Fractional-Step Methods: Theory and Practice

Raymond J. Spiteri

Department of Computer Science
University of Saskatchewan

TIM 2023 Los Alamos Workshop on Time Integration for Multiphysics

August 8–11, 2023

Orientation

Saskatchewan

Orientation

Saskatchewan

Easy to draw, hard to spell.

Acknowledgements

- Other guilty parties

S. Wei

- Support from

CIROH

Environment and Climate Change Canada

NSERC CRSNG

Outline

What?

Consider the initial-value problem

$$\frac{d\mathbf{y}}{dt} = \mathbf{f}(t, \mathbf{y}) := \sum_{i=1}^{N} \mathbf{f}^{[i]}(t, \mathbf{y}), \quad \mathbf{y}(0) = \mathbf{y}_0.$$

What?

Consider the initial-value problem

$$\frac{d\mathbf{y}}{dt} = \mathbf{f}(t, \mathbf{y}) := \sum_{i=1}^{2} \mathbf{f}^{[i]}(t, \mathbf{y}), \quad \mathbf{y}(0) = \mathbf{y}_0.$$

What?

Consider the initial-value problem

$$\frac{d\mathbf{y}}{dt} = \mathbf{f}(t, \mathbf{y}) := \sum_{i=1}^{2} \mathbf{f}^{[i]}(t, \mathbf{y}), \quad \mathbf{y}(0) = \mathbf{y}_0.$$

One step of Godunov / Lie–Trotter splitting (first-order accurate) is

Step Δt: $\quad \frac{d\mathbf{y}^{[1]}}{dt} = \mathbf{f}^{[1]}(t, \mathbf{y}^{[1]}), \qquad \mathbf{y}_n^{[1]} = \mathbf{y}_n.$

Step Δt: $\quad \frac{d\mathbf{y}^{[2]}}{dt} = \mathbf{f}^{[2]}(t, \mathbf{y}^{[2]}), \qquad \mathbf{y}_n^{[2]} = \mathbf{y}_{n+1}^{[1]}.$

Set: $\quad \mathbf{y}_{n+1} = \mathbf{y}_{n+1}^{[2]}.$

Bubble Diagram

Godunov / Lie–Trotter

Why?

- feasibility
- efficiency

Why?

- feasibility
- efficiency

Caveat emptor!

How?

- linear / non-linear
- physics
- stiff / non-stiff (includes geometry)
- scale
- exact flow
- co-simulation

How?

- linear / non-linear
- physics
- stiff / non-stiff (includes geometry)
- scale
- exact flow
- co-simulation

Fractional steps

Define $\psi_{\Delta t}^{(j)}$ recursively by

$$\psi_{\Delta t}^{(0)} = \mathrm{Id},$$
$$\psi_{\Delta t}^{(j)} = \phi^{[N]}{}_{\alpha_j^{[N]}\Delta t} \circ \cdots \circ \phi^{[1]}{}_{\alpha_j^{[1]}\Delta t} \circ \psi_{\Delta t}^{(j-1)}, \qquad j = 1, 2, \ldots, s.$$

Then

$$\psi_{\Delta t}^{(s)} \approx \phi_{\Delta t}.$$

Fractional steps

Define $\psi_{\Delta t}^{(j)}$ recursively by

$$\psi_{\Delta t}^{(0)} = \mathrm{Id},$$
$$\psi_{\Delta t}^{(j)} = \phi^{[N]}_{\alpha_j^{[N]}\Delta t} \circ \cdots \circ \phi^{[1]}_{\alpha_j^{[1]}\Delta t} \circ \psi_{\Delta t}^{(j-1)}, \qquad j = 1, 2, \ldots, s.$$

Then

$$\psi_{\Delta t}^{(s)} \approx \phi_{\Delta t}.$$

Fractional steps

Define $\psi_{\Delta t}^{(j)}$ recursively by

$$\psi_{\Delta t}^{(0)} = \mathrm{Id},$$
$$\psi_{\Delta t}^{(j)} = \phi^{[N]}{}_{\alpha_j^{[N]}\Delta t} \circ \cdots \circ \phi^{[1]}{}_{\alpha_j^{[1]}\Delta t} \circ \psi_{\Delta t}^{(j-1)}, \qquad j = 1, 2, \ldots, s.$$

Then

$$\psi_{\Delta t}^{(s)} \approx \phi_{\Delta t}.$$

Fractional steps

Define $\psi_{\Delta t}^{(j)}$ recursively by

$$\psi_{\Delta t}^{(0)} = \mathrm{Id},$$
$$\psi_{\Delta t}^{(j)} = \phi^{[N]}{}_{\alpha_j^{[N]}\Delta t} \circ \cdots \circ \phi^{[1]}{}_{\alpha_j^{[1]}\Delta t} \circ \psi_{\Delta t}^{(j-1)}, \qquad j = 1, 2, \ldots, s.$$

Then

$$\psi_{\Delta t}^{(s)} \approx \phi_{\Delta t}.$$

Practical matters

Order conditions on $\alpha_j^{[1]}$, $\alpha_j^{[2]}$,..., $\alpha_j^{[N]}$, $j = 1, 2, \ldots, s$, so that

$$\|\phi_{\Delta t} - \psi_{\Delta t}^{(s)}\| = \mathcal{O}((\Delta t)^{p+1}).$$

Typically, we must approximate $\phi_t^{[i]}$ numerically to $\mathcal{O}((\Delta t)^p)$; e.g.,

- $p = 1$: forward Euler for $\phi_{\Delta t}^{[1]}$, forward Euler for $\phi_{\Delta t}^{[2]}$,
- $p = 1$: forward Euler for $\phi_{\Delta t}^{[1]}$, backward Euler for $\phi_{\Delta t}^{[2]}$.

If $\phi_{\Delta t}^{[i]}$ are approximated by Runge–Kutta methods, FS methods can be described by (generalized) additive Runge–Kutta methods.

Practical matters

Order conditions on $\alpha_j^{[1]}$, $\alpha_j^{[2]}$,..., $\alpha_j^{[N]}$, $j = 1, 2, \ldots, s$, so that

$$\|\phi_{\Delta t} - \psi_{\Delta t}^{(s)}\| = \mathcal{O}((\Delta t)^{p+1}).$$

Typically, we must approximate $\phi_t^{[i]}$ numerically to $\mathcal{O}((\Delta t)^p)$; e.g.,

- $p = 1$: forward Euler for $\phi_{\Delta t}^{[1]}$, forward Euler for $\phi_{\Delta t}^{[2]}$,
- $p = 1$: forward Euler for $\phi_{\Delta t}^{[1]}$, backward Euler for $\phi_{\Delta t}^{[2]}$.

If $\phi_{\Delta t}^{[i]}$ are approximated by Runge–Kutta methods, FS methods can be described by (generalized) additive Runge–Kutta methods.

Classical low-order splittings

- Original Strang–Marchuk (second order):

j	$\alpha_j^{[1]}$	$\alpha_j^{[2]}$
1	$\frac{1}{2}$	1
2	$\frac{1}{2}$	0

Classical low-order splittings

Original Strang–Marchuk

Classical low-order splittings

- Symmetric Strang–Marchuk (second order):

j	$\alpha_j^{[1]}$	$\alpha_j^{[2]}$
1	$\frac{1}{2}$	$\frac{1}{2}$
2	0	$\frac{1}{2}$
3	$\frac{1}{2}$	0

Classical low-order splittings

Symmetric Strang–Marchuk

Classical low-order splittings

- Symmetric Strang–Marchuk (second order):

j	$\alpha_j^{[1]}$	$\alpha_j^{[2]}$	$\mathcal{O}$
1	$\frac{1}{2}$	$\frac{1}{2}$	$\{[1],[2]\}$
2	$\frac{1}{2}$	$\frac{1}{2}$	$\{[2],[1]\}$

Classical high-order splitting

Third order: Ruth

j	$\alpha_j^{[1]}$	$\alpha_j^{[2]}$
1	$\frac{7}{24}$	$\frac{2}{3}$
2	$\frac{3}{4}$	$-\frac{2}{3}$
3	$-\frac{1}{24}$	1

Classical high-order splitting

Third order: Ruth

j	$\alpha_j^{[1]}$	$\alpha_j^{[2]}$
1	$\frac{7}{24}$	$\frac{2}{3}$
2	$\frac{3}{4}$	$-\frac{2}{3}$
3	$-\frac{1}{24}$	1

Theory (Sheng–Suzuki; Goldman–Kaper)

- Sheng–Suzuki Theorem:

  Real-valued OS methods of order greater than two require backward time integration.

Theory (Sheng–Suzuki; Goldman–Kaper)

- Goldman–Kaper Theorem:

  Real-valued OS methods of order greater than two require backward time integration in each operator.

Theory (Sheng–Suzuki; Goldman–Kaper)

- Goldman–Kaper Theorem:

  Real-valued OS methods of order greater than two require backward time integration in each operator.

  Unstable for some equations, e.g., parabolic.

Theory (Sheng–Suzuki; Goldman–Kaper)

- Goldman–Kaper Theorem:

  Real-valued OS methods of order greater than two require backward time integration in each operator.

  Unstable for some equations, e.g., parabolic.

  Is this a deal breaker?

Theory (Sheng–Suzuki; Goldman–Kaper)

- Goldman–Kaper Theorem:

  Real-valued OS methods of order greater than two require backward time integration in each operator.

  Unstable for some equations, e.g., parabolic.

Is this a deal breaker?

Use (G)ARK theory to understand.

FSRK Representation

Theorem 1

An N-split FSRK method can be written as an extended Butcher tableau of the form

$$\begin{array}{c:c:c:c|c:c:c:c} \mathbf{c}^{[1]} & \mathbf{c}^{[2]} & \dots & \mathbf{c}^{[N]} & \mathbf{A}^{[1]} & \mathbf{A}^{[2]} & \dots & \mathbf{A}^{[N]} \\ \hline & & & & \mathbf{b}^{[1]} & \mathbf{b}^{[2]} & \dots & \mathbf{b}^{[N]} \end{array}$$

FSRK Representation

where

N operators

$$
\begin{array}{c|c}
\mathbf{c}^{[\ell]} & \mathbf{A}^{[\ell]} \\ \hline
 & \mathbf{b}^{[\ell]}
\end{array}
=
\begin{array}{c|ccc:ccc:c:ccc}
0 & 0 & \cdots & 0 & & & & & & & \\
\alpha_1^{[\ell]}\tilde{\mathbf{c}}_1^{[\ell]} & \vdots & \alpha_1^{[\ell]}\tilde{\mathbf{A}}_1^{[\ell]} & \vdots & & & & & & & \\
\alpha_1^{[\ell]}\mathbb{1} & \vdots & \alpha_1^{[\ell]}\mathbb{1}\tilde{\mathbf{b}}_1^{[\ell]} & \vdots & & & & & & & \\
\vdots & \vdots & \vdots & \vdots & & & & & & & \\ \hdashline
\vdots & \vdots & \alpha_1^{[\ell]}\mathbb{1}\tilde{\mathbf{b}}_1^{[\ell]} & \vdots & 0 & \cdots & 0 & & & & \\
\alpha_1^{[\ell]}\mathbb{1} + \alpha_2^{[\ell]}\tilde{\mathbf{c}}_2^{[\ell]} & \vdots & \vdots & \vdots & \vdots & \alpha_2^{[\ell]}\tilde{\mathbf{A}}_2^{[\ell]} & \vdots & & & & \\
\left(\alpha_1^{[\ell]} + \alpha_2^{[\ell]}\right)\mathbb{1} & \vdots & \vdots & \vdots & \vdots & \alpha_2^{[\ell]}\mathbb{1}\tilde{\mathbf{b}}_2^{[\ell]} & \vdots & & & & \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & & & & \\ \hdashline
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & & & \\ \hdashline
\left(\sum_{i=1}^{s-1}\alpha_i^{[\ell]}\right)\mathbb{1} & \vdots & \alpha_1^{[\ell]}\mathbb{1}\tilde{\mathbf{b}}_1^{[\ell]} & \vdots & \vdots & \alpha_2^{[\ell]}\mathbb{1}\tilde{\mathbf{b}}_2^{[\ell]} & \vdots & \ddots & 0 & \cdots & 0 \\
\left(\sum_{i=1}^{s-1}\alpha_i^{[\ell]}\right)\mathbb{1} + \alpha_s^{[\ell]}\tilde{\mathbf{c}}_s^{[\ell]} & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \alpha_s^{[\ell]}\tilde{\mathbf{A}}_s^{[\ell]} & \vdots \\
\sum_{i=1}^{s}\alpha_i^{[\ell]}\mathbb{1} & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \alpha_s^{[\ell]}\mathbb{1}\tilde{\mathbf{b}}_s^{[\ell]} & \vdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
\sum_{i=1}^{s}\alpha_i^{[\ell]}\mathbb{1} & 0 & \alpha_1^{[\ell]}\mathbb{1}\tilde{\mathbf{b}}_1^{[\ell]} & 0 & 0 & \alpha_2^{[\ell]}\mathbb{1}\tilde{\mathbf{b}}_2^{[\ell]} & 0 & \cdots & 0 & \alpha_s^{[\ell]}\mathbb{1}\tilde{\mathbf{b}}_s^{[\ell]} & 0 \\ \hline
 & 0 & \alpha_1^{[\ell]}\tilde{\mathbf{b}}_1^{[\ell]} & 0 & 0 & \alpha_2^{[\ell]}\tilde{\mathbf{b}}_2^{[\ell]} & 0 & \cdots & 0 & \alpha_s^{[\ell]}\tilde{\mathbf{b}}_s^{[\ell]} & 0
\end{array}
$$

s stages

FSRK Representation Example

Godunov / Lie–Trotter with tableau

Linear Stability Analysis

Theorem 2

Consider the linear test equation $\frac{\mathrm{d}y}{\mathrm{d}t} = \sum_{\ell=1}^{N} \lambda^{[\ell]} y$*. The stability function of the FSRK method is given by*

$$R(z^{[1]}, z^{[2]}, \ldots, z^{[N]}) = \prod_{k=1}^{s} \prod_{\ell=1}^{N} R_k^{[\ell]}(\alpha_k^{[\ell]} z^{[\ell]}),$$

where $z^{[\ell]} = \Delta t \lambda^{[\ell]}$ *and* $R_k^{[\ell]}(z^{[\ell]})$ *is the stability function of the Runge–Kutta method at stage* k *applied to operator* ℓ*.*

Linear Stability Analysis

Theorem 2

Consider the linear test equation $\frac{\mathrm{d}y}{\mathrm{d}t} = \sum_{\ell=1}^{N} \lambda^{[\ell]} y$. *The stability function of the FSRK method is given by*

$$R(z^{[1]}, z^{[2]}, \ldots, z^{[N]}) = \prod_{k=1}^{s} \prod_{\ell=1}^{N} R_k^{[\ell]}(\alpha_k^{[\ell]} z^{[\ell]}),$$

where $z^{[\ell]} = \Delta t \lambda^{[\ell]}$ *and* $R_k^{[\ell]}(z^{[\ell]})$ *is the stability function of the Runge–Kutta method at stage* k *applied to operator* ℓ.

Linear Stability Analysis

Theorem 2

Consider the linear test equation $\frac{\mathrm{d}y}{\mathrm{d}t} = \sum_{\ell=1}^{N} \lambda^{[\ell]} y$. *The stability function of the FSRK method is given by*

$$R(z^{[1]}, z^{[2]}, \ldots, z^{[N]}) = \prod_{k=1}^{s} \prod_{\ell=1}^{N} R_k^{[\ell]}(\alpha_k^{[\ell]} z^{[\ell]}),$$

where $z^{[\ell]} = \Delta t \lambda^{[\ell]}$ *and* $R_k^{[\ell]}(z^{[\ell]})$ *is the stability function of the Runge–Kutta method at stage* k *applied to operator* ℓ.

Brusselator

The Brusselator problem:

$$\frac{\partial T}{\partial t} = D_1 \frac{\partial^2 T}{\partial x^2} + \alpha - (\beta + 1)T + T^2 C,$$
$$\frac{\partial C}{\partial t} = D_2 \frac{\partial^2 C}{\partial x^2} + \beta T - T^2 C,$$

where T and C are concentrations of different chemical species.

Brusselator

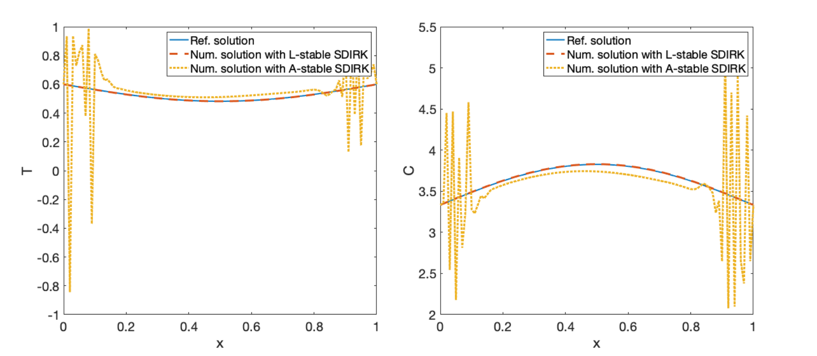
The Brusselator problem:

$$\frac{\partial T}{\partial t} = D_1 \frac{\partial^2 T}{\partial x^2} + \alpha - (\beta + 1)T + T^2 C,$$
$$\frac{\partial C}{\partial t} = D_2 \frac{\partial^2 C}{\partial x^2} + \beta T - T^2 C,$$

where T and C are concentrations of different chemical species.

Numerical PDE people "know" that integrating the diffusion operator with an L-stable method can "better" control high-wavenumber instability than with an A-stable method.

Brusselator



Brusselator

$$R_{\text{Heun}}(z) = 1 + z + \frac{z^2}{2}, \qquad R_{\text{SDIRK}(2,2)}(z) = \frac{1 + z(1 - 2\gamma)}{(1 - \gamma z)^2}$$

$$R_{\text{FSRK}}(z) = R_{\text{Heun}}(1 z_R)\ R^2_{\text{SDIRK}(2,2)}\left(\frac{1}{2} z_D\right)$$

Brusselator

$$R_{\text{Heun}}(z) = 1 + z + \frac{z^2}{2}, \qquad R_{\text{SDIRK(2,2)}}(z) = \frac{1 + z(1-2\gamma)}{(1-\gamma z)^2}$$

$$R_{\text{FSRK}}(z) = R_{\text{Heun}}(1 z_R)\ R^2_{\text{SDIRK(2,2)}}\left(\frac{1}{2} z_D\right)$$

Let $z_D = \frac{1}{r} z_R$. Then

$$\lim_{|z|\to\infty} |R_{\text{FSRK}}(z)| = \frac{2r^2(1-2\gamma)^2}{\gamma^4}$$

Brusselator

$$R_{\text{Heun}}(z) = 1 + z + \frac{z^2}{2}, \qquad R_{\text{SDIRK}(2,2)}(z) = \frac{1 + z(1-2\gamma)}{(1-\gamma z)^2}$$

$$R_{\text{FSRK}}(z) = R_{\text{Heun}}(1 z_R)\ R^2_{\text{SDIRK}(2,2)}\left(\frac{1}{2} z_D\right)$$

Let $z_D = \frac{1}{r} z_R$. Then

$$\lim_{|z|\to\infty} |R_{\text{FSRK}}(z)| = \frac{2r^2(1-2\gamma)^2}{\gamma^4} < 1 \quad \text{for } r < \frac{\gamma^2}{\sqrt{2}(1-2\gamma)}$$

Brusselator

$$R_{\text{Heun}}(z) = 1 + z + \frac{z^2}{2}, \qquad R_{\text{SDIRK}(2,2)}(z) = \frac{1 + z(1-2\gamma)}{(1-\gamma z)^2}$$

$$R_{\text{FSRK}}(z) = R_{\text{Heun}}(1 z_R)\ R^2_{\text{SDIRK}(2,2)}\left(\frac{1}{2} z_D\right)$$

Let $z_D = \frac{1}{r} z_R$. Then for $\gamma = 1 + 1/\sqrt{2}$,

$$\lim_{|z| \to \infty} |R_{\text{FSRK}}(z)| = \frac{2r^2(1-2\gamma)^2}{\gamma^4} < 1 \quad \text{for } r < \frac{\gamma^2}{\sqrt{2}(1-2\gamma)} \approx 0.85.$$

Brusselator

$$R_{\text{Heun}}(z) = 1 + z + \frac{z^2}{2}, \qquad R_{\text{SDIRK(2,2)}}(z) = \frac{1 + z(1-2\gamma)}{(1-\gamma z)^2}$$

$$R_{\text{FSRK}}(z) = R_{\text{Heun}}(1 z_R)\ R^2_{\text{SDIRK(2,2)}}\left(\frac{1}{2} z_D\right)$$

Let $z_D = \frac{1}{r}\, z_R$. Then for $\gamma = 1 + 1/\sqrt{2}$,

$$\lim_{|z|\to\infty} |R_{\text{FSRK}}(z)| = \frac{2r^2(1-2\gamma)^2}{\gamma^4} < 1 \quad \text{for } r < \frac{\gamma^2}{\sqrt{2}(1-2\gamma)} \approx 0.85.$$

L-stable + ERK can be A-stable

Negative coefficients

Consider solving the linear equation

$$\frac{\mathrm{d}y}{\mathrm{d}t} = \lambda^{[1]}y + \lambda^{[2]}y.$$

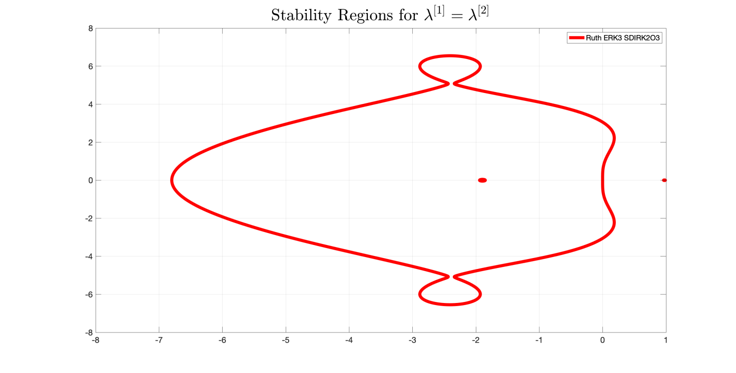
For simplicity, let $\lambda^{[1]} = \lambda^{[2]}$.

Negative coefficients

Use Ruth's method with Kutta ERK3 for operator 1 and A-stable SDIRK(2,3) for operator 2:

$$R(z) = R_{\text{ERK3}}\left(\frac{7}{24}z\right) R_{\text{ERK3}}\left(\frac{3}{4}z\right) R_{\text{ERK3}}\left(-\frac{1}{24}z\right)$$
$$R_{\text{SDIRK(2,3)}}\left(\frac{2}{3}z\right) R_{\text{SDIRK(2,3)}}\left(-\frac{2}{3}z\right) R_{\text{SDIRK(2,3)}}\left(1z\right)$$

Negative coefficients



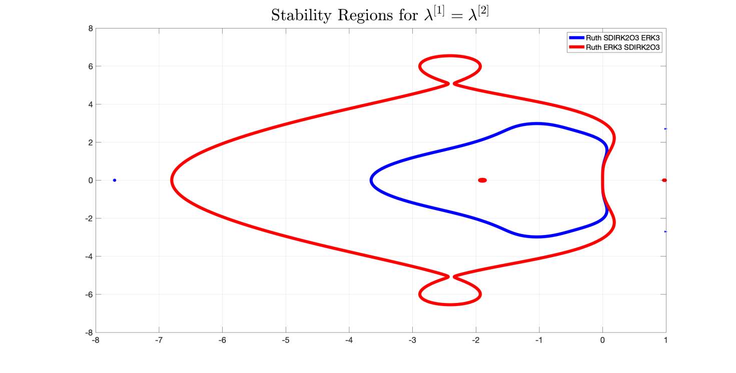
There's a hole in my stability region (for $z \approx -1.9$).

Negative coefficients

Now use Ruth's method with A-stable SDIRK(2,3) for operator 1 and Kutta ERK3 for operator 2:

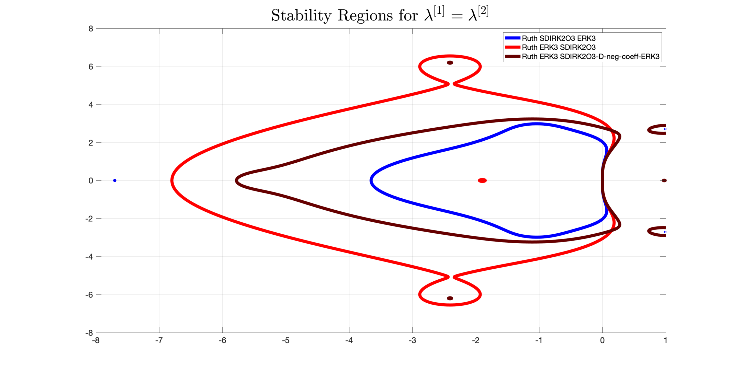
$$R(z) = R_{\text{SDIRK(2,3)}}\left(\frac{7}{24}z\right) R_{\text{SDIRK(2,3)}}\left(\frac{3}{4}z\right) R_{\text{SDIRK(2,3)}}\left(-\frac{1}{24}z\right)$$
$$R_{\text{ERK3}}\left(\frac{2}{3}z\right) R_{\text{ERK3}}\left(-\frac{2}{3}z\right) R_{\text{ERK3}}\left(1z\right)$$

Negative coefficients



No more hole in my stability region ($z \approx -7.7$).

Negative coefficients



Explicit method for unstable integration.

FSRK method anatomy

Require: $\left\{\mathbf{F}^{[k]}\right\}$, $\left\{\alpha_j^{[k]}\right\}$, $\left\{\Phi_j^{[k]}\right\}$, $\{\mathcal{O}_j\}$, t_n, t_{n+1}, $\mathbf{U}_n$.

$\tilde{\mathbf{U}}_0 = \mathbf{U}_n$; $\Delta t_n = t_{n+1} - t_n$; $t^{[k]} = t_n$ for $k = 1, 2, \dots, N$

for $j = 1$ to n_s **do**

    for $k \in \mathcal{O}_j$ **do**

        $(k, \alpha) = \left(k_j, \alpha_j^{[k]}\right)$

        Solve $\left\{\dot{\tilde{\mathbf{U}}}\right\}^{[k]} = \mathbf{F}^{[k]}\left(t, \left\{\tilde{\mathbf{U}}\right\}^{[k]}\right)$, $\left\{\tilde{\mathbf{U}}(t^{[k]})\right\} = \tilde{\mathbf{U}}_0^{[k]}$,

        $t \in \left[t^{[k]}, t^{[k]} + \alpha \Delta t_n\right]$, using $\Phi_j^{[k]}$

        $t^{[k]} = t^{[k]} + \alpha \Delta t_n$

        $\left\{\tilde{\mathbf{U}}_0\right\}^{[k]} = \left\{\tilde{\mathbf{U}}\right\}^{[k]}(t^{[k]})$

    end for

end for

Return $\mathbf{U}_{n+1} = \tilde{\mathbf{U}}_0$

FSRK method design principles

- match desirable characteristics of sub-integrator to operator
- minimize unstable sub-integration / maximize method stability
- maximize accuracy
- minimize computational expense

Niederer benchmark: problem

Monodomain model, 3D, TTP cell model
PDE:

$$\chi C_m \frac{\partial v}{\partial t} = \nabla \cdot \left(\frac{\lambda}{1+\lambda} \sigma_i \nabla v \right) - \chi \Big(I_{\mathrm{ion}}(\mathbf{s}, v) + I_{\mathrm{stim}}(t, \mathbf{x}) \Big)$$

$$\frac{\partial \mathbf{s}}{\partial t} = \mathbf{g}(\mathbf{s}, v)$$

Discretized and split:

$$\begin{bmatrix} \dot{\mathbf{V}} \\ \dot{\mathbf{S}} \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{1}{C_m \chi} \sigma^{-1} \mathbf{D} \mathbf{V} \\ \mathbf{0} \end{bmatrix}}_{\mathbf{F}^{[1]}} + \underbrace{\begin{bmatrix} -\frac{1}{C_m} \left(\mathbf{I}_{\mathrm{ion}}(\mathbf{S}, \mathbf{V}) + \mathbf{I}_{\mathrm{stim}}(t) \right) \\ \mathbf{G}(\mathbf{S}, \mathbf{V}) \end{bmatrix}}_{\mathbf{F}^{[2]}}$$

Niederer benchmark: problem

Monodomain model, 3D, TTP cell model

Niederer benchmark: ingredients

OS methods:

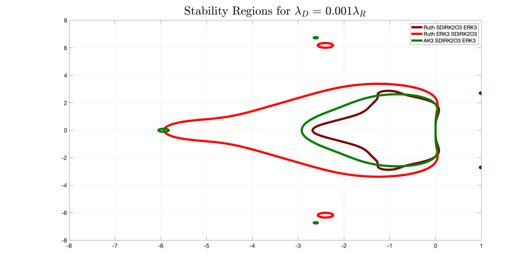
- Ruth (6 sub-integrations)
- main method from Emb 3/2 AKS (6 sub-integrations, palindromic, optimized LEM)
- OS(4,3)[7] (7 sub-integrations, optimized LEM)

Sub-integrators:

- reaction: explicit Kutta method (3 stages)
- diffusion: SDIRK (2 stages, A-stable)

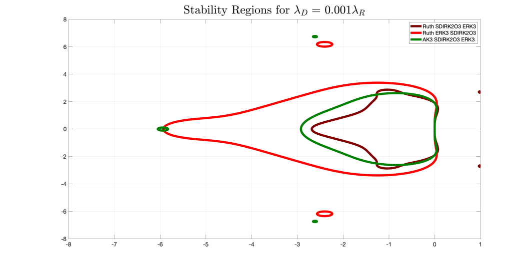
Niederer benchmark: stability (Ruth, AKS3)

$\lambda_D = 0.001\lambda_R$



Niederer benchmark: stability (Ruth, AKS3)

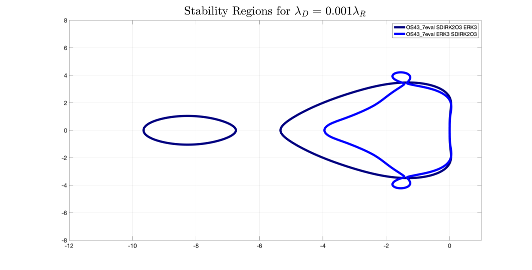
$\lambda_D = 0.001\lambda_R$



Method	Δt	Error (%)	CPU (s)
Ruth DR	0.0028	0.07	10,794
Ruth RD	0.0062	0.039	4,314
AKS3	0.0031	2.3	9,970

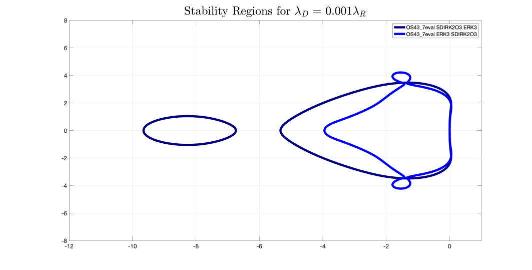
Niederer benchmark: stability (OS(4,3)[7])

$\lambda_D = 0.001\lambda_R$



Niederer benchmark: stability (OS(4,3)[7])

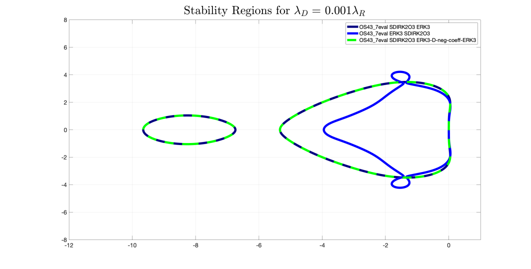
$\lambda_D = 0.001\lambda_R$



Method	Δt	Error (%)	CPU (s)
DR	0.0057	1.50	6,290
RD	0.0041	0.081	8,555

Niederer benchmark: stability (OS(4,3)[7])

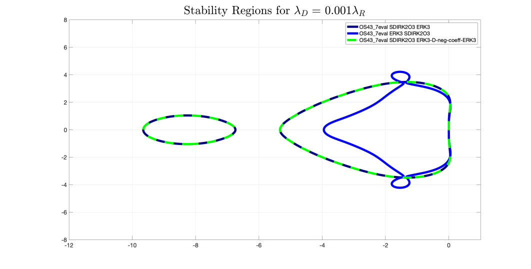
$\lambda_D = 0.001\lambda_R$



Method	Δt	Error (%)	CPU (s)
DR	0.0057	1.50	6,290
RD	0.0041	0.081	8,555

Niederer benchmark: stability (OS(4,3)[7])

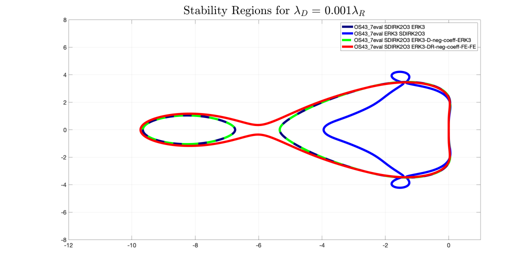
$\lambda_D = 0.001\lambda_R$



Method	Δt	Error (%)	CPU (s)
DR	0.0057	1.50	6,290
RD	0.0041	0.081	8,555
DR ERK3$^-$	0.0057	1.50	5,157

Niederer benchmark: stability (OS(4,3)[7])

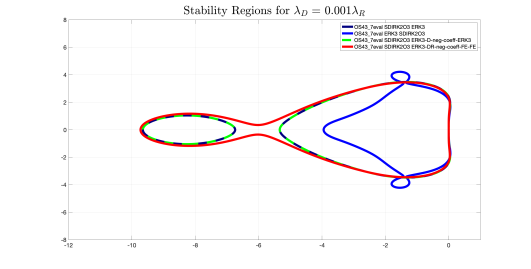
$\lambda_D = 0.001\lambda_R$



Method	Δt	Error (%)	CPU (s)
DR	0.0057	1.50	6,290
RD	0.0041	0.081	8,555
DR ERK3$^-$	0.0057	1.50	5,157

Niederer benchmark: stability (OS(4,3)[7])

$\lambda_D = 0.001\lambda_R$



Method	Δt	Error (%)	CPU (s)
DR	0.0057	1.50	6,290
RD	0.0041	0.081	8,555
DR ERK3$^-$	0.0057	1.50	5,157
DR FE$^-$	0.01	1.3	2,228

Niederer benchmark: preliminary observations

Summary of preliminary observations:

- simulations are stability constrained at 5% MRMS error level
- using explicit methods for unstable implicit sub-integration removes poles in linear stability regions
- using low-order integration for unstable sub-integration reduces computation time without loss of accuracy

Conclusions

- FSRK methods can be represented as (G)ARK methods.
- Linear stability of FSRK methods is the product of individual sub-integrators with modified argument.
- Order of sub-integrations matters for linear stability if $\alpha^{[\ell]}$ are not permutations of each other.
- High-order splitting not doomed to fail; negative coefficients can lead to poles in bad places, but there is a potential fix.
- Splitting/sub-integrators/design principles not crystal clear, but some progress and working on it.

Conclusions

- FSRK methods can be represented as (G)ARK methods.
- Linear stability of FSRK methods is the product of individual sub-integrators with modified argument.
- Order of sub-integrations matters for linear stability if $\alpha^{[\ell]}$ are not permutations of each other.
- High-order splitting not doomed to fail; negative coefficients can lead to poles in bad places, but there is a potential fix.
- Splitting/sub-integrators/design principles not crystal clear, but some progress and working on it.

Collaborators welcome!

Future Work

- Applications in hydrology and plasma physics