Implicit Multirate GARK Methods

Steven Roberts, John Loffeld, Arash Sarshar, Adrian Sandu, and Carol Woodward

"Compute the Future!",
Department of Computer Science,
Virginia Polytechnic Institute and State University
Blacksburg, VA 24060

SciCADE, July 22, 2019

Compute the Future!

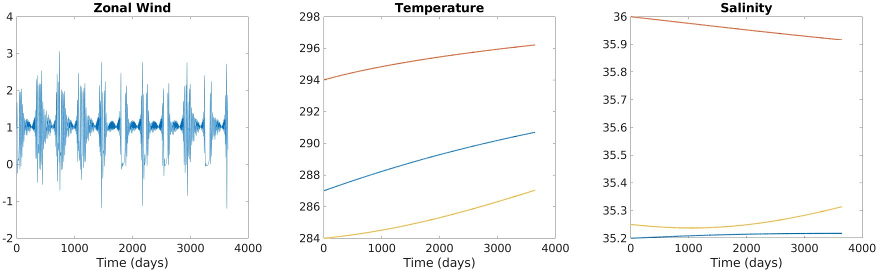
VIRGINIA TECH

Why use multirate methods?

- Many dynamical systems exhibit multiple characteristic timescales.

$$y' = f(y) = f^{\{\mathfrak{f}\}}(y) + f^{\{\mathfrak{s}\}}(y), \qquad y(t_0) = y_0$$

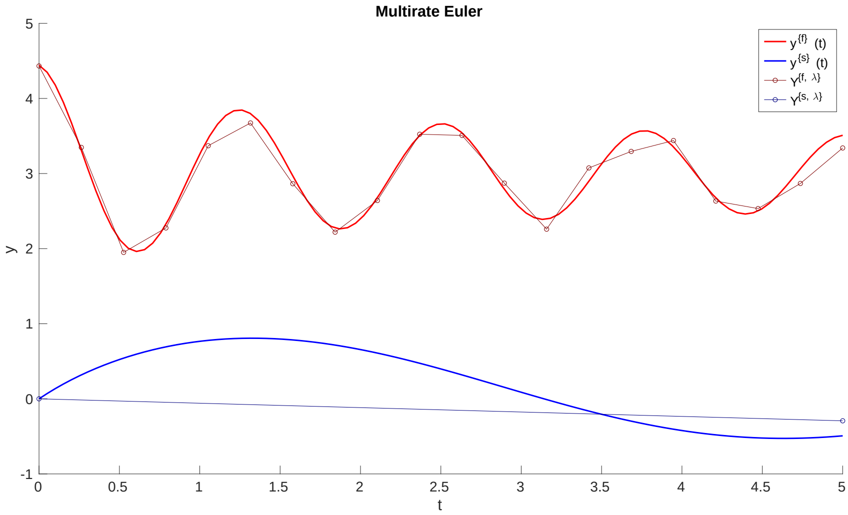
- Example: Wind, temperature, and salinity in a simplified climate model



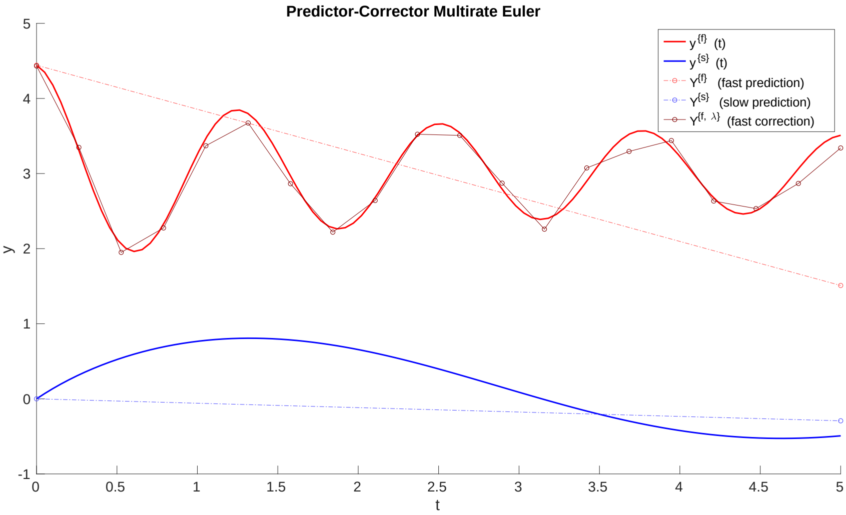
What are multirate methods?

- Integrate the slow partition with Runge–Kutta method $\left(A^{\{\mathfrak{s},\mathfrak{s}\}}, b^{\{\mathfrak{s}\}}\right)$ using a stepsize H
- Integrate the fast partition with Runge–Kutta method $\left(A^{\{\mathfrak{f},\mathfrak{f}\}}, b^{\{\mathfrak{f}\}}\right)$ using a stepsize $h = H/M$
- M is called the multirate ratio
- Coupling information needs to be shared between slow and fast integrations.
- Why use implicit method for both fast and slow dynamics?
    - Adapting timesteps to accuracy requirements can improve efficiency.
    - Decoupled methods simplify Newton iterations.
    - Certain parts of system may slow down Newton iterations.

Multirate Runge–Kutta



Predictor-corrector multirate Runge–Kutta



GARK provides a theoretical foundation

- A generalized-structure additively partitioned Runge–Kutta (GARK)[1] method with two partitions reads

$$
\begin{aligned}
Y_i^{\{\mathfrak{f}\}} &= y_n + H \sum_{j=1}^{s^{\{\mathfrak{f}\}}} a_{i,j}^{\{\mathfrak{f},\mathfrak{f}\}} f^{\{\mathfrak{f}\}}\left(Y_j^{\{\mathfrak{f}\}}\right) + H \sum_{j=1}^{s^{\{\mathfrak{s}\}}} a_{i,j}^{\{\mathfrak{f},\mathfrak{s}\}} f^{\{\mathfrak{s}\}}\left(Y_j^{\{\mathfrak{s}\}}\right), \qquad i = 1, \dots s^{\{\mathfrak{f}\}}, \\
Y_i^{\{\mathfrak{s}\}} &= y_n + H \sum_{j=1}^{s^{\{\mathfrak{f}\}}} a_{i,j}^{\{\mathfrak{s},\mathfrak{f}\}} f^{\{\mathfrak{f}\}}\left(Y_j^{\{\mathfrak{f}\}}\right) + H \sum_{j=1}^{s^{\{\mathfrak{s}\}}} a_{i,j}^{\{\mathfrak{s},\mathfrak{s}\}} f^{\{\mathfrak{s}\}}\left(Y_j^{\{\mathfrak{f}\}}\right), \qquad i = 1, \dots s^{\{\mathfrak{s}\}}, \\
y_{n+1} &= y_n + H \sum_{j=1}^{s^{\{\mathfrak{f}\}}} b_j^{\{\mathfrak{f}\}} f^{\{\mathfrak{f}\}}\left(Y_j^{\{\mathfrak{f}\}}\right) + H \sum_{j=1}^{s^{\{\mathfrak{s}\}}} b_j^{\{\mathfrak{s}\}} f^{\{\mathfrak{s}\}}\left(Y_j^{\{\mathfrak{s}\}}\right).
\end{aligned}
$$

[1] Sandu & Günther, "A generalized-structure approach to additive Runge-Kutta methods".

GARK provides a theoretical foundation

- A generalized-structure additively partitioned Runge–Kutta (GARK)[1] method with two partitions reads

$$
\begin{aligned}
Y_i^{\{\mathfrak{f}\}} &= y_n + H \sum_{j=1}^{s^{\{\mathfrak{f}\}}} a_{i,j}^{\{\mathfrak{f},\mathfrak{f}\}} f^{\{\mathfrak{f}\}}\left(Y_j^{\{\mathfrak{f}\}}\right) + H \sum_{j=1}^{s^{\{\mathfrak{s}\}}} a_{i,j}^{\{\mathfrak{f},\mathfrak{s}\}} f^{\{\mathfrak{s}\}}\left(Y_j^{\{\mathfrak{s}\}}\right), \qquad i = 1, \dots s^{\{\mathfrak{f}\}}, \\
Y_i^{\{\mathfrak{s}\}} &= y_n + H \sum_{j=1}^{s^{\{\mathfrak{f}\}}} a_{i,j}^{\{\mathfrak{s},\mathfrak{f}\}} f^{\{\mathfrak{f}\}}\left(Y_j^{\{\mathfrak{f}\}}\right) + H \sum_{j=1}^{s^{\{\mathfrak{s}\}}} a_{i,j}^{\{\mathfrak{s},\mathfrak{s}\}} f^{\{\mathfrak{s}\}}\left(Y_j^{\{\mathfrak{f}\}}\right), \qquad i = 1, \dots s^{\{\mathfrak{s}\}}, \\
y_{n+1} &= y_n + H \sum_{j=1}^{s^{\{\mathfrak{f}\}}} b_j^{\{\mathfrak{f}\}} f^{\{\mathfrak{f}\}}\left(Y_j^{\{\mathfrak{f}\}}\right) + H \sum_{j=1}^{s^{\{\mathfrak{s}\}}} b_j^{\{\mathfrak{s}\}} f^{\{\mathfrak{s}\}}\left(Y_j^{\{\mathfrak{s}\}}\right).
\end{aligned}
$$

- The corresponding tableau is

$$
\begin{array}{c|c}
\mathbf{A}^{\{\mathfrak{f},\mathfrak{f}\}} & \mathbf{A}^{\{\mathfrak{f},\mathfrak{s}\}} \\
\hline
\mathbf{A}^{\{\mathfrak{s},\mathfrak{f}\}} & \mathbf{A}^{\{\mathfrak{s},\mathfrak{s}\}} \\
\hline
\mathbf{b}^{\{\mathfrak{f}\}\,T} & \mathbf{b}^{\{\mathfrak{s}\}\,T}
\end{array}.
$$

[1] Sandu & Günther, "A generalized-structure approach to additive Runge-Kutta methods".

GARK provides a theoretical foundation

- A generalized-structure additively partitioned Runge–Kutta (GARK)[1] method with two partitions reads

$$
\begin{aligned}
Y_i^{\{\mathfrak{f}\}} &= y_n + H \sum_{j=1}^{s^{\{\mathfrak{f}\}}} a_{i,j}^{\{\mathfrak{f},\mathfrak{f}\}} f^{\{\mathfrak{f}\}}\left(Y_j^{\{\mathfrak{f}\}}\right) + H \sum_{j=1}^{s^{\{\mathfrak{s}\}}} a_{i,j}^{\{\mathfrak{f},\mathfrak{s}\}} f^{\{\mathfrak{s}\}}\left(Y_j^{\{\mathfrak{s}\}}\right), \qquad i = 1, \dots s^{\{\mathfrak{f}\}}, \\
Y_i^{\{\mathfrak{s}\}} &= y_n + H \sum_{j=1}^{s^{\{\mathfrak{f}\}}} a_{i,j}^{\{\mathfrak{s},\mathfrak{f}\}} f^{\{\mathfrak{f}\}}\left(Y_j^{\{\mathfrak{f}\}}\right) + H \sum_{j=1}^{s^{\{\mathfrak{s}\}}} a_{i,j}^{\{\mathfrak{s},\mathfrak{s}\}} f^{\{\mathfrak{s}\}}\left(Y_j^{\{\mathfrak{f}\}}\right), \qquad i = 1, \dots s^{\{\mathfrak{s}\}}, \\
y_{n+1} &= y_n + H \sum_{j=1}^{s^{\{\mathfrak{f}\}}} b_j^{\{\mathfrak{f}\}} f^{\{\mathfrak{f}\}}\left(Y_j^{\{\mathfrak{f}\}}\right) + H \sum_{j=1}^{s^{\{\mathfrak{s}\}}} b_j^{\{\mathfrak{s}\}} f^{\{\mathfrak{s}\}}\left(Y_j^{\{\mathfrak{s}\}}\right).
\end{aligned}
$$

- The corresponding tableau is

$$
\begin{array}{c|c}
\mathbf{A}^{\{\mathfrak{f},\mathfrak{f}\}} & \mathbf{A}^{\{\mathfrak{f},\mathfrak{s}\}} \\
\hline
\mathbf{A}^{\{\mathfrak{s},\mathfrak{f}\}} & \mathbf{A}^{\{\mathfrak{s},\mathfrak{s}\}} \\
\hline
\mathbf{b}^{\{\mathfrak{f}\}\,T} & \mathbf{b}^{\{\mathfrak{s}\}\,T}
\end{array}.
$$

- Internal consistency: $c^{\{\mathfrak{f}\}} \equiv A^{\{\mathfrak{f},\mathfrak{f}\}} \mathbb{1}_{s^{\{\mathfrak{f}\}}} = A^{\{\mathfrak{f},\mathfrak{s}\}} \mathbb{1}_{s^{\{\mathfrak{s}\}}}$ and $c^{\{\mathfrak{s}\}} \equiv A^{\{\mathfrak{s},\mathfrak{f}\}} \mathbb{1}_{s^{\{\mathfrak{f}\}}} = A^{\{\mathfrak{s},\mathfrak{s}\}} \mathbb{1}_{s^{\{\mathfrak{s}\}}}$

[1] Sandu & Günther, "A generalized-structure approach to additive Runge-Kutta methods".

Multirate Runge–Kutta methods are GARK methods

Standard MrGARK[2]:

$$
\begin{array}{ccc|c}
\frac{1}{M}A & \dots & 0 & A^{\{\mathfrak{f},\mathfrak{s},1\}} \\
\vdots & \ddots & \vdots & \vdots \\
\frac{1}{M}\mathbb{1}_s b^T & \dots & \frac{1}{M}A & A^{\{\mathfrak{f},\mathfrak{s},M\}} \\
\hline
\frac{1}{M}A^{\{\mathfrak{s},\mathfrak{f},1\}} & \dots & A^{\{\mathfrak{s},\mathfrak{f},M\}} & A \\
\hline
\frac{1}{M}b^T & \dots & \frac{1}{M}b^T & b^T
\end{array}.
$$

Predictor-corrector MrGARK:

$$
\begin{array}{cccc|c}
A & 0 & \dots & 0 & A \\
0 & \frac{1}{M}A & \dots & 0 & A^{\{\mathfrak{f},\mathfrak{s},1\}} \\
0 & \vdots & \ddots & \vdots & \vdots \\
0 & \frac{1}{M}\mathbb{1}_s b^T & \dots & \frac{1}{M}A & A^{\{\mathfrak{f},\mathfrak{s},M\}} \\
\hline
A & 0 & \dots & 0 & A \\
\hline
0 & \frac{1}{M}b^T & \dots & \frac{1}{M}b^T & b^T
\end{array}.
$$

[2] Günther & Sandu, "Multirate generalized additive Runge Kutta methods".

Challenges in developing implicit multirate methods

- Order conditions grow quickly in quantity and complexity.
- How can we balance the cost of solving nonlinear equations with stability?
- Linear stability is surprisingly complex and there are many open research questions.
- Many results on stability are limited to particular methods.

Challenges in developing implicit multirate methods

- Order conditions grow quickly in quantity and complexity.
- How can we balance the cost of solving nonlinear equations with stability?
- Linear stability is surprisingly complex and there are many open research questions.
- Many results on stability are limited to particular methods.

"Stability properties of various multirate schemes have been disussed . . . . However, most of these discussions are not very detailed, nor very conclusive." Kværnø, "Stability of multirate Runge–Kutta schemes"

Challenges in developing implicit multirate methods

- Order conditions grow quickly in quantity and complexity.
- How can we balance the cost of solving nonlinear equations with stability?
- Linear stability is surprisingly complex and there are many open research questions.
- Many results on stability are limited to particular methods.

"Stability properties of various multirate schemes have been disussed .... However, most of these discussions are not very detailed, nor very conclusive." Kværnø, "Stability of multirate Runge–Kutta schemes"

"Even though the multirate scheme considered in this paper is quite simple, the stability analysis will turn out to be complicated." Hundsdorfer & Savcenco, "Analysis of a Multirate Theta-method for Stiff ODEs"

MrGARK Order Conditions

- The MrGARK order conditions follow from substituting tableau structure into GARK order conditions.
- Assuming internal consistency, the cummulative number of order conditions is

Method	Order 1	Order 2	Order 3	Order 4
Standard MrGARK[3]	2	4	10	36
Predictor-corrector MrGARK	2	4	9	29

- Predictor-corrector order conditions more precise than usual technique of finding dense output of sufficient accuracy. The third order coupling condition, for example, is

$$\frac{M}{6} = \sum_{\lambda=1}^{M} b^T A^{\{\mathfrak{f},\mathfrak{s},\lambda\}} c.$$

[3] Sarshar *et al.*, "Design of High-Order Decoupled Multirate GARK Schemes".

Newton iterations

- The most computationally expensive part of implicit multirate methods
- Decoupled methods
  - Implicitness only comes from base methods
  - Only requires decompositions of $I - h\gamma J^{\{\mathfrak{f}\}}$ and $I - H\gamma J^{\{\mathfrak{s}\}}$
  - Efficient for component partitioned problems
- Coupled methods
  - Fast and slow stages solved together
  - Potentially very expensive
  - Practical methods require linear solves no more expensive than those of their singlerate counterparts.
  - Potential for better stability

Scalar stability function

- We can generalize the Dahlquist test problem by

$$y' = f^{\{\mathfrak{f}\}}(y) + f^{\{\mathfrak{s}\}}(y) \xRightarrow{\text{linearize}} y' = J^{\{\mathfrak{f}\}}\, y + J^{\{\mathfrak{s}\}}\, y \xRightarrow{\text{change basis*}} y' = \lambda^{\{\mathfrak{f}\}}\, y + \lambda^{\{\mathfrak{s}\}}\, y$$

[4] Gear, *Multirate methods for ordinary differential equations.*

Scalar stability function

- We can generalize the Dahlquist test problem by

$$y' = f^{\{\mathfrak{f}\}}(y) + f^{\{\mathfrak{s}\}}(y) \quad \xRightarrow{\text{linearize}} \quad y' = J^{\{\mathfrak{f}\}}\, y + J^{\{\mathfrak{s}\}}\, y \quad \xRightarrow{\text{change basis*}} \quad y' = \lambda^{\{\mathfrak{f}\}}\, y + \lambda^{\{\mathfrak{s}\}}\, y$$

- *Only if $J^{\{\mathfrak{f}\}}(y)$ and $J^{\{\mathfrak{s}\}}(y)$ are simultaneously triangularizable
- *Multirate methods are not invariant under change of basis[4].

[4] Gear, *Multirate methods for ordinary differential equations.*

Scalar stability function

- We can generalize the Dahlquist test problem by

$$y' = f^{\{\mathfrak{f}\}}(y) + f^{\{\mathfrak{s}\}}(y) \xrightarrow{\text{linearize}} y' = J^{\{\mathfrak{f}\}}\, y + J^{\{\mathfrak{s}\}}\, y \xrightarrow{\text{change basis*}} y' = \lambda^{\{\mathfrak{f}\}}\, y + \lambda^{\{\mathfrak{s}\}}\, y$$

- *Only if $J^{\{\mathfrak{f}\}}(y)$ and $J^{\{\mathfrak{s}\}}(y)$ are simultaneously triangularizable
- *Multirate methods are not invariant under change of basis[4].
- Applying the scalar test problem yields a stability function $R_1(z^{\{\mathfrak{f}\}}, z^{\{\mathfrak{s}\}})$ with $z^{\{\mathfrak{f}\}} = H\lambda^{\{\mathfrak{f}\}}$ and $z^{\{\mathfrak{s}\}} = H\lambda^{\{\mathfrak{s}\}}$.
- Stability criteria
  - A-Stability: $\left|R_1(z^{\{\mathfrak{f}\}}, z^{\{\mathfrak{s}\}})\right| \leq 1$ for all $z^{\{\mathfrak{f}\}}, z^{\{\mathfrak{s}\}} \in \mathbb{C}^-$
  - L-Stability: A-stability and $R_1(\infty, z^{\{\mathfrak{s}\}}) = R_1(z^{\{\mathfrak{f}\}}, \infty) = 0$
  - A(α)- and L(α)-stability: A 4D wedge fits in stability region

[4] Gear, *Multirate methods for ordinary differential equations.*

2D stability function

- At least two variables are needed for a component partitioned test problem:

$$\begin{bmatrix} y^{\{\mathfrak{f}\}} \\ y^{\{\mathfrak{s}\}} \end{bmatrix}' = \underbrace{\begin{bmatrix} \lambda^{\{\mathfrak{f}\}} & \eta^{\{\mathfrak{s}\}} \\ \eta^{\{\mathfrak{f}\}} & \lambda^{\{\mathfrak{s}\}} \end{bmatrix}}_{\Lambda} \begin{bmatrix} y^{\{\mathfrak{f}\}} \\ y^{\{\mathfrak{s}\}} \end{bmatrix}.$$

2D stability function

- At least two variables are needed for a component partitioned test problem:

$$\begin{bmatrix} y^{\{\mathfrak{f}\}} \\ y^{\{\mathfrak{s}\}} \end{bmatrix}' = \underbrace{\begin{bmatrix} \lambda^{\{\mathfrak{f}\}} & \eta^{\{\mathfrak{s}\}} \\ \eta^{\{\mathfrak{f}\}} & \lambda^{\{\mathfrak{s}\}} \end{bmatrix}}_{\Lambda} \begin{bmatrix} y^{\{\mathfrak{f}\}} \\ y^{\{\mathfrak{s}\}} \end{bmatrix}.$$

- Applying the scalar test problem yields a stability function $R_2(Z) \in \mathbb{C}^{2 \times 2}$ with $Z = H\Lambda$.
- Stability criteria
  - A-Stability: $R_2(Z)$ power bounded for all Z exponentially bounded with $z^{\{\mathfrak{f}\}}, z^{\{\mathfrak{s}\}} \in \mathbb{C}^-$
  - Many have restricted the problem to real entries to simplify analysis.

Even more ways to assess stability

- Others have looked at block test problems:

$$\begin{bmatrix} y^{\{\mathfrak{f}\}} \\ y^{\{\mathfrak{s}\}} \end{bmatrix}' = \begin{bmatrix} \Lambda^{\{\mathfrak{f}\}} & E^{\{\mathfrak{s}\}} \\ E^{\{\mathfrak{f}\}} & \Lambda^{\{\mathfrak{s}\}} \end{bmatrix} \begin{bmatrix} y^{\{\mathfrak{f}\}} \\ y^{\{\mathfrak{s}\}} \end{bmatrix}.$$

- Algebraic stability: If $f^{\{\mathfrak{f}\}}$ and $f^{\{\mathfrak{s}\}}$ are dissipative, then $\|y_{n+1} - \widetilde{y}_{n+1}\| \leq \|y_n - \widetilde{y}_n\|$.
- **Which stability cirteria should we use?**

Our findings on stability analysis

- E-Polynomial can be generalized for scalar test problem

Our findings on stability analysis

- E-Polynomial can be generalized for scalar test problem
- The scalar and 2D stability functions are related:

$$R_1\left(z^{\{\mathfrak{f}\}}, z^{\{\mathfrak{s}\}}\right) = \begin{bmatrix} 1 & 1 \end{bmatrix} R_2\left(\begin{bmatrix} z^{\{\mathfrak{f}\}} & z^{\{\mathfrak{f}\}} \\ z^{\{\mathfrak{s}\}} & z^{\{\mathfrak{s}\}} \end{bmatrix}\right) \begin{bmatrix} \alpha \\ 1 - \alpha \end{bmatrix}.$$

Our findings on stability analysis

- E-Polynomial can be generalized for scalar test problem
- The scalar and 2D stability functions are related:

$$R_1\left(z^{\{\mathfrak{f}\}}, z^{\{\mathfrak{s}\}}\right) = \begin{bmatrix} 1 & 1 \end{bmatrix} R_2\left(\begin{bmatrix} z^{\{\mathfrak{f}\}} & z^{\{\mathfrak{f}\}} \\ z^{\{\mathfrak{s}\}} & z^{\{\mathfrak{s}\}} \end{bmatrix}\right) \begin{bmatrix} \alpha \\ 1-\alpha \end{bmatrix}.$$

Theorem

If a GARK method is A-stable with respect to the 2D test problem, then it is A-stable with respect to the scalar test problem.

Our findings on stability analysis

- E-Polynomial can be generalized for scalar test problem
- The scalar and 2D stability functions are related:

$$R_1\left(z^{\{\mathfrak{f}\}}, z^{\{\mathfrak{s}\}}\right) = \begin{bmatrix} 1 & 1 \end{bmatrix} R_2\left(\begin{bmatrix} z^{\{\mathfrak{f}\}} & z^{\{\mathfrak{f}\}} \\ z^{\{\mathfrak{s}\}} & z^{\{\mathfrak{s}\}} \end{bmatrix}\right) \begin{bmatrix} \alpha \\ 1-\alpha \end{bmatrix}.$$

Theorem

If a GARK method is A-stable with respect to the 2D test problem, then it is A-stable with respect to the scalar test problem.

Theorem

A decoupled GARK method is conditionally stable for the real 2D test problem.

GARK stability hierarchy

$$
\begin{tikzcd}
\text{2D A-stability} \arrow[r, Rightarrow] \arrow[dr, Rightarrow] & \text{Real 2D A-stability} & \\
\text{Algebraic stability} \arrow[r, Rightarrow] & \text{Scalar A-stability} \arrow[r, Rightarrow] & \text{Scalar A}(\alpha)\text{-stability} \\
\text{Scalar L-stability} \arrow[r, Rightarrow] \arrow[ur, Rightarrow] & \text{Scalar L}(\alpha)\text{-stability} \arrow[ur, Rightarrow] &
\end{tikzcd}
$$

- In general, no implication arrows are reversible.

New general stability function for predictor-corrector MrGARK

- Using the particular structure of predictor-corrector coupling, we found the scalar stability function is

$$R_1\left(z^{\{\mathfrak{f}\}}, z^{\{\mathfrak{s}\}}\right) = R\left(\frac{z^{\{\mathfrak{f}\}}}{M}\right)^M + z^{\{\mathfrak{s}\}} \left(b^T + \frac{z^{\{\mathfrak{f}\}}}{M} b^T \left(I_{s\times s} - \frac{z^{\{\mathfrak{f}\}}}{M} A\right)^{-1} \sum_{\lambda=1}^{M} R\left(\frac{z^{\{\mathfrak{f}\}}}{M}\right)^{M-\lambda} A^{\{\mathfrak{f},\mathfrak{s},\lambda\}}\right) R_{\text{int}}(z),$$

  with $z = z^{\{\mathfrak{f}\}} + z^{\{\mathfrak{s}\}}$.
- If $R(\infty) = 0$ for the base method, then the condition

$$A^{\{\mathfrak{f},\mathfrak{s},\lambda\}} A^{-1} \mathbb{1}_s = \mathbb{1}_s$$

  ensures $R_1(\infty, z^{\{\mathfrak{s}\}}) = 0$.

First order multirate methods

- Many coupling structures have been explored.
- Surprising stability limitation:

Theorem

An internally consistent MrGARK method of order exactly one has conditional scalar stability for all but a finite number of multirate ratios.

Higher order multirate methods

- We found a decoupled multirate midpoint method that preserves the algebraic stability, symmetry, and symplecticity of the midpoint method.
- New predictor-corrector up to order four that are close to scalar A-stable:

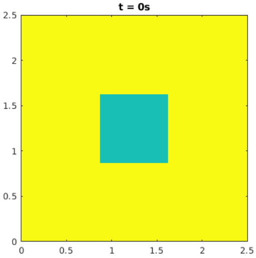
Method	$M = 2$	$M = 3$	$M = 4$	$M = 8$	$M = 16$	$M = 32$
SDIRK 2	84.6°	83.5°	83.2°	83.0°	83.0°	83.0°
SDIRK 3	88.6°	87.8°	87.3°	86.9°	86.8°	86.8°
SDIRK 4	81.7°	81.2°	81.2°	81.2°	81.2°	81.2°

Table: Scalar $L(\alpha)$-stability for new predictor-corrector MrGARK methods.

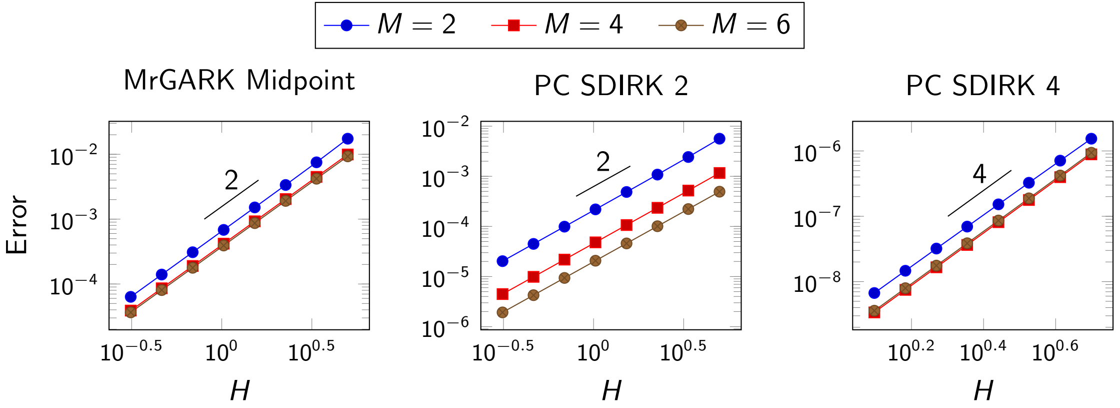
- Internal consistency seems to inhibit stability.

The Gray–Scott model

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

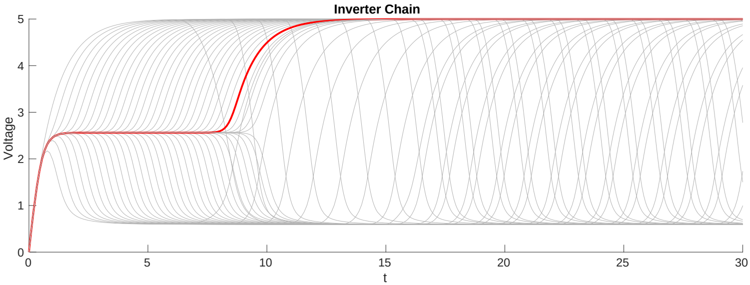


Gray–Scott convergence test



Inverter chain: a classic multirate test problem

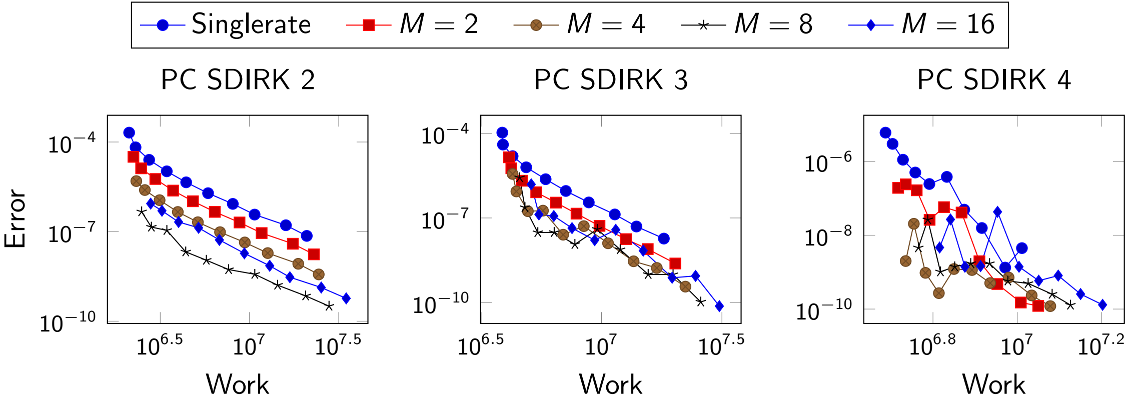
$$
\begin{aligned}
U_1' &= U_{op} - U_1 - g(U_{in}, U_1, U_0), \\
U_i' &= U_{op} - U_i - g(U_{i-1}, U_i, U_0), \qquad i = 2, \dots m, \\
g(U_g, U_D, U_S) &= (\max(U_G - U_S - U_T, 0))^2 - (\max(U_G - U_D - U_T, 0))^2
\end{aligned}
$$



Setup for inverter chain performace results

- Dynamic partitioning is used to select fast parts of circuit
- Performance depends heavily on implementation details
  - Linear solver
  - Stage value predictor
  - Newton tolerances
  - Programming language
- Work is measured by accumulating the dimension of each linear solve performed across integration.

Inverter chain performance results



Conclusions

- Linear stability is surprisingly challenging for multirate methods.
- GARK provides overarching framework to analyze multirate Runge–Kutta methods.
  - Order conditions
  - Stability
- We derive general stability results and fundamental stability limitations.
- New methods are derived up to order four.

Bibliography I

- Gear, C. *Multirate methods for ordinary differential equations.* Tech. rep. (Illinois Univ., Urbana (USA). Dept. of Computer Science, 1974).
- Günther, M. & Sandu, A. Multirate generalized additive Runge Kutta methods. *Numerische Mathematik* **133,** 497–524. ISSN: 0945-3245 (2016).
- Hundsdorfer, W. & Savcenco, V. Analysis of a Multirate Theta-method for Stiff ODEs. *Appl. Numer. Math.* **59,** 693–706. ISSN: 0168-9274 (Mar. 2009).
- Kværnø, A. Stability of multirate Runge–Kutta schemes. (2000).
- Sandu, A. & Günther, M. A generalized-structure approach to additive Runge-Kutta methods. *SIAM Journal on Numerical Analysis* **53,** 17–42 (2015).
- Sarshar, A., Roberts, S. & Sandu, A. Design of High-Order Decoupled Multirate GARK Schemes. *SIAM Journal on Scientific Computing* **41,** A816–A847 (2019).

Questions?

- Slides available at `https://steven-roberts.github.io/`