\}}$ fast stages, denoted by $Y_i^{\{f,\lambda\}}$:

$$(4a) \quad \begin{aligned} Y_i^{\{s\}} = y_n + H \sum_{j=1}^{s^{\{s\}}} a_{i,j}^{\{s,s\}} f^{\{s\}} \left(Y_j^{\{s\}} \right) \\ + h \sum_{\lambda=1}^M \sum_{j=1}^{s^{\{f\}}} a_{i,j}^{\{s,f,\lambda\}} f^{\{f\}} \left(Y_j^{\{f,\lambda\}} \right), \quad i = 1, \dots, s^{\{s\}}, \end{aligned}$$

$$(4b) \quad \begin{aligned} Y_i^{\{f,\lambda\}} = \tilde{y}_{n+(\lambda-1)/M} + H \sum_{j=1}^{s^{\{s\}}} a_{i,j}^{\{f,s,\lambda\}} f^{\{s\}} \left(Y_j^{\{s\}} \right) \\ + h \sum_{j=1}^{s^{\{f\}}} a_{i,j}^{\{f,f\}} f^{\{f\}} \left(Y_j^{\{f,\lambda\}} \right), \quad \begin{array}{l} i=1,\dots,s^{\{f\}} \\ \lambda=1,\dots,M, \end{array} \end{aligned}$$

$$(4c) \quad \tilde{y}_{n+\lambda/M} = \tilde{y}_{n+(\lambda-1)/M} + h \sum_{i=1}^{s^{\{f\}}} b_i^{\{f\}} f^{\{f\}} \left(Y_i^{\{f,\lambda\}} \right).$$

The full step is then formed with:

$$(4d) \quad y_{n+1} = \tilde{y}_{n+M/M} + H \sum_{i=1}^{s^{\{s\}}} b_i^{\{s\}} f^{\{s\}}(Y_i^{\{s\}}).$$

Let the coupling between the two methods be described by $A^{\{s,f,\lambda\}}$ and $A^{\{f,s,\lambda\}}$ for $\lambda \in \{1, 2, \dots, M\}$. The GARK Butcher tableau for the method (4) is [19]:

$$(5) \quad \begin{array}{c|c} \mathbf{A}^{\{f,f\}} & \mathbf{A}^{\{f,s\}} \\ \mathbf{A}^{\{s,f\}} & \mathbf{A}^{\{s,s\}} \\ \mathbf{b}^{\{f\}T} & \mathbf{b}^{\{s\}T} \end{array} := \begin{array}{c|ccc|c} \frac{1}{M}A^{\{f,f\}} & 0 & \dots & 0 & A^{\{f,s,1\}} \\ \frac{1}{M}\mathbb{1}b^{\{f\}T} & \frac{1}{M}A^{\{f,f\}} & \dots & 0 & A^{\{f,s,2\}} \\ \vdots & & \ddots & & \vdots \\ \frac{1}{M}\mathbb{1}b^{\{f\}T} & \frac{1}{M}\mathbb{1}b^{\{f\}T} & \dots & \frac{1}{M}A^{\{f,f\}} & A^{\{f,s,M\}} \\ \hline \frac{1}{M}A^{\{s,f,1\}} & \frac{1}{M}A^{\{s,f,2\}} & \dots & \frac{1}{M}A^{\{s,f,M\}} & A^{\{s,s\}} \\ \frac{1}{M}b^{\{f\}T} & \frac{1}{M}b^{\{f\}T} & \dots & \frac{1}{M}b^{\{f\}T} & b^{\{s\}T} \end{array}.$$

REMARK 2.1 (Slow and fast stage numbers). *The structure of the Butcher tableau implies that the fast stage ℓ of the fast micro-step λ corresponds to row $(\lambda - 1)s^{\{f\}} + \ell$ in (5), and the slow stage j corresponds to row $Ms^{\{f\}} + j$ in (5).*

DEFINITION 2.1 (Telescopic MrGARK schemes). *A telescopic MrGARK method [19] uses the same base scheme for the slow and fast partitions:*

$$(6) \quad A^{\{f,f\}} = A^{\{s,s\}} = A, \quad b^{\{f\}} = b^{\{s\}} = b.$$

In this paper we will focus on schemes with telescopic property since it allows a simple extension of the MrGARK method to an arbitrary number of partitions (time scales) using successively larger time steps, each an integer multiple of the previous one.

3. Order conditions for multirate GARK methods. We consider the following *internal consistency* conditions [19, 32] to ensure that fast and slow right-hand side evaluations are performed at the same points in time:

$$(7) \quad \mathbf{A}^{\{s,f\}} \mathbb{1}^{\{s\}} = \mathbf{A}^{\{s,s\}} \mathbb{1}^{\{s\}} := \mathbf{c}^{\{s\}}, \quad \mathbf{A}^{\{f,s\}} \mathbb{1}^{\{f\}} = \mathbf{A}^{\{f,f\}} \mathbb{1}^{\{f\}} := \mathbf{c}^{\{f\}},$$

where $\mathbb{1}^{\{\sigma\}} := [1, 1, \dots, 1]^T \in \mathbb{R}^{s^{\{\sigma\}}}$.

Assume that the internal consistency conditions (7) hold, and further assume that each individual method $(A^{\{s,s\}}, b^{\{s\}})$ and $(A^{\{f,f\}}, b^{\{f\}})$ has at least order 4. Then the MrGARK scheme (4)–(5) has order four if and only if the following coupling conditions hold [19]:

$$(8a) \quad \frac{1}{6} = \mathbf{b}^{\{f\}T} \mathbf{A}^{\{f,s\}} \mathbf{c}^{\{s\}}, \quad (\text{order } 3)$$

$$(8b) \quad \frac{1}{6} = \mathbf{b}^{\{s\}T} \mathbf{A}^{\{s,f\}} \mathbf{c}^{\{f\}}, \quad (\text{order } 3)$$

$$(8c) \quad \frac{1}{8} = \mathbf{b}^{\{f\}T} \left(\mathbf{c}^{\{f\}} \times \mathbf{A}^{\{f,s\}} \mathbf{c}^{\{s\}} \right), \quad (\text{order } 4)$$

$$(8d) \quad \frac{1}{8} = \mathbf{b}^{\{s\}T} \left(\mathbf{c}^{\{s\}} \times \mathbf{A}^{\{s,f\}} \mathbf{c}^{\{f\}} \right), \quad (\text{order } 4)$$

$$\begin{aligned}
(8e) \quad & \frac{1}{12} = \mathbf{b}^{\{f\}T} \mathbf{A}^{\{f,s\}} \left(\mathbf{c}^{\{s\}} \times \mathbf{c}^{\{s\}} \right), & (\text{order } 4) \\
(8f) \quad & \frac{1}{12} = \mathbf{b}^{\{s\}T} \mathbf{A}^{\{s,f\}} \left(\mathbf{c}^{\{f\}} \times \mathbf{c}^{\{f\}} \right), & (\text{order } 4) \\
(8g) \quad & \frac{1}{24} = \mathbf{b}^{\{s\}T} \mathbf{A}^{\{s,s\}} \mathbf{A}^{\{s,f\}} \mathbf{c}^{\{f\}}, & (\text{order } 4) \\
(8h) \quad & \frac{1}{24} = \mathbf{b}^{\{s\}T} \mathbf{A}^{\{s,f\}} \mathbf{A}^{\{f,s\}} \mathbf{c}^{\{s\}}, & (\text{order } 4) \\
(8i) \quad & \frac{1}{24} = \mathbf{b}^{\{s\}T} \mathbf{A}^{\{s,f\}} \mathbf{A}^{\{f,f\}} \mathbf{c}^{\{f\}}, & (\text{order } 4) \\
(8j) \quad & \frac{1}{24} = \mathbf{b}^{\{f\}T} \mathbf{A}^{\{f,f\}} \mathbf{A}^{\{f,s\}} \mathbf{c}^{\{s\}}, & (\text{order } 4) \\
(8k) \quad & \frac{1}{24} = \mathbf{b}^{\{f\}T} \mathbf{A}^{\{f,s\}} \mathbf{A}^{\{s,s\}} \mathbf{c}^{\{s\}}, & (\text{order } 4) \\
(8l) \quad & \frac{1}{24} = \mathbf{b}^{\{f\}T} \mathbf{A}^{\{f,s\}} \mathbf{A}^{\{s,f\}} \mathbf{c}^{\{f\}}. & (\text{order } 4)
\end{aligned}$$

Here “ \times ” denotes component-wise multiplication of two vectors.

Without imposing any special structure on coupling matrices, we can rewrite (8) using the block structure shown in (5). Considering the following intermediate simplifications:

$$\begin{aligned}
(9a) \quad & \mathbf{c}^{\{f\}} = \frac{1}{M} \left[c^{\{f\}} + (\lambda - 1) \mathbb{1} \right]_{\lambda=1,\dots,M}, \\
(9b) \quad & \mathbf{A}^{\{f,f\}} \mathbf{c}^{\{f\}} = \left[\frac{1}{M^2} \left(\frac{(\lambda - 1)^2}{2} \mathbb{1} + (\lambda - 1) c^{\{f\}} + A^{\{f,f\}} c^{\{f\}} \right) \right]_{\lambda=1,\dots,M}, \\
(9c) \quad & \mathbf{A}^{\{s,f\}} \mathbf{c}^{\{f\}} = \frac{1}{M^2} \sum_{\lambda=1}^M A^{\{s,f,\lambda\}} \left((\lambda - 1) \mathbb{1} + c^{\{f\}} \right), \\
(9d) \quad & \mathbf{A}^{\{f,f\}} \mathbf{A}^{\{f,s\}} \mathbf{c}^{\{s\}} = \frac{1}{M} \left[\sum_{k=1}^{\lambda-1} \mathbb{1} b^{\{f\}T} A^{\{f,s,k\}} c^{\{s\}} + A^{\{f,f\}} A^{\{f,s,\lambda\}} c^{\{s\}} \right]_{\lambda=1,\dots,M}.
\end{aligned}$$

The order conditions (8) can be written in terms of base methods and coupling coefficients as follows:

$$\begin{aligned}
(10a) \quad & \frac{M}{6} = \sum_{\lambda=1}^M b^{\{f\}T} A^{\{f,s,\lambda\}} c^{\{s\}}, & (\text{order } 3) \\
(10b) \quad & \frac{M^2}{6} = \sum_{\lambda=1}^M b^{\{s\}T} A^{\{s,f,\lambda\}} \left((\lambda - 1) \mathbb{1} + c^{\{f\}} \right), & (\text{order } 3) \\
(10c) \quad & \frac{M^2}{8} = \sum_{\lambda=1}^M (\lambda - 1) b^{\{f\}T} A^{\{f,s,\lambda\}} c^{\{s\}} & (\text{order } 4) \\
& + \sum_{\lambda=1}^M b^{\{f\}T} \left(c^{\{f\}} \times A^{\{f,s,\lambda\}} c^{\{s\}} \right), \\
(10d) \quad & \frac{M^2}{8} = b^{\{s\}T} \sum_{\lambda=1}^M \left(c^{\{s\}} \times \left(A^{\{s,f,\lambda\}} \left((\lambda - 1) \mathbb{1} + c^{\{f\}} \right) \right) \right), & (\text{order } 4) \\
(10e) \quad & \frac{M}{12} = \sum_{\lambda=1}^M b^{\{f\}T} A^{\{f,s,\lambda\}} c^{\{s\} \times 2}, & (\text{order } 4)
\end{aligned}$$

$$(10f) \quad \frac{M^3}{12} = \sum_{\lambda=1}^M b^{\{s\}T} A^{\{s,f,\lambda\}} c^{\{f\} \times 2} + \sum_{\lambda=1}^M (\lambda-1)^2 b^{\{s\}T} A^{\{s,f,\lambda\}} \mathbb{1} \\ + 2 \sum_{\lambda=1}^M (\lambda-1) b^{\{s\}T} A^{\{s,f,\lambda\}} c^{\{f\}}, \quad (\text{order } 4)$$

$$(10g) \quad \frac{M^2}{24} = \sum_{\lambda=1}^M b^{\{s\}T} A^{\{s,s\}} A^{\{s,f,\lambda\}} \left((\lambda-1) \mathbb{1} + c^{\{f\}} \right), \quad (\text{order } 4)$$

$$(10h) \quad \frac{M}{24} = \sum_{\lambda=1}^M b^{\{s\}T} A^{\{s,f,\lambda\}} A^{\{f,s,\lambda\}} c^{\{s\}}, \quad (\text{order } 4)$$

$$(10i) \quad \frac{M^3}{24} = \sum_{\lambda=1}^M \frac{(\lambda-1)^2}{2} b^{\{s\}T} A^{\{s,f,\lambda\}} \mathbb{1} \\ + \sum_{\lambda=1}^M (\lambda-1) b^{\{s\}T} A^{\{s,f,\lambda\}} c^{\{f\}} + \sum_{\lambda=1}^M b^{\{s\}T} A^{\{s,f,\lambda\}} A^{\{f,f\}} c^{\{f\}}, \quad (\text{order } 4)$$

$$(10j) \quad \frac{M^2}{24} = \sum_{\lambda=1}^M \sum_{k=1}^{\lambda-1} b^{\{f\}T} A^{\{f,s,k\}} c^{\{s\}} + \sum_{\lambda=1}^M b^{\{f\}T} A^{\{f,f\}} A^{\{f,s,\lambda\}} c^{\{s\}}, \quad (\text{order } 4)$$

$$(10k) \quad \frac{M}{24} = \sum_{\lambda=1}^M b^{\{f\}T} A^{\{f,s,\lambda\}} A^{\{s,s\}} c^{\{s\}}, \quad (\text{order } 4)$$

$$(10l) \quad \frac{M^3}{24} = \sum_{\lambda=1}^M \sum_{k=1}^M b^{\{f\}T} A^{\{f,s,\lambda\}} A^{\{s,f,k\}} \left((k-1) \mathbb{1} + c^{\{f\}} \right). \quad (\text{order } 4)$$

REMARK 3.1 (Design process). *A practical design procedure for MrGARK schemes is to first select the base slow and fast methods of desired order, and then solve the order conditions (10) for the coupling coefficients $\mathbf{A}^{\{f,s\}}$ and $\mathbf{A}^{\{s,f\}}$.*

4. Linear stability analysis. Following [19, 32] we consider the following scalar model problem:

$$(11) \quad y' = \lambda^{\{f\}} y + \lambda^{\{s\}} y.$$

where the ratio of the fast to the slow variable is M , the step size ratio. Denote $z^{\{f\}} = H \lambda^{\{f\}}$, $z^{\{s\}} = H \lambda^{\{s\}}$, and

$$(12) \quad s = M s^{\{f\}} + s^{\{f\}}, \quad Z = \begin{bmatrix} z^{\{f\}} \mathbf{I}_{M s^{\{f\}} \times M s^{\{f\}}} & \mathbf{0} \\ \mathbf{0} & z^{\{s\}} \mathbf{I}_{s^{\{s\}} \times s^{\{s\}}} \end{bmatrix}.$$

It was shown in [19, 32] that application of MrGARK method (5) to (11) leads to the solution

$$y_{n+1} = R(z^{\{f\}}, z^{\{s\}}) y_n,$$

with the stability function:

$$(13) \quad R(z^{\{f\}}, z^{\{s\}}) = 1 + \mathbf{b}_{\text{GARK}}^T \cdot Z \cdot (\mathbf{I}_{s \times s} - \mathbf{A}_{\text{GARK}} \cdot Z)^{-1} \cdot \mathbf{1}_{s \times 1}.$$

In order to visualize the stability region of a method, we choose:

$$(14) \quad z^{\{f\}} = M \rho e^{-i \theta^{\{f\}}}, \quad z^{\{s\}} = \rho e^{-i \theta^{\{s\}}}, \quad \frac{\pi}{2} \leq \theta^{\{f\}}, \theta^{\{s\}} \leq \frac{3\pi}{2},$$

such that the ratio of the variable magnitudes matches the ratio of the step sizes. This reduces (13) to a function of three real variables by assuming the fast eigenvalue is M times larger in magnitude than the slow eigenvalue. The stability region is the volume in the $\{\theta^{\{f\}}, \theta^{\{s\}}, \rho\}$ space where the magnitude of the stability function (13) is less than or equal to one. Stability regions for each method developed here are provided in Appendix A. For some of the methods the region decreases with increasing M . The plots for different values of M correspond to different test problems: when M increases, the scale separation of the test problem also increases (we apply smaller micro-steps to faster problems). The increasing stiffness of the fast component restricts the macro-step, a phenomenon known as stiffness leakage. Coupling coefficients whose magnitude increases rapidly with M can lead to a degradation of stability.

5. Decoupled MrGARK methods. We now discuss in detail the structure of the coupling coefficient matrices, and how this structure defines the way the fast and slow stage computations (4) are carried out, and therefore determines the practicality of the MrGARK method. In order to construct practical MrGARK methods we need to avoid complex couplings between multiple fast and slow stages. To this end we define *decoupled MrGARK methods*.

DEFINITION 5.1 (Decoupled MrGARK methods). *An MrGARK method is decoupled if the computation of its stages proceeds in sequence, such that each slow stage uses only information from other slow stages and the already computed fast stages, and vice-versa. There is no coupling that requires fast and slow stages to be solved together. Any form of implicitness is entirely within the fast or within the slow system.*

5.1. Structure of the slow-fast coupling (including fast information into the slow stage calculations). We first introduce a notation for the order of computation of the slow stages with respect to the fast stages. Consider the j -th slow stage (4a) at abscissa $c_j^{\{s\}}$, i.e., the row $M s^{\{f\}} + j$ in the Butcher tableau (5). We denote its order of computation by (L_j, I_j) , i.e., the j -th slow stage is computed immediately after the I_j -th stage of the L_j -th micro-step is computed. This means that the first $L_j - 1$ micro-steps have been completed, and the L_j -th micro-step has partially progressed to compute stage I_j , when the j -th slow stage is evaluated.

When the slow stage $c_j^{\{s\}}$ is evaluated after the last stage of micro-step $\lambda - 1$, but before the first stage of the λ -th micro-step, we have $(L_j, I_j) = (\lambda - 1, s^{\{f\}})$. Equivalently, this situation can be represented by $(L_j, I_j) = (\lambda, 0)$.

In order to construct practical MrGARK methods we require that the evaluation of the j -th slow stage depends only on those fast stages that have been completed; in the GARK Butcher tableau (5), the j -th slow stage depends only on the rows $1 : (L_j - 1)s^{\{f\}} + I_j$.

The j -th row of the slow-fast coupling matrix:

$$(15a) \quad \mathbf{A}^{\{s,f\}} = \frac{1}{M} [A^{\{s,f,1\}} \quad \dots \quad A^{\{s,f,\lambda\}} \quad \dots \quad A^{\{s,f,M\}}] \in \mathbb{R}^{s^{\{s\}} \times M s^{\{f\}}}$$

contains the coefficients that bring fast information into the computation of slow stage j . A decoupled MrGARK scheme enjoys the following properties:

- The coupling matrices $A^{\{s,f,\lambda\}}$ with the completed microsteps $\lambda = 1 : L_j - 1$ can have full rows:

$$a_{j,k}^{\{s,f,\lambda\}} \neq 0, \quad \text{for } \lambda = 1 : L_j - 1.$$

- Rows of the coupling matrix $A^{\{s,f,L_j\}}$ can have non-zero entries only in the first I_j positions:

$$\begin{aligned} a_{j,k}^{\{s,f,L_j\}} &\neq 0, & k = 1, \dots, I_j, \\ a_{j,k}^{\{s,f,L_j\}} &= 0, & k = I_j + 1, \dots, s^{\{f\}}. \end{aligned}$$

- The coupling matrices $A^{\{s,f,\lambda\}}$ with the future microsteps $\lambda = L_j + 1 : M$ are zero:

$$a_{j,k}^{\{s,f,\lambda\}} = 0, \quad \text{for } \lambda = L_j + 1 : M.$$

Consequently, for decoupled MrGARK schemes, the slow-fast coupling matrix is lower block triangular:

$$(15b) \quad (\mathbf{A}^{\{s,f\}})_{j,k} = 0, \quad \text{for } 1 \leq j \leq s^{\{s\}}, \quad (L_j - 1)s^{\{f\}} + I_j + 1 \leq k \leq Ms^{\{f\}},$$

where we used the fact that the fast stage I_j of micro-step L_j has GARK stage number $(L_j - 1)s^{\{f\}} + I_j$ in the Butcher tableau (5).

5.2. Structure of the fast-slow coupling (including slow information into the fast stage calculations). We next introduce a notation for the order of computation of the fast stages with respect to the slow stages. We denote by $J_{L,i}$ the index of the last slow stage computed before starting the fast stage i of micro-step L . Specifically, the fast stage i at micro-step L is computed after the slow stage $J_{L,i}$, but before the slow stage $J_{L,i} + 1$. For a decoupled MrGARK this stage can only use information from slow stages 1 to $J_{L,i}$. Consequently, the i -th row of the coupling matrix $A^{\{f,s,\lambda\}}$ can have nonzero entries only in the first $J_{L,i}$ columns:

$$a_{i,j}^{\{f,s,\lambda\}} = 0, \quad \text{for } J_{L,i} + 1 \leq j \leq s^{\{s\}}.$$

The coupling matrix:

$$(16a) \quad \mathbf{A}^{\{f,s\}} = [A^{\{f,s,1\}T} \quad \dots \quad A^{\{f,s,\lambda\}T} \quad \dots \quad A^{\{f,s,M\}T}]^T \in \mathbb{R}^{Ms^{\{f\}} \times s^{\{s\}}}$$

has a block lower triangular structure:

$$(16b) \quad \mathbf{A}_{(L-1)s^{\{f\}}+i,j}^{\{f,s\}} = 0 \quad \text{for } J_{L,i} + 1 \leq j \leq s^{\{s\}},$$

where we used the fact that the fast stage i of micro-step L has GARK stage number $(L - 1)s^{\{f\}} + i$ in the Butcher tableau (5).

5.3. Relation between the coupling matrices. From (15) and (16) we see that the two coupling matrices $\mathbf{A}^{\{f,s\}}$ and $\mathbf{A}^{\{s,f\}}$ have related sparsity structures. For *decoupled MrGARK methods* the sparsity structures are complementary, in the sense that one must have zeros in the entries where the other matrix can have nonzero elements:

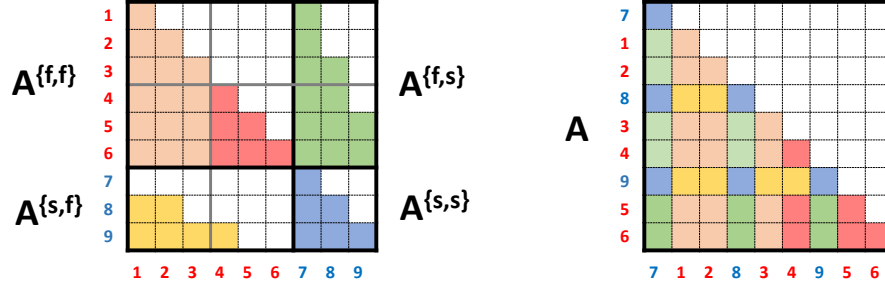
$$(17) \quad \mathbf{A}^{\{s,f\}} \times \mathbf{A}^{\{f,s\}T} = \mathbf{0}_{s^{\{s\}} \times Ms^{\{f\}}},$$

where \times denotes element-by-element multiplication. This is schematically illustrated in [Figure 1a](#).

For *coupled MrGARK methods* the sparsity structures can overlap, in the sense that they both can have non-zeros entries in the same location:

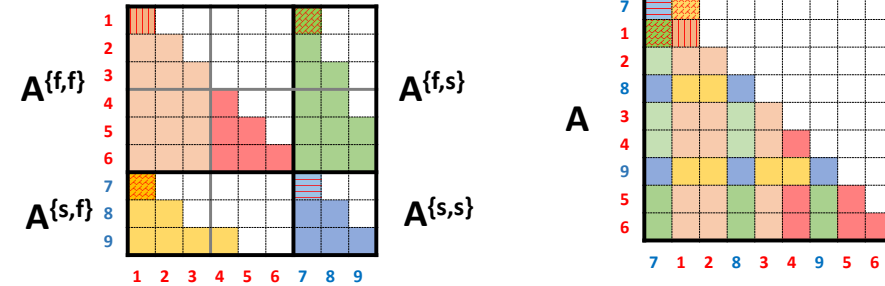
$$(18) \quad \mathbf{A}^{\{s,f\}} \times \mathbf{A}^{\{f,s\}T} \neq \mathbf{0}_{s\{s\} \times M s\{f\}}.$$

This is illustrated in [Figure 1c](#). The overlapping non-zero coupling coefficients, indicated by dashed boxes in [Figure 1c](#), imply that the corresponding fast and slow stages need to be computed together, in a step that involves the entire non-partitioned system. Coupled methods are not pursued further in this paper.



(a) Butcher tableau of a decoupled MrGARK. The coupling matrices have complementary sparsity patterns.

(b) Permuted Butcher tableau of a decoupled MrGARK shows a lower diagonal structure.



(c) Butcher tableau of a coupled MrGARK. The dashed entries of the coupling matrices violate the sparsity complementarity.

(d) Permuted Butcher tableau of a coupled MrGARK. The first slow and the first fast steps, dashed, need to be computed together in a coupled manner.

FIG. 1: Example of decoupled and coupled MrGARK with $M = 2$ and $s = 3$. Blue is the slow method, pink the first fast step, dark pink the second fast step, and green and yellow are the couplings. The permuted versions of the tableaux reflect the sequential order of stage computations. Note the entry above the diagonal for the coupled, permuted tableau due to the non-complementary coupling structure.

5.4. Order of the slow and fast stage evaluation. The sparsity structure of the coupling matrices $\mathbf{A}^{\{s,f\}}$ and $\mathbf{A}^{\{f,s\}}$ determines the order in which the fast and slow stages can be evaluated such as to respect the data dependencies between them. Using the notation defined in [subsection 5.1](#) and [subsection 5.2](#), the general order in which stages are evaluated is follows:

1. Start by evaluating the fast stages i for which $a_{1,i}^{\{s,f\}} \neq 0$; these are the fast stages needed by the computation of the first slow stage $c_1^{\{s\}}$. If the entire first row $a_{1,:}^{\{s,f\}} = 0$ then proceed with evaluating the first slow stage $c_1^{\{s\}}$.
2. After stage I_j of the L_j -th micro-step is computed, the j -th stage $c_j^{\{s\}}$ of the slow method is evaluated.
3. Continue with evaluating stages of the fast method, and stop after stage I_{j+1} of the L_{j+1} -st micro-step to compute stage $c_{j+1}^{\{s\}}$ of the slow method.
4. Continue until all fast and slow stages are evaluated.

REMARK 5.1 (Time ordering). *Assuming that the slow stage abscissae are increasing, and that fast stage abscissae are non-decreasing in time:*

$$(19) \quad c_1^{\{s\}} < \dots < c_{s^{\{s\}}}^{\{s\}}, \quad c_1^{\{f\}} \leq \dots \leq c_{s^{\{f\}}}^{\{f\}},$$

a natural order to evaluate the MrGARK stages follows the time ordering of their abscissae. Note that the j -th slow stage approximates the solution at time $T_{Ms^{\{f\}}+j} = t_n + c_j^{\{s\}} H$, while the j -th fast stage of the L -th microstep approximates the solution at time $T_{(L-1)s^{\{f\}}+j} = t_n + (L-1 + c_j^{\{f\}})(H/M)$. The pairs (L_j, I_j) are chosen such that the stages are evaluated in increasing order of their approximation times:

$$(20a) \quad \frac{L_j - 1 + c_{I_j}^{\{f\}}}{M} \leq c_j^{\{s\}} \leq \frac{L_j - 1 + c_{I_j+1}^{\{f\}}}{M} \quad \text{for } 1 \leq I_j < s^{\{f\}},$$

$$(20b) \quad \frac{L_j + c_{s^{\{f\}}}^{\{f\}}}{M} \leq c_j^{\{s\}} \leq \frac{L_j + 1 + c_1^{\{f\}}}{M} \quad \text{for } I_j = s^{\{f\}},$$

$$\text{or } \frac{L_j - 1 + c_{s^{\{f\}}}^{\{f\}}}{M} \leq c_j^{\{s\}} \leq \frac{L_j + c_1^{\{f\}}}{M} \quad \text{for } I_j = 0.$$

REMARK 5.2 (Simpler time ordering). *It is possible to simplify the time ordering such that the slow stages are evaluated at the end of full micro-steps. The stage $c_j^{\{s\}}$ is evaluated after the end of micro-step $\lambda - 1$, but before micro-step λ , if $(\lambda - 1)/M \leq c_j^{\{s\}} < \lambda/M$, in which case $L_j := \lambda$ and $I_j = 0$. When $c_j^{\{s\}} \geq 1$ we take $L_j = M, I_j = 0$.*

5.5. Reordering the GARK Butcher tableau. In the standard form of the GARK Butcher tableau (5) the first $Ms^{\{f\}}$ stages are for the fast method, and stages $Ms^{\{f\}} + 1$ to $Ms^{\{f\}} + s^{\{s\}}$ are for the slow method. Let IC be the vector of GARK tableau stage indices sorted in the order of computations, as summarized in Figure 2. A renumbering of stages leads to a row and column permutation of the Butcher tableau (5). The reordered Butcher matrix is $\mathbf{A}_{\text{GARK}}(\text{IC}, \text{IC})$. The reordering is illustrated in Figure 1. We distinguish the following cases:

- If the reordered Butcher tableau is strictly lower triangular then the MrGARK method is explicit, and each stage uses only previously computed information.
- If the reordered Butcher tableau is lower triangular, with some non-zero diagonal entries, then the MrGARK method is implicit, but decoupled. The non-zero diagonal entries correspond to implicit fast or slow stages. This case is illustrated in Figure 1b.
- Finally, if the reordered Butcher tableau has block-diagonal entries then the MrGARK method is coupled, in the sense that several fast and slow stages need to be solved together, in a coupled manner. This case is illustrated in Figure 1d.

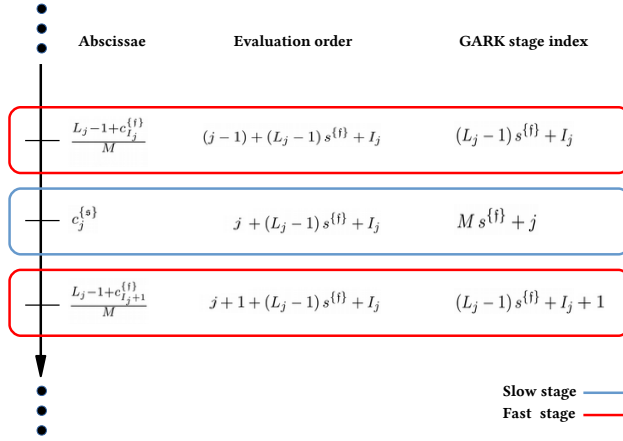


FIG. 2: An example of the order of computing stages in MrGARK with stage sequence $\{(L_j, I_j)\}$.

As an example let us consider the GARK Butcher matrix for the explicit method EX2-EX2 2(1)[A] introduced in Appendix A.1 for $M = 3$:

$$\mathbf{A} = \left(\begin{array}{cccccc|cc}
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 \frac{2}{9} & 0 & 0 & 0 & 0 & 0 & \frac{2}{9} & 0 \\
 \frac{1}{4} & \frac{3}{4} & 0 & 0 & 0 & 0 & \frac{11}{60} & \frac{3}{20} \\
 \frac{1}{4} & \frac{3}{4} & \frac{2}{9} & 0 & 0 & 0 & -\frac{19}{180} & \frac{9}{20} \\
 \frac{1}{4} & \frac{3}{4} & \frac{1}{4} & \frac{3}{4} & 0 & 0 & \frac{31}{60} & \frac{3}{20} \\
 \frac{1}{4} & \frac{3}{4} & \frac{1}{4} & \frac{3}{4} & \frac{2}{9} & 0 & -\frac{79}{180} & \frac{9}{20} \\
 \hline
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 2 & 0 & 0 & 0 & 0 & 0 & \frac{2}{3} & 0
 \end{array} \right), \quad \mathbf{A}^{(ic, ic)} = \left(\begin{array}{cccccccc}
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 \frac{2}{9} & \frac{2}{9} & 0 & 0 & 0 & 0 & 0 & 0 \\
 \frac{2}{3} & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\
 \frac{11}{60} & \frac{1}{4} & \frac{3}{4} & \frac{3}{20} & 0 & 0 & 0 & 0 \\
 -\frac{19}{180} & \frac{1}{4} & \frac{3}{4} & \frac{9}{20} & \frac{2}{9} & 0 & 0 & 0 \\
 \frac{31}{60} & \frac{1}{4} & \frac{3}{4} & \frac{3}{20} & \frac{1}{4} & \frac{3}{4} & 0 & 0 \\
 -\frac{79}{180} & \frac{1}{4} & \frac{3}{4} & \frac{9}{20} & \frac{1}{4} & \frac{3}{4} & \frac{2}{9} & 0
 \end{array} \right),$$

where $ic = [7, 1, 2, 8, 3, 4, 5, 6]$. The permuted Butcher tableau verifies the explicit nature of the method, and also provides a progression of stage computations that is practically easy to implement.

6. Design of practical decoupled MrGARK methods. In this section we discuss several desirable properties of practical MrGARK methods, and the design methodologies to incorporate them in the construction of new high order schemes.

6.1. Design principles. The following set of design principles incorporates properties that are important for ensuring the practicality of MrGARK schemes:

1. Practical MrGARK methods need to be *high order* (have an order of accuracy larger than two), and *generic* in the multirate ratio (have the same order of accuracy for any M .) This typically requires that the coupling coefficients are functions of M . We have addressed these requirements via the order conditions derived in section 3.
2. Methods where the fast and slow base schemes are both either explicit or implicit should be *telescopic* (6) in order to be easily applicable to multi-partitioned systems with multiple time scales. All explicit-explicit methods proposed in this paper are telescopic.

3. In order to maintain computational efficiency, a complex coupling between multiple fast and slow stages needs to be avoided. In this paper, we focus on *decoupled* multirate methods satisfying equation (17), as they are far less computationally demanding than coupled methods. However, the overall stability of the scheme may be affected by decoupling.
4. Methods should be optimized for general-purpose time integration, i.e. have small principal error terms and large stability regions. Note that the principal error for MrGARK methods is a function of M . If not properly controlled, the coupling errors can grow with M , thereby reducing the overall efficiency of the method. All schemes derived herein have coefficients optimized for small principal error and large stability region.
5. Consider the case where one of the base methods $q \in \{f, s\}$ is implicit. A favorable property of the entire MrGARK method is stiff accuracy [19]:

$$(21) \quad \mathbf{e}_{s\{q\}}^T \mathbf{A}^{\{q,f\}} = \mathbf{b}^{\{f\}T}, \quad \mathbf{e}_{s\{q\}}^T \mathbf{A}^{\{q,s\}} = \mathbf{b}^{\{s\}T}.$$

This property implies that the base implicit Runge–Kutta method is stiffly accurate, and that the MrGARK stability function approaches zero as the q -th component of the system becomes infinitely stiff [19]. All schemes derived herein having an implicit base method enjoy the property (21).

6. Practical use of MrGARK methods demands an error control mechanism that is capable of adapting both the step size H and the multirate ratio M . Therefore the error control problem in multirate integration is fundamentally more complex than in traditional, single rate integration. We fully address the error control issue in [section 7](#).
7. It is desirable to derive multirate methods with reduced coupling errors by requiring that both the main and the embedded methods satisfy higher order coupling conditions. This strategy isolates the dominant local truncation errors to the solution of the slow and fast components and greatly simplifies the task of adaptively choosing H and M , as discussed in [section 7](#).

DEFINITION 6.1 (Naturally adaptive schemes). *A MrGARK scheme of order p is naturally adaptive if the fast and slow local truncation error terms (corresponding to N -trees with nodes of the same color) are $\mathcal{O}(h^{p+1})$, and the coupling local truncation error terms (corresponding to N -trees with nodes of different colors) are $\mathcal{O}(h^{p+2})$.*

We construct *naturally adaptive* second and third order schemes in [Appendix A](#).

8. Methods of type S are optimized for simplicity and stability. First, by design, one seeks to maximize the sparsity of the coupling coefficient matrices, such as to simplify the implementation and decrease data dependencies. Next, the coupling coefficients can potentially become large in magnitude when M increases, and this can lead to large cancellation errors in the context of finite precision arithmetic, as well as to a degradation of numerical stability. It is desirable that the coupling coefficients are optimized such that their magnitude remains bounded for large values M .

6.2. Design process. The order conditions (8) and the additional constraints associated with the design principles of [subsection 6.1](#) lead to large, nonlinear systems of equations that need to be solved for the method coefficients. The resulting coupling coefficients $\mathbf{A}^{\{s,f\}}$ and $\mathbf{A}^{\{f,s\}}$ are *functions* of the micro-step number λ and the multirate step size ratio M . Our proposed approach for designing MrGARK methods of order p is a three-step process, as follows.

1. The first step is to construct optimized base slow $(A^{\{s,s\}}, b^{\{s\}})$ and fast $(A^{\{f,f\}}, b^{\{f\}})$ schemes of order p . Implicit base methods are selected to be stiffly-accurate SDIRK schemes. Explicit base methods are chosen to have small principal error components. When both the slow and the fast methods are of the same type (explicit or implicit) we choose the same discretization coefficients $A^{\{s,s\}} = A^{\{f,f\}}$, $b^{\{s\}} = b^{\{f\}}$ such as to obtain a telescopic MrGARK method.
2. The second step is to define the sparsity patterns of the coupling matrices $\mathbf{A}^{\{s,f\}}$ and $\mathbf{A}^{\{f,s\}}$. These patterns determine the computational flow of the method and its implementation complexity, and influence the overall stability and accuracy properties. We manually test various sparsity patterns to balance all of these properties.
3. The third step computes the coupling coefficients $\mathbf{A}^{\{s,f\}}$ and $\mathbf{A}^{\{f,s\}}$ such as to satisfy the coupling conditions (8) up to order p . Any free parameters in the family are used to minimize the Euclidean norm of the residuals of the order $p+1$ coupling conditions; for naturally adaptive MrGARK methods all these residuals are cancelled. In this work the solution of order conditions and the minimization of the error coefficients are carried out with Mathematica Version 11.2.

An alternative to this derivation strategy is a monolithic constrained optimization procedure that minimizes the residuals of the order $p+1$ coupling conditions, subject to solving the conditions (8) up to order p , and subject to structural constraints such as decoupling (17) and stiff accuracy (21).

The proposed three-step procedure is preferable for two reasons. First, we expect MrGARK methods to be applied to problems with a rather weak coupling between partitions, where the primary sources of error are the base methods and not the coupling. Our approach gives precedence to the base errors first. Second, our procedure is practical as it reduces the number of nonlinear equations that need to be solved together during the design.

7. Error estimation and adaptive MrGARK methods. Adaptivity of traditional (single rate) methods adjusts the step size such as to ensure the desired accuracy of the solution at a minimal computational effort. In the context of Runge–Kutta schemes a second “embedded” method is used to provide an a posteriori estimate of the local truncation error [20, CH II.4]. The step size is adjusted to bring the local error estimate to the user prescribed level using the asymptotic relation that this error is proportional to $\propto H^{p+1}$.

Adaptivity of multirate methods is more complex, as there are two independent parameters that control the solution accuracy and efficiency: the macro-step size H and the multirate ratio M (or, equivalently, the macro-step H and the micro-step h .) In order to achieve adaptivity of both H and M we propose to construct error estimates using not one, but several embedded methods, such as to obtain additional information about the structure of the local truncation error. Analytical asymptotic formulas are used to quantify the dependency of the local error terms on both H and M . Armed with this information, we develop several criteria for adaptively selecting both the macro-step size and the multirate ratio.

7.1. Local error structure for a second order MrGARK method. In order to understand the structure of the local truncation error we first consider a second order, internally consistent MrGARK scheme. The principal error terms of order $\mathcal{O}(H^3)$ are associated with the third order conditions, and are provided in

Table 1.

Error type	Error term	Elementary differential	Order condition	Residuals for IM2-EX2 2(1)[A]
Slow	$e_{3,1}^{\{s\}}$	$f_{y,y}^{\{s\}}(f^{\{s\}}, f^{\{s\}})$	$\mathbf{b}^{\{s\}T} \mathbf{c}^{\{s\} \times 2} = \frac{1}{3}$	0
Slow	$e_{3,2}^{\{s\}}$	$f_y^{\{s\}} f_y^{\{s\}} f^{\{s\}}$	$\mathbf{b}^{\{s\}T} \mathbf{A}^{\{s,s\}} \mathbf{c}^{\{s\}} = \frac{1}{6}$	$\frac{1}{6}$
Fast	$e_{3,1}^{\{f\}}$	$f_{y,y}^{\{f\}}(f^{\{f\}}, f^{\{f\}})$	$\mathbf{b}^{\{f\}T} \mathbf{c}^{\{f\} \times 2} = \frac{1}{3}$	$\frac{4-3\sqrt{2}}{12M^2}$
Fast	$e_{3,2}^{\{f\}}$	$f_y^{\{f\}} f_y^{\{f\}} f^{\{f\}}$	$\mathbf{b}^{\{f\}T} \mathbf{A}^{\{f,f\}} \mathbf{c}^{\{f\}} = \frac{1}{6}$	$\frac{4-3\sqrt{2}}{6M^2}$
Coupling	$e_{3,1}^{\{c\}}$	$f_y^{\{s\}} f_y^{\{f\}} f^{\{f\}}$	$\mathbf{b}^{\{s\}T} \mathbf{A}^{\{s,f\}} \mathbf{c}^{\{f\}} = \frac{1}{6}$	$\frac{3\sqrt{2}-3-M}{12M}$
Coupling	$e_{3,2}^{\{c\}}$	$f_y^{\{f\}} f_y^{\{s\}} f^{\{s\}}$	$\mathbf{b}^{\{f\}T} \mathbf{A}^{\{f,s\}} \mathbf{c}^{\{s\}} = \frac{1}{6}$	$\frac{1}{6}$

Table 1: Principal error terms of $\mathcal{O}(H^3)$ for second order MrGARK schemes.

As an example, consider the second order MrGARK method IM2-EX2 2(1)[A] from [Appendix A.4](#). We note from [Table 1](#) that the third order residuals for this method are all asymptotically bounded as M increases, and the local truncation error behaves like:

$$\begin{aligned}
(22) \quad \text{LTE} &= \left(0 \cdot f_{y,y}^{\{s\}}(f^{\{s\}}, f^{\{s\}}) + \frac{1}{6} \cdot f_y^{\{s\}} f_y^{\{s\}} f^{\{s\}} \right) H^3 \\
&+ \left(\frac{4-3\sqrt{2}}{12M^2} \cdot f_{y,y}^{\{f\}}(f^{\{f\}}, f^{\{f\}}) + \frac{4-3\sqrt{2}}{6M^2} \cdot f_y^{\{f\}} f_y^{\{f\}} f^{\{f\}} \right) H^3 \\
&+ \left(\frac{3\sqrt{2}-3-M}{12M} \cdot f_y^{\{s\}} f_y^{\{f\}} f^{\{f\}} + \frac{1}{6} \cdot f_y^{\{f\}} f_y^{\{s\}} f^{\{s\}} \right) H^3 + \mathcal{O}(H^4), \\
\|\text{LTE}\| &\approx \left(K_1 + K_2 \frac{1}{M} + K_3 \frac{1}{M^2} \right) H^3 + \mathcal{O}(H^4).
\end{aligned}$$

In some cases, when there are sufficient degrees of freedom left after solving the order conditions, it is possible for a method of order p to either cancel out coupling errors of order $\mathcal{O}(H^{p+1})$ ([Figure 3a](#)) or to minimize them to assure better coupling behavior ([Figure 3b](#)). [Figure 3](#) shows how leading error coefficients change with M for two optimized MrGARK methods.

7.2. General structure of the MrGARK local truncation error. As the analysis in [subsection 7.1](#) reveals, the local truncation error of the MrGARK method (4) has three components associated with the slow integration, the fast integration, and with the coupling:

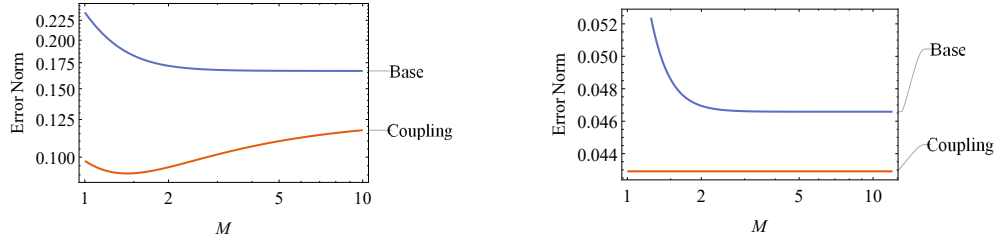
$$(23) \quad \text{LTE} = \text{LTE}^{\{s\}} + \text{LTE}^{\{f\}} + \text{LTE}^{\{c\}}.$$

For a method of order p the slow truncation error is that of applying one step with the base slow Runge–Kutta method with a step size H :

$$(24a) \quad \|\text{LTE}^{\{s\}}\| \approx C^{\{s\}} \cdot H^{p+1} + \mathcal{O}(H^{p+2}).$$

The fast truncation error is that of applying M consecutive steps with the base fast Runge–Kutta method with a step size H/M . We use the global error estimate [20, CH II.3]:

$$\|\text{LTE}^{\{f\}}\| \leq \left(\frac{H}{M} \right)^p \frac{C'}{L} (\exp(LH) - 1) = \left(\frac{H}{M} \right)^p \frac{C'}{L} (LH + \mathcal{O}(H^2)),$$



(a) EX2-EX2 2(1)[A]. The base error is $\mathcal{O}(H^3)$, but the coupling is $\mathcal{O}(H^4)$ since the method is naturally adaptive.

(b) EX3-EX3 3(2)[A]. Both the base and coupling errors are $\mathcal{O}(H^4)$.

FIG. 3: Behavior of the local truncation error components for two MrGARK schemes as the multirate ratio M increases.

where $\|\partial f^{\{f\}}/\partial y\| \leq L$ and C' is a constant, to obtain:

$$(24b) \quad \|\text{LTE}^{\{f\}}\| \leq \frac{C^{\{f\}}}{M^p} \cdot H^{p+1} + \mathcal{O}(H^{p+2}).$$

The principal terms of the coupling errors correspond to N-trees having $p + 1$ nodes of two colors. A tree with k fast nodes and $p + 1 - k$ slow nodes corresponds to a residual term constructed from multiplying k fast matrix blocks in the Butcher tableau (5) and $p + 1 - k$ slow matrix blocks. Note that each of the fast matrix blocks $(\mathbf{b}^{\{f\}T}, \mathbf{A}^{\{s,f\}}, \mathbf{A}^{\{f,f\}})$ in (5) carries a scaling factor of $1/M$. Assuming that the coupling coefficients remain bounded for large M , a product of k fast blocks is $\mathcal{O}(1/M^k)$. Since we have coupling trees with $p \geq k \geq 1$ fast nodes, the corresponding products of fast blocks have scalings ranging from $1/M$ to $1/M^p$. Residuals contain sums of M elementary coupling blocks as seen in (10); sums of M fast blocks have scaling factors ranging from 1 to $1/M^{p-1}$, but some of the sums can have a small, fixed number of terms. In addition, the magnitude of the coupling coefficients, as resulted from the solution of order conditions, can scale as M^ℓ (for some $\ell \geq 0$). Consequently, a generic expression for the local coupling error is:

$$(24c) \quad \|\text{LTE}^{\{c\}}\| \approx \left(\sum_{i=-\ell}^p C_i^{\{c\}} M^{-i} \right) \cdot H^{p+1} + \mathcal{O}(H^{p+2}),$$

While the slow and fast errors are simple functions of H and M , the dependency of the coupling error on M is more difficult to describe even with access to the residuals of each of the $p + 1$ order conditions. These residuals can be positive or negative. Moreover, the evolution of the coupling error as a function of M depends not only on residuals, but also on elementary differentials, since the constants $C_i^{\{c\}}$ in (24c) are linear combinations of products of method residuals and norms of elementary differentials. Therefore, in an adaptive method, the effect of changing M on the coupling error is more difficult to quantify.

7.3. Use of multiple embedded methods to estimate the local truncation error. Following the traditional Runge–Kutta strategy, we look to use embedded methods to obtain estimates of the local truncation error. Specifically, we design

pairs of main weight vectors $(b^{\{f\}}, b^{\{s\}})$ and embedded weight vectors $(\widehat{b}^{\{f\}}, \widehat{b}^{\{s\}})$ that produce solutions of different orders when paired with $(\mathbf{A}^{\{f,f\}}, \mathbf{A}^{\{f,s\}}, \mathbf{A}^{\{s,f\}}, \mathbf{A}^{\{s,s\}})$. The choice of the embedded weights is made such that additional function evaluations are avoided.

In the traditional Runge–Kutta approach, a single embedded method is used to provide a single estimate of the error, and a step size scaling factor is computed (for the next step) such that the corresponding scaled error estimate satisfies the accuracy requirements at hand. In multirate integration the solution accuracy depends on two parameters, H and M , that can be adjusted independently. We have seen in [subsection 7.2](#) that changes in these parameters affect differently the slow, fast, and coupling components of the local truncation error.

Our proposed strategy to select H and M adaptively relies on multiple embedded methods to independently estimate different parts of the error. Specifically, consider an MrGARK scheme that produces a main solution y_{n+1} of order p , i.e., cancels all residuals for two-trees with up to p nodes. We seek to build an embedded solution $\widehat{y}_{n+1}^{\{s\}}$ that cancels all residuals up to order $p - 1$, as well as all fast and all coupling residuals of order p . Similarly, we seek to build a second embedded solution $\widehat{y}_{n+1}^{\{f\}}$ that cancels all residuals up to order $p - 1$, as well as all slow and all coupling residuals of order p . The three components of the local truncation error can then be estimated as follows:

$$(25) \quad \begin{aligned} \text{LTE}_{n+1} &\approx y_{n+1} - \widehat{y}_{n+1}, & \text{LTE}_{n+1}^{\{s\}} &\approx y_{n+1} - \widehat{y}_{n+1}^{\{s\}}, & \text{LTE}_{n+1}^{\{f\}} &\approx y_{n+1} - \widehat{y}_{n+1}^{\{f\}}, \\ \text{LTE}_{n+1}^{\{c\}} &= \text{LTE}_{n+1} - \text{LTE}_{n+1}^{\{s\}} - \text{LTE}_{n+1}^{\{f\}} &&\approx \widehat{y}_{n+1}^{\{s\}} + \widehat{y}_{n+1}^{\{f\}} - \widehat{y}_{n+1} - y_{n+1}. \end{aligned}$$

The construction of three different embedded methods that capture precisely some components of the error can be very difficult to achieve for high order schemes. For this reason we now discuss a simplified error estimation strategy that utilizes only two embedded methods. Consider an MrGARK scheme with primary weights $(b^{\{f\}}, b^{\{s\}})$ that produces a main solution y_{n+1} of order p . The residuals corresponding to two-trees with up to p nodes are zero. We construct three different embedded solutions, as follows:

- A generic embedded solution \widehat{y}_{n+1} of order $p - 1$, obtained with weights $(\widehat{b}^{\{f\}}, \widehat{b}^{\{s\}})$, that captures all the error terms and is used to approximate the overall local truncation error. The residuals associated with two-trees with up to $p - 1$ nodes are zero, while the residuals of two-trees with p nodes can be nonzero.
- An embedded solution $\widehat{y}_{n+1}^{\{s\}}$, generated with the weights $(b^{\{f\}}, \widehat{b}^{\{s\}})$, captures all of the slow error, part of the coupling error, and none of the fast error. The weight vector $b^{\{f\}}$ cancels the residuals for all two-trees with up to p nodes with a fast-colored root. The weight vector $\widehat{b}^{\{s\}}$ cancels the residuals for all two-trees with up to $p - 1$ nodes with a slow-colored root. However, the residuals of the two-trees with p nodes and a slow-colored root, and the associated $\mathcal{O}(H^p)$ error terms, can be nonzero. Trees with a slow-colored root correspond to either slow or to coupling trees, and the solution difference $y_{n+1} - \widehat{y}_{n+1}^{\{s\}}$ captures the sum of the corresponding errors terms.
- Similarly, an embedded solution $\widehat{y}_{n+1}^{\{f\}}$, generated with $(\widehat{b}^{\{f\}}, b^{\{s\}})$, captures all of the fast error, part of the coupling error, and none of the slow error.

In general, these mixed embeddings do not exactly isolate the slow, fast, and coupling error, but they can serve as approximations in (25). Note that the resulting

approximate $\text{LTE}^{\{\text{s}\}}$ and $\text{LTE}^{\{\text{f}\}}$ contain the parts of the coupling error corresponding to trees of order p with slow-colored roots and with fast colored-roots, respectively. Consequently the $\text{LTE}^{\{\text{c}\}}$ estimated by the solution difference in (25) is zero.

However, for naturally adaptive methods (Definition 6.1), the simplified error estimation strategy does separate the main components of the fast and slow errors, as explained next. For a naturally adaptive method of order p with weights $(b^{\{\text{f}\}}, b^{\{\text{s}\}})$ the residuals corresponding to all two-trees with up to p nodes are zero, and the coupling residuals for two-trees with $p+1$ nodes are also zero. The non-zero residuals of order p correspond to either fast or low trees with p nodes.

A naturally adaptive embedded solution \hat{y}_{n+1} of order $p-1$, obtained with weights $(\hat{b}^{\{\text{f}\}}, \hat{b}^{\{\text{s}\}})$, cancels all residuals of order up to $p-1$; it also cancels the coupling residuals for two-trees of up to p nodes. Consequently, the difference $y_{n+1} - \hat{y}_{n+1}^{\{\text{s}\}}$ contains only $\mathcal{O}(H^p)$ terms corresponding to slow-colored trees, and $y_{n+1} - \hat{y}_{n+1}^{\{\text{f}\}}$ contains only $\mathcal{O}(H^p)$ terms corresponding to fast-colored trees. Consequently, naturally adaptive embedded methods *do* isolate the slow and the fast errors. This property allows to construct more accurate adaptivity mechanisms.

7.4. Controlling errors by adapting both the macro-step and the micro-step. Based on our understanding of the structure of errors in the MrGARK framework, we now look into practical approaches to adaptivity of multirate methods. The following error estimates are available via the set of embedded methods:

$$\hat{\varepsilon}_{n+1} := \|y_{n+1} - \hat{y}_{n+1}\|_{\text{err}}, \quad \hat{\varepsilon}_{n+1}^{\{\text{s}\}} := \|y_{n+1} - \hat{y}_{n+1}^{\{\text{s}\}}\|_{\text{err}}, \quad \hat{\varepsilon}_{n+1}^{\{\text{f}\}} := \|y_{n+1} - \hat{y}_{n+1}^{\{\text{f}\}}\|_{\text{err}},$$

where the error is measured by the following relative error norm [20, CH II.4]:

$$\|x - y\|_{\text{err}} := \sqrt{\frac{1}{d} \sum_{i=1}^d \left(\frac{x_i - y_i}{\text{AbsTol}_i + \text{RelTol}_i \cdot \max(|x_i|, |y_i|)} \right)^2}.$$

Several heuristic strategies that use these error estimates to adapt both H and M are discussed below.

7.4.1. Balancing error strategy. In this approach, we first use the estimated total truncation error $\hat{\varepsilon}_{n+1}$ to control the macro-step size using the traditional mechanism based on the asymptotic error behavior [20, CH II.4]:

$$\hat{\varepsilon}_{n+2} \leq 1 \quad \Rightarrow \quad H_{\text{new}} = \text{fac} \cdot H \cdot (\hat{\varepsilon}_{n+1})^{-\frac{1}{p}},$$

where $\text{fac} < 1$ is a safety factor.

Next, the multirate ratio M is adjusted such that the estimated slow and fast error components over the next step are equal to each other, i.e., the slow and fast contributions to the error are balanced. Assuming $q = \min(p, \hat{p})$ and using the asymptotic formulas (24) we have that:

$$(26) \quad \begin{aligned} \hat{\varepsilon}_{n+2}^{\{\text{s}\}} &= \hat{\varepsilon}_{n+2}^{\{\text{f}\}}, \\ C^{\{\text{s}\}} \cdot H_{\text{new}}^{q+1} &\approx \frac{C^{\{\text{f}\}}}{M_{\text{new}}^q} \cdot H_{\text{new}}^{q+1}, & \Rightarrow & \quad M_{\text{new}} \approx M \cdot \left(\frac{\hat{\varepsilon}_{n+1}^{\{\text{f}\}}}{\hat{\varepsilon}_{n+1}^{\{\text{s}\}}} \right)^{\frac{1}{q}}, \\ \hat{\varepsilon}_{n+1}^{\{\text{s}\}} \cdot \frac{H_{\text{new}}^{q+1}}{H^{q+1}} &\approx \hat{\varepsilon}_{n+1}^{\{\text{f}\}} \cdot \frac{M^q}{M_{\text{new}}^q} \cdot \frac{H_{\text{new}}^{q+1}}{H^{q+1}}, \end{aligned}$$

In this strategy, as well as the others, M can be rounded up, down, or to the nearest integer. Also in practice, the re-scaling of H and M for the next step can be bounded up and down in order to avoid large jumps and oscillations. This adaptivity strategy serves as a very simple heuristic. However, the choices of the new H and of the new M are made independently of each other, and the approach does not account for how their interaction impacts the error; the only mechanism for controlling the coupling error is the change in H .

7.4.2. Efficiency optimization strategy. This approach focuses on the important aspect of the overall cost of multirate integration. Evaluation of the slow and fast partitions can have very different computational costs in some applications. Moreover, an implicit method (e.g., applied to solve the fast component) is likely to be much more expensive than an explicit method (e.g., applied to the slow component). We require the adaptive selection of H and M to satisfy the error tolerance criteria at a minimal overall computational cost.

Let $t^{\{s\}}$ and $t^{\{f\}}$ represent the computational costs of a slow macro-step and a fast micro-step, respectively. We define the computational efficiency of a step as the progress made during the step (H) divided by the total cost of executing step ($t^{\{s\}} + M_{\text{new}}t^{\{f\}}$).

The new values of H and M are selected such as to achieve the desired accuracy while maximizing the computational efficiency. This requires solving the following constrained optimization problem to minimize the inverse of efficiency:

$$(27) \quad \min_{H_{\text{new}}, M_{\text{new}}} \frac{t^{\{s\}} + M_{\text{new}} t^{\{f\}}}{H_{\text{new}}},$$

subject to $\widehat{\varepsilon}_{n+2} = 1$.

Expanding the constraint yields:

$$1 = \widehat{\varepsilon}_{n+2} \approx \widehat{\varepsilon}_{n+1}^{\{s\}} \cdot \frac{H_{\text{new}}^{q+1}}{H^{q+1}} + \widehat{\varepsilon}_{n+1}^{\{f\}} \cdot \frac{M^q}{M_{\text{new}}^q} \cdot \frac{H_{\text{new}}^{q+1}}{H^{q+1}},$$

where we have used the fact that, for naturally adaptive methods, the coupling component of the local truncation error is negligible in the asymptotic regime. The constraint equation can be solved explicitly for H_{new} :

$$(28) \quad H_{\text{new}} = H \cdot \left(\widehat{\varepsilon}_{n+1}^{\{s\}} + \widehat{\varepsilon}_{n+1}^{\{f\}} \cdot \frac{M^q}{M_{\text{new}}^q} \right)^{-\frac{1}{q+1}}.$$

After eliminating the constraint the optimization problem (27) simplifies to:

$$(29) \quad \min_{M_{\text{new}}} \frac{t^{\{s\}} + M_{\text{new}} t^{\{f\}}}{H} \left(\widehat{\varepsilon}_{n+1}^{\{s\}} + \widehat{\varepsilon}_{n+1}^{\{f\}} \cdot \frac{M^q}{M_{\text{new}}^q} \right)^{\frac{1}{q+1}}.$$

Note this is an integer minimization problem. One can solve it as a continuous optimization problem, then round the result to an integer to find the optimal M_{new} . Afterwards H_{new} is computed from (28).

REMARK 7.1 (Timing). *The CPU times $t^{\{s\}}$ and $t^{\{f\}}$ can be evaluated online by timing the slow macro-steps and the fast micro-steps during their execution. The algorithm adjusts automatically if these compute times vary during the application lifetime.*

8. New high-order MrGARK methods. Using the design process outlined above we construct several MrGARK methods for use in practical applications. We use the naming convention *FASTf-SLOWs* $p(\hat{p})[type]$, where p is the method order, \hat{p} is the embedded order, f is the number of stages in the fast base method, and s is the number of stages in the slow base method. Each component method is either explicit or implicit: FAST, SLOW \in {EX,IM}. We distinguish between methods of type A (optimized for accuracy and for better step size control) and methods of type S (optimized for simplicity and for stability), therefore $type \in$ {A,S}. The newly developed methods are as follows:

- EX2-EX2 2(1)[A], EX2-EX2 2(1)[S], EX3-EX3 3(2)[A], EX4-EX4 3(2)[A], EX3-EX3 3(2)[S], and EX5-EX5 4(3)[A] are explicit-explicit, schemes of order two, three, three and four respectively.
- EX2-IM2 2(1)[A], EX3-IM3 3(2)[A], and EX6-IM5 4(3)[A] are methods with an explicit fast part and an implicit slow part of order two, three, and four respectively.
- IM2-EX2 2(1)[A], IM3-EX3 3(2)[A], and IM6-EX5 4(3)[A] are methods with an implicit fast part and an explicit slow part of order two, three, and four, respectively.

The coefficients of these methods are given in [Appendix A](#).

9. Numerical experiments. In this section we carry out numerical experiments to validate and test the newly derived MrGARK methods.

9.1. Additive partitioning tests. The first experiment is carried out using a two-dimensional unsteady convection-diffusion equation [10, Ch. 3] in a square spatial domain $\Omega = \{0 \leq x, y \leq 1\}$ and with a circular wind profile:

$$(30a) \quad u_t - \varepsilon \nabla^2 u + w \cdot \nabla u = 0 \text{ in } \Omega, \quad u = 0 \text{ on } \partial\Omega, \quad w = \begin{bmatrix} 2y(1-x^2) \\ -2x(1-y^2) \end{bmatrix}.$$

A Streamline Upwind Petrov-Galerkin (SUPG) spatial discretization is used, which leads to a semi-discrete system of linear ODEs:

$$(30b) \quad \mathbf{M}^h u_t^h = \mathbf{A} u^h + (\bar{n} + \bar{n}^{stab}) u^h,$$

where \mathbf{M}^h and \mathbf{A} are mass and stiffness matrices and $\bar{n} + \bar{n}^{stab}$ represent linear forms of the convective term and SUPG stabilization.

In the multirate experiments we designate the first term in (30b) as the slow component, $f^{\{s\}} = (\mathbf{M}^h)^{-1} \mathbf{A} u^h$, and the second term as the fast component, $f^{\{f\}} = (\mathbf{M}^h)^{-1} (\bar{n} + \bar{n}^{stab}) u^h$. In practice the splitting choice is informed by inspecting the spectral radius of the right hand side operators. The weak form of the PDE and the corresponding MrGARK schemes are implemented in the FEniCS package [2], which is used to carry out the convergence experiments.

The convergence diagrams for this test are shown in [Figure 4](#). The numerical orders of convergence for all schemes match their theoretical orders for the multirate step ratios tested. [Figure 5](#) shows the evolution of the model solution over time.

9.2. Timing experiments. Timing experiments are performed in MATLAB using the Gray-Scott model [26]. Here, we are interested in the application of MrGARK methods to additive splittings of the right hand side into linear and nonlinear terms. This reaction-diffusion PDE is:

$$(31) \quad \begin{cases} u_t = \nabla \cdot (\varepsilon_u \nabla u) - u v^2 + f(1-u), \\ v_t = \nabla \cdot (\varepsilon_v \nabla v) + u v^2 - (f + \mathfrak{k}) v. \end{cases}$$

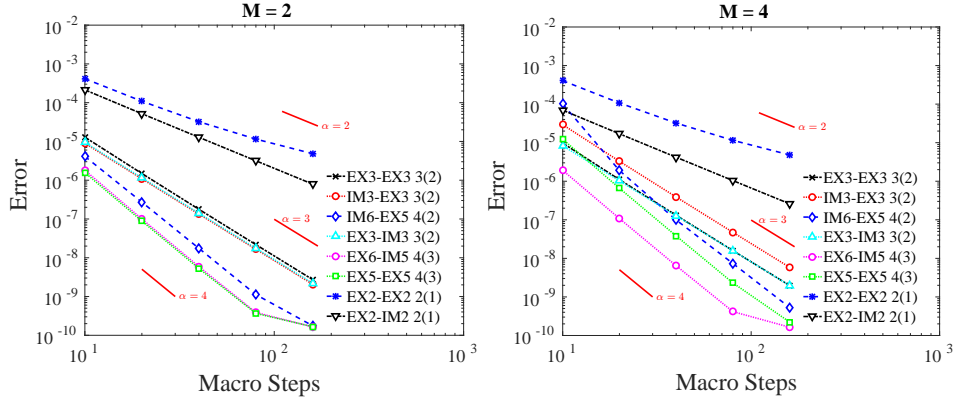


FIG. 4: Convergence plots for the unsteady convection-diffusion test (30) over the time span $T = [0, 10]$ seconds. A fixed macro-step time integration is carried out with varying multirate step ratios M using MrGARK type A methods.

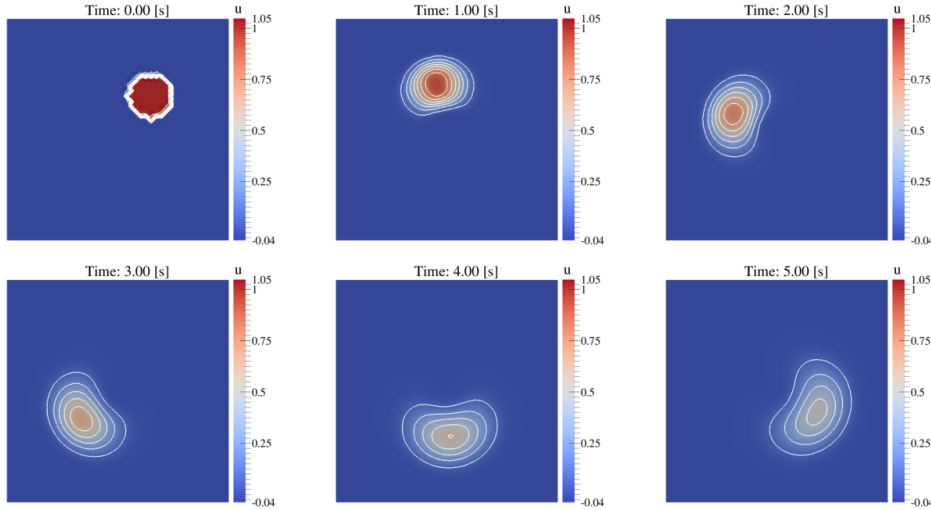


FIG. 5: Evolution of the unsteady convection-diffusion problem (30) solution in time.

The domain is the unit square discretized with second order finite differences. The reaction parameters are $\mathfrak{k} = 0.0520$ and $\mathfrak{f} = 0.0180$.

In the following experiments the nonlinear reaction terms on the right hand side are considered the fast system, and the diffusion terms are regarded as the slow one.

A first version of the model, used to test explicit MrGARK methods, includes nonlinear diffusion terms:

$$(32) \quad \varepsilon_u = 0.0625 e^{-\frac{u}{100}} \sin(\pi x) \sin(\pi y), \quad \varepsilon_v = 0.0312 e^{-\frac{v}{100}} \sin(\pi x) \sin(\pi y).$$

Figure 6 presents the evolution of quantity u over time. Figures 7a and 7b present performance diagrams for a fixed-step time integration of Gray-Scott model with nonlinear diffusion. The MrGARK method EX3-EX3 3(2)[A] from Appendix A.5

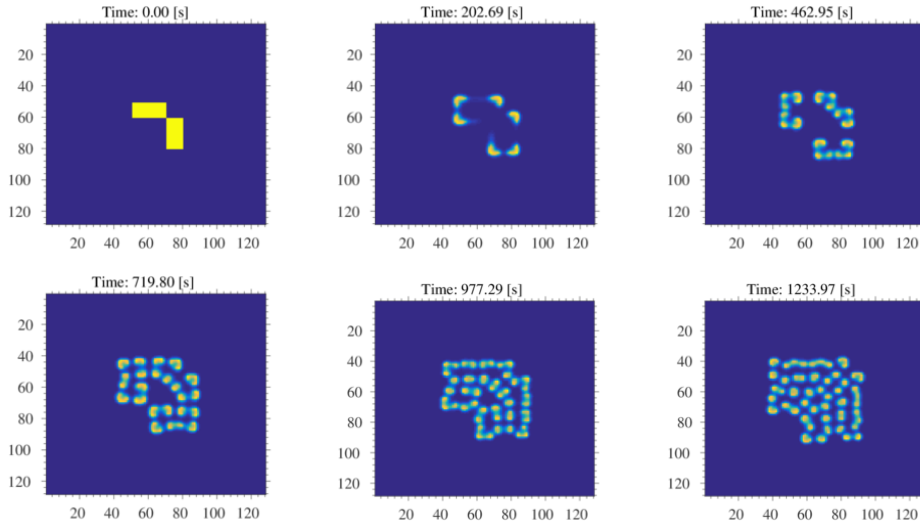


FIG. 6: Evolution of Gray-Scott model (31) solution u in time. A nonlinear diffusion is used, and integration is performed with explicit MrGARK methods.

with different multirate step ratios is used. Notice that as M increases, it is possible to increase the macro-step size without violating the CFL conditions. The results shown in Figure 7a indicate that, for a fixed error level, the multirate method shows better performance than single rate one for $M = 2$ and 3.

A second version of the model, used to test EX-IM MrGARK methods, includes linear diffusive terms with parameters $\varepsilon_u = 0.0625$ and $\varepsilon_v = 0.0312$. The diffusion is considered the slow process and is treated implicitly; the constant diffusion operator is leveraged in the solution of linear stage equations. Figures 7c and 7d show the performance diagrams for the MrGARK method EX3-IM3 3(2)[A] from Appendix A.8 compared to single rate implicit method of the same order. In Figure 7d we note the reduced order of convergence for the single rate implicit method. Being a DIRK scheme, this method has a Newton-Krylov iteration for each stage that involves Jacobian-vector products of the full right hand side. Here, as in many practical cases, these are approximated by finite differences. Therefore, the quality of the approximation and the Krylov solver affect the convergence rate and efficiency of the SR-IM method [33]. On the other hand, the stages for multirate methods use only the Jacobian of the linear term, and show full order of convergence and faster computations than both single rate implicit and explicit methods. As M increases the performance of the multirate method improves incurring negligible increase in error.

9.3. Numerical experiments for H and M adaptivity. We experiment with different adaptivity strategies described in subsection 7.4 using the Gray-Scott model (31) with nonlinear diffusion term (32). All experiments in this section are performed for a time span of $T = [0, 2]$ seconds.

Figure 8 shows the automatically selected macro-step size (H) and multirate step ratio (M) for the naturally adaptive method EX4-EX4 3(2)[A]. The efficiency optimization strategy (subsection 7.4.2) is used to select both H and M . As the relative cost of evaluating the fast and slow right hand side functions changes in

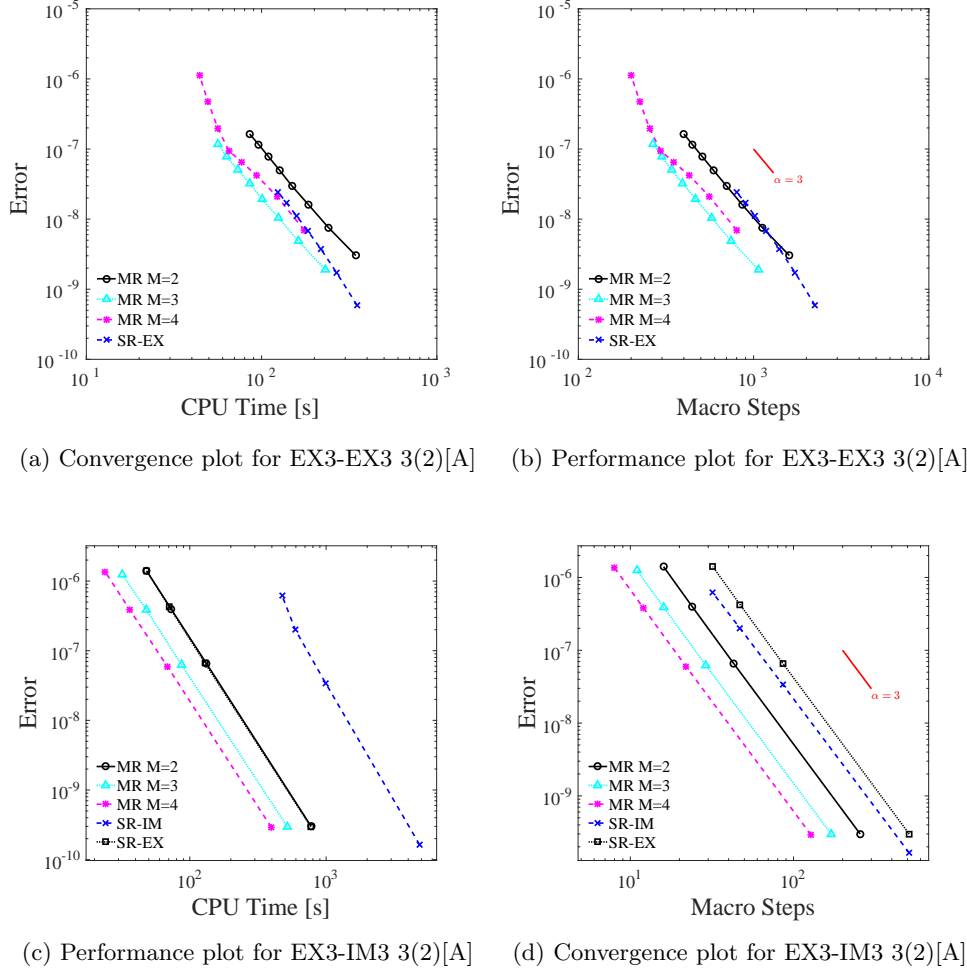


FIG. 7: Results for MrGARK schemes applied to Gray-Scott model (31).

Figures 8a to 8c, the choice of M varies to keep the overall efficiency high. The costs of slow and fast system are computed by timing their respective right hand side evaluations. The one-dimensional integer optimization in (29) is then approximated by limiting the possible increase or decrease of M_{new} to

$$\max(1, M - 1) \leq M_{\text{new}} \leq M + 2,$$

which reduces the computational cost of the optimization and increases the robustness of the algorithm.

In Figure 9 the adaptivity strategy based on balancing the fast and slow errors (subsection 7.4.1) is tested. By interchanging the roles of the fast and slow partitions the choice of multirate step ratio M changes from the maximum allowed value of ten to its minimum allowed value of two, in an attempt to keep the fast and slow errors balanced. The macro-step size selection by the traditional error controller is the same

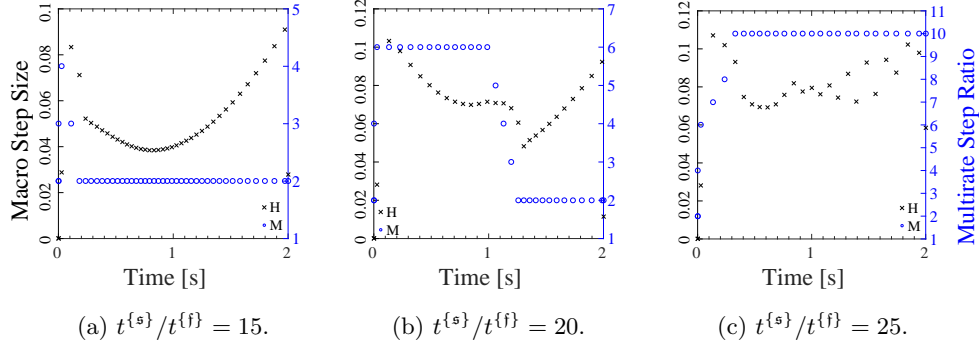


FIG. 8: Automatically selected macro-step size and multirate step ratio for the Gray-Scott model (31) integrated over the time span $T = [0, 2]$ using EX4-EX4 3(2)[A] method. The efficiency optimization strategy (subsection 7.4.2) is used with AbsTol = RelTol = 10^{-4} .

in both cases.

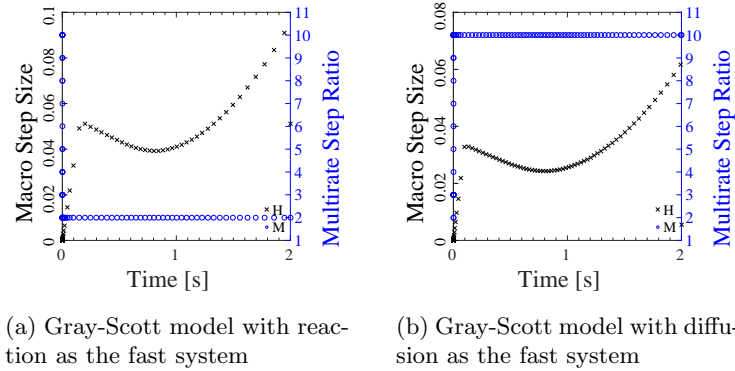


FIG. 9: Automatically selected macro-step size and multirate step ratio for the Gray-Scott model (31) integrated over the time span $T = [0, 2]$ seconds using EX4-EX4 3(2)[A] method. The strategy of balancing the slow and fast errors (subsection 7.4.1) is used with AbsTol = RelTol = 10^{-2} .

Finally, it is instructive to see how different H - M adaptivity strategies developed in subsection 7.4 compare against the conventional H adaptivity. We carry out this experiment using the Gray-Scott model. Figure 10a shows the error in the final solution scaled by the computation time when the EX2-EX2 2(1)[A] method from Appendix A.1 is used. Figure 10a repeats the experiment using the EX5-EX5 4(3)[A] method from Appendix A.10. In these experiments the efficient single rate base method is employed when the adaptive strategy selects $M = 1$. The parameters of the adaptivity algorithm such as macro-step rejection factor are optimized for maximum

efficiency. The results indicate that the H - M adaptivity strategies are perform better than the classical H error control.

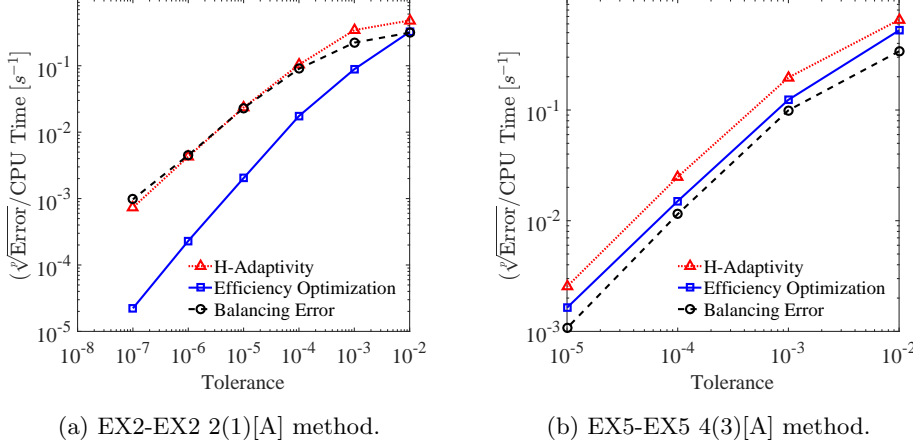


FIG. 10: Efficiency of different adaptivity strategies for explicit-explicit methods applied to Gray-Scott model with a compute time ratio $t^{\{s\}}/t^{\{i\}} = 2$.

9.4. Component partitioning experiment. We consider the BSVD reaction-diffusion problem on the unit square Ω [21] with boundary $\partial\Omega$ and outward boundary normal vector \vec{n} :

$$(33a) \quad u_t = \nabla \cdot (D(x, y)\nabla u) + 10(1 - u^2)(u + 0.6),$$

$$(33b) \quad u(x, y, 0) = 2 \exp(-10(x - 0.5)^2 - 10(y + 0.1)^2), \quad x, y \in \Omega,$$

$$(33c) \quad D(x, y) = 0.1 \sum_{i=1}^3 e^{-100(x-0.5)^2 + (y-y_i)^2},$$

$$(33d) \quad D(x, y)\nabla u \cdot \vec{n} = 0, \quad x, y \in \partial\Omega, t \in [0, t_F].$$

The parameters values are $y_1 = 0.6, y_2 = 0.75, y_3 = 0.9$ as prescribed in [21]. The domain is partitioned into two sub-domains:

$$\Omega_1 = \{(x, y) \in \Omega : |x - 0.5| \leq 0.125, y \leq 0.125\}, \quad \Omega_2 = \Omega \setminus \Omega_1.$$

A continuous finite element semi-discretization in space with Lagrange polynomial basis of order four leads to a two-way partitioned system of ODEs. We discretized this model using the FEniCS package [2] and the evolution of its solution over the time span of $T = [0, 5]$ seconds is shown in Figure 11. The subsystem with fewer degrees of freedom (DOFs) is designated as the fast one, and the remaining variables are considered to be the slow system. In this experiment the ratio of slow to fast DOFs is 28. Figure 12 shows the error for fixed macro-step time integration using explicit-explicit multirate methods of orders 2, 3 and 4. The largest macro-steps are chosen close to the maximum stable step size for the method. Results reported in Figure 12 show that, although the numerical order of convergence of the methods

fluctuate slightly, they follow their theoretical values closely. Note that the errors are reported only for stable macro-step sizes for different methods and multirate step ratios.

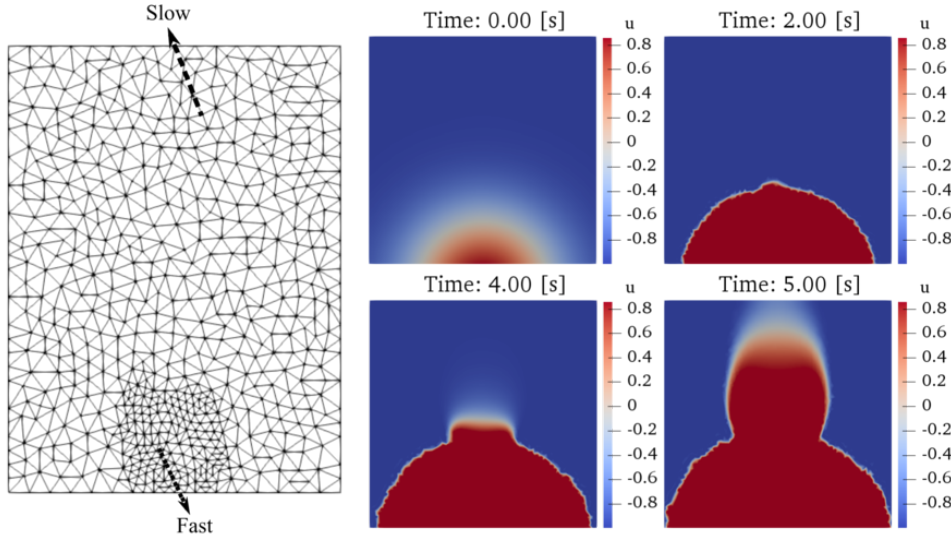


FIG. 11: Evolution of the BSVD problem (33) solution in time.

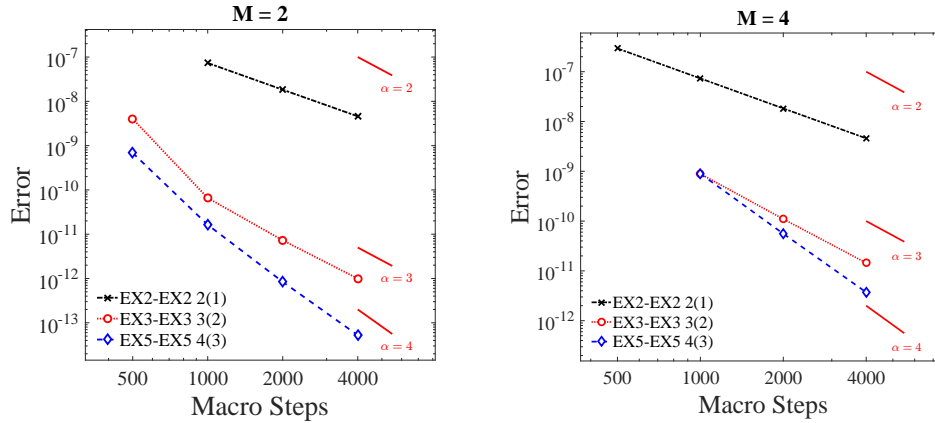


FIG. 12: Convergence results for BSVD test (33) over the time span $T = [0, 0.2]$ seconds. A fixed macro-step time integration is carried out with varying multirate step ratios M using explicit MrGARK type A methods.

10. Conclusions. This work develops a coherent design strategy for high order MrGARK, and constructs several particular schemes of explicit-explicit, explicit-implicit and implicit-explicit types up to order four. Adopting a two-way split system for simplicity, we identify coupling structures for the fast and slow components that

strike a balance between stability and computational efficiency of MrGARK methods. Furthermore, in the process of designing schemes we explore a variety of optimizations such as local truncation error minimization, FSAL, and stiff accuracy of base and overall implicit methods. When possible, our methods are endowed with telescopic properties to facilitate their application to multi-scale, multi-domain problems. Different design criteria lead to different types of methods, e.g., the A-type is optimized for accuracy, and the S-type is optimized for simplicity and stability. A novel concept of H - M adaptivity, based on multiple error estimators and the property of natural adaptivity, is presented.

Our numerical experiments demonstrate several applications of these methods with finite element and finite difference discretized PDEs, where different time-scales in system variables (component partitioning), or in various physical processes of the problem (additive partitioning) are treated with different time steps. In all cases, the proposed methods converge at their theoretical orders. In all cases, speedups are obtained from the multirate approach when compared to the single rate integration. Since the multirate performance depends critically on the efficiency of the implementation, and a high quality implementation is not within the scope of this work, we do not report these speedups, as they are likely considerably smaller than what can realistically be achieved. The future work plans of the authors include the development of high quality implementations of MrGARK schemes. Moreover, for many of the new schemes derived here the stability region decreases with increasing step size ratio; future work will focus on the development of S-type schemes where the stability region remains large for any M .

This contribution focuses on *decoupled* multirate methods satisfying (17), as they are far less computationally demanding than coupled methods. However, the overall stability of the scheme may be affected by this choice. In future work the authors plan to explore coupled implicit-implicit MrGARK methods, and to study their stability via an extended, two-dimensional test problem [9].

REFERENCES

- [1] R. ALEXANDER, *Diagonally implicit Runge–Kutta methods for stiff ODE’s*, SIAM Journal on Numerical Analysis, 14 (1977), pp. 1006–1021.
- [2] M. S. ALNÆS, J. BLECHTA, J. HAKE, A. JOHANSSON, B. KEHLET, A. LOGG, C. RICHARDSON, J. RING, M. E. ROGNES, AND G. N. WELLS, *The FEniCS project version 1.5*, Archive of Numerical Software, 3 (2015), <https://doi.org/10.11588/ans.2015.100.20553>.
- [3] J. ANDRUS, *Numerical solution for ordinary differential equations separated into subsystems*, SIAM Journal on Numerical Analysis, 16 (1979), pp. 605–611.
- [4] J. ANDRUS, *Stability of a multirate method for numerical integration of ODEs*, Computers Math. Applic, 25 (1993), pp. 3–14.
- [5] A. BARTEL AND M. GÜNTHER, *A multirate W-method for electrical networks in state-space formulation*, Journal of Computational Applied Mathematics, 147 (2002), pp. 411–425, [https://doi.org/10.1016/S0377-0427\(02\)00476-4](https://doi.org/10.1016/S0377-0427(02)00476-4).
- [6] S. BREMICKER-TRÜBELHORN AND S. ORTLEB, *On multirate GARK schemes with adaptive micro-step sizes for fluid-structure interaction: order conditions and preservation of the geometric conservation law*, Aerospace, 4 (2017), <https://doi.org/10.3390/aerospace4010008>.
- [7] E. CONSTANTINESCU AND A. SANDU, *Multirate timestepping methods for hyperbolic conservation laws*, Journal on Scientific Computing, 33 (2007), pp. 239–278, <https://doi.org/10.1007/s10915-007-9151-y>, <http://dx.doi.org/10.1007/s10915-007-9151-y>.
- [8] E. CONSTANTINESCU AND A. SANDU, *On extrapolated multirate methods*, in Progress in Industrial Mathematics at ECMI 2008, H.-G. Bock, F. Hoog, A. Friedman, A. Gupta, H. Neunzert, W. R. Pulleyblank, T. Rusten, F. Santosa, A.-K. Tornberg, V. Capasso, R. Mattheij, H. Neunzert, O. Scherzer, A. D. Fitt, J. Norbury, H. Ockendon, and E. Wilson, eds., vol. 15

- of Mathematics in Industry, Springer Berlin Heidelberg, 2010, pp. 341–347, https://doi.org/10.1007/978-3-642-12110-4_52, http://dx.doi.org/10.1007/978-3-642-12110-4_52.
- [9] E. CONSTANTINESCU AND A. SANDU, *Extrapolated multirate methods for differential equations with multiple time scales*, Journal of Scientific Computing, 56 (2013), pp. 28–44, <https://doi.org/10.1007/s10915-012-9662-z>, <http://dx.doi.org/10.1007/s10915-012-9662-z>.
- [10] H. C. ELMAN, D. J. SILVESTER, AND A. J. WATHEN, *Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics*, Numerical Mathematics and Scientific Computation, 2014.
- [11] B. ENQUIST AND R. TSAI, *Heterogeneous multiscale methods for stiff ordinary differential equations*, Mathematics of Computation, 74 (2005), pp. 1707–1742.
- [12] C. ENGSTLER AND C. LUBICH, *Multirate extrapolation methods for differential equations with different time scales*, Computing, 58 (1997), pp. 173–185, <https://doi.org/10.1007/BF02684438>, <https://doi.org/10.1007/BF02684438>.
- [13] E. FEHLBERG, *Low-order classical Runge–Kutta formulas with stepsize control and their application to some heat transfer problems*, NASA Technical Report, NASA-TR-R-315, (1969).
- [14] C. GEAR AND D. WELLS, *Multirate linear multistep methods*, BIT, 24 (1984), pp. 484–502.
- [15] M. GÜNTHER, C. HACHTEL, AND A. SANDU, *Multirate GARK schemes for multiphysics problems*, in 10th International Conference on Scientific Computing in Electrical Engineering, 2014, https://doi.org/10.1007/978-3-319-30399-4_12, http://dx.doi.org/10.1007/978-3-319-30399-4_12.
- [16] M. GÜNTHER AND M. HOSCHEK, *ROW methods adapted to electric circuit simulation packages*, in ICCAM '96: Proceedings of the seventh international congress on Computational and applied mathematics, Elsevier Science Publishers B. V., 1997, pp. 159–170, [https://doi.org/http://dx.doi.org/10.1016/S0377-0427\(97\)00043-5](https://doi.org/http://dx.doi.org/10.1016/S0377-0427(97)00043-5).
- [17] M. GÜNTHER, A. KVÆRNØ, AND P. RENTROP, *Multirate partitioned Runge-Kutta methods*, BIT Numerical Mathematics, 41 (2001), pp. 504–514, <https://doi.org/10.1023/A:1021967112503>, <https://doi.org/10.1023/A:1021967112503>.
- [18] M. GÜNTHER AND P. RENTROP, *Partitioning and multirate strategies in latent electric circuits*, Birkhäuser Basel, Basel, 1994, pp. 33–60, https://doi.org/10.1007/978-3-0348-8528-7_3, https://doi.org/10.1007/978-3-0348-8528-7_3.
- [19] M. GÜNTHER AND A. SANDU, *Multirate generalized additive Runge-Kutta methods*, Numerische Mathematik, 133 (2016), pp. 497–524, <https://doi.org/10.1007/s00211-015-0756-z>, <http://dx.doi.org/10.1007/s00211-015-0756-z>.
- [20] E. HAIRER, S. NORSETT, AND G. WANNER, *Solving ordinary differential equations I: Nonstiff problems*, no. 8 in Springer Series in Computational Mathematics, Springer-Verlag Berlin Heidelberg, 1993.
- [21] W. HEINEKEN AND G. WARNECKE, *Partitioning methods for reaction–diffusion problems*, Applied numerical mathematics, 56 (2006), pp. 981–1000.
- [22] T. KATO AND T. KATAOKA, *Circuit analysis by a new multirate method*, Electrical Engineering in Japan, 126 (1999), pp. 55–62.
- [23] C. A. KENNEDY AND M. H. CARPENTER, *Diagonally implicit Runge–Kutta methods for ordinary differential equations. A review*, Tech. Report NASA/TM-2016-219173, NASA, 2016.
- [24] A. KVÆRNØ, *Stability of multirate Runge-Kutta schemes*, International Journal of Differential Equations and Applications, 1 (2000), pp. 97–105.
- [25] A. KVÆRNØ AND P. RENTROP, *Low order multirate Runge-Kutta methods in electric circuit simulation*, 1999, <citeseer.ist.psu.edu/629589.html>.
- [26] K. J. LEE, W. MCCORMICK, Q. OUYANG, AND H. L. SWINNEY, *Pattern formation by interacting chemical fronts*, Science, 261 (1993), pp. 192–194.
- [27] A. LOGG, *Multi-adaptive Galerkin methods for ODEs I*, SIAM Journal on Scientific Computing, 24 (2003), pp. 1879–1902.
- [28] A. RALSTON, *Runge–Kutta methods with minimum error bounds*, Mathematics of computation, 16 (1962), pp. 431–437.
- [29] J. RICE, *Split Runge-Kutta methods for simultaneous equations*, Journal of Research of the National Institute of Standards and Technology, 60 (1960).
- [30] A. SANDU AND E. CONSTANTINESCU, *Multirate Adams methods for hyperbolic equations*, Journal of Scientific Computing, 38 (2009), pp. 229–249, <https://doi.org/doi:10.1007/s10915-008-9235-3>, <http://dx.doi.org/10.1007/s10915-008-9235-3>.
- [31] A. SANDU AND E. CONSTANTINESCU, *Multirate time discretizations for hyperbolic partial differential equations*, in International Conference of Numerical Analysis and Applied Mathematics (ICNAAM 2009), vol. 1168-1 of American Institute of Physics (AIP) Conference Proceedings, 2009, pp. 1411–1414, <https://doi.org/10.1063/1.3241354>, <http://dx.doi.org/10.1063/1.3241354>.

- [32] A. SANDU AND M. GÜNTHER, *A generalized-structure approach to additive Runge-Kutta methods*, SIAM Journal on Numerical Analysis, 53 (2015), pp. 17–42, <https://doi.org/10.1137/130943224>, <http://dx.doi.org/10.1137/130943224>.
- [33] A. SARSHAR, P. TRANQUILLI, B. PICKERING, A. MCCALL, C. J. ROY, AND A. SANDU, *A numerical investigation of matrix-free implicit time-stepping methods for large {CFD} simulations*, Computers & Fluids, 159 (2017), pp. 53–63, <https://doi.org/10.1016/j.compfluid.2017.09.014>, <http://www.sciencedirect.com/science/article/pii/S0045793017303493>.
- [34] M. SOFRONIOU AND G. SPALETTA, *Construction of explicit Runge–Kutta pairs with stiffness detection*, Mathematical and Computer Modelling, 40 (2004), pp. 1157–1169.

Appendix A. Decoupled multirate GARK schemes.

We use the naming convention *FASTf-SLOWs p(p̂)[type]*, where p is the method order, \hat{p} is the embedded order, f is the number of stages in the fast base method, and s is the number of stages in the slow base method. Each component method is either explicit or implicit: FAST, SLOW $\in \{\text{EX,IM}\}$. We distinguish between methods of type A (optimized for accuracy and for better step size control) and methods of type S (optimized for simplicity and for stability), therefore $type \in \{\text{A,S}\}$.

A.1. MrGARK EX2-EX2 2(1)[A]. This explicit method uses a base method from [28] and has telescopic (6) and naturally adaptive (Definition 6.1) properties.

$$A^{\{f,f\}} = \begin{bmatrix} 0 & 0 \\ \frac{2}{3} & 0 \end{bmatrix}, \quad A^{\{s,s\}} = \begin{bmatrix} 0 & 0 \\ \frac{2}{3} & 0 \end{bmatrix}, \quad A^{\{f,s,1\}} = \begin{bmatrix} 0 & 0 \\ \frac{2}{3M} & 0 \end{bmatrix},$$

$$A^{\{f,s,\lambda\}} = \begin{bmatrix} \frac{3M^3 - 11M^2 + 20\lambda M - 20M - 20\lambda + 20}{20(M-1)M} & -\frac{M(3M-11)}{20(M-1)} \\ \frac{-3M^3 - 9M^2 + 60\lambda M - 20M - 60\lambda + 20}{60(M-1)M} & \frac{M(M+3)}{20(M-1)} \end{bmatrix}, \quad \lambda = 2, \dots, M,$$

$$A^{\{s,f,1\}} = \begin{bmatrix} 0 & 0 \\ -\frac{1}{3}(M-2)M & \frac{M^2}{3} \end{bmatrix}, \quad A^{\{s,f,\lambda\}} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad \lambda = 2, \dots, M,$$

$$b^{\{f\}} = b^{\{s\}} = \begin{bmatrix} \frac{1}{4} & \frac{3}{4} \end{bmatrix}^T, \quad \hat{b}^{\{f\}} = \hat{b}^{\{s\}} = \begin{bmatrix} 1 & 0 \end{bmatrix}^T.$$

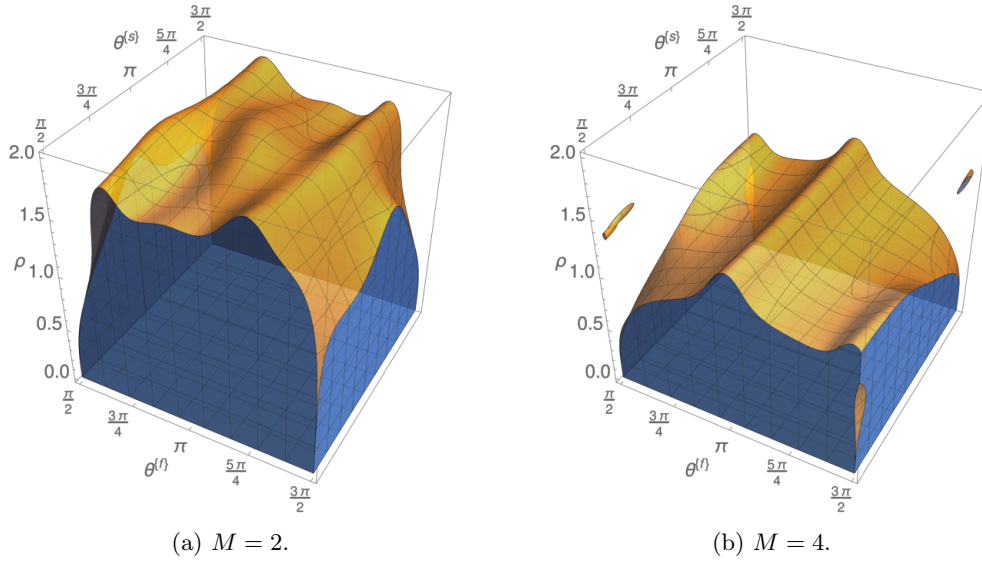


FIG. 13: EX2-EX2 2(1)[A] stability regions.

A.2. MrGARK EX2-EX2 2(1)[S]. This explicit method uses a two stage base method and has telescopic (6) and naturally adaptive (Definition 6.1) properties. The scheme computes the first slow stage, followed by L_2 fast steps, then the second slow stage, followed by $M - L_2$ fast steps. For example, we can take $L_2 = \text{floor}(c_2 M)$.

$$A^{\{f,f\}} = \begin{bmatrix} 0 & 0 \\ c_2 & 0 \end{bmatrix},$$

$$A^{\{s,s\}} = \begin{bmatrix} 0 & 0 \\ c_2 & 0 \end{bmatrix},$$

$$A^{\{f,s,\lambda\}} = \begin{bmatrix} \frac{\lambda-1}{M} & 0 \\ \frac{\lambda+c_2-1}{M} & 0 \end{bmatrix},$$

$$\lambda = 1, \dots, L_2,$$

$$A^{\{f,s,\lambda\}} = \begin{bmatrix} \frac{(\lambda-1)(2c_2-1)}{2Mc_2} & \frac{\lambda-1}{2Mc_2} \\ \frac{M}{3(L_2-M)} + \frac{-\lambda+2c_2(2\lambda+c_2-2)+1}{2Mc_2} & \frac{M}{3M-3L_2} + \frac{\lambda-1}{2Mc_2} + \frac{1-\lambda}{M} \end{bmatrix}, \quad \lambda = L_2 + 1, \dots, M,$$

$$A^{\{s,f,\lambda\}} = \begin{bmatrix} 0 & 0 \\ \frac{M(-2M+6c_2+3L_2-3)}{6L_2} & \frac{M(2M-3L_2+3)}{6L_2} \end{bmatrix},$$

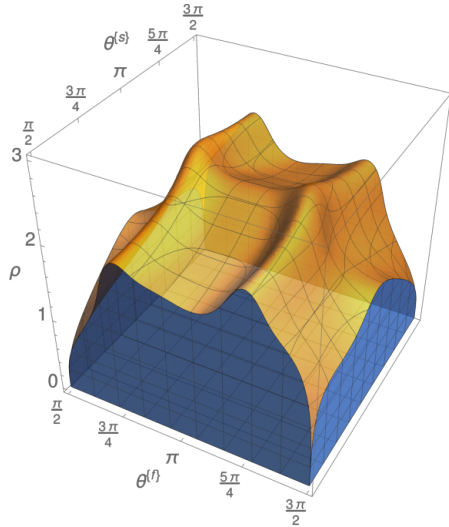
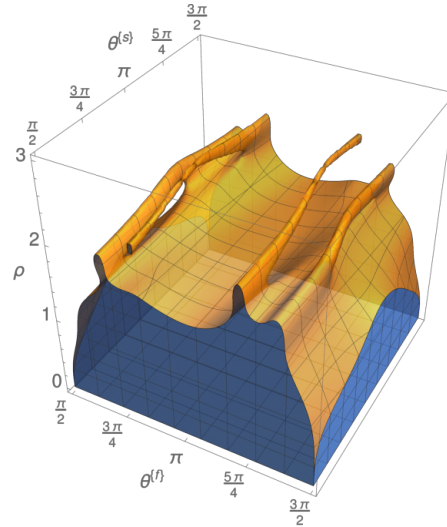
$$\lambda = 1, \dots, L_2,$$

$$A^{\{s,f,\lambda\}} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix},$$

$$\lambda = L_2 + 1, \dots, M,$$

$$b^{\{f\}} = b^{\{s\}} = \begin{bmatrix} \frac{2c_2-1}{2c_2} & \frac{1}{2c_2} \end{bmatrix}^T,$$

$$\widehat{b}^{\{f\}} = \widehat{b}^{\{s\}} = \begin{bmatrix} 1 & 0 \end{bmatrix}^T.$$

(a) $M = 2$.(b) $M = 4$.FIG. 14: EX2-EX2 2(1)[S] stability regions with $c_2 = \frac{2}{3}$.

A.3. MrGARK EX2-IM2 2(1)[A]. This explicit-implicit method uses the fast method from [28] and slow method from [1]. The multirate scheme is stiffly accurate (21) in the slow partition and the coupling error is independent of M .

$$A^{\{f,f\}} = \begin{bmatrix} 0 & 0 \\ \frac{2}{3} & 0 \end{bmatrix}, \quad A^{\{s,s\}} = \begin{bmatrix} 1 - \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & 1 - \frac{1}{\sqrt{2}} \end{bmatrix}, \quad A^{\{s,f,1\}} = \begin{bmatrix} M - \frac{M}{\sqrt{2}} & 0 \\ \frac{1}{4} & \frac{3}{4} \end{bmatrix},$$

$$A^{\{f,s,\lambda\}} = \begin{bmatrix} \frac{\lambda-1}{M} & 0 \\ \frac{3\lambda-1}{3M} & 0 \end{bmatrix}, \quad \lambda = 1, \dots, M,$$

$$A^{\{s,f,\lambda\}} = \begin{bmatrix} 0 & 0 \\ \frac{1}{4} & \frac{3}{4} \end{bmatrix}, \quad \lambda = 2, \dots, M,$$

$$b^{\{f\}} = \begin{bmatrix} \frac{1}{4} & \frac{3}{4} \end{bmatrix}^T, \quad b^{\{s\}} = \begin{bmatrix} \frac{1}{\sqrt{2}} & 1 - \frac{1}{\sqrt{2}} \end{bmatrix}^T,$$

$$\widehat{b}^{\{f\}} = \begin{bmatrix} 1 & 0 \end{bmatrix}^T, \quad \widehat{b}^{\{s\}} = \begin{bmatrix} \frac{3}{5} & \frac{2}{5} \end{bmatrix}^T.$$

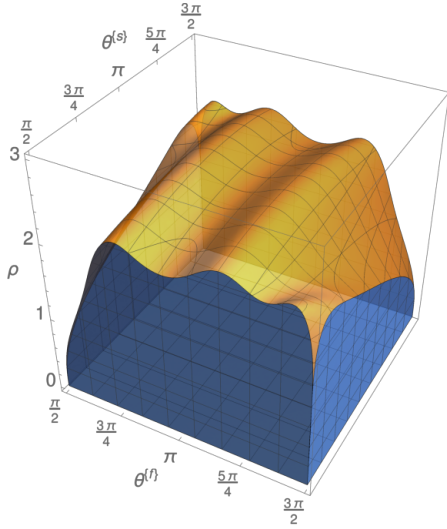
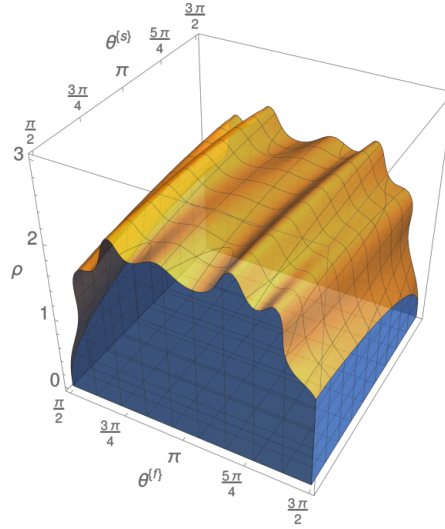
(a) $M = 2$.(b) $M = 4$.

FIG. 15: EX2-IM2 2(1)[A] stability regions.

A.4. MrGARK IM2-EX2 2(1)[A]. This implicit-explicit method uses the fast method from [1] and the slow method from [28]. The multirate scheme is stiffly accurate (21) in the fast partition.

$$A^{\{f,f\}} = \begin{bmatrix} 1 - \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & 1 - \frac{1}{\sqrt{2}} \end{bmatrix}, \quad A^{\{s,s\}} = \begin{bmatrix} 0 & 0 \\ \frac{2}{3} & 0 \end{bmatrix}, \quad A^{\{f,s,M\}} = \begin{bmatrix} \frac{2M-\sqrt{2}}{2M} & 0 \\ \frac{1}{4} & \frac{3}{4} \end{bmatrix},$$

$$A^{\{f,s,\lambda\}} = \begin{bmatrix} \frac{2\lambda-\sqrt{2}}{2M} & 0 \\ \frac{\lambda}{M} & 0 \end{bmatrix}, \quad \lambda = 1, \dots, M-1,$$

$$A^{\{s,f,\lambda\}} = \begin{bmatrix} 0 & 0 \\ \frac{2}{3} & 0 \end{bmatrix}, \quad \lambda = 1, \dots, M,$$

$$b^{\{f\}} = \left[\frac{1}{\sqrt{2}} \quad 1 - \frac{1}{\sqrt{2}} \right]^T, \quad b^{\{s\}} = \left[\frac{1}{4} \quad \frac{3}{4} \right]^T,$$

$$\widehat{b}^{\{f\}} = \left[\frac{3}{5} \quad \frac{2}{5} \right]^T, \quad \widehat{b}^{\{s\}} = \left[1 \quad 0 \right]^T,$$

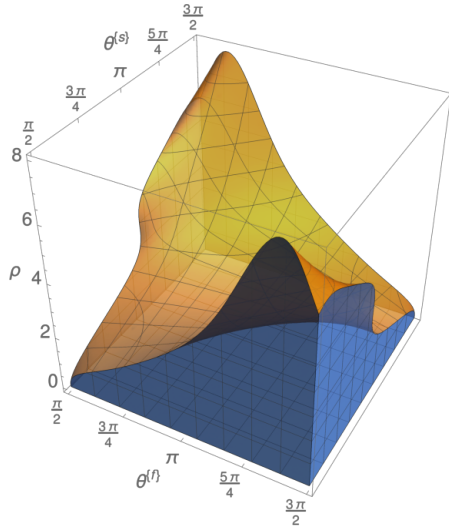
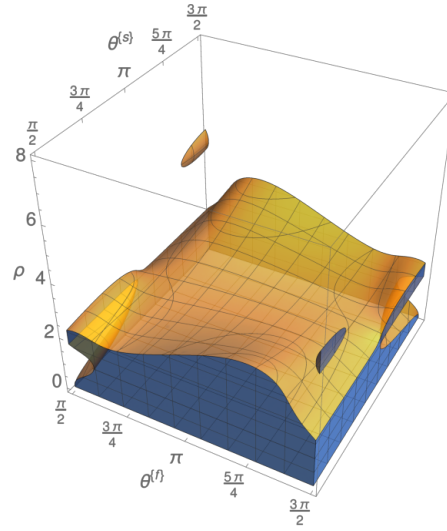
(a) $M = 2$.(b) $M = 4$.

FIG. 16: IM2-EX2 2(1)[A] stability regions.

A.5. MrGARK EX3-EX3 3(2)[A]. This explicit method uses a base method from [28] and has telescopic (6) properties. The coupling error is independent of M .

$$A^{(f,f)} = \begin{bmatrix} 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 \\ 0 & \frac{3}{4} & 0 \end{bmatrix}, \quad A^{(s,s)} = \begin{bmatrix} 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 \\ 0 & \frac{3}{4} & 0 \end{bmatrix}, \quad A^{(f,s,1)} = \begin{bmatrix} 0 & 0 & 0 \\ \frac{1}{2M} & 0 & 0 \\ 0 & \frac{3}{4M} & 0 \end{bmatrix},$$

$$A^{(f,s,\lambda)} = \begin{bmatrix} \frac{3M^2-8M^2+6\lambda M-6\lambda+6}{6(M-1)M} & \frac{-3M^2+8M-6}{6(M-1)} & 0 \\ \frac{-2M^2+6\lambda M-3M-6\lambda+3}{6(M-1)M} & \frac{M}{3(M-1)} & 0 \\ \frac{-3M^3+2M^2+12\lambda M-9M-12\lambda+12}{12(M-1)M} & \frac{3M^2-2M^2+6M-9}{12(M-1)M} & 0 \end{bmatrix}, \quad \lambda = 2, \dots, M,$$

$$A^{(s,f,1)} = \begin{bmatrix} 0 & 0 & 0 \\ -\frac{1}{66}M(16M-33) & \frac{8M^2}{33} & 0 \\ \frac{1}{384}(11M^4-22M^3+26M^2+11M+44) & \frac{1}{88}(-11M^4+22M^3-16M^2-11M+22) & \frac{1}{12}(M^4-2M^3+M^2+M+4) \end{bmatrix},$$

$$A^{(s,f,\lambda)} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{-M^4+2M^3+2M^2+3M-4}{24(M-1)} & \frac{1}{8}(M^3-M^2-M+2) & \frac{-M^4+2M^3-M^2+3M-4}{12(M-1)} \end{bmatrix}, \quad \lambda = 2, \dots, M,$$

$$b^{(f)} = b^{(s)} = \begin{bmatrix} \frac{2}{9} & \frac{1}{3} & \frac{4}{9} \end{bmatrix}^T, \quad \hat{b}^{(f)} = \hat{b}^{(s)} = \begin{bmatrix} \frac{1}{40} & \frac{37}{40} & \frac{1}{20} \end{bmatrix}^T.$$

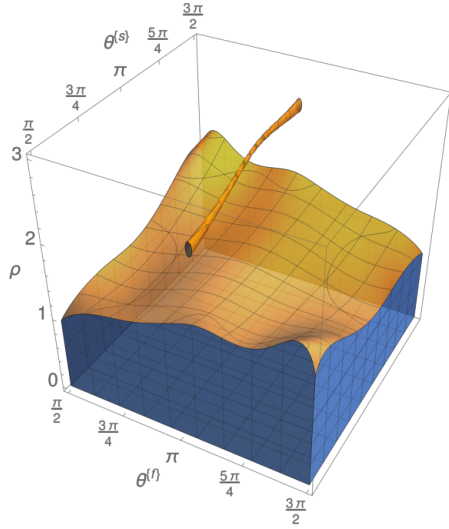
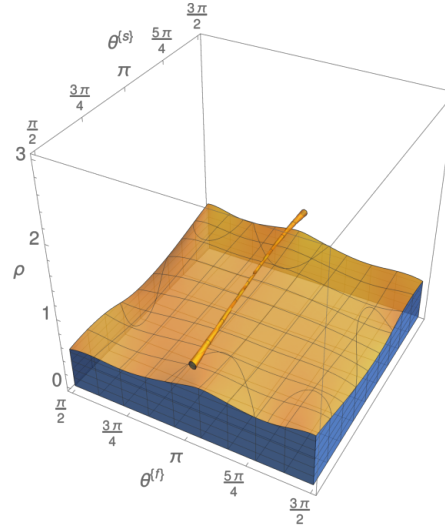
(a) $M = 2$.(b) $M = 4$.

FIG. 17: EX3-EX3 3(2)[A] stability regions.

A.6. MrGARK EX4-EX4 3(2)[A]. This explicit method is telescopic (6) and naturally adaptive (Definition 6.1).

$$\begin{aligned}
 A^{(f,f)} &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{1}{3} & 0 & 0 & 0 \\ 0 & \frac{2}{9} & 0 & 0 \\ \frac{833}{7680} & \frac{833}{9216} & \frac{3213}{5120} & 0 \end{bmatrix}, & A^{(s,s)} &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{1}{3} & 0 & 0 & 0 \\ 0 & \frac{2}{9} & 0 & 0 \\ \frac{833}{7680} & \frac{833}{9216} & \frac{3213}{5120} & 0 \end{bmatrix}, & A^{(s,f,\lambda)} &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, & \lambda &= 2, \dots, M, \\
 A^{(f,s,1)} &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{1}{3M} & 0 & 0 & 0 \\ \frac{5(518M^3-2140M^2+2399M-777)}{2331M(3M-4)} & -\frac{5(518M^3-2140M^2+1622M+259)}{2331M(3M-4)} & 0 & 0 \\ \frac{17(141932M^3-445231M^2+481160M-178710)}{852480M(3M-4)} & -\frac{17(94535M^3-228442M^2+142736M-5180)}{340992M(3M-4)} & \frac{3213M}{5120} & 0 \end{bmatrix}, \\
 A^{(f,s,\lambda)} &= \begin{bmatrix} \frac{\lambda-1}{M} & 0 & 0 & 0 \\ \frac{3\lambda-2}{3M} & 0 & 0 & 0 \\ \frac{777(12\lambda-7)+(6993\lambda+12092)M^2-5965M^3-3(5439\lambda+286)M}{2331M(3M^2-7M+4)} & \frac{5(1193M^3-3040M^2+1622M+259)}{2331M(3M^2-7M+4)} & 0 & 0 \\ \frac{222(3072\lambda-335)+(511488\lambda+1937719)M^2-867119M^3-72(16576\lambda+13973)M}{170496M(3M^2-7M+4)} & \frac{17(51007M^3-119207M^2+71368M-2590)}{170496M(3M^2-7M+4)} & 0 & 0 \end{bmatrix}, & \lambda &= 2, \dots, M, \\
 A^{(s,f,1)} &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{M}{3} - \frac{34M^2}{361} & \frac{34M^2}{361} & 0 & 0 \\ 0 & \frac{5(1805-981M)M}{6498} & \frac{5M(327M-361)}{2166} & 0 \\ \frac{M(1480461M^2-3944118M+3007130)}{2772480} & -\frac{119M(3249M^2-20358M+18050)}{3326976} & -\frac{119M(66063M^2-78954M-18050)}{5544960} & (M-1)M^2 \end{bmatrix},
 \end{aligned}$$

$$b^{(f)} = b^{(s)} = \begin{bmatrix} \frac{101}{714} & \frac{1}{3} & \frac{1}{6} & \frac{128}{357} \end{bmatrix}^T, \quad \widehat{b}^{(f)} = \widehat{b}^{(s)} = \begin{bmatrix} \frac{7}{40} & -\frac{425}{8784} & \frac{100037}{131760} & \frac{188}{1647} \end{bmatrix}^T.$$

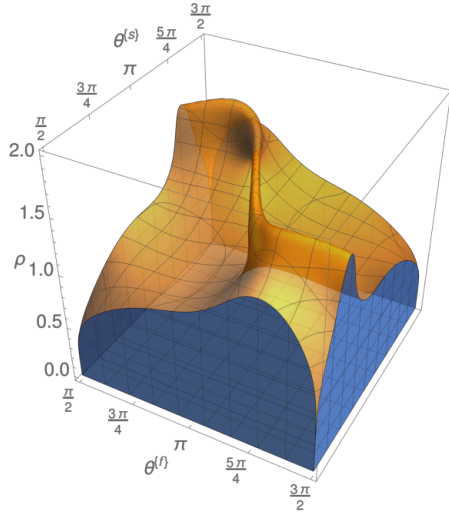
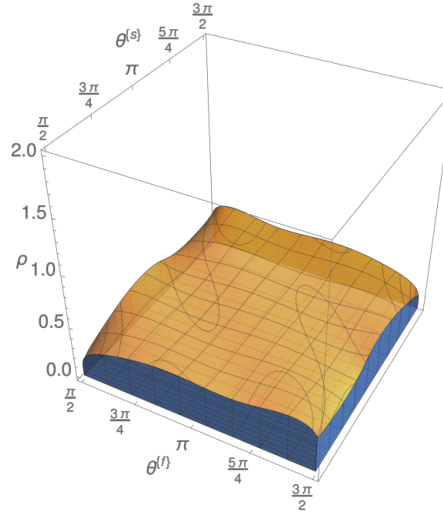
(a) $M = 2$.(b) $M = 4$.

FIG. 18: EX4-EX4 3(2)[A] stability regions.

A.7. MrGARK EX3-EX3 3(2)[S]. This explicit method uses a three stage base method and has telescopic (6) property. Once again, we can take $L_2 = \text{floor}(c_2 M)$.

$$\begin{aligned}
 A^{(f,f)} = A^{(s,s)} &= \begin{bmatrix} 0 & 0 & 0 \\ c_2 & 0 & 0 \\ \frac{(3c_2^2-3c_2+1)}{c_2(3c_2-2)} & \frac{(c_2-1)}{c_2(3c_2-2)} & 0 \end{bmatrix}, \\
 A^{(f,s,\lambda)} &= \begin{bmatrix} \frac{\lambda-1}{M} & 0 & 0 \\ \frac{c_2+\lambda-1}{M} & 0 & 0 \\ \frac{\lambda}{M} & 0 & 0 \end{bmatrix}, & \lambda = 1, \dots, L_2, \\
 A^{(f,s,\lambda)} &= \begin{bmatrix} \frac{2\lambda-1}{12c_2(L_2-M)} + \frac{1-2\lambda}{12c_2(L_2+M)} + \frac{2\lambda-1}{2M} & \frac{2\lambda-1}{12c_2(L_2+M)} + \frac{1-2\lambda}{12c_2(L_2-M)} - \frac{1}{2M} & 0 \\ \frac{2\lambda-1}{12c_2(L_2-M)} + \frac{1-2\lambda}{12c_2(L_2+M)} + \frac{2\lambda-1}{2M} & \frac{2\lambda-1}{12c_2(L_2+M)} + \frac{1-2\lambda}{12c_2(L_2-M)} + \frac{2c_2-1}{2M} & 0 \\ \frac{2\lambda-1}{12c_2(L_2-M)} + \frac{1-2\lambda}{12c_2(L_2+M)} + \frac{2\lambda-1}{2M} & \frac{2\lambda-1}{12c_2(L_2+M)} + \frac{1-2\lambda}{12c_2(L_2-M)} + \frac{1}{2M} & 0 \end{bmatrix}, & \lambda = L_2 + 1, \dots, M, \\
 A^{(s,f,\lambda)} &= \begin{bmatrix} 0 & 0 & 0 \\ c_2 \left(\frac{2\lambda(c_2(4L_2-3)-3L_2+3)}{(c_2-1)(3c_2^2+4c_2+1)(L_2+1)} + \frac{M}{L_2} \right) & -\frac{c_2\lambda(c_2(4L_2-3)-3L_2+3)}{(c_2-1)(3c_2^2+4c_2+1)(L_2+1)} & -\frac{c_2\lambda(c_2(4L_2-3)-3L_2+3)}{(c_2-1)(3c_2^2+4c_2+1)(L_2+1)} \\ \frac{2c_2\lambda(c_2(4L_2-3)-3L_2+3)}{(c_2-1)(3c_2^2+4c_2+1)(L_2+1)} & -\frac{c_2\lambda(c_2(4L_2-3)-3L_2+3)}{(c_2-1)(3c_2^2+4c_2+1)(L_2+1)} & -\frac{c_2\lambda(c_2(4L_2-3)-3L_2+3)}{(c_2-1)(3c_2^2+4c_2+1)(L_2+1)} \end{bmatrix}, & \lambda = 1, \dots, L_2, \\
 A^{(s,f,\lambda)} &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{\lambda}{3c_2-2} + \frac{c_2(3L_2-4)-3L_2+3}{6c_2-4} + \frac{M}{M-L_2} & \frac{\lambda}{2-3c_2} & \frac{c_2(4-3L_2)+3(L_2-1)}{6c_2-4} \end{bmatrix}, & \lambda = L_2 + 1, \dots, M, \\
 b^{(f)} = b^{(s)} &= \begin{bmatrix} \frac{3c_2-1}{6c_2} & -\frac{1}{6(c_2-1)c_2} & \frac{3c_2-2}{6(c_2-1)} \end{bmatrix}^T, \quad \hat{b}^{(f)} = \hat{b}^{(s)} = \begin{bmatrix} \hat{b}_2(c_2-1) + \frac{1}{2} & \hat{b}_2 & \frac{1-2\hat{b}_2c_2}{2} \end{bmatrix}^T.
 \end{aligned}$$

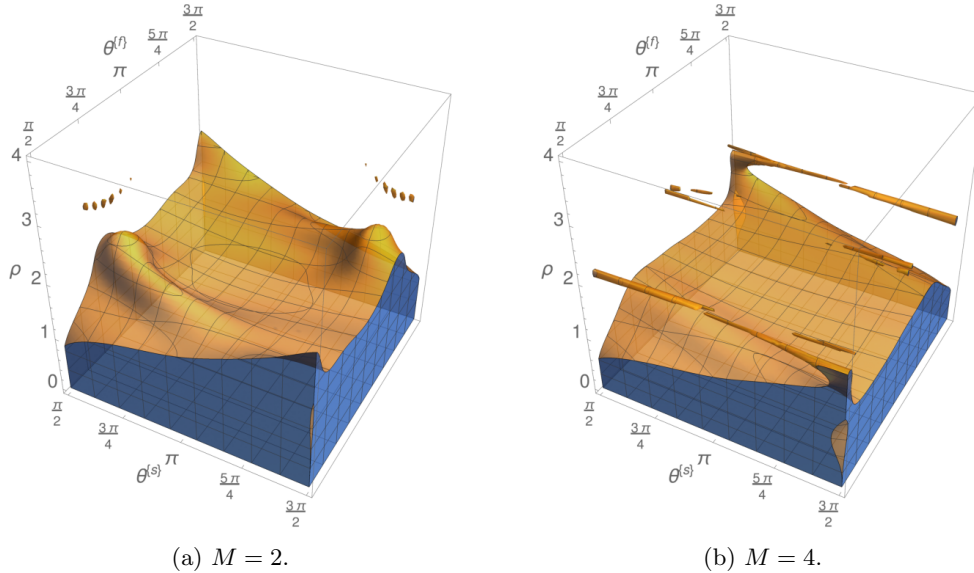


FIG. 19: Stability plots for MrGARK EX3-EX3 3(2)[S] method with $c_2 = \frac{1}{2}$.

A.8. MrGARK EX3-IM3 3(2)[A]. This explicit method uses a fast method from [28] and a slow method from [1] and is stiffly accurate (21) in the slow partition.

$$A^{(f,f)} = \begin{bmatrix} 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 \\ 0 & \frac{3}{4} & 0 \end{bmatrix}, \quad A^{(s,s)} = \begin{bmatrix} \gamma & 0 & 0 \\ -\frac{2(3\gamma^3-9\gamma^2+6\gamma-1)}{3(2\gamma^2-4\gamma+1)} & \gamma & 0 \\ \frac{4\gamma-1}{4(3\gamma^3-9\gamma^2+6\gamma-1)} & -\frac{3(2\gamma^2-4\gamma+1)^2}{4(3\gamma^3-9\gamma^2+6\gamma-1)} & \gamma \end{bmatrix},$$

$$A^{(f,s,\lambda)} = \begin{bmatrix} \frac{\lambda-1}{M} & 0 & 0 \\ \frac{2\lambda-1}{2M} & 0 & 0 \\ \frac{-60\lambda\gamma^3+42\gamma^3+18M\gamma^2+72\lambda\gamma^2-72\gamma^2-36M\gamma+42\lambda\gamma+3\gamma+9M-16\lambda+4}{16M(3\gamma^3-9\gamma^2+6\gamma-1)} & -\frac{9(2\gamma^2-4\gamma+1)(M+3\gamma-6\gamma\lambda)}{16M(3\gamma^3-9\gamma^2+6\gamma-1)} & 0 \end{bmatrix}, \quad \lambda = 1, \dots, M,$$

$$A^{(s,f,1)} = \begin{bmatrix} M\gamma & 0 & 0 \\ -\frac{M(36M\gamma^4-36\gamma^4-120M\gamma^3+126\gamma^3+108M\gamma^2-138\gamma^2-36M\gamma+51\gamma+4M-6)}{9(2\gamma^2-4\gamma+1)^2} & \frac{4M^2(9\gamma^4-30\gamma^3+27\gamma^2-9\gamma+1)}{9(2\gamma^2-4\gamma+1)^2} & 0 \\ \frac{2}{9} & \frac{1}{3} & \frac{4}{9} \end{bmatrix},$$

$$A^{(s,f,\lambda)} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{2}{9} & \frac{1}{3} & \frac{4}{9} \end{bmatrix}, \quad \lambda = 2, \dots, M,$$

$$b^{(f)} = \begin{bmatrix} \frac{2}{9} & \frac{1}{3} & \frac{4}{9} \end{bmatrix}^T, \quad b^{(s)} = \begin{bmatrix} \frac{4\gamma-1}{4(3\gamma^3-9\gamma^2+6\gamma-1)} & -\frac{3(2\gamma^2-4\gamma+1)^2}{4(3\gamma^3-9\gamma^2+6\gamma-1)} & \gamma \end{bmatrix}^T,$$

$$\widehat{b}^{(f)} = \begin{bmatrix} \frac{1}{40} & \frac{37}{40} & \frac{1}{20} \end{bmatrix}^T, \quad \widehat{b}^{(s)} = \begin{bmatrix} \frac{-6\gamma^2+6\gamma-1}{4(3\gamma^3-9\gamma^2+6\gamma-1)} & \frac{3(4\gamma^3-10\gamma^2+6\gamma-1)}{4(3\gamma^3-9\gamma^2+6\gamma-1)} & 0 \end{bmatrix}^T,$$

$$\gamma = \frac{1}{2} (2 + \sqrt{6} \sin(\frac{1}{3} \cot^{-1} 2\sqrt{2}) - \sqrt{2} \cos(\frac{1}{3} \cot^{-1} 2\sqrt{2})) \approx 0.43586652150845899942.$$

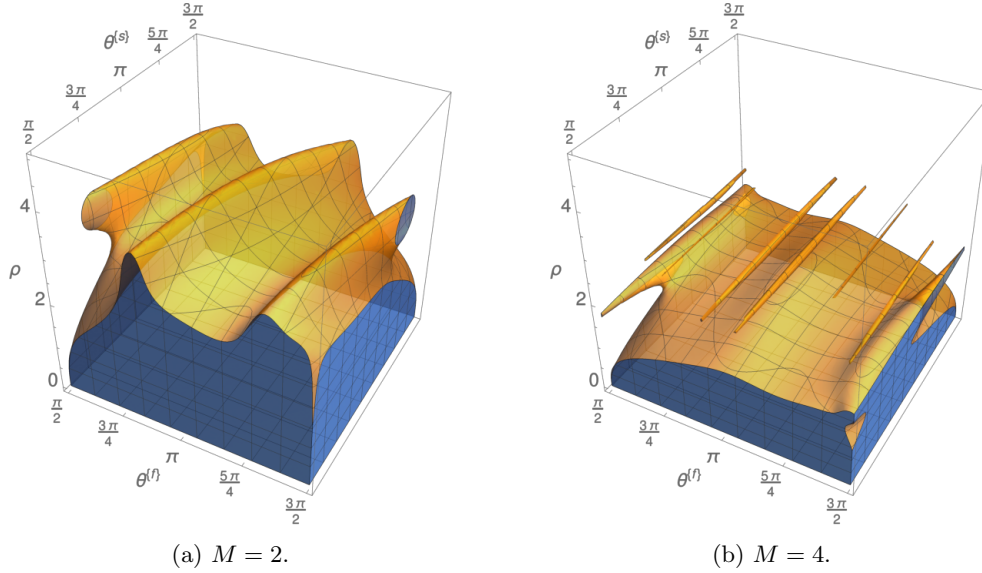


FIG. 20: EX3-IM3 3(2)[A] stability regions.

A.9. MrGARK IM3-EX3 3(2)[A]. This implicit-explicit method uses a fast method from [1] and a slow method from [28] and is stiffly accurate (21) in the fast partition.

$$A^{\{f,f\}} = \begin{bmatrix} \gamma & 0 & 0 \\ -\frac{2(3\gamma^3-9\gamma^2+6\gamma-1)}{3(2\gamma^2-4\gamma+1)} & \gamma & 0 \\ \frac{4\gamma-1}{4(3\gamma^3-9\gamma^2+6\gamma-1)} & -\frac{3(2\gamma^2-4\gamma+1)^2}{4(3\gamma^3-9\gamma^2+6\gamma-1)} & \gamma \end{bmatrix}, \quad A^{\{s,s\}} = \begin{bmatrix} 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 \\ 0 & \frac{3}{4} & 0 \end{bmatrix},$$

$$A^{\{f,s,\lambda\}} = \begin{bmatrix} \frac{\gamma+\lambda-1}{M} & 0 & 0 \\ \frac{6\lambda\gamma^2-12\lambda\gamma+3\gamma+3\lambda-1}{3M(2\gamma^2-4\gamma+1)} & 0 & 0 \\ \frac{\lambda}{M} & 0 & 0 \end{bmatrix}, \quad \lambda = 1, \dots, M-1,$$

$$A^{\{f,s,M\}} = \begin{bmatrix} \frac{M+\gamma-1}{M} & 0 & 0 \\ \frac{12M^2\gamma^3-36M\gamma^3+18\gamma^3-36M^2\gamma^2+108M\gamma^2-42\gamma^2+24M^2\gamma-60M\gamma+21\gamma-4M^2+9M-3}{9M(2\gamma^2-4\gamma+1)^2} & -\frac{4(M-3\gamma)(3\gamma^3-9\gamma^2+6\gamma-1)}{9(2\gamma^2-4\gamma+1)^2} & 0 \\ \frac{2}{9} & \frac{1}{3} & \frac{4}{9} \end{bmatrix},$$

$$A^{\{s,f,\lambda\}} = \begin{bmatrix} 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 \\ -\frac{3(12\gamma^3+6M\gamma^2-18\gamma^2-12M\gamma+6\gamma+3M-1)}{32(3\gamma^3-9\gamma^2+6\gamma-1)} & \frac{9(M+6\gamma-3)(2\gamma^2-4\gamma+1)}{32(3\gamma^3-9\gamma^2+6\gamma-1)} & 0 \end{bmatrix}, \quad \lambda = 1, \dots, M,$$

$$b^{\{f\}} = \begin{bmatrix} \frac{4\gamma-1}{4(3\gamma^3-9\gamma^2+6\gamma-1)} & -\frac{3(2\gamma^2-4\gamma+1)^2}{4(3\gamma^3-9\gamma^2+6\gamma-1)} & \gamma \end{bmatrix}^T, \quad b^{\{s\}} = \begin{bmatrix} \frac{2}{9} & \frac{1}{3} & \frac{4}{9} \end{bmatrix}^T,$$

$$\widehat{b}^{\{f\}} = \begin{bmatrix} \frac{-6\gamma^2+6\gamma-1}{4(3\gamma^3-9\gamma^2+6\gamma-1)} & \frac{3(4\gamma^3-10\gamma^2+6\gamma-1)}{4(3\gamma^3-9\gamma^2+6\gamma-1)} & 0 \end{bmatrix}^T, \quad \widehat{b}^{\{s\}} = \begin{bmatrix} \frac{1}{40} & \frac{37}{40} & \frac{1}{20} \end{bmatrix}^T.$$

$$\gamma = \frac{1}{2} (2 + \sqrt{6} \sin(\frac{1}{3} \cot^{-1} 2\sqrt{2}) - \sqrt{2} \cos(\frac{1}{3} \cot^{-1} 2\sqrt{2})) \approx 0.43586652150845899942.$$

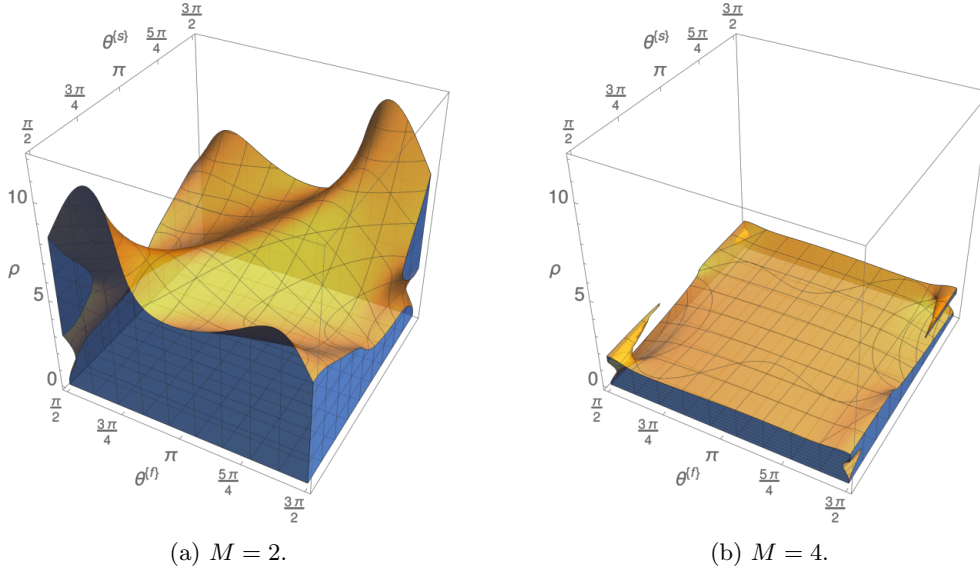


FIG. 21: IM3-EX3 3(2)[A] stability regions.

A.10. MrGARK EX5-EX5 4(3)[A]. This explicit method uses a base method from [34], is telescopic (6) and has the First same as last (FSAL) property.

$$\begin{aligned}
 A^{(5,1)} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ \frac{2}{5} & 0 & 0 & 0 & 0 \\ -\frac{19}{44} & \frac{3}{4} & 0 & 0 & 0 \\ \frac{19}{44} & -\frac{15}{44} & \frac{10}{11} & 0 & 0 \\ \frac{11}{72} & \frac{25}{72} & \frac{11}{72} & 0 & 0 \end{bmatrix}, & A^{(5,s)} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ \frac{2}{5} & 0 & 0 & 0 & 0 \\ -\frac{19}{44} & \frac{3}{4} & 0 & 0 & 0 \\ \frac{19}{44} & -\frac{15}{44} & \frac{10}{11} & 0 & 0 \\ \frac{11}{72} & \frac{25}{72} & \frac{11}{72} & 0 & 0 \end{bmatrix}, & A^{(5,\lambda)} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \frac{11}{72} & \frac{25}{72} & \frac{11}{72} & 0 & 0 \end{bmatrix}, & \lambda &= 2, \dots, M, \\
 A^{(5,s,1)} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ \frac{2}{5M} & 0 & 0 & 0 & 0 \\ \frac{3(10M^2-30M^2+22M-1)}{20M(3M-4)} & -\frac{3(2M^2-6M^2+2M+3)}{4M(3M-4)} & 0 & 0 & 0 \\ 0 & \frac{3(10M^2-50M^2+116M-83)}{22M(3M-4)} & -\frac{30M^2+150M^2-282M+161}{22M(3M-4)} & 0 & 0 \\ \frac{11}{72M} & \frac{25}{72M} & \frac{11}{72M} & 0 & 0 \end{bmatrix}, \\
 A^{(5,\lambda,1)} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ \frac{11(\lambda-1)}{72M} & \frac{25(\lambda-1)}{72M} & \frac{11(\lambda-1)}{72M} & 0 & 0 \\ -\frac{450M^2+966M-497M-966\lambda+776}{430(M-1)M} & \frac{450M^2-506\lambda M+227M+506\lambda-506}{430(M-1)M} & 0 & 0 & 0 \\ -\frac{900M^2+1239\lambda M^2+2217M^2-2891\lambda M-97M+1652\lambda-1562}{600M(3M^2-7M+4)} & \frac{900M^2+56\lambda M^2-2937M^2-1309\lambda M+1777M+748\lambda+602}{600M(3M^2-7M+4)} & \frac{3240M^2-825\lambda M^2-7455M^2+1025\lambda M+8227M-1100\lambda-4696}{792M(3M^2-7M+4)} & 0 & 0 \\ 0 & \frac{33\lambda}{72M} & \frac{33\lambda}{72M} & 0 & 0 \end{bmatrix}, & \lambda &= 2, \dots, M, \\
 A^{(5,s,1)} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ \frac{2M}{5} & 0 & 0 & 0 & 0 \\ -\frac{3}{20}M(5M-4) & \frac{3M^2}{4} & 0 & 0 & 0 \\ \frac{1}{14}M(56M^2-81M+44) & -\frac{5}{11}M^2(16M-13) & -\frac{5}{11}(M-3)M^2 & (M-1)M^2 & 0 \\ \frac{11}{72} & \frac{25}{72} & \frac{11}{72} & 0 & 0 \end{bmatrix}, \\
 b^{(1)} = b^{(s)} &= \begin{bmatrix} \frac{11}{72} & \frac{25}{72} & \frac{11}{72} & 0 \\ \frac{1251515}{8070912} & \frac{3710105}{8070912} & \frac{2519695}{8070912} & \frac{51105}{8070912} \\ \frac{19001}{747516} & \frac{11001}{747516} & \frac{11001}{747516} & 0 \end{bmatrix}^T, & \hat{b}^{(1)} = \hat{b}^{(s)} &= \begin{bmatrix} \frac{11}{72} & \frac{25}{72} & \frac{11}{72} & 0 \\ \frac{1251515}{8070912} & \frac{3710105}{8070912} & \frac{2519695}{8070912} & \frac{51105}{8070912} \\ \frac{19001}{747516} & \frac{11001}{747516} & \frac{11001}{747516} & 0 \end{bmatrix}^T.
 \end{aligned}$$

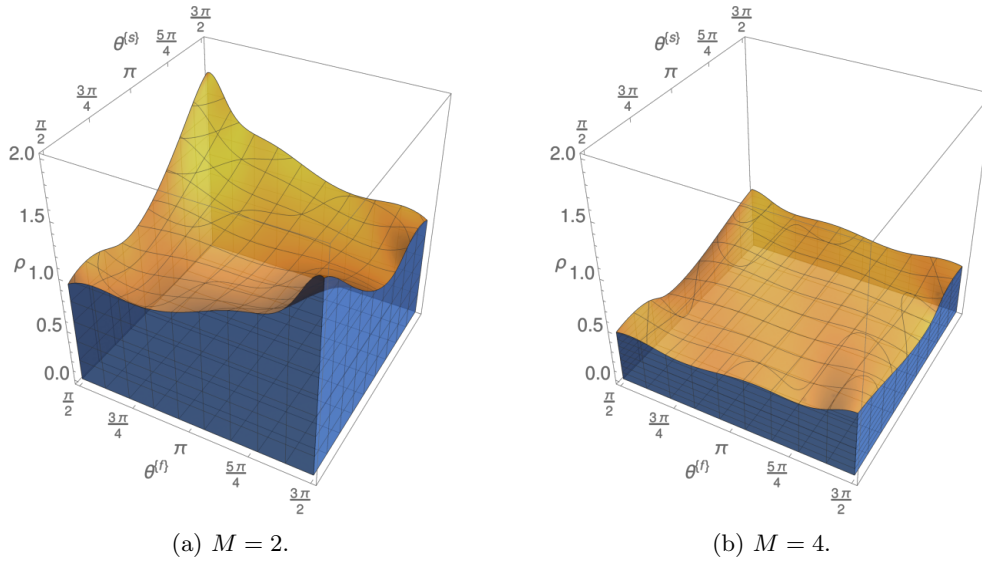


FIG. 22: EX5-EX5 4(3)[A] stability regions.

A.11. MrGARK EX6-IM5 4(3)[A]. This explicit-implicit method uses a fast method from [13] and a slow method from [23]. The multirate scheme is stiffly accurate (21) in the slow partition.

$$\begin{aligned}
A(f) &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{32}{2107} & \frac{2}{2107} & \frac{7296}{2107} & 0 & 0 & 0 & 0 \\ \frac{439}{216} & -8 & \frac{3680}{2107} & 0 & 0 & 0 & 0 \\ -\frac{8}{27} & 2 & -\frac{3544}{2565} & \frac{1859}{4104} & -\frac{11}{40} & 0 & 0 \end{bmatrix}, & A(g^*) &= \begin{bmatrix} \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{13}{20} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 \\ \frac{3727}{3727} & -\frac{575}{3727} & \frac{1}{4} & 0 & 0 & 0 & 0 \\ \frac{12698}{37275} & -\frac{201}{11500} & \frac{801}{11500} & \frac{1}{4} & 0 & 0 & 0 \\ \frac{944}{1365} & -\frac{400}{819} & \frac{99}{35} & -\frac{575}{252} & \frac{1}{4} & 0 & 0 \end{bmatrix}, \\
A(f^*) &= \begin{bmatrix} \frac{M}{4} & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{100}{169}M(169M-90) & \frac{169M^2}{100} & 0 & 0 & 0 & 0 & 0 \\ -\frac{108}{135}M(155M-132) & \frac{135M^2}{108} & 0 & 0 & 0 & 0 & 0 \\ -\frac{1}{820}M(497M-552) & \frac{141M^2}{23} & -\frac{806M^2}{10925} & \frac{1183M^2}{87400} & 0 & 0 & 0 \\ \frac{25}{216} & 0 & \frac{1408}{2565} & \frac{2107}{4104} & -\frac{1}{5} & 0 & 0 \end{bmatrix}, \\
A(f^{\ast,\lambda}) &= \begin{bmatrix} \frac{\lambda-1}{M} & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{4\lambda-3}{4M} & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{45M^2-90M^2+551\lambda M-335M+90\lambda-90}{4163M^2} & -\frac{15(3M^3-6M^2+9\lambda M-5M+6\lambda-6)}{4163M^2} & 0 & 0 & 0 & 0 & 0 \\ \frac{1440M^3-2880M^2+6517\lambda M-3709M-3960\lambda+3960}{2197M^2} & -\frac{60(24M^3-48M^2+72\lambda M-50M-66\lambda+66)}{2197M^2} & 0 & 0 & 0 & 0 & 0 \\ \frac{560M^2-362M^2+386\lambda M-529M-1155\lambda+1155}{2183M^2} & -\frac{5(672M^3-2439M^2+10405\lambda M-2590M-1386\lambda+1386)}{1638M^2} & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \\
A(g^{\ast,\lambda}) &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{25}{216} & 0 & \frac{1408}{2565} & \frac{2107}{4104} & -\frac{1}{5} & 0 & 0 \end{bmatrix}, & \lambda &= 2, \dots, M, \\
b(f) &= \begin{bmatrix} \frac{25}{216} & 0 & \frac{1408}{2565} & \frac{2107}{4104} & -\frac{1}{5} & 0 \end{bmatrix}^T, & b(g^*) &= \begin{bmatrix} \frac{944}{1365} & -\frac{400}{819} & \frac{99}{35} & -\frac{575}{252} & \frac{1}{4} \end{bmatrix}^T, \\
\widehat{b}(f) &= \begin{bmatrix} \frac{16}{135} & 0 & \frac{6556}{12825} & \frac{28561}{56430} & -\frac{9}{50} & \frac{2}{55} \end{bmatrix}^T, & \widehat{b}(g^*) &= \begin{bmatrix} \frac{41911}{60060} & -\frac{83975}{144174} & \frac{3363}{1120} & -\frac{27025}{11088} & \frac{103}{352} \end{bmatrix}^T.
\end{aligned}$$

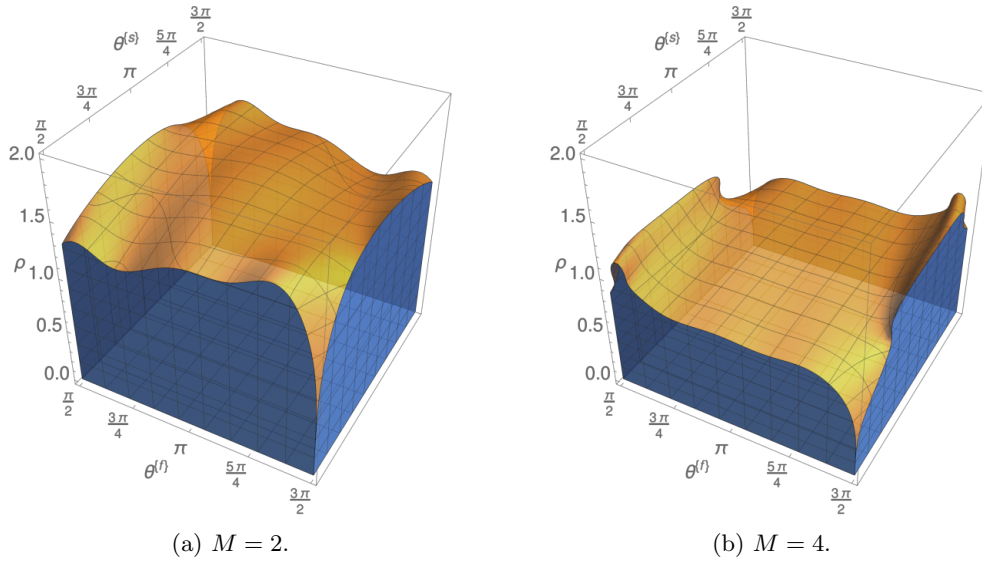


FIG. 23: EX6-IM5 4(3)[A] stability regions.

A.12. MrGARK IM6-EX4 4(2)[A]. This implicit-explicit method uses a slow method from [34] and is stiffly accurate (21) in the fast partition.

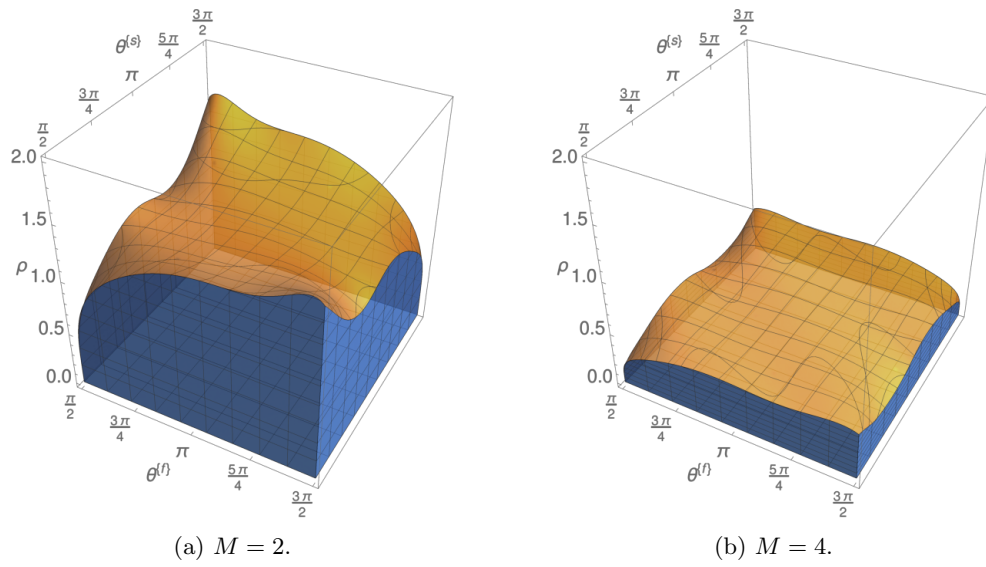


FIG. 24: IM6-EX4 4(2)[A] stability regions.