PRACTICAL MULTIRATE TIME INTEGRATION METHODS

Steven Roberts
Advisor: Dr. Adrian Sandu
Virginia Polytechnic Institute and State University

Abstract

Multirate methods are designed to efficiently solve systems of ordinary differential equations that have fast and slow dynamics. This is done by using different timesteps for different parts of a system. In this work, we develop a new theoretical framework for constructing and analyzing multirate methods based on general linear methods. The newly derived methods strike a balance between stability and accuracy that cannot be achieved by existing multirate methods based on the popular Runge–Kutta and linear multistep methods. Numerical experiments are performed to confirm the theoretical properties of the methods. Finally, we introduce coupled multirate infinitesimal methods, which offer a different perspective on multirate methods and have great stability and flexibility in implementation.

Introduction

In this report, we consider the autonomous system of ordinary differential equations (ODEs)

$$y' = f(y), \qquad y(t_0) = y_0, \tag{1}$$

where $f\colon \mathbb{R}^d \to \mathbb{R}^d$. Equations of this form are ubiquitous in the modeling and simulation of physical phenomena such as fluid flow, chemical reactions, and electric circuits. For these reasons, NASA has a long history of developing methods to solve eq. (1).[1,4–7,10–12,14,15,21] Some of the most common families of method for numerically solving ODEs are the Runge–Kutta and linear multistep methods. One limitation of these methods is that they use a global timestep to move from one state to the next. Many systems of practical interest combine complex processes that move on vastly different time scales. The fastest dynamics of the system impose a relatively small global step size even if they comprise a small portion of the system. Also, there may be optimal time-stepping methods for individual parts of a system but no satisfactory method for the whole problem.

Instead of treating f entirely as a black box, it can be additively partitioned into subfunctions:

$$y' = f(y) = \sum_{m=1}^{N} f^{\{m\}}(y). \tag{2}$$

Partitioned methods use different base methods for the different partitions, allowing for a more targeted and efficient method overall. Some types include implicit-explicit (IMEX) and multirate methods, the latter of which will be the primary focus of this paper. Multirate methods use different timesteps for each $f^{\{m\}}$ to overcome the aforementioned challenges faced by singlerate methods. The idea traces back to the 1960s[17] and has struggled to gain traction outside a small number of fields such as electric circuit simulation. The lack of widespread adoption is due to several factors. Deriving multirate methods is significantly

more challenging than singlerate methods, especially at high orders. In previous work,[20] we derive multirate Runge–Kutta methods up to order four, but extending this to fifth order and higher starts to become very challenging due to large number of order conditions. Deriving multirate linear multistep methods is much easier[13], but the so-called Dahlquist barriers limit their stability at high orders.[8,9] In addition, multirate methods introduce coupling error into the numerical solution, which if not properly controlled, ruins any speedup benefits.

We present new types of multirate methods designed to overcome these limitations. First, we discuss partitioned general linear method as a theoretical foundation for the primary focus of the report: multirate general linear methods. New multirate general linear methods up to order four are derived and tested. Finally, coupled multirate infinitesimal methods are described. For brevity, we do not include the full analyses and omit theorem proofs in this report. Interested readers can find the full details in corresponding papers.

Partitioned General Linear Methods

General linear methods (GLMs) present a large and flexible family of methods for solving eq. (1). One step is given by:

$$Y_i^{[n]} = H\sum_{j=1}^{s} a_{i,j} f\left(Y_j^{[n]}\right) + \sum_{j=1}^{r} u_{i,j}\xi_j^{[n-1]}, \quad i = 1,\dots,s, \tag{3a}$$

$$\xi_i^{[n]} = H\sum_{j=1}^{s} b_{i,j} f\left(Y_j^{[n]}\right) + \sum_{j=1}^{r} v_{i,j}\xi_j^{[n-1]}, \quad i = 1,\dots,r. \tag{3b}$$

The internal stage $Y_i^{[n]}$ approximates the solution $y(t_{n-1} + c_i H)$ and the external stage $\xi_i^{[n]}$ contains derivative information of the solution:

$$\xi_i^{[n-1]} = \sum_{k=0}^{p} q_{i,k} H^k y^{(k)}(t_{n-1}) + \mathcal{O}\big(H^{p+1}\big). \tag{3c}$$

With high stage order, the order conditions for GLMs are relatively simple.[2,3] Further, they can still have excellent stability properties. For these reasons, GLMs can serve as a solid foundation for practical multirate time integration methods.

The theory for partitioned GLMs is relatively new and is mostly limited to IMEX methods.[22] This existing work is not general enough to describe multirate GLMs. Instead of directly developing the order conditions, stability analysis, and theory for multirate GLMs, we first describe a very general framework for partitioned GLMs. Much of the results on multirate GLMs will follow as simple special cases.

Method Definition

One step of a partitioned GLM reads:

$$Y_i^{\{\sigma\}[n]} = H\sum_{\nu=1}^{N}\sum_{j=1}^{s^{\{\nu\}}} a_{i,j}^{\{\sigma,\nu\}} f^{\{\nu\}}\left(Y_j^{\{\nu\}[n]}\right) + \sum_{\nu=1}^{N}\sum_{j=1}^{r^{\{\nu\}}} u_{i,j}^{\{\sigma,\nu\}}\xi_j^{\{\nu\}[n-1]}, \quad i = 1,\dots,s^{\{\sigma\}}, \quad \sigma = 1,\dots,N, \tag{4a}$$

$$\xi_i^{\{\sigma\}[n]} = H\sum_{\nu=1}^{N}\sum_{j=1}^{s^{\{\nu\}}} b_{i,j}^{\{\sigma,\nu\}} f^{\{\nu\}}\left(Y_j^{\{\nu\}[n]}\right) + \sum_{\nu=1}^{N}\sum_{j=1}^{r^{\{\nu\}}} v_{i,j}^{\{\sigma,\nu\}}\xi_j^{\{\nu\}[n-1]}, \quad i = 1,\dots,r^{\{\sigma\}}, \quad \sigma = 1,\dots,N. \tag{4b}$$

Compared to eq. (3), both the internal and external stages are now partitioned and combine information from all other partitions. Similar to the unpartitioned setting, the internal stage $Y_i^{\{\sigma\}[n]}$ approximates the solution $y(t_{n-1} +$

$c_i^{\{\sigma\}} H$) and the external stage $\xi_i^{\{\sigma\}[n]}$ contains derivative information of the solution:

$$\xi_i^{\{\sigma\}[n]} = \sum_{\nu=1}^{N} \sum_{k=0}^{p} q_{i,k}^{\{\sigma,\nu\}} H^k \frac{\mathrm{d}^k y^{\{\nu\}}}{\mathrm{d}t^k}(t_{n-1}) + \mathcal{O}\big(H^{p+1}\big). \tag{5}$$

The coefficients can be organized into an extended tableau shown in fig. 1.

Order Conditions

Here, we consider the conditions on the method coefficients that ensure a method is of a certain order of accuracy. It is often the case that higher order methods are more efficient, but they are more challenging to derive.

First, we consider the conditions required for eq. (4) to solve the simple system

$$y' = 0, \qquad y(t_0) = \kappa = \sum_{\sigma=1}^{N} \kappa^{\{\sigma\}}. \tag{6}$$

When a partitioned GLM is applied to eq. (6), the internal stages are

$$Y^{\{\sigma\}[n]} = \sum_{\mu=1}^{N} \left(\sum_{\nu=1}^{N} \mathbf{U}^{\{\sigma,\nu\}} \mathbf{q}_0^{\{\nu,\mu\}} \right) \kappa^{\{\mu\}} + \mathcal{O}(H),$$

and the external stages are

$$\xi^{\{\sigma\}[n]} = \sum_{\mu=1}^{N} \left(\sum_{\nu=1}^{N} \mathbf{V}^{\{\sigma,\nu\}} \mathbf{q}_0^{\{\nu,\mu\}} \right) \kappa^{\{\mu\}} + \mathcal{O}(H).$$

The exact solution to eq. (6) is $y(t) = \kappa$, so we expect a partitioned GLM to satisfy $Y^{\{\sigma\}[n]} = \kappa \mathbb{1}_{s^{\{\sigma\}}} + \mathcal{O}(H)$ and $\xi^{\{\sigma\}[n]} = \sum_{\nu=1}^{N} \mathbf{q}^{\{\sigma,\nu\}} \kappa^{\{\nu\}} + \mathcal{O}(H)$. This motivates the following definition.

Definition 1 *A partitioned GLM is preconsistent if*

$$\sum_{\nu=1}^{N} \mathbf{U}^{\{\sigma,\nu\}} \mathbf{q}_0^{\{\nu,\mu\}} = \mathbb{1}_{s^{\{\sigma\}}},$$

$$\sum_{\nu=1}^{N} \mathbf{V}^{\{\sigma,\nu\}} \mathbf{q}_0^{\{\nu,\mu\}} = \mathbf{q}_0^{\{\sigma\}},$$

for $\sigma, \mu = 1, \ldots, N$.

Theorem 1 *Assume that* $\xi_i^{\{\sigma\}[n-1]}$ *satisfies eq.* (5). *A preconsistent partitioned GLM eq.* (4) *has order* p *and stage order* $q = p - 1$ *or* $q = p$ *if and only if for all* $\sigma, \nu = 1, \ldots, N$*:*

$$0 = \frac{\mathbf{c}^{\{\sigma\}\times k}}{k!} - \frac{\mathbf{A}^{\{\sigma,\nu\}} \mathbf{c}^{\{\nu\}\times(k-1)}}{(k-1)!} - \sum_{\mu=1}^{N} \mathbf{U}^{\{\sigma,\mu\}} \mathbf{q}_k^{\{\mu,\nu\}}, \quad k = 1, \ldots, q,$$

$$0 = \sum_{\ell=0}^{k} \frac{\mathbf{q}_{k-\ell}^{\{\sigma,\nu\}}}{\ell!} - \frac{\mathbf{B}^{\{\sigma,\nu\}} \mathbf{c}^{\{\nu\}\times(k-1)}}{(k-1)!} - \sum_{\mu=1}^{N} \mathbf{V}^{\{\sigma,\mu\}} \mathbf{q}_k^{\{\mu,\nu\}}, \quad k = 1, \ldots, p.$$

Linear Stability

A standard assessment of a time integration method is the linear stability analysis, which considers the amplification of errors from step to step for a scalar, linear test problem. In the context of partitioned methods, the standard test problem generalization is

$$y' = \sum_{\sigma=1}^{N} \lambda^{\{\sigma\}} y,$$

where each $\lambda^{\{\sigma\}} \in \mathbb{C}$. Applying this problem to eq. (4) yields

$$\begin{aligned} \xi^{[n]} &= \mathbf{M}(\mathbf{Z}) \xi^{[n-1]}, \\ \mathbf{M}(\mathbf{Z}) &\coloneqq \mathbf{V} + \mathbf{B}\mathbf{Z}(I - \mathbf{A}\mathbf{Z})^{-1}\mathbf{U}, \end{aligned} \tag{7}$$

where

$$\mathbf{Z} = \operatorname{diag}\left(z^{\{1\}} I, \ldots, z^{\{N\}} I\right),$$

and each $z^{\{\sigma\}} = H\lambda^{\{\sigma\}}$.

Definition 2 *The stability region of a partitioned GLM is the set*

$$\left\{ z^{\{1\}}, \ldots, z^{\{N\}} \in \mathbb{C} : \mathbf{M}(\mathbf{Z}) \textit{ power bounded} \right\}.$$

A large stability region is a desirable property as it allows for stiffer problems to be solved without error accumulation ruining the solution.

$$\left[\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{U} & \mathbf{V} \end{array}\right] := \left[\begin{array}{ccc|ccc} \mathbf{A}^{\{1,1\}} & \cdots & \mathbf{A}^{\{1,N\}} & \mathbf{U}^{\{1,1\}} & \cdots & \mathbf{U}^{\{1,N\}} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}^{\{N,1\}} & \cdots & \mathbf{A}^{\{N,N\}} & \mathbf{U}^{\{N,1\}} & \cdots & \mathbf{U}^{\{N,N\}} \\ \hline \mathbf{B}^{\{1,1\}} & \cdots & \mathbf{B}^{\{1,N\}} & \mathbf{V}^{\{1,1\}} & \cdots & \mathbf{V}^{\{1,N\}} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{B}^{\{N,1\}} & \cdots & \mathbf{B}^{\{N,N\}} & \mathbf{V}^{\{N,1\}} & \cdots & \mathbf{V}^{\{N,N\}} \end{array}\right].$$

Figure 1: Tableau for a partitioned GLM.

Partitioned GLM Example

To demonstrate the flexibility of this new partitioned GLM framework, we present a method that combines the third order Adams–Bashforth method with the third order Radau IIA method:

$$\left[\begin{array}{ccc|cccc} 0 & 0 & 0 & 1 & \frac{23}{12} & -\frac{4}{3} & \frac{5}{12} & 1 \\ \hline \frac{17}{20} & \frac{5}{12} & -\frac{1}{12} & 1 & \frac{839}{1620} & -\frac{49}{1620} & -\frac{7}{1620} & 1 \\ 2 & \frac{3}{4} & \frac{1}{4} & 1 & \frac{1}{2} & -1 & \frac{1}{2} & 1 \\ \hline 0 & 0 & 0 & 1 & \frac{23}{12} & -\frac{4}{3} & \frac{5}{12} & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ \hline 0 & \frac{3}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 & 1 \end{array}\right].$$

This method preserves the third order of the base methods and has stage order two. To our knowledge, this is the first high-order method combining a Runge–Kutta and linear multistep method. Moreover, this demonstrates the potential of this framework outside of multirate GLMs.

Multirate General Linear Methods

In the context of multirate methods, it is common to consider a two-partitioned version of eq. (2):

$$y' = f^{\{\mathfrak{f}\}}(y) + f^{\{\mathfrak{s}\}}(y). \tag{8}$$

The fast dynamics are contained in $f^{\{\mathfrak{f}\}}$, and the slow dynamics are contained in $f^{\{\mathfrak{s}\}}$. Now, we define a fast GLM method $\mathcal{G}^{\{\mathfrak{f}\}}$ with the coefficients $A^{\{\mathfrak{f},\mathfrak{f}\}}$, $B^{\{\mathfrak{f},\mathfrak{f}\}}$, $U^{\{\mathfrak{f},\mathfrak{f}\}}$, $V^{\{\mathfrak{f},\mathfrak{f}\}}$, $Q^{\{\mathfrak{f},\mathfrak{f}\}}$ and $c^{\{\mathfrak{f}\}}$ and a slow base method $\mathcal{G}^{\{\mathfrak{s}\}}$ with $A^{\{\mathfrak{s},\mathfrak{s}\}}$, $B^{\{\mathfrak{s},\mathfrak{s}\}}$, $U^{\{\mathfrak{s},\mathfrak{s}\}}$, $V^{\{\mathfrak{s},\mathfrak{s}\}}$, $Q^{\{\mathfrak{s},\mathfrak{s}\}}$, and $c^{\{\mathfrak{s}\}}$. Following the multirate idea, we would like to integrate $f^{\{\mathfrak{s}\}}$ with $\mathcal{G}^{\{\mathfrak{s}\}}$ using a time step H and integrate $f^{\{\mathfrak{f}\}}$ with $\mathcal{G}^{\{\mathfrak{f}\}}$ using a time step $h = H/M$. The multirate ratio M is a positive integer.

One step of a multirate GLM is as follows. The slow stages are given by

$$\begin{aligned} Y_i^{\{\mathfrak{s}\}[n]} = {} & H \sum_{j=1}^{s^{\{\mathfrak{s}\}}} a_{i,j}^{\{\mathfrak{s},\mathfrak{s}\}} f^{\{\mathfrak{s}\}}\left(Y_j^{\{\mathfrak{s}\}[n]}\right) \\ & + h \sum_{\lambda=1}^{M} \sum_{j=1}^{s^{\{\mathfrak{f}\}}} a_{i,j}^{\{\mathfrak{s},\mathfrak{f},\lambda\}} f^{\{\mathfrak{f}\}}\left(Y_j^{\{\mathfrak{f},\lambda\}[n]}\right) \\ & + \sum_{j=1}^{r^{\{\mathfrak{s}\}}} u_{i,j}^{\{\mathfrak{s},\mathfrak{s}\}} \xi_j^{\{\mathfrak{s}\}[n-1]} \\ & + \sum_{j=1}^{r^{\{\mathfrak{f}\}}} u_{i,j}^{\{\mathfrak{s},\mathfrak{f}\}} \xi_j^{\{\mathfrak{f}\}[n-1]}, \end{aligned} \tag{9a}$$

where $i = 1, \ldots, s^{\{\mathfrak{s}\}}$. The fast stages for mi-

crostep λ are given by

$$\begin{aligned} Y_i^{\{\mathfrak{f},\lambda\}[n]} &= h\sum_{j=1}^{s^{\{\mathfrak{f}\}}} a_{i,j}^{\{\mathfrak{f},\mathfrak{f}\}} f^{\{\mathfrak{f}\}}\left(Y_j^{\{\mathfrak{f},\lambda\}[n]}\right) \\ &\quad + H\sum_{j=1}^{s^{\{\mathfrak{s}\}}} a_{i,j}^{\{\mathfrak{f},\mathfrak{s},\lambda\}} f^{\{\mathfrak{s}\}}\left(Y_j^{\{\mathfrak{s}\}[n]}\right) \\ &\quad + \sum_{j=1}^{r^{\{\mathfrak{f}\}}} u_{i,j}^{\{\mathfrak{f},\mathfrak{f}\}} \widetilde{\xi}_j^{\{\mathfrak{f}\}[n-1+(\lambda-1)/M]} \\ &\quad + \sum_{j=1}^{r^{\{\mathfrak{s}\}}} u_{i,j}^{\{\mathfrak{f},\mathfrak{s},\lambda\}} \xi_j^{\{\mathfrak{s}\}[n-1]}, \end{aligned} \tag{9b}$$

where $\lambda = 1, \ldots, M$ and $i = 1, \ldots, s^{\{\mathfrak{f}\}}$. The intermediate external stages are updated each microstep by

$$\begin{aligned} \widetilde{\xi}_i^{\{\mathfrak{f}\}[n-1+\lambda/M]} &= h\sum_{j=1}^{s^{\{\mathfrak{f}\}}} b_{i,j}^{\{\mathfrak{f},\mathfrak{f}\}} f^{\{\mathfrak{f}\}}\left(Y_j^{\{\mathfrak{f},\lambda\}[n]}\right) \\ &\quad + \sum_{j=1}^{r^{\{\mathfrak{f}\}}} v_{i,j}^{\{\mathfrak{f},\mathfrak{f}\}} \widetilde{\xi}_j^{\{\mathfrak{f}\}[n-1+(\lambda-1)/M]}, \end{aligned} \tag{9c}$$

where $i = 1, \ldots, s^{\{\mathfrak{f}\}}$. Finally, the fast external stages are simply

$$\xi_i^{\{\mathfrak{f}\}[n]} = \widetilde{\xi}_i^{\{\mathfrak{f}\}[n-1+M/M]}, \tag{9d}$$

where $i = 1, \ldots, r^{\{\mathfrak{f}\}}$, and the slow external stages are computed by

$$\begin{aligned} \xi_i^{\{\mathfrak{s}\}[n]} &= H\sum_{j=1}^{s^{\{\mathfrak{s}\}}} b_{i,j}^{\{\mathfrak{s},\mathfrak{s}\}} f^{\{\mathfrak{s}\}}\left(Y_j^{\{\mathfrak{s}\}[n]}\right) \\ &\quad + \sum_{j=1}^{r^{\{\mathfrak{s}\}}} v_{i,j}^{\{\mathfrak{s},\mathfrak{s}\}} \xi_i^{\{\mathfrak{s}\}[n-1]}, \end{aligned} \tag{9e}$$

where $i = 1, \ldots, r^{\{\mathfrak{s}\}}$.

The partitioned GLM tableau for eq. (9) takes the form

$$\left[\begin{array}{cc|cc} \mathbf{A}^{\{\mathfrak{f},\mathfrak{f}\}} & \mathbf{A}^{\{\mathfrak{f},\mathfrak{s}\}} & \mathbf{U}^{\{\mathfrak{f},\mathfrak{f}\}} & \mathbf{U}^{\{\mathfrak{f},\mathfrak{s}\}} \\ \mathbf{A}^{\{\mathfrak{s},\mathfrak{f}\}} & \mathbf{A}^{\{\mathfrak{s},\mathfrak{s}\}} & \mathbf{U}^{\{\mathfrak{s},\mathfrak{f}\}} & \mathbf{U}^{\{\mathfrak{s},\mathfrak{s}\}} \\ \hline \mathbf{B}^{\{\mathfrak{f},\mathfrak{f}\}} & \mathbf{0} & \mathbf{V}^{\{\mathfrak{f},\mathfrak{f}\}} & \mathbf{0} \\ \mathbf{0} & \mathbf{B}^{\{\mathfrak{s},\mathfrak{s}\}} & \mathbf{0} & \mathbf{V}^{\{\mathfrak{s},\mathfrak{s}\}} \end{array}\right]. \tag{10}$$

The fast blocks of eq. (10) come from expressing M steps of $\mathcal{G}^{\{\mathfrak{f}\}}$ with time step h as one step of a GLM $\mathbf{G}^{\{\mathfrak{f}\}}$ with time step H. The coefficients of $\mathbf{G}^{\{\mathfrak{f}\}}$ are given in fig. 2. The slow blocks are much simpler, as the coefficients map directly to the base method:

$$\left[\begin{array}{c|c} \mathbf{A}^{\{\mathfrak{s},\mathfrak{s}\}} & \mathbf{U}^{\{\mathfrak{s},\mathfrak{s}\}} \\ \hline \mathbf{B}^{\{\mathfrak{s},\mathfrak{s}\}} & \mathbf{V}^{\{\mathfrak{s},\mathfrak{s}\}} \end{array}\right] := \left[\begin{array}{c|c} A^{\{\mathfrak{s},\mathfrak{s}\}} & U^{\{\mathfrak{s},\mathfrak{s}\}} \\ \hline B^{\{\mathfrak{s},\mathfrak{s}\}} & V^{\{\mathfrak{s},\mathfrak{s}\}} \end{array}\right].$$

Order Conditions

We can now leverage the order condition results from the previous section to determine the order conditions for a multirate GLM.

Theorem 2 *Assume that $\xi_i^{\{\sigma\}[n-1]}$ satisfies eq.* (5)*. A preconsistent partitioned GLM eq.* (4) *has order p and stage order $q \in \{p, p-1\}$ if and only if $\mathcal{G}^{\{\mathfrak{f}\}}$ and $\mathcal{G}^{\{\mathfrak{s}\}}$ have order p and stage order q and for $k = 1, \ldots, q$:*

$$\begin{aligned} 0 &= \frac{c^{\{\mathfrak{f}\}\times k}}{k!} - \frac{A^{\{\mathfrak{f},\mathfrak{s},\lambda\}} c^{\{\mathfrak{s}\}\times(k-1)}}{(k-1)!} \\ &\quad - U^{\{\mathfrak{f},\mathfrak{s},\lambda\}} q_k^{\{\mathfrak{s},\mathfrak{s}\}}, \quad \lambda = 1, \ldots, M \\ 0 &= \frac{M^k c^{\{\mathfrak{s}\}\times k}}{k!} - U^{\{\mathfrak{s},\mathfrak{f}\}} q_k^{\{\mathfrak{f},\mathfrak{f}\}} \\ &\quad - \sum_{\lambda=1}^{M} \frac{A^{\{\mathfrak{s},\mathfrak{f},\lambda\}} \left(c^{\{\mathfrak{f}\}} + (\lambda-1)\mathbb{1}_{s^{\{\mathfrak{f}\}}}\right)^{k-1}}{(k-1)!}. \end{aligned}$$

New Methods

Explicit methods are the first type of multirate GLM derived, with implicit methods being a subject of future work. We choose $\mathcal{G}^{\{\mathfrak{f}\}}$ and $\mathcal{G}^{\{\mathfrak{s}\}}$ to be the same since it allows the multirate methods to easily be applied telescopically to more than two partitions. Another desirable feature is method coefficients that are bounded for all multirate ratios. With these design principles in mind, we derive parameterized families of methods up to order four.

We start with the a parameterized GLM with $c = [0, 1]^T$ of order two or possibly order

$$\begin{array}{c|c} \mathbf{A}^{\{\mathfrak{f},\mathfrak{f}\}} & \mathbf{U}^{\{\mathfrak{f},\mathfrak{f}\}} \\ \hline \mathbf{B}^{\{\mathfrak{f},\mathfrak{f}\}} & \mathbf{V}^{\{\mathfrak{f},\mathfrak{f}\}} \end{array} := \left[\begin{array}{cccc|c} \frac{1}{M}A^{\{\mathfrak{f},\mathfrak{f}\}} & & & & U^{\{\mathfrak{f},\mathfrak{f}\}} \\ \frac{1}{M}U^{\{\mathfrak{f},\mathfrak{f}\}}B^{\{\mathfrak{f},\mathfrak{f}\}} & \frac{1}{M}A^{\{\mathfrak{f},\mathfrak{f}\}} & & & U^{\{\mathfrak{f},\mathfrak{f}\}}V^{\{\mathfrak{f},\mathfrak{f}\}} \\ \vdots & \vdots & \ddots & & \vdots \\ \frac{1}{M}U^{\{\mathfrak{f},\mathfrak{f}\}}V^{\{\mathfrak{f},\mathfrak{f}\}\times(M-2)}B^{\{\mathfrak{f},\mathfrak{f}\}} & \frac{1}{M}U^{\{\mathfrak{f},\mathfrak{f}\}}V^{\{\mathfrak{f},\mathfrak{f}\}\times(M-3)}B^{\{\mathfrak{f},\mathfrak{f}\}} & \dots & \frac{1}{M}A^{\{\mathfrak{f},\mathfrak{f}\}} & U^{\{\mathfrak{f},\mathfrak{f}\}}V^{\{\mathfrak{f},\mathfrak{f}\}\times(M-1)} \\ \hline \frac{1}{M}V^{\{\mathfrak{f},\mathfrak{f}\}\times(M-1)}B^{\{\mathfrak{f},\mathfrak{f}\}} & \frac{1}{M}V^{\{\mathfrak{f},\mathfrak{f}\}\times(M-2)}B^{\{\mathfrak{f},\mathfrak{f}\}} & \dots & \frac{1}{M}B^{\{\mathfrak{f},\mathfrak{f}\}} & V^{\{\mathfrak{f},\mathfrak{f}\}\times M} \end{array}\right]$$

Figure 2: Tableau generated by composition of M steps.

three:

$$\begin{array}{cc|cc} 0 & 0 & 1 & 0 \\ a_{2,1} & 0 & 0 & 1 \\ \hline \frac{(2a_{2,1}-1)v_{1,2}+1}{2} & \frac{1-v_{1,2}}{2} & 1-v_{1,2} & v_{1,2} \\ \frac{(2a_{2,1}-1)v_{2,2}}{2} & \frac{4-2a_{2,1}-v_{2,2}}{2} & 1-v_{2,2} & v_{2,2} \end{array}. \tag{11a}$$

The corresponding coupling coefficients to preserve the order are

$$A^{\{\mathfrak{f},\mathfrak{s},\lambda\}} = \begin{bmatrix} \frac{(\lambda-1)(M-\lambda+(\lambda-1)a_{2,1}+1)}{M^2} & 0 \\ \frac{\lambda(M-\lambda+\lambda a_{2,1})}{M^2} & 0 \end{bmatrix}, \tag{11b}$$

$$A^{\{\mathfrak{s},\mathfrak{f},\lambda\}} = \begin{bmatrix} 0 & 0 \\ \frac{\lambda-(\lambda-4)a_{2,1}-1}{3} & a_{2,2}^{\{\mathfrak{s},\mathfrak{f},\lambda\}} \end{bmatrix},$$
$$a_{2,2}^{\{\mathfrak{s},\mathfrak{f},\lambda\}} = \frac{(\lambda-1)(a_{2,1}-1)(4(M-5)a_{2,1}-4M+23)}{3(4(M+1)a_{2,1}-4M-1)}, \tag{11c}$$

$$U^{\{\mathfrak{f},\mathfrak{s},\lambda\}} = \begin{bmatrix} \frac{M^2-(\lambda-1)^2}{M^2} & \frac{(\lambda-1)^2}{M^2} \\ 1-\frac{\lambda^2}{M^2} & \frac{\lambda^2}{M^2} \end{bmatrix}, \tag{11d}$$

$$U^{\{\mathfrak{s},\mathfrak{f}\}} = \begin{bmatrix} 1 & 0 \\ \frac{(1-M)(4a_{2,1}-1)}{4(M+1)a_{2,1}-4M-1} & u_{2,2}^{\{\mathfrak{s},\mathfrak{f}\}} \end{bmatrix},$$
$$u_{2,2}^{\{\mathfrak{s},\mathfrak{f}\}} = \frac{M(8a_{2,1}-5)}{4(M+1)a_{2,1}-4M-1}. \tag{11e}$$

We choose $a_{2,1} = 2$ and $v_{1,2} = v_{2,2} = 1/2$ for a second order method. Using $a_{2,1} = 20/27$ and $v_{1,2} = v_{2,2} = 1/6$ yields third order. For brevity we omit the more complex fourth order family, which will be provided in the subsequent publication.

Linear Stability

The stability analysis follows from substituting the particular tableau structure in eq. (10) into the general stability function eq. (7). To simplify the visualization of the stability regions, we consider the linear test problem

$$y' = \underbrace{\frac{M\lambda}{2}y}_{\text{fast}} + \underbrace{\frac{\lambda}{2}y}_{\text{slow}}, \tag{12}$$

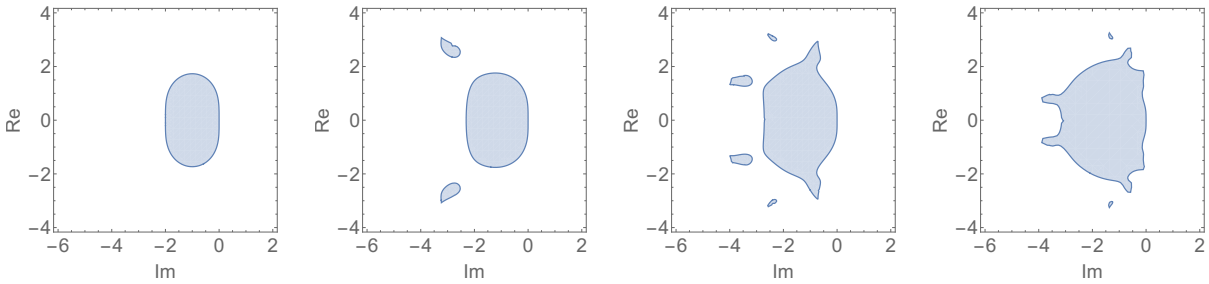
where $\lambda \in \mathbb{C}$. In the spirit of multirate methods, the fast partition is M times stiffer than the slow partition. Further, eq. (12) is scaled such that when $M = 1$, we recover the standard test function $y' = \lambda y$. We are interested in the set of $z = H\lambda$ for which a multirate GLM is stable when applied to eq. (12). For the second order method, the stability regions are plotted in fig. 3. As M increases, the stability region actually becomes slightly larger, indicating excellent scaling. The third and fourth orders methods see similar stability behavior. For comparison, a singlerate method applied to eq. (12) will have a stability region that shrinks to the origin as M grows.

Numerical Experiment: Gray–Scott

To confirm the theoretical properties of the new multirate GLMs and to test their efficiency, we apply them to the Gray–Scott partial differential equation:

$$\underbrace{\begin{bmatrix} u \\ v \end{bmatrix}'}_{y'} = \underbrace{\begin{bmatrix} \varepsilon_u \Delta u \\ \varepsilon_v \Delta v \end{bmatrix}}_{f^{\{\mathfrak{s}\}}(y)} + \underbrace{\begin{bmatrix} -uv^2 + f(1-u) \\ uv^2 - (f+k) \end{bmatrix}}_{f^{\{\mathfrak{f}\}}(y)}. \tag{13}$$

The parameters are taken to be $\varepsilon_u = 2\times10^{-6}$, $\varepsilon_v = 10^{-6}$, $f = 0.04$, and $k = 0.06$. The



(a) $M = 1$ (b) $M = 2$ (c) $M = 4$ (d) $M = 8$

Figure 3: Multirate stability regions for second order method

domain is periodic over $[0, 2.5]^2$ and is discretized into a 100×100 grid with second order central differences. Experiments are run from $t = 0$ to $t = 100$ using the third order method.

In fig. 4, the error versus the number of steps is provided. We can see the method matches the expected order for each multirate ratio M, and that increasing M reduces the error.

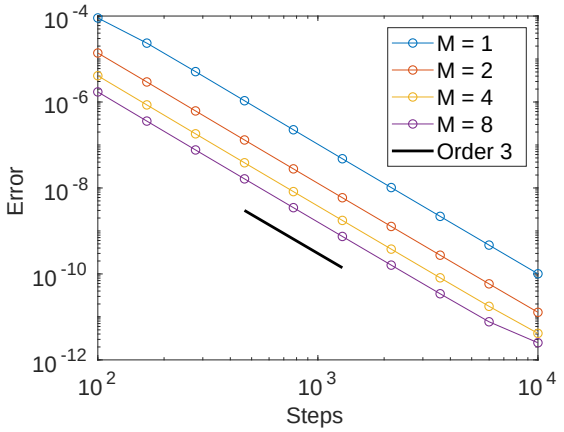


Figure 4: Convergence plot for third order multirate GLM.

In fig. 5, we compare the error versus time to solution. The multirate method consistently reaches a desired accuracy in less time than the singlerate base method ($M = 1$).

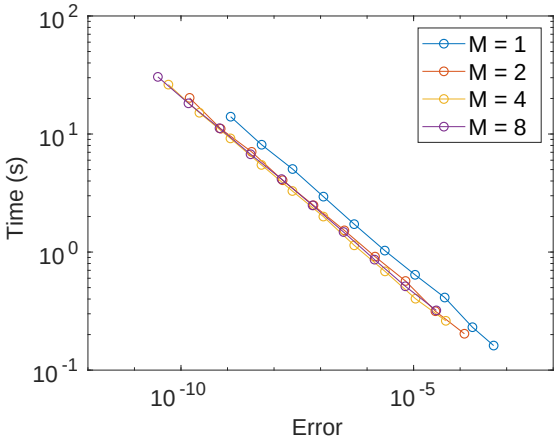


Figure 5: Work precision plot for third order multirate GLM.

Coupled Multirate Infinitesimal GARK Methods

While not mentioned in the original proposal, another topic of research on practical multirate time integration methods was coupled multirate infinitesimal GARK (MRI-GARK) methods. This work builds on the decoupled MRI-GARK methods of Sandu[19] and the original work of Knoth and Wolke on infinitesimal methods[16]. The defining characteristic of infinitesimal methods is that they combine a discretization of the slow dynamics, say with a Runge–Kutta method, with "exact" solves of modified ODEs for the fast dynamics. Except in the simplest of cases, it is infeasible to exactly solve these modified ODEs, so instead, a discretization method is used with an "infinitely" small time step, thus motivating the name. These schemes are

especially well-suited for problems with extreme timescale differences. One of the most appealing features of these methods, is that *any* fast discretization method can be used provided it is convergent.

Along with the recent work of Sandu[19], these are the first implicit infinitesimal methods. The primary benefit of coupled MRI-GARK methods over other flavors of infinitesimal schemes is improved stability. We explore two different coupling approaches, which we briefly describe below. Interested readers can see the full publication for additional details, including methods up to order four.[18]

Step Predictor Corrector MRI-GARK

A step predictor corrector MRI-GARK (SCP-MIR-GARK) scheme applied to eq. (8) advances the solution from t_n to t_{n+1} as follows:

$$
\begin{aligned}
&Y_i = y_n + H \sum_{j=1}^{s^{\{\mathfrak{s}\}}} a_{i,j}^{\{\mathfrak{s},\mathfrak{s}\}} f(Y_j), \\
&\begin{cases}
v(0) = y_n, \\
\quad v' = f^{\{\mathfrak{f}\}}(v) + \sum_{j=1}^{s^{\{\mathfrak{s}\}}} \gamma_j\left(\frac{\theta}{H}\right) f^{\{\mathfrak{s}\}}(Y_j), \\
y_{n+1} = v(H).
\end{cases}
\end{aligned}
$$

SCP-MIR-GARK methods start with traditional Runge–Kutta stages before solving a single modified fast ODE to obtain a solution. Unlike previous infinitesimal schemes, there is no restriction on the type of base method. It can be explicit, diagonally implicit, or even fully implicit.

Internal Stage Predictor Corrector MRI-GARK

One step of an internal stage predictor corrector MRI-GARK (IPC-MRI-GARK) scheme proceeds as follows:

$$
\begin{aligned}
&Y_0 \coloneqq y_n, \quad c_0^{\{\mathfrak{s}\}} \coloneqq 0, \\
&\begin{cases}
Y_i^* = y_n + H \sum_{j=1}^{i-1} a_{i,j}^{\{\mathfrak{s},\mathfrak{s}\}} f(Y_j) \\
\qquad\quad + H a_{i,i}^{\{\mathfrak{s},\mathfrak{s}\}} f(Y_i^*) \\
v_i(0) = Y_{i-1}, \\
v_i' = \Delta c_i^{\{\mathfrak{s}\}} f^{\{\mathfrak{f}\}}(v_i) \\
\qquad + \sum_{j=1}^{i-1} \gamma_{i,j}\left(\frac{\theta}{H}\right) f^{\{\mathfrak{s}\}}(Y_j) \\
\qquad + \sum_{j=1}^{i} \psi_{i,j}\left(\frac{\theta}{H}\right) f^{\{\mathfrak{s}\}}(Y_i^*), \\
Y_i = v_i(H), \qquad i = 1, \ldots, s^{\{\mathfrak{s}\}},
\end{cases} \\
&y_{n+1} = Y_{s^{\{\mathfrak{s}\}}}.
\end{aligned}
$$

IPC-MRI-GARK methods interleave Runge–Kutta stages and infinitesimal integrations. For this reason, the base method must have nonincreasing abscissa and an explicit or diagonally implicit structure.

Conclusion

Multirate GLMs and coupled MRI-GARK methods offer efficient, practical ways to solve ODEs that exhibit multiple characteristic timescales. The former is built on a new theoretical foundation of partitioned GLMs, which is currently the most general framework for time integration and opens the door to new types of multi-methods.

Areas of future work include multirate GLMs with one or more implicit base methods and methods of order five. In addition to the Gray–Scott test problem, we plan to apply these schemes to more sophisticated test problems to examine properties such as order reduction.

Acknowledgement

I would like to express my gratitude to the Virginia Space Grant Consortium for providing

the fellowship that helped to make this work possible. I would also like to thank my advisor Dr. Adrian Sandu and fellow Ph.D. student Arash Sarshar for their support and feedback.

References

[1] Hester Bijl, Mark H Carpenter, Veer N Vatsa, and Christopher A Kennedy. Implicit time integration schemes for the unsteady compressible Navier–Stokes equations: laminar flow. *Journal of Computational Physics*, 179(1):313–329, 2002.

[2] JC Butcher and Zdzislaw Jackiewicz. Diagonally implicit general linear methods for ordinary differential equations. *BIT Numerical Mathematics*, 33(3):452–472, 1993.

[3] John C Butcher. Diagonally-implicit multi-stage integration methods. *Applied Numerical Mathematics*, 11(5):347–363, 1993.

[4] Mark H Carpenter, CA Kennedy, Hester Bijl, SA Viken, and Veer N Vatsa. Fourth-order Runge–Kutta schemes for fluid mechanics applications. *Journal of Scientific Computing*, 25(1-2):157–194, 2005.

[5] Mark H Carpenter and Christopher A Kennedy. Fourth-order 2n-storage Runge–Kutta schemes. 1994.

[6] Mark H Carpenter and Christopher A Kennedy. Third-order 2n-storage Runge–Kutta schemes with error control. 1994.

[7] Kennedy Christopher A and Carpenter Mark H. Additive Runge–Kutta schemes for convection-diffusion-reaction equations. 2001.

[8] Germund Dahlquist. Convergence and stability in the numerical integration of ordinary differential equations. *Mathematica Scandinavica*, pages 33–53, 1956.

[9] Germund G Dahlquist. A special stability problem for linear multistep methods. *BIT Numerical Mathematics*, 3(1):27–43, 1963.

[10] E Fehlberg. Runge–Kutta type formulas of high-order accuracy and their application to the numerical integration of the restricted problem of three bodies. 1964.

[11] Erwin Fehlberg. Classical fifth-, sixth-, seventh-, and eighth-order Runge–Kutta formulas with stepsize control. 1968.

[12] Erwin Fehlberg. Low-order classical Runge–Kutta formulas with stepsize control and their application to some heat transfer problems. 1969.

[13] C.W. Gear and D.R. Wells. Multirate linear multistep methods. *BIT*, 24:484–502, 1984.

[14] Christopher A Kennedy and Mark H Carpenter. Diagonally implicit Runge–Kutta methods for ordinary differential equations. a review. 2016.

[15] Christopher A Kennedy, Mark H Carpenter, and R Michael Lewis. Low-storage, explicit Runge–Kutta schemes for the compressible Navier–Stokes equations. *Applied numerical mathematics*, 35(3):177–219, 2000.

[16] Oswald Knoth and Ralf Wolke. Implicit-explicit Runge–Kutta methods for computing atmospheric reactive flows. *Applied numerical mathematics*, 28(2-4):327–341, 1998.

[17] J.R. Rice. Split Runge-Kutta methods for simultaneous equations. *Journal of Research of the National Institute of Standards and Technology*, 60(B), 1960.

[18] Steven Roberts, Arash Sarshar, and Adrian Sandu. Coupled multirate infinitesimal GARK schemes for stiff systems with multiple time scales. *arXiv preprint arXiv:1812.00808*, 2018.

[19] Adrian Sandu. A class of multirate infinitesimal gark methods. *arXiv preprint arXiv:1808.02759*, 2018.

[20] A. Sarshar, S. Roberts, and A. Sandu. Design of high-order decoupled multirate GARK schemes. *SIAM Journal on Scientific Computing*, 41(2):A816–A847, 2019.

[21] Veer Vatsa, Mark Carpenter, and David Lockard. Re-evaluation of an optimized second order backward difference (bdf2opt) scheme for unsteady flow applications. In *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, page 122, 2010.

[22] Hong Zhang, Adrian Sandu, and Sebastien Blaise. Partitioned and implicit–explicit general linear methods for ordinary differential equations. *Journal of Scientific Computing*, 61(1):119–144, 2014.