Strong Stability Preserving Integrating Factor Runge-Kutta Methods

Sigal Gottlieb

Joint work with: Leah Isherwood and Zachary J. Grant

Advances in PDEs: Theory, Computation and Application to CFD
ICERM August 20-24, 2018

UMass Dartmouth

Center for Scientific Computing & Visualization Research

Overview

1. In memory of Saul (Shalom) Abarbanel
2. Strong Stability Preserving (SSP)
   - Strong Stability Preserving Runge-Kutta (SSPRK) methods
3. Exponential Integrators Motivation
   - Integrating Factor (IF) methods
   - Integrating Factor Runge-Kutta (IFRK) methods
   - Strong Stability Preserving Integrating Factor Runge-Kutta (SSPIFRK) methods
4. Numerical Results

- **Transformations:** Saul/Shalom
- **Communication:** Joan and Bob
- **Optimality:** Postdoc positions
- **Relativity:** Uncle Shalom to me and my brothers
- **Uniqueness:** Many Friday night dinners over 40+ years
- **Positivity preservation:** ICASE names, insult-work sessions
- **Energy maximization** Travel and NSF panels

Strong Stability Preserving (SSP) Motivation

Consider the hyperbolic partial differential equation

$$U_t + f(U)_x = 0.$$

Method of lines approach: we discretize the problem in space, to obtain some ODE of the form

$$\boldsymbol{u}_t = F(\boldsymbol{u})$$

and we evolve this ODE in time using standard time-stepping methods such as Runge–Kutta methods.

Strong Stability Preserving (SSP) Motivation

- Given a *linear* differential equation and consistent *linear* numerical method, the Lax-Richtmyer equivalence theorem tells us that linear stability is necessary and sufficient for convergence.

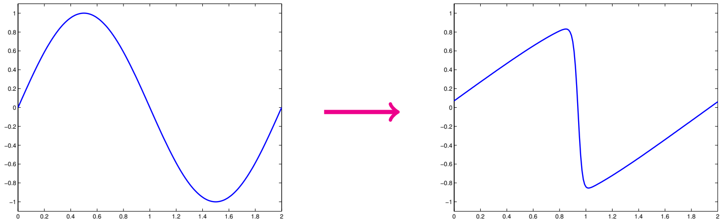
$$\textit{consistency} + \textit{stability} \iff \textit{convergence}$$

- For *nonlinear* PDEs, if a numerical method is consistent and its linearization is L_2 stable and adequately dissipative, then for **sufficiently smooth** problems the nonlinear approximation is convergent (Strang 1964).

$\Rightarrow$ For **smooth** solutions, we look at L_2 linear stability (plus some dissipativity) to prove convergence.

Strong Stability Preserving (SSP) Motivation

Hyperbolic conservation laws tend to have solutions which start with or develop discontinuities or steep gradients over time.



If the solution is **discontinuous**, then we can get oscillations, non-physical negative values, etc.
For a nonlinear problem with discontinuous solutions, linear L_2 stability analysis is is not sufficient for convergence.
We build spatial discretizations which satisfy some **nonlinear, non-inner product** stability properties.

Strong Stability Preserving (SSP)

Consider ODE system

$$\boldsymbol{u}_t = F(\boldsymbol{u}),$$

where spatial discretization $F(\boldsymbol{u})$ is carefully chosen,
(e.g. TVD, TVB, ENO, WENO, positivity or maximum principle preserving)
so that the solution from the forward Euler method

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^n + \Delta t F(\boldsymbol{u}^n),$$

satisfies the monotonicity (or strong stability) requirement

$$||\boldsymbol{u}^{n+1}|| \leq ||\boldsymbol{u}^n||,$$

in some norm, semi-norm, or convex functional $||\cdot||$, for a suitably restricted timestep

$$\Delta t \leq \Delta t_{\text{FE}}.$$

Strong Stability Preserving (SSP) Motivation

We now have a **spatial discretization** that is stable **when coupled with forward Euler**. But in practice, we need to use higher order methods. There has been much work designing spatial discretizations which satisfy certain **nonlinear**, **non-inner product** stability properties when coupled with Forward Euler.

- Forward Euler is only first order accurate error$=O(\Delta t)$
- Linear stability region fails to capture the imaginary axis

These issues can be handled by using a **higher order** time integrator.

How can these time integrators also preserve the **monotonicity** property guaranteed by the Forward Euler time step?

Strong Stability Preserving (SSP)

Given an operator $F(u)$ with forward Euler monotonicity condition

$$\|\mathbf{u}^{n+1}\| = \|\mathbf{u}^n + \Delta t F(\mathbf{u}^n)\| \leq \|\mathbf{u}^n\|$$

under time-step restriction $\Delta t \leq \Delta t_{\text{FE}}$.
Creating the higher order time integrator is done by

- Decomposing a higher order time-stepping method into convex combinations of forward Euler so that any convex-functional monotonicity property $\|\mathbf{u}^{n+1}\| \leq \|\mathbf{u}^n\|$ will be preserved under a time step restriction $\Delta t \leq \mathcal{C} \Delta t_{\text{FE}}$.
- Any higher order method that can be decomposed in this way is called strong stability preserving with SSP coefficient $\mathcal{C}$.
- This convex combination condition is both necessary and sufficient.

Decoupling the analysis

$\mathcal{C}$ is a property only of the time-integrator.
Δt_{FE} is a property of the spatial discretization.

Example: Shu-Osher third order method

For example, Shu-Osher's third order SSP method:
eSSPRK(3,3):

$$\begin{aligned}
u^{(1)} &= u^n + \Delta t F(u^n) \\
u^{(2)} &= \frac{3}{4} u^n + \frac{1}{4} \left(u^{(1)} + \Delta t F(u^{(1)}) \right) \\
u^{n+1} &= \frac{1}{3} u^n + \frac{2}{3} \left(u^{(2)} + \Delta t F(u^{(2)}) \right)
\end{aligned}$$

This three stage method has an SSP coefficient $\mathcal{C}=1$. i.e $\Delta t = \Delta t_{FE}$

The notation eSSPRK(s,p) denotes an explicit SSP Runge–Kutta method with s stages and of order p.

SSP Runge-Kutta methods

For example, an s-stage explicit Runge–Kutta method can be written as:

$$\begin{aligned} u^{(0)} &= u^n, \\ u^{(i)} &= \sum_{j=0}^{i-1} \left(\alpha_{i,j} u^{(j)} + \Delta t \beta_{i,j} F(u^{(j)}) \right), \quad i = 1, ..., s \\ u^{n+1} &= u^{(s)}. \end{aligned}$$

If all the coefficients $\alpha_{i,j}$ and $\beta_{i,j}$ are non-negative, and a given $\alpha_{i,j}$ is zero only if its corresponding $\beta_{i,j}$ is zero,then each stage can be rearranged into a convex combination of forward Euler steps

$$\|u^{(i)}\| = \left\| \sum_{j=0}^{i-1} \left(\alpha_{i,j} u^{(j)} + \Delta t \beta_{i,j} F(u^{(j)}) \right) \right\| \leq \sum_{j=0}^{i-1} \alpha_{i,j} \left\| u^{(j)} + \Delta t \frac{\beta_{i,j}}{\alpha_{i,j}} F(u^{(j)}) \right\| \leq \|u^n\|$$

provided that the time-step satisfies

$$\Delta t \leq \min_{i,j} \frac{\alpha_{i,j}}{\beta_{i,j}} \Delta t_{\text{FE}}.$$

Note that for consistency $\sum_{j=0}^{i-1} \alpha_{i,j} = 1$.

SSP Runge–Kutta Methods: bounds and barriers

- Explicit SSP Runge–Kutta methods
  - Order barrier $p \leq 4$
  - SSP coefficent $\mathcal{C} \leq s$
- Implicit SSP Runge–Kutta methods
  - Order barrier $p \leq 6$
  - SSP coefficient $\mathcal{C} \leq 2s$ for order $p \geq 2$
- Implicit-explicit (IMEX) SSP Runge–Kutta methods
  - Order barrier $p \leq 4$
  - SSP coefficient between $\mathcal{C} \leq s$ and $\mathcal{C} \leq 2s$

Exponential Integrators Motivation

We consider a problem of the form

$$u_t = Lu + N(u)$$

As before, the two components satisfy

$$\|u^n + \Delta t N(u^n)\| \leq \|u^n\| \qquad \text{for} \qquad \Delta t \leq \Delta t_{\text{FE}}$$

while taking a forward Euler step using the linear component Lu results in the strong stability condition

$$\|u^n + \Delta t L u^n\| \leq \|u^n\| \qquad \text{for} \qquad \Delta t \leq \tilde{\Delta} t_{\text{FE}}$$

where $\tilde{\Delta} t_{\text{FE}} << \Delta t_{\text{FE}}$.

Our focus is how to preserve the nonlinear, non-inner product monotonicity properties, while avoiding the severe time-step restriction coming from the linear component.

Motivating example

Consider

$$u_t + 10u_x + \left(\frac{1}{2}u^2\right)_x = 0 \qquad u(0,x) = \begin{cases} 1, & \text{if } 0 \le x \le 1/2 \\ 0, & \text{if } x > 1/2 \end{cases}$$

using first order upwind finite difference.
The relevant TVD time-step restrictions are:

Explicit SSPRK(3,3): $\Delta t \le 0.09\Delta x$
Implicit SSPRK(2,3): $\Delta t \le 0.248\Delta x$
IMEX SSP(3,3,$\frac{1}{10}$): $\Delta t \le 0.149\Delta x$

- The time-step restriction is driven by the **linear component** – even when handled implicitly!
- When linear L_2 stability is the concern, implicit or IMEX methods completely resolve the stiffness coming from the linear case.
- However, when we require the SSP condition to hold, implicit or IMEX do not significantly alleviate the time-step restriction.

Exponential Integrators background

- Emerged in the 1960s as an alternative to implicit methods for stiff problems
  - Rather than solving large linear systems these methods require the exponential of a matrix (typically linear operator or Jacobian)
  - Using direct methods to evaluate the matrix exponential such as Taylor or Padé approximations
    - Direct approximations made exponential integrator methods too costly for large scale problems
    - Focus was on small scale problems
- Re-emerged in 1980s
  - Using iterative techniques to compute the matrix exponential
  - Enables exponential integrator methods to be used on large scale problems

Exponential Integrators

Starting from the equation

$$u_t = Lu + N(u)$$

we solve the linear component by an integrating factor

$$\begin{aligned} e^{-Lt}u_t - e^{-Lt}Lu &= e^{-Lt}N(u) \\ \left(e^{-Lt}u\right)_t &= e^{-Lt}N(u). \end{aligned}$$

The following classes of exponential integrators

- Integrating factor methods (IF)[1]
- Exponential time differencing methods (ETD)

differ in how they approximate the nonlinear component.

[1] Also known as Lawson methods

Integrating Factor methods

The integrating factor approach uses a transformation of variables on

$$\left(e^{-Lt}u\right)_t = e^{-Lt}N(u).$$

By letting $w = e^{-Lt}u$ we obtain the ODE system

$$w_t = e^{-Lt}N(e^{Lt}w).$$

then a numerical time stepping method can be applied on transformed equation.

Integrating Factor Runge-Kutta methods

The transformed equation

$$w_t = e^{-Lt} N(e^{Lt} w)$$

is evolved forward in time using, for example, an explicit Runge–Kutta method.

$$\begin{aligned} w^{(0)} &= w^n, \\ w^{(i)} &= \sum_{j=0}^{i-1} \left(\alpha_{i,j} w^{(j)} + \Delta t \beta_{i,j} e^{-Lt_j} N(e^{Lt_j} w^{(j)}) \right), \quad i = 1, ..., s \\ w^{n+1} &= w^{(s)} \end{aligned}$$

Will the result be SSP?

Motivating example

Consider

$$u_t + 5u_x + \left(\frac{1}{2}u^2\right)_x = 0 \qquad u(0,x) = \begin{cases} 1, & \text{if } 0 \le x \le 1/2 \\ 0, & \text{if } x > 1/2 \end{cases}$$

- u_x: first order upwind difference
- $\left(\frac{1}{2}u^2\right)_x$: fifth order WENO finite difference

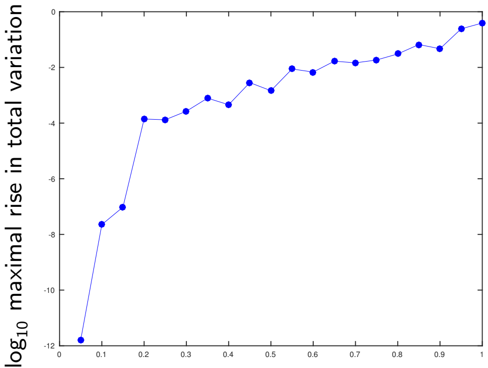
Step the transformed problem in time using the eSSPRK(3,3) Shu-Osher method:

$$\begin{aligned} u^{(1)} &= e^{L\Delta t}\left(u^n + \Delta t N(u^n)\right) \\ u^{(2)} &= \frac{3}{4}e^{\frac{1}{2}L\Delta t}u^n + \frac{1}{4}e^{-\frac{1}{2}L\Delta t}\left(u^{(1)} + \Delta t N(u^{(1)})\right) \\ u^{n+1} &= \frac{1}{3}e^{L\Delta t}u^n + \frac{2}{3}e^{\frac{1}{2}L\Delta t}\left(u^{(2)} + \Delta t N(u^{(2)})\right). \end{aligned}$$

Motivating example

The SSP coefficient of eSSPRK(3,3) is $\mathcal{C} = 1$, implying

$$\|u^{n+1}\|_{TV} \leq \|u^n\|_{TV} \quad \text{for} \quad \Delta t \leq \frac{\Delta x}{2}$$



$$\lambda = \frac{\Delta t}{\Delta x}$$

IFRK with eSSPRK(3,3) has a large maximal rise in total variation.

Motivating example: analyzing IFRK methods

Transforming back to the original variable, the integrating factor Runge–Kutta method becomes

$$e^{-Lt_i} u^{(i)} = \sum_{j=0}^{i-1} \left(\alpha_{i,j} e^{-Lt_j} u^{(j)} + \Delta t \beta_{i,j} e^{-Lt_j} N(u^{(j)}) \right),$$

where $u^{(i)}$ corresponds to the solution at time $t_i = t^n + c_i \Delta t$ (each c_i is the abscissa of the method at the ith stage),

$$\begin{aligned} u^{(i)} &= \sum_{j=0}^{i-1} \left(\alpha_{i,j} e^{L(t_i - t_j)} u^{(j)} + \Delta t \beta_{i,j} e^{L(t_i - t_j)} N(u^{(j)}) \right) \\ &= \sum_{j=0}^{i-1} \alpha_{i,j} e^{L(c_i - c_j)\Delta t} \left(u^{(j)} + \Delta t \frac{\beta_{i,j}}{\alpha_{i,j}} N(u^{(j)}) \right). \end{aligned}$$

Motivating example: What went wrong?

Observe the appearance of negative values in the exponential.

$$
\begin{aligned}
u^{(1)} &= e^{L\Delta t}\left(u^n + \Delta t N(u^n)\right) \\
u^{(2)} &= \frac{3}{4} e^{\frac{1}{2}L\Delta t} u^n + \frac{1}{4} e^{-\frac{1}{2}L\Delta t}\left(u^{(1)} + \Delta t N(u^{(1)})\right) \\
u^{n+1} &= \frac{1}{3} e^{L\Delta t} u^n + \frac{2}{3} e^{\frac{1}{2}L\Delta t}\left(u^{(2)} + \Delta t N(u^{(2)})\right).
\end{aligned}
$$

Motivating example: What went wrong?

Observe the appearance of negative values in the exponential.

$$u^{(1)} = e^{L\Delta t}\left(u^n + \Delta t N(u^n)\right)$$

$$u^{(2)} = \frac{3}{4} e^{\frac{1}{2}L\Delta t} u^n + \frac{1}{4} e^{-\frac{1}{2}L\Delta t}\left(u^{(1)} + \Delta t N(u^{(1)})\right)$$

$$u^{n+1} = \frac{1}{3} e^{L\Delta t} u^n + \frac{2}{3} e^{\frac{1}{2}L\Delta t}\left(u^{(2)} + \Delta t N(u^{(2)})\right).$$

which came from the decreasing abscissas c_i on $t_i = t^n + c_i \Delta t$

$$u^{(i)} = \sum_{j=0}^{i-1} \alpha_{i,j} e^{L(c_i - c_j)\Delta t}\left(u^{(j)} + \Delta t \frac{\beta_{i,j}}{\alpha_{i,j}} N(u^{(j)})\right)$$

Motivating example

To avoid this, we wish to use an explicit SSP Runge–Kutta method that has non-decreasing abscissas c_i

$$
\begin{aligned}
u^{(1)} &= \frac{1}{2}u^n + \frac{1}{2}\left(u^n + \frac{4}{3}\Delta t N(u^n)\right) \\
u^{(2)} &= \frac{2}{3}u^n + \frac{1}{3}\left(u^{(1)} + \frac{4}{3}\Delta t N(u^{(1)})\right) \\
u^{n+1} &= \frac{59}{128}u^n + \frac{15}{128}\left(u^n + \frac{4}{3}\Delta t N(u^n)\right) \\
&\quad + \frac{27}{64}\left(u^{(2)} + \frac{4}{3}\Delta t N(u^{(2)})\right).
\end{aligned}
$$

We call these eSSPRK$^+$ methods.

Motivating example

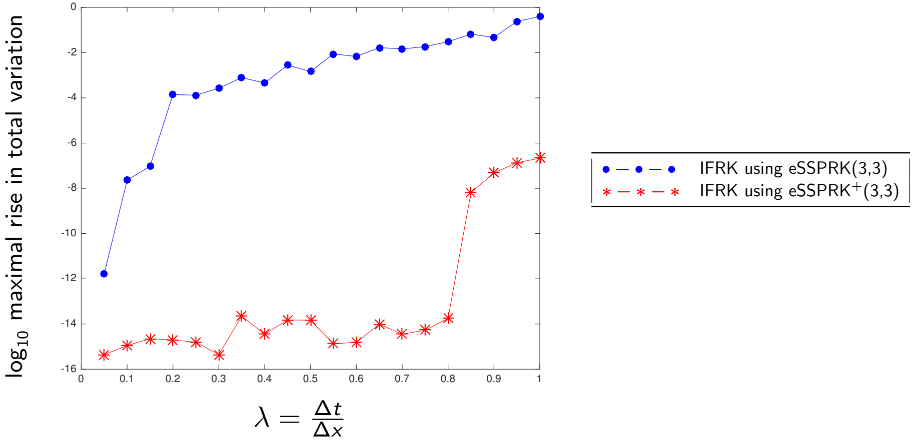
So when you step the transformed problem in time using eSSPRK$^+$(3,3):

$$
\begin{aligned}
u^{(1)} &= \frac{1}{2} e^{\frac{2}{3}\Delta t L} u^n + \frac{1}{2} e^{\frac{2}{3}\Delta t L} \left(u^n + \frac{4}{3} \Delta t N(u^n) \right) \\
u^{(2)} &= \frac{2}{3} e^{\frac{2}{3}\Delta t L} u^n + \frac{1}{3} \left(u^{(1)} + \frac{4}{3} \Delta t N(u^{(1)}) \right) \\
u^{n+1} &= \frac{59}{128} e^{\Delta t L} u^n + \frac{15}{128} e^{\Delta t L} \left(u^n + \frac{4}{3} \Delta t N(u^n) \right) \\
&\quad + \frac{27}{64} e^{\frac{1}{3}\Delta t L} \left(u^{(2)} + \frac{4}{3} \Delta t N(u^{(2)}) \right) .
\end{aligned}
$$

There are no negative values in the exponential. This method has a SSP coefficient of $\mathcal{C} = \frac{3}{4}$.

Motivating example

When IFRK uses eSSPRK$^+$(3,3), a small maximal rise in total variation is observed up to $\lambda \approx 0.8$.



Explicit SSP integrating factor Runge–Kutta methods

The optimal eSSPRK$^+$ methods can be used to develop explicit SSP integrating factor Runge–Kutta (eSSPIFRK) methods of the form

$$\begin{aligned} u^{(0)} &= u^n \\ u^{(i)} &= \sum_{j=0}^{i-1} \alpha_{ij} e^{L(c_i - c_j)\Delta t} \left(u^{(j)} + \Delta t \frac{\beta_{ij}}{\alpha_{ij}} N(u^{(j)}) \right), \quad i = 1, \ldots, s \\ u^{n+1} &= u^{(s)} \end{aligned}$$

which are provably SSP.

Leah Isherwood, S. Gotlieb, and Zachary J. Grant (2018) Strong Stability Preserving Integrating Factor Runge-Kutta Methods
SINUM (to appear).
Available online at https://arxiv.org/abs/1708.02595

The price: SSP coefficients $\mathcal{C}$ of the optimal methods

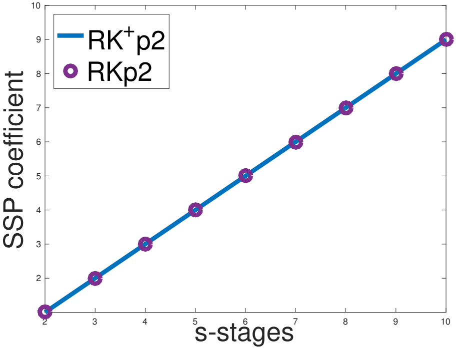
s \ p	2	3	4
1	-	-	-
2	1.0000	-	-
3	2.0000	1.0000	-
4	3.0000	2.0000	-
5	4.0000	2.6506	1.5082
6	5.0000	3.5184	2.2945
7	6.0000	4.2879	3.3209
8	7.0000	5.1071	4.1459
9	8.0000	6.0000	4.9142
10	9.0000	6.7853	6.0000

Table: SSP coefficients of the optimal eSSPRK(s,p) methods.

s \ p	2	3	4
1	-	-	-
2	1.0000	-	-
3	2.0000	0.7500	-
4	3.0000	1.8182	-
5	4.0000	2.6351	1.3466
6	5.0000	3.5184	2.2738
7	6.0000	4.2857	3.0404
8	7.0000	5.1071	3.8926
9	8.0000	6.0000	4.6048
10	9.0000	6.7853	5.2997

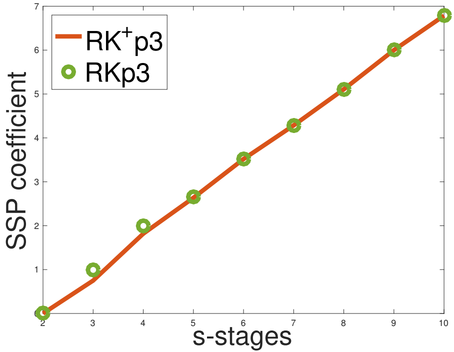
Table: SSP coefficients of the optimal eSSPRK$^+$(s,p) methods.

The price: SSP coefficients $\mathcal{C}$ of the optimal methods



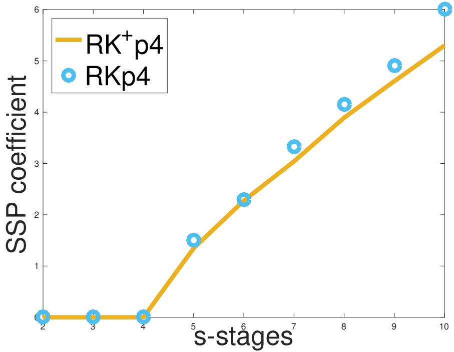
- The optimal second order eSSPRK(s,2) methods have non-decreasing coefficients, so they are automatically eSSPRK$^+$(s,2) methods.

The price: SSP coefficients $\mathcal{C}$ of the optimal methods



- The optimal eSSPRK$^+$(s,3) methods have smaller SSP coefficients than the eSSPRK(s,3) methods for small s.

The price: SSP coefficients $\mathcal{C}$ of the optimal methods



- The optimal eSSPRK$^+$(s,4) methods have smaller SSP coefficients than the eSSPRK(s,4) methods. This does not significantly improve as s increases.
- eSSPRK$^+$(10,4) has SSP coefficient $\mathcal{C} = 5.3$, a reduction of over 10% from eSSPRK(10,4) which has an SSP coefficient of $\mathcal{C} = 6$
- eSSPRK$^+$(6,4) has only a 1% reduction of the SSP coefficient compared to the eSSPRK(6,4).

Numerical tests

How do these methods perform in practice?

The price: SSP coefficients $\mathcal{C}$ of the optimal methods

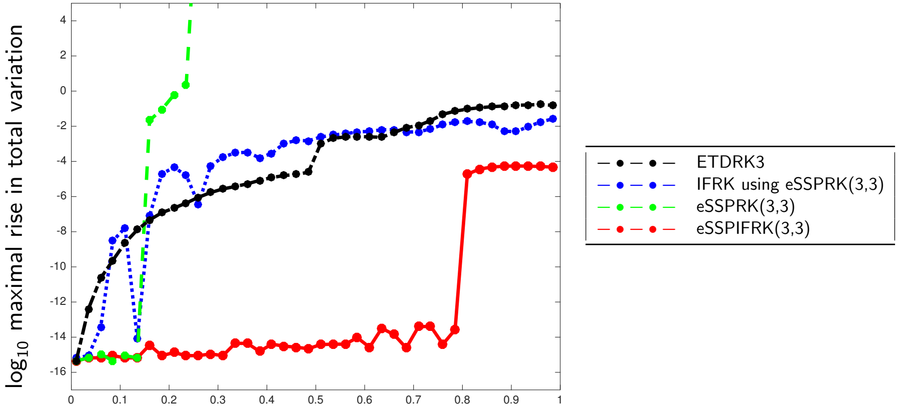
Consider

$$u_t + au_x + \left(\frac{1}{2}u^2\right)_x = 0 \qquad u(0,x) = \begin{cases} 1, & \text{if } 0 \le x \le 1/2 \\ 0, & \text{if } x > 1/2 \end{cases}$$

on the domain $[0,1]$ with periodic boundary conditions.

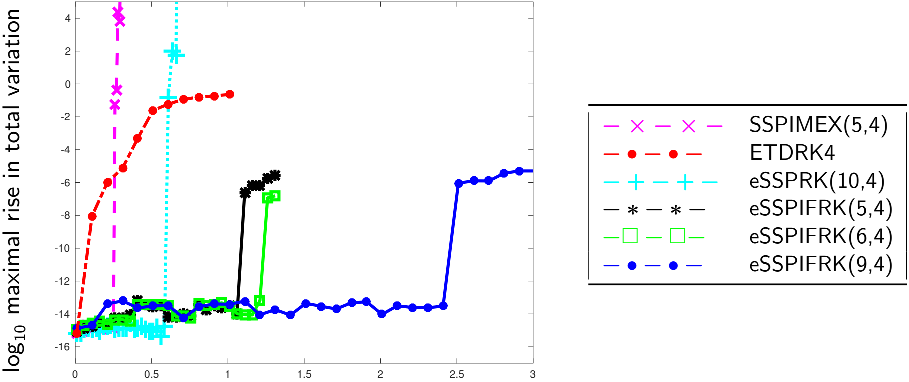
- au_x: First-order upwind difference
- $\left(\frac{1}{2}u^2\right)_x$: Fifth order WENO finite difference
- Spatial grid with 400 points and evolved forward 25 time-steps using $\Delta t = \lambda \Delta x$.
- We measured the total variation at each stage, and calculated the maximal rise in total variation over each stage for these 25 time-steps.

Third order methods with $a = 5$:



$$\lambda = \frac{\Delta t}{\Delta x}$$

Fourth order methods with $a = 10$:



$$\lambda = \frac{\Delta t}{\Delta x}$$

Sharpness of SSP time-step for a linear problem

Consider

$$u_t + au_x + u_x = 0 \qquad\qquad u(0,x) = \begin{cases} 1, & \text{if } 1/4 \leq x \leq 3/4 \\ 0, & \text{else} \end{cases}$$

on the domain $[0,1]$ with periodic boundary conditions.

- au_x & u_x: First-order upwind difference
- IMEX: au_x is treated implicitly & u_x is treated explicitly
- IF: au_x is treated exactly & u_x is treated explicitly
- Spatial grid with 1000 points and evolved forward 10 time-steps using $\Delta t = \lambda \Delta x$.
- We measured the total variation at each stage, and calculated the maximal rise in total variation over each stage for these 10 time-steps.

Observed SSP Coefficient

The observed value of the SSP coefficient $\mathcal{C}_{obs} = \frac{\Delta t_{obs}^{TVD}}{\Delta x}$

Wavespeed	$a = 0$	$a = 1$	$a = 2$	$a = 10$	$a = 20$	$a = 100$
eSSPRK(4,3)	2.000	1.000	0.666	0.181	0.095	0.019
SSPIMEX(4,3,K)	2.000	1.476	1.192	0.310	0.162	0.033
eSSPIFRK(4,3)	1.818	1.818	1.818	1.818	1.818	4.200

The contribution from the faster wave negatively impacts the observed SSP coefficient of the explicit Runge–Kutta method and even of the IMEX method.

However, **eSSPIFRK methods are unaffected by the faster wave!**

Comparison to linear stability properties.

Method	$\lambda^{L_2}_{obs}$	λ^{TVD}_{pred}	λ^{TVD}_{obs}
eSSPRK(3,3)	0.114	$1/11$	0.090
eSSPRK+(3,3)	0.114	$3/44$	0.090
IMEXSSP(3,3,$K=0.1$)	0.448	0.149	0.236
IMEXSSP(3,3,$K=\infty$)	1.198	0.000	0.000
eSSPIFRK(3,3)	*	0.750	1.500
eSSPRK(5,3)	0.260	0.240	0.240
eSSPRK+(5,3)	0.261	0.239	0.239
eSSPKG+(5,3)	0.138	$1/11$	0.090
IMEXSSP(5,3,$K=0.1$)	0.683	0.407	0.407
eSSPIFRK(5,3)	*	2.635	2.635
eSSPRK(6,4)	0.273	0.208	0.208
eSSPRK+(6,4)	0.270	0.206	0.206
eSSPIFRK(6,4)	*	2.273	2.273

Table: An * indicates that these methods were linearly L_2 stable for the largest values tested, $\lambda \leq 27$.

Consider

$$u_t + 10u_x + u_x = 0$$

The values of the observed CFL number $\lambda^{L_2} = \frac{\Delta t}{\Delta x}$ required for L_2 linear stability compared to the predicted and observed values λ^{TVD}.

Next steps

- Consider downwinding instead of non-decreasing abscissas (see poster)
- Combine with efficient exponentiation approaches
- Test on more challenging problems of interest (not TVD due to linearity)
- Identify splittings that would benefit from this approach
- Use this approach as a type of additive preconditioning?

Conclusions

- The integrating factor approach can eliminate the severe time-step restriction resulting from a linear operator, leaving only the mild restriction coming from $N(u)$.
- The cost of solving and storing the few exponentials may be very reasonable, considering it only needs to be done once per simulation. For very large problems this may become problematic in terms of storage.
- We have established the SSP properties of integrating factor Runge–Kutta methods and developed suitable methods which out-perform explicit, IMEX, and ETD methods in terms of total variation stability.
- This promising approach needs to be considered on a case-by-case basis; co-development with spatial discretizations on particular applications is advisable.

Thank You!

Motivating example

Recall the appearance of negative values in the exponential.

$$
\begin{aligned}
u^{(1)} &= e^{L\Delta t}u^n + e^{L\Delta t}\Delta t N(u^n) \\
u^{(2)} &= \frac{3}{4}e^{\frac{1}{2}L\Delta t}u^n + \frac{1}{4}e^{-\frac{1}{2}L\Delta t}u^{(1)} + \frac{1}{4}e^{-\frac{1}{2}L\Delta t}\Delta t N(u^{(1)}) \\
u^{n+1} &= \frac{1}{3}e^{L\Delta t}u^n + \frac{2}{3}e^{\frac{1}{2}L\Delta t}u^{(2)} + \frac{2}{3}e^{\frac{1}{2}L\Delta t}\Delta t N(u^{(2)}).
\end{aligned}
$$

which came from the decreasing abscissas c_i

$$
u^{(i)} = \sum_{j=0}^{i-1}\left(\alpha_{i,j}e^{L(c_i-c_j)\Delta t}u^{(j)} + \Delta t\beta_{i,j}e^{L(c_i-c_j)\Delta t}N(u^{(j)})\right)
$$

Downwinding

Provided that whenever the term $c_i - c_j$ is negative, the operator L is replaced by an operator $\tilde{L}$ that satisfies the condition

$$\|e^{-\tau \tilde{L}} u^n\| \leq \|u^n\| \quad \forall\, \tau \geq 0.$$

For hyperbolic partial differential equations, this is accomplished by using the spatial discretization that is stable for downwind problem.
This approach is similar to the one employed in the classical SSP literature, where negative coefficients $\beta_{i,j}$ may be allowed if the corresponding operator is replaced by a downwind operator.

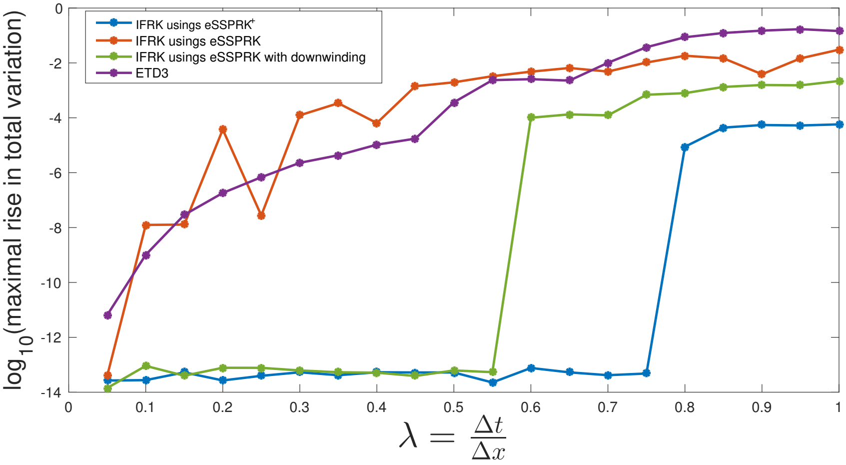
Downwinding

Preserving the strong stability property of an existing SSPRK methods by using downwinding, $\tilde{L}$, wherever there is a negative in the exponential. The 3 stage 3rd order looks like:

$$
\begin{aligned}
u^{(1)} &= e^{L\Delta t}u^n + e^{L\Delta t}\Delta t N(u^n) \\
u^{(2)} &= \frac{3}{4}e^{\frac{1}{2}L\Delta t}u^n + \frac{1}{4}e^{-\frac{1}{2}\tilde{L}\Delta t}u^{(1)} + \frac{1}{4}e^{-\frac{1}{2}\tilde{L}\Delta t}\Delta t N(u^{(1)}) \\
u^{n+1} &= \frac{1}{3}e^{L\Delta t}u^n + \frac{2}{3}e^{\frac{1}{2}L\Delta t}u^{(2)} + \frac{2}{3}e^{\frac{1}{2}L\Delta t}\Delta t N(u^{(2)}).
\end{aligned}
$$

We call these eDWSSPRK methods.

TVD 3s 3p methods with $a = 5$:



Order of convergence

Consider

$$u_t + 10u_x + \left(\frac{1}{2}u^2\right)_x = 0 \qquad\qquad u(0,x) = e^{\sin(2\pi x)}$$

on the domain $[0,1]$ with periodic boundary conditions.

- au_x: First-order upwind difference
- $\left(\frac{1}{2}u^2\right)_x$: Fifth order WENO finite difference
- Spatial grid with 64 points and ODE15s with $AbsTol = 10^{-15}$ and $RelTol = 5 \times 10^{-14}$ to compute the reference solution
- Compared ETD Runge–Kutta methods of orders $p = 2, 3, 4$ (the schemes by Cox and Matthews called ETDRK2, ETDRK3, and ETDRK4 in EXPINT) and our eSSPIFRK methods with $(s,p) = (2,2), (3,3), (5,4)$

Order of convergence

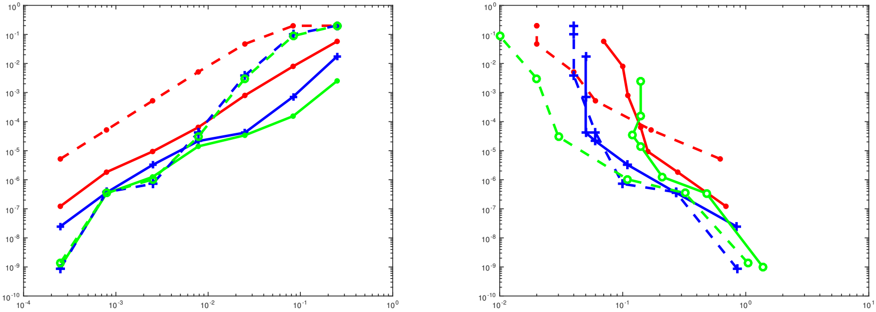


Figure: The second order methods are in red, third order in blue, and fourth order in green, dashed lines represent the ETD methods while solid lines are the eSSPIFRK methods. Left: on x-axis is the step-size while on the y-axis is the error. Right: on x-axis is the CPU time while on the y-axis is the error.

References

S. Gotlieb, D. I. Ketcheson and C.-W. Shu (2011)
Strong Stability Preserving Runge-Kutta and Multistep Time Discretizations
World Scientific Press.

S. Gotlieb and C.-W. Shu (1998)
Total variation diminishing runge-kutta methods
Mathematics of Computation 67, 73 – 85.

S. Gotlieb, C.-W. Shu, and E. Tadmor (2001)
Strong Stability Preserving High-Order Time Discretization methods
SIAM Review 43, 89 – 112.

C.-W. Shu (1988)
Total-variation diminishing time discretizations
SIAM Journal Scientific Statistical Computing 9, 1073 – 1084.

C.-W. Shu and Shu Osher (1988)
Efficient implementation of essentially non-oscillatory shock-capturing schemes
Journal of Computational Physics 77, 439 – 471.