A new perspective on time-stepping schemes: Beyond strong stability

Alexandre Ern

ENPC and INRIA, Paris, France
joint work with Jean-Luc Guermond (TAMU)

WIAS CFD Workshop, 05-08 July 2022
Setting

Why (and how to) go beyond strong stability (SSP)?

New perspective on explicit Runge–Kutta (ERK)

New perspective on implicit-explicit (IMEX)
Cauchy problem

\[ \begin{cases} \partial_t u + \nabla \cdot f(u) + \nabla \cdot g(u, \nabla u) = S(u) & \text{in } D \times \mathbb{R}_+ \\ u(\cdot, 0) = u_0 & \text{in } D \end{cases} \]

\( D \subset \mathbb{R}^d \) (open Lipschitz polytope)
Cauchy problem

\[
\begin{aligned}
\begin{cases}
\partial_t u + \nabla \cdot f(u) + \nabla \cdot g(u, \nabla u) = S(u) & \text{in } D \times \mathbb{R}_+
\\
u(\cdot, 0) = u_0 & \text{in } D
\end{cases}
\end{aligned}
\]

\(D \subset \mathbb{R}^d\) (open Lipschitz polytope)

- Field \(u\) takes values in \(\mathbb{R}^m\), i.e., \(u : D \times \mathbb{R}_+ \to \mathbb{R}^m\)
- \(f \in C^1(\mathbb{R}^m; \mathbb{R}^{m \times d})\): hyperbolic flux
- \(g \in C^1(\mathbb{R}^m \times \mathbb{R}^{m \times d}; \mathbb{R}^{m \times d})\): parabolic/diffusive flux
- \(S \in C^1(\mathbb{R}^m; \mathbb{R}^m)\): source/relaxation term
Cauchy problem

\[
\begin{aligned}
\left\{ 
\begin{array}{l}
\partial_t u + \nabla \cdot f(u) + \nabla \cdot g(u, \nabla u) = S(u) \quad \text{in } D \times \mathbb{R}_+ \\
u(\cdot, 0) = u_0 \quad \text{in } D
\end{array}
\right.
\end{aligned}
\]

\(D \subset \mathbb{R}^d\) (open Lipschitz polytope)

- Field \(u\) takes values in \(\mathbb{R}^m\), i.e., \(u : D \times \mathbb{R}_+ \rightarrow \mathbb{R}^m\)
- \(f \in C^1(\mathbb{R}^m; \mathbb{R}^{m \times d})\): hyperbolic flux
- \(g \in C^1(\mathbb{R}^m \times \mathbb{R}^{m \times d}; \mathbb{R}^{m \times d})\): parabolic/diffusive flux
- \(S \in C^1(\mathbb{R}^m; \mathbb{R}^m)\): source/relaxation term
- \(u_0\): admissible initial data
- BCs not discussed herein
Exemple 1: Scalar advection-diffusion-reaction

- Find $u : D \times \mathbb{R}_+ \to \mathbb{R}$ such that

$$\partial_t u + \nabla \cdot f(x, u) + \nabla \cdot (\kappa(u) \nabla u) = S(u), \quad u(\cdot, 0) = u_0$$

- Hyperbolic and parabolic fluxes

$$f(x, u) := \begin{cases} (f_1(u), \ldots, f_d(u)) \\ \beta(x)u \text{ with } \nabla \cdot \beta = 0 \end{cases} \quad g(u, \nabla u) := \kappa(u) \nabla u$$

- Source term, for instance

$$S(u) := \mu \phi(u) u(1 - u), \quad \phi \in C^0([0, 1]; \mathbb{R}), \quad \mu \geq 0$$
Exemple 2: Navier–Stokes

Find \( u := (\rho, m^\top, E)^\top : D \times \mathbb{R}_+ \rightarrow \mathbb{R}^{d+2} \) such that

\[
\begin{align*}
\partial_t \rho + \nabla \cdot (v \rho) &= 0 \\
\partial_t m + \nabla \cdot (v \otimes m + p(u)I - s(v)) &= 0 \\
\partial_t E + \nabla \cdot (v(E + p(u)) - v \cdot s(v) + q(u)) &= 0
\end{align*}
\]

with velocity \( v := m/\rho \) and pressure \( p(u) \)
Exemple 2: Navier–Stokes

- Find \( u := (\rho, m^T, E)^T : D \times \mathbb{R}_+ \rightarrow \mathbb{R}^{d+2} \) such that

\[
\begin{align*}
\partial_t \rho + \nabla \cdot (v \rho) &= 0 \\
\partial_t m + \nabla \cdot (v \otimes m + p(u)I - s(v)) &= 0 \\
\partial_t E + \nabla \cdot (v(E + p(u)) - v\cdot s(v) + q(u)) &= 0
\end{align*}
\]

with velocity \( v := m/\rho \) and pressure \( p(u) \)

- Hyperbolic (Euler) and parabolic fluxes

\[
f(u) := \begin{pmatrix} v\rho \\ v \otimes m + p(u)I \\ v(E + p(u)) \end{pmatrix}, \quad g(u, \nabla u) := \begin{pmatrix} 0 \\ -s(v) \\ -v \cdot s(v) + q(u) \end{pmatrix}
\]

with viscous stress tensor and heat flux such that

\[
s(v) = 2\mu e(v) + (\lambda - \frac{2}{3}\mu)(\nabla \cdot v)I, \quad q(u) = -\kappa \nabla e(u)
\]

with (linearized) strain tensor \( e(v) := \frac{1}{2}(\nabla v + \nabla v^T) \) and specific internal energy \( e(u) := E/\rho - \frac{1}{2} \|v\|^2_{\ell^2} \)

- \( \mu, \lambda, \kappa \) constant for simplicity
Find \( u := (\rho, m^T, E, E_R)^T : D \times \mathbb{R}_+ \rightarrow \mathbb{R}^{d+3} \) such that

\[
\begin{align*}
\partial_t \rho + \nabla \cdot (v \rho) &= 0 \\
\partial_t m + \nabla \cdot (v \otimes m + (p(u) + p_R(E_R))I) &= 0 \\
\partial_t E + \nabla \cdot (v(E + p(u) + p_R(E_R))) - \nabla \cdot \left( \frac{c}{3 \sigma_t} \nabla E_R \right) &= 0 \\
\partial_t E_R + \nabla \cdot (v(E_R + p_R(E_R))) - v \cdot \nabla p_R(E_R) - \nabla \cdot \left( \frac{c}{3 \sigma_t} \nabla E_R \right) &= \sigma_a c (a_R T(u)^4 - E_R)
\end{align*}
\]

with radiation energy \( E_R \), radiation pressure \( p_R(E_R) \), and temp. \( T(u) \)

\( (c: \text{speed of light}, \sigma_a, \sigma_t \text{ absorption and total cross sections}, a_R := \frac{4 \sigma}{c}: \text{radiation constant, } \sigma: \text{Stefan–Boltzmann constant}) \)

Possible definitions: \( p_R(E_R) := E_R/3 \) and \( T(u) := e(u)/c_v \)
Invariant domain

- **Key assumption:** There exists a convex subset $\mathcal{A} \subseteq \mathbb{R}^m$, depending on the initial data $u_0$ s.t. the entropy/admissible solution to the Cauchy problem takes values in $\mathcal{A}$ for a.e. $(x, t) \in D \times \mathbb{R}_+$.

$$\{u_0(x) \in \mathcal{A} \text{ for a.e. } x \in D\} \implies \{u(x, t) \in \mathcal{A} \text{ for a.e. } (x, t) \in D \in \mathbb{R}_+\}$$
Invariant domain

- **Key assumption:** There exists a convex subset $\mathcal{A} \subseteq \mathbb{R}^m$, depending on the initial data $u_0$ s.t. the entropy/admissible solution to the Cauchy problem takes values in $\mathcal{A}$ for a.e. $(x, t) \in D \times \mathbb{R}_+$

\[\{u_0(x) \in \mathcal{A} \text{ for a.e. } x \in D\} \implies \{u(x, t) \in \mathcal{A} \text{ for a.e. } (x, t) \in D \in \mathbb{R}_+\}\]

- This is a generalization of the maximum principle
Invariant domain

- **Key assumption:** There exists a convex subset $\mathcal{A} \subseteq \mathbb{R}^m$, depending on the initial data $u_0$ s.t. the entropy/admissible solution to the Cauchy problem takes values in $\mathcal{A}$ for a.e. $(x, t) \in D \times \mathbb{R}_+$

\[
\{u_0(x) \in \mathcal{A} \text{ for a.e. } x \in D\} \implies \{u(x, t) \in \mathcal{A} \text{ for a.e. } (x, t) \in D \in \mathbb{R}_+\}
\]

- This is a generalization of the maximum principle

- Scalar conservation equations without reaction

\[\mathcal{A} := [\text{ess inf } u_0, \text{ess sup } u_0]\]

- Scalar conservation equations with $S(u) := \mu \phi(u)u(1 - u)$

\[\mathcal{A} := [0, 1]\]

- Navier–Stokes and Euler equations ($s(u)$: specific entropy)

\[\mathcal{A}_{NS} := \{(\rho, m^\top, E)^\top \in \mathbb{R}^m \mid 0 < \rho, \ 0 < e(u)\}\]

\[\mathcal{A}_{Eu} := \{(\rho, m^\top, E)^\top \in \mathbb{R}^m \mid 0 < \rho, \ 0 < e(u), \ \text{ess inf } s(u_0) \leq s(u)\}\]
Invariant-domain preserving (IDP) approximation methods

- Approximation methods that preserve invariant domains are called **Invariant domain preserving (IDP)**
Invariant-domain preserving (IDP) approximation methods

- Approximation methods that preserve invariant domains are called Invariant domain preserving (IDP)

- Space semi-discrete problem: Find $\mathbf{U} \in C^1(\mathbb{R}_+; (\mathbb{R}^m)^I)$ s.t.

$$\mathbb{M} \partial_t \mathbf{U} = \mathbf{F}(\mathbf{U}) + \mathbf{G}(\mathbf{U}), \quad \mathbf{U}(0) = \mathbf{U}_0$$

- $I$: #dofs for space approximation ($C^0$-FEM, dG, FV, FD, ...)
- $\mathbb{M}$: mass matrix (invertible)
- $\mathbf{F} : (\mathbb{R}^m)^I \to (\mathbb{R}^m)^I$: space approximation of $-\nabla \cdot \mathbf{f}(\mathbf{u})$
- $\mathbf{G} : (\mathbb{R}^m)^I \to (\mathbb{R}^m)^I$: space approximation of $-\nabla \cdot \mathbf{g}(\mathbf{u}, \nabla \mathbf{u}) + S(\mathbf{u})$
- $\mathbf{U}_i$ approximates $\mathbf{u}$ at some point $\mathbf{x}_i \in D \implies$ natural requirement is $\mathbf{U} \in \mathcal{A}^I$
Invariant-domain preserving (IDP) approximation methods

- Approximation methods that preserve invariant domains are called **Invariant domain preserving (IDP)**

- Space semi-discrete problem: Find $U \in C^1(\mathbb{R}_+; (\mathbb{R}^m)^I)$ s.t.

$$M \partial_t U = F(U) + G(U), \quad U(0) = U_0$$

  - $I$: #dofs for space approximation ($C^0$-FEM, dG, FV, FD, ...)
  - $M$: mass matrix (invertible)
  - $F : (\mathbb{R}^m)^I \rightarrow (\mathbb{R}^m)^I$: space approximation of $-\nabla \cdot f(u)$
  - $G : (\mathbb{R}^m)^I \rightarrow (\mathbb{R}^m)^I$: space approximation of $-\nabla \cdot g(u, \nabla u) + S(u)$
  - $U_i$ approximates $u$ at some point $x_i \in D \implies$ natural requirement is $U \in \mathcal{A}^I$

- Time-stepping scheme produces a sequence $U^0, U^1, \ldots, U^n, \ldots$

- Time-stepping scheme is IDP if

$$\{U_0 \in \mathcal{A}^I\} \implies \{U^n \in \mathcal{A}^I \ \forall n \geq 0\}$$

How to achieve this goal?
Let us focus first on **hyperbolic problems**

**Key idea:** [Shu & Osher 88] SSPRK are ERK methods where all updates are **convex combinations** of previous updates computed with forward Euler method (recall $\mathcal{A}$ convex)
Let us focus first on hyperbolic problems

**Key idea:** [Shu & Osher 88] SSPRK are ERK methods where all updates are convex combinations of previous updates computed with forward Euler method (recall $A$ convex)

**Key assumption:** $\exists \tau^* > 0$ s.t. $\forall \tau \in (0, \tau^*],$

$$\{V \in A^I\} \implies \{V + \tau(M)^{-1}F(V) \in A^I\}$$

In other words, $A^I$ is invariant under the forward Euler method and CFL condition $\tau \in (0, \tau^*]$
SSP paradigm for hyperbolic problems

- Let us focus first on **hyperbolic problems**

- **Key idea:** [Shu & Osher 88] SSPRK are ERK methods where all updates are convex combinations of previous updates computed with forward Euler method (recall \( A \) convex)

- **Key assumption:** \( \exists \tau^* > 0 \text{ s.t. } \forall \tau \in (0, \tau^*], \)

\[
\{ V \in A^I \} \implies \{ V + \tau(M)^{-1}F(V) \in A^I \}
\]

In other words, \( A^I \) is invariant under the forward Euler method and CFL condition \( \tau \in (0, \tau^*] \)

- Theory of SSP methods applied to ODEs is well understood
  [Kraaijevanger 91; S Ruuth & Spiteri 02; Ferracina & Spijker 05; Higueras 05]
Examples (for $\partial_t u = L(t, u)$)

- **Notation:** $\text{SSPRK}(s, p)$ for $s$-stage, $p$th-order method

- **SSPRK(2,2)** (two-stage, second-order) [Heun’s second-order method]

  $$w^{(1)} = u^n + \tau L(t^n, u^n)$$
  $$u^{n+1} = \frac{1}{2} u^n + \frac{1}{2} (w^{(1)} + \tau L(t^{n+1}, w^{(1)}))$$

- **SSPRK(3,3)** (three-stage, third-order) [Fehlberg’s method]

  $$w^{(1)} = u^n + \tau L(t^n, u^n)$$
  $$w^{(2)} = \frac{3}{4} u^n + \frac{1}{4} (w^{(1)} + \tau L(t^{n+1}, w^{(1)}))$$
  $$u^{n+1} = \frac{1}{3} u^n + \frac{2}{3} (w^{(2)} + \tau L(t^{n+\frac{1}{2}}, w^{(2)}))$$

- **SSPRK(4,3)** and **SSPRK(5,4)** also available
Why (and how to) go beyond SSP?
Restriction in accuracy: SSPRK are restricted to fourth-order (if one insists on never stepping backward in time) [Ruuth & Spiteri 02]
Limitations of SSP paradigm (1/2)

- **Restriction in accuracy**: SSPRK are restricted to fourth-order (if one insists on never stepping backward in time) [Ruuth & Spiteri 02]

- **Difficult to accommodate implicit and explicit substeps**
  - implicit RK schemes of order $\geq 2$ cannot be SSP [Gottlieb, Shu, Tadmor 01]
  - explicit methods suffer from parabolic CFL restriction $\tau \leq ch^2$
  - schemes with two time-derivatives [Gottlieb, Grant, Hu, Shu 22]
Definition: efficiency ratio of any $s$-stage ERK method

- $\tau^*$: maximal time step that makes forward Euler method IDP
- $\tilde{\tau}$: maximal time step that makes $s$-stage ERK method IDP

$$c_{\text{eff}} := \frac{\tilde{\tau}}{s\tau^*}$$ (usually, $c_{\text{eff}} \leq 1$)
● **Definition:** efficiency ratio of any $s$-stage ERK method
  - $\tau^*$: maximal time step that makes forward Euler method IDP
  - $\tilde{\tau}$: maximal time step that makes $s$-stage ERK method IDP

  $$c_{\text{eff}} := \frac{\tilde{\tau}}{s\tau^*} \quad \text{(usually, } c_{\text{eff}} \leq 1)$$

● Do we care? Under the **same CFL constraint**, # flux evaluations to reach some $T$ for $s$-stage ERK is $\frac{1}{c_{\text{eff}}} \times$ that for forward Euler method
**Definition:** efficiency ratio of any $s$-stage ERK method
- $\tau^*$: maximal time step that makes forward Euler method IDP
- $\tilde{\tau}$: maximal time step that makes $s$-stage ERK method IDP

\[
c_{\text{eff}} := \frac{\tilde{\tau}}{s\tau^*}
\] (usually, $c_{\text{eff}} \leq 1$)

Do we care? Under the **same CFL constraint**, the number of flux evaluations to reach some $T$ for $s$-stage ERK is $\frac{1}{c_{\text{eff}}} \times$ that for forward Euler method

**SSPRK methods are usually inefficient!**
- $c_{\text{eff}} = \frac{1}{2}$ for SSPRK(2,2)
- $c_{\text{eff}} = \frac{1}{3}$ for SSPRK(3,3)
- $c_{\text{eff}} = \frac{1}{2}$ for SSPRK(4,3)
Limitations of SSP paradigm (2/2)

- **Definition:** efficiency ratio of any $s$-stage ERK method
  - $\tau^*$: maximal time step that makes forward Euler method IDP
  - $\tilde{\tau}$: maximal time step that makes $s$-stage ERK method IDP
  
  \[ c_{\text{eff}} := \frac{\tilde{\tau}}{s\tau^*} \quad \text{(usually, } c_{\text{eff}} \leq 1) \]

- Do we care? Under the same CFL constraint, $\#$ flux evaluations to reach some $T$ for $s$-stage ERK is $\frac{1}{c_{\text{eff}}} \times$ that for forward Euler method

- SSPRK methods are usually inefficient!
  - $c_{\text{eff}} = \frac{1}{2}$ for SSPRK(2,2)
  - $c_{\text{eff}} = \frac{1}{3}$ for SSPRK(3,3)
  - $c_{\text{eff}} = \frac{1}{2}$ for SSPRK(4,3)

- **Notation:** $\text{RK}(s, p; e)$ for $s$-stage, $p$th-order method, efficiency ratio $e$
  
  SSPRK(2,2; $\frac{1}{2}$)  SSPRK(3,3; $\frac{1}{3}$)  SSPRK(4,3; $\frac{1}{2}$)
Introduce a new methodology that makes any ERK scheme IDP
Our contribution

- Introduce a new methodology that makes any ERK scheme IDP
- Introduce a new methodology that makes any IMEX scheme IDP
Our contribution

- Introduce a new methodology that makes any ERK scheme IDP
- Introduce a new methodology that makes any IMEX scheme IDP

Benefits
- employ optimally efficient methods
- break order barriers
- introduce IDP-IMEX schemes of order $p \geq 2$
Examples of optimally efficient ERK methods

We will see that for an ERK-IDP scheme, maximal efficiency with $c_{\text{eff}} = 1$ is reached for equi-distributed sub-stages.
Examples of optimally efficient ERK methods

- We will see that for an ERK-IDP scheme, maximal efficiency with $c_{\text{eff}} = 1$ is reached for equi-distributed sub-stages

- RK(2,2;1) (midpoint), RK(3,3;1) (Heun), RK(4,3;1) [fourth-order on linear pb.]

\[
\begin{array}{c|cc}
0 & 0 & 0 \\
\frac{1}{2} & \frac{1}{2} & 0 \\
1 & 0 & 1 \\
\end{array}
\quad
\begin{array}{c|ccc}
0 & 0 & 0 \\
\frac{1}{3} & \frac{1}{3} & 0 \\
\frac{2}{3} & 0 & \frac{2}{3} & 0 \\
1 & \frac{1}{4} & 0 & \frac{3}{4} \\
\end{array}
\quad
\begin{array}{c|ccc}
0 & 0 & 0 \\
\frac{1}{4} & \frac{1}{4} & 0 \\
\frac{1}{2} & 0 & \frac{1}{2} & 0 \\
\frac{3}{4} & 0 & \frac{1}{4} & \frac{1}{2} & 0 \\
1 & 0 & \frac{2}{3} & -\frac{1}{3} & \frac{2}{3} \\
\end{array}
\]

- RK(5,4;1), RK(6,4;1) [fifth-order on linear pb.] and RK(7,5;1) [AE & JLG 22]
New perspective on ERK schemes
The beauty of SSP is that the forward Euler substep is a black box.
The beauty of SSP is that the forward Euler substep is a black box.

This black box involves two fluxes (not just one as one might think):
- low-order in space: flux $F^L$ and mass matrix $M^L$
- high-order in space: flux $F^H$ and mass matrix $M^H$
The beauty of SSP is that the forward Euler substep is a black box

This black box involves two fluxes (not just one as one might think)
- low-order in space: flux $F^L$ and mass matrix $M^L$
- high-order in space: flux $F^H$ and mass matrix $M^H$

Some details

\[
M^L U^{L,n+1} := M^L U^n + \tau F^L(U^n)
\]
\[
M^H U^{H,n+1} := M^H U^n + \tau F^H(U^n)
\]
The beauty of SSP is that the forward Euler substep is a black box

This black box involves two fluxes (not just one as one might think)
  - low-order in space: flux $F^L$ and mass matrix $M^L$
  - high-order in space: flux $F^H$ and mass matrix $M^H$

Some details

$$M^L U^{L,n+1} := M^L U^n + \tau F^L(U^n)$$
$$M^H U^{H,n+1} := M^H U^n + \tau F^H(U^n)$$

Starting from $U^n \in \mathcal{A}^I$,
  - $U^{L,n+1} \in \mathcal{A}^I$ under CFL, but is low-order accurate ...
  - $U^{H,n+1}$ departs from $\mathcal{A}^I$ but is high-order accurate ...

$\Rightarrow$ employ a limiter to construct new update $U^{n+1} \in \mathcal{A}^I$ as close as possible to $U^{H,n+1}$
**Assumption 1.** [forward Euler with low-order flux is IDP under CFL condition]

\[ \exists \tau^* \text{ s.t. } \forall \tau \in (0, \tau^*) \text{ and all } V \in (\mathbb{R}^m)^I, \]

\[ \{ V \in \mathcal{A}^I \} \implies \{ V + \tau (M^L)^{-1} F^L(V) \in \mathcal{A}^I \} \]
**Assumption 1.** [forward Euler with low-order flux is IDP under CFL condition]

\[ \exists \tau^* \text{ s.t. } \forall \tau \in (0, \tau^*) \text{ and all } V \in (\mathbb{R}^m)^I, \]

\[ \{ V \in \mathcal{A}^I \} \implies \{ V + \tau (M_L^{-1}) F_L(V) \in \mathcal{A}^I \} \]

**Assumption 2.** [nonlinear limiting operator]

\[ \ell : \mathcal{A}^I \times (\mathbb{R}^m)^I \times (\mathbb{R}^m)^I \to (\mathbb{R}^m)^I \text{ s.t. for all } (V, F^L, F^H), \]

\[ \{ V + \tau (M_L^{-1}) F_L(V) \in \mathcal{A}^I \} \implies \{ \ell(V, F^L, F^H) \in \mathcal{A}^I \} \]

Key idea: \( \ell(V, F^L, F^H) \) is built as convex combination of

\[ V + \tau (M_L^{-1}) F_L \text{ and } V + \tau (M_L^{-1}) F_H \]
Assumption 1. [forward Euler with low-order flux is IDP under CFL condition]

\[ \exists \tau^* \text{ s.t. } \forall \tau \in (0, \tau^*] \text{ and all } V \in (\mathbb{R}^m)^I, \]

\[ \{ V \in A^I \} \implies \{ V + \tau (M_L)^{-1} F^L(V) \in A^I \} \]

Assumption 2. [nonlinear limiting operator]

\[ \ell : A^I \times (\mathbb{R}^m)^I \times (\mathbb{R}^m)^I \to (\mathbb{R}^m)^I \text{ s.t. for all } (V, F^L, F^H), \]

\[ \{ V + \tau (M_L)^{-1} F^L(V) \in A^I \} \implies \{ \ell(V, F^L, F^H) \in A^I \} \]

Key idea: \( \ell(V, F^L, F^H) \) is built as convex combination of \( V + \tau (M_L)^{-1} F^L \) and \( V + \tau (M_L)^{-1} F^H \)

Notice that both low/high-order updates start from same vector \( V \)
• Given \( U^n \) in the invariant set \( \mathcal{A}^I \)

• The forward Euler step proceeds as follows:
  • compute low-order flux \( F^L(U^n) \)
  • compute high-order flux \( F^H(U^n) \)
  • compute update by limiting

\[
U^{n+1} := \ell(U^n, F^L(U^n), F^H(U^n))
\]
Given $U^n$ in the invariant set $\mathcal{A}^I$

The forward Euler step proceeds as follows:
- compute low-order flux $F_L(U^n)$
- compute high-order flux $F_H(U^n)$
- compute update by limiting

$$U^{n+1} := \ell(U^n, F_L(U^n), F_H(U^n))$$

Proposition. [Forward Euler is IDP]
Let Assumptions 1 and 2 be met. Assume $U^n \in \mathcal{A}^I$. Then, $U^{n+1} \in \mathcal{A}^I$ for all $\tau \in (0, \tau^*)$
The two key ideas of IDP-ERK

- We are ready to go high-order in time!
The two key ideas of IDP-ERK

- We are ready to go high-order in time!
- **Externalize** the limiting process at each ERK stage
The two key ideas of IDP-ERK

- We are ready to go high-order in time!
- **Externalize** the limiting process at each ERK stage
- Rewrite ERK scheme in **incremental form**: at each stage,
  - compute low/high-order updates using a **common** previous IDP-update
  - apply limiter
Butcher tableau of $s$-stage ERK method

- Generic form of Butcher tableau

<table>
<thead>
<tr>
<th>$c_1$</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>$c_2$</td>
<td>$a_{2,1}$ 0</td>
</tr>
<tr>
<td>$c_3$</td>
<td>$a_{3,1}$ $a_{3,2}$ 0</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>$c_s$</td>
<td>$a_{s,1}$ $a_{s,2}$ ... $a_{s,s-1}$ 0</td>
</tr>
<tr>
<td></td>
<td>$b_1$  $b_2$ ... $b_{s-1}$ $b_s$</td>
</tr>
</tbody>
</table>

1. Assume $c_k \geq 0$ for all $k \in \{1, \ldots, s\}$.
2. For all $l \in \{2, \ldots, s\}$, set $l'(l) : = \max\{k < l | c_k \leq c_l\}$.

Think of $l'(l) : = l - 1$ if sequence $(c_l)_{l \in \{1, \ldots, s+1\}}$ is increasing.
Butcher tableau of $s$-stage ERK method

- Generic form of Butcher tableau

\[
\begin{array}{c|ccc}
  c_1 & 0 & & \\
  c_2 & a_{2,1} & 0 & \\
  c_3 & a_{3,1} & a_{3,2} & 0 \\
  \vdots & \vdots & \ddots & \ddots \\
  c_s & a_{s,1} & a_{s,2} & \cdots & a_{s,s-1} & 0 \\
  b_1 & b_2 & \cdots & b_{s-1} & b_s \\
\end{array}
\]

- Rename last line, set $c_1 := 0$ and $c_{s+1} := 1$

\[
\begin{array}{c|ccc}
  0 & 0 & & \\
  c_2 & a_{2,1} & 0 & \\
  c_3 & a_{3,1} & a_{3,2} & 0 \\
  \vdots & \vdots & \ddots & \ddots \\
  c_s & a_{s,1} & a_{s,2} & \cdots & a_{s,s-1} & 0 \\
  1 & a_{s+1,1} & a_{s+1,2} & \cdots & a_{s+1,s-1} & a_{s+1,s} \\
\end{array}
\]
Butcher tableau of \( s \)-stage ERK method

- **Generic form of Butcher tableau**

  \[
  \begin{array}{c|ccc}
  c_1 & 0 \\
  c_2 & a_{2,1} & 0 \\
  c_3 & a_{3,1} & a_{3,2} & 0 \\
  \vdots & \vdots & \ddots & \ddots \\
  c_s & a_{s,1} & a_{s,2} & \cdots & a_{s,s-1} & 0 \\
  \hline
  b_1 & b_2 & \cdots & b_{s-1} & b_s \\
  \end{array}
  \]

- Rename last line, set \( c_1 := 0 \) and \( c_{s+1} := 1 \)

  \[
  \begin{array}{c|ccc}
  0 & 0 \\
  c_2 & a_{2,1} & 0 \\
  c_3 & a_{3,1} & a_{3,2} & 0 \\
  \vdots & \vdots & \ddots & \ddots \\
  c_s & a_{s,1} & a_{s,2} & \cdots & a_{s,s-1} & 0 \\
  \hline
  1 & a_{s+1,1} & a_{s+1,2} & \cdots & a_{s+1,s-1} & a_{s+1,s} \\
  \end{array}
  \]

- Assume \( c_k \geq 0 \) for all \( k \in \{1:s+1\} \)
- For all \( l \in \{2:s+1\} \), set

  \[
  l'(l) := \max\{k < l \mid c_k \leq c_l\}
  \]

  Think of \( l'(l) := l - 1 \) if sequence \((c_l)_{l \in \{1:s+1\}}\) is increasing
Let $U^n \in \mathcal{A}$ and set $U^{n,1} := U^n$
Let $U^n \in \mathcal{A}^l$ and set $U^{n,1} := U^n$

Loop over $l \in \{2:s+1\}$ (stage index)
Details

- Let $U^n \in \mathcal{A}^l$ and set $U^{n,1} := U^n$

- Loop over $l \in \{2:s + 1\}$ (stage index)

- Compute low-order update starting from $U^{n,l''}$ (think of $l' = l - 1$)

$$
\mathcal{M}^L U^{L,l} := \mathcal{M}^L U^{n,l''} + \tau \left( c_l - c_{l'} \right) F^L(U^{n,l''})
\quad := \Phi^L
$$
Let $U^n \in \mathcal{H}^l$ and set $U^{n,1} := U^n$

Loop over $l \in \{2:s + 1\}$ (stage index)

Compute low-order update starting from $U^{n,l''}$ (think of $l' = l - 1$)

$$M^L U^{L,l} := M^L U^{n,l''} + \tau (c_l - c_{l'}) F^L(U^{n,l''})$$

Compute high-order update using same starting point $U^{n,l''}$ (incremental form)

$$M^H U^{H,l} := M^H U^{n,l''} + \tau \sum_{k \in \{1:l-1\}} (a_{l,k} - a_{l',k}) F^H(U^{n,k})$$

Apply limiter:

$$U^{n,l} := \ell(M^L U^{L,l}, M^H U^{H,l})$$

End of loop: return $U^{n+1,s+1} := U^n$
Let $U^n \in \mathcal{A}^l$ and set $U^{n,1} := U^n$

Loop over $l \in \{2:s + 1\}$ (stage index)

Compute low-order update starting from $U^{n,l''}$ (think of $l' = l - 1$)

$$M^L U^{L,l} := M^L U^{n,l''} + \tau \left( (c_l - c_{l'}) F^L(U^{n,l''}) \right) \cdot \Phi^L$$

Compute high-order update using same starting point $U^{n,l''}$ (incremental form)

$$M^H U^{H,l} := M^H U^{n,l''} + \tau \sum_{k \in \{1:l-1\}} (a_{l,k} - a_{l',k}) F^H(U^{n,k}) \cdot \Phi^H$$

Apply limiter: $U^{n,l} := \ell(U^{n,l''}, \Phi^L, \Phi^H)$
Let $U^n \in A^l$ and set $U^{n,1} := U^n$

Loop over $l \in \{2:s+1\}$ (stage index)

Compute low-order update starting from $U^{n,l'}$ (think of $l' = l - 1$)

$$
M^L U^{L,l} := M^L U^{n,l'} + \tau \left( c_I - c_{l'} \right) F^L (U^{n,l'}) := \Phi^L
$$

Compute high-order update using same starting point $U^{n,l'}$ (incremental form)

$$
M^H U^{H,l} := M^H U^{n,l'} + \tau \sum_{k \in \{1:l-1\}} (a_{l,k} - a_{l',k}) F^H (U^{n,k}) := \Phi^H
$$

Apply limiter: $U^{n,l} := \ell(U^{n,l'}, \Phi^L, \Phi^H)$

End of loop: return $U^{n+1} := U^{n,s+1}$
Theorem. [IDP-ERK scheme]

Let Assumptions 1 and 2 be met. Assume $U^n \in \mathcal{A}^l$. Then, $U^{n+1} \in \mathcal{A}^l$ (as well as all intermediate stages) for all

$$\tau \in (0, \tau^*/ \max_{l \in \{2:s+1\}} (c_l - c_r))$$
Main results

- **Theorem.** [IDP-ERK scheme]
  Let Assumptions 1 and 2 be met. Assume $U^n \in \mathcal{A}^l$. Then, $U^{n+1} \in \mathcal{A}^l$ (as well as all intermediate stages) for all

  $\tau \in (0, \tau^* / \max_{l \in \{2:s+1\}} (c_l - c_{l'})]

- **Corollary.** [Optimal efficiency]
  - $c_{\text{eff}} = 1 / (s \max_{l \in \{2:s+1\}} (c_l - c_{l'}))$
  - optimal efficiency (with $c_{\text{eff}} = 1$) reached when points $(c_l)_{l \in \{1:s+1\}}$ are equi-distributed in $[0, 1]$
Examples: second- and third-order methods

- Some optimal methods: RK(2,2;1), RK(3,3;1), RK(4,3;1)

\[
\begin{array}{c|ccc}
0 & 0 & \frac{1}{2} & 0 \\
\hline
\frac{1}{2} & \frac{1}{2} & 0 & 0 \\
\hline
1 & 0 & 1 & 0 \\
\end{array}
\quad
\begin{array}{c|ccc}
0 & 0 & \frac{1}{3} & 0 \\
\hline
\frac{1}{3} & \frac{1}{3} & 0 & 0 \\
\hline
\frac{2}{3} & 0 & \frac{2}{3} & 0 \\
\hline
1 & \frac{1}{4} & 0 & \frac{3}{4} \\
\end{array}
\quad
\begin{array}{c|ccc}
0 & 0 & \frac{1}{4} & 0 \\
\hline
\frac{1}{4} & \frac{1}{4} & 0 & 0 \\
\hline
\frac{1}{2} & 0 & \frac{1}{2} & 0 \\
\hline
\frac{3}{4} & 0 & \frac{1}{4} & \frac{1}{2} \\
\hline
1 & 0 & \frac{5}{3} & \frac{1}{3} & \frac{2}{3} \\
\end{array}
\]
Examples: second- and third-order methods

- Some optimal methods: RK(2,2;1), RK(3,3;1), RK(4,3;1)

- Some non-optimal methods: SSPRK(2,2;\(\frac{1}{2}\)), SSPRK(3,3;\(\frac{1}{3}\))
Examples: fourth-order methods

- Two popular but **sub-optimal** methods: $\text{RK}(4,4; \frac{1}{2})$ and $\text{RK}(4,4; \frac{3}{4})$

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1/2</th>
<th>0</th>
<th>1/2</th>
<th>0</th>
<th>1</th>
<th>0</th>
<th>1/6</th>
<th>1/3</th>
<th>1/3</th>
<th>1/6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1/2</td>
<td>0</td>
<td>1/2</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1/2</td>
<td>1/2</td>
<td>0</td>
<td>1/2</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1/2</td>
<td>0</td>
<td>1/2</td>
<td>0</td>
<td>1/2</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1/3</th>
<th>0</th>
<th>1/3</th>
<th>−1/3</th>
<th>1</th>
<th>0</th>
<th>1/8</th>
<th>3/8</th>
<th>3/8</th>
<th>1/8</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1/3</td>
<td>0</td>
<td>1/3</td>
<td>−1/3</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1/3</td>
<td>1/3</td>
<td>0</td>
<td>1/3</td>
<td>−1/3</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1/3</td>
<td>0</td>
<td>1/3</td>
<td>−1/3</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>−1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>
Examples: fourth-order methods

- Two popular but sub-optimal methods: RK(4,4; $\frac{1}{2}$) and RK(4,4; $\frac{3}{4}$)

- Optimal RK(5,4; 1) and RK(6,4; 1) devised in [AE & JLG 22]  
  [both can be used within an IMEX scheme]
**Examples: fifth-order methods**

- Butcher’s method RK(6,5; $\frac{2}{3}$) (requires $c_6 = 1$)

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1/4</th>
<th>1/4</th>
<th>1/2</th>
<th>3/4</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>$\frac{1}{4}$</td>
<td>$\frac{1}{4}$</td>
<td>$\frac{1}{8}$</td>
<td>$\frac{1}{8}$</td>
<td>0</td>
</tr>
<tr>
<td>1/4</td>
<td>$\frac{1}{16}$</td>
<td>$\frac{1}{8}$</td>
<td>$\frac{1}{8}$</td>
<td>0</td>
<td>$\frac{9}{16}$</td>
<td>0</td>
</tr>
<tr>
<td>1/2</td>
<td>0</td>
<td>$-\frac{1}{2}$</td>
<td>1</td>
<td>0</td>
<td>$\frac{9}{16}$</td>
<td>0</td>
</tr>
<tr>
<td>3/4</td>
<td>$\frac{3}{7}$</td>
<td>0</td>
<td>0</td>
<td>$\frac{12}{7}$</td>
<td>$-\frac{12}{7}$</td>
<td>$\frac{8}{7}$</td>
</tr>
<tr>
<td>1</td>
<td>$-\frac{3}{7}$</td>
<td>$\frac{2}{7}$</td>
<td>$\frac{12}{7}$</td>
<td>$-\frac{12}{7}$</td>
<td>$\frac{8}{7}$</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>$\frac{7}{90}$</td>
<td>0</td>
<td>$\frac{32}{90}$</td>
<td>$\frac{12}{90}$</td>
<td>$\frac{32}{90}$</td>
<td>$\frac{7}{90}$</td>
</tr>
</tbody>
</table>
Examples: fifth-order methods

- Butcher’s method RK(6,5;\(\frac{2}{3}\)) (requires \(c_6 = 1\))

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(\frac{1}{4})</td>
<td>(\frac{1}{4})</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(\frac{1}{4})</td>
<td>(\frac{1}{8})</td>
<td>(\frac{1}{8})</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(\frac{1}{2})</td>
<td>0</td>
<td>(-\frac{1}{2})</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(\frac{3}{4})</td>
<td>(\frac{3}{16})</td>
<td>0</td>
<td>0</td>
<td>(\frac{9}{7})</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>(-\frac{3}{7})</td>
<td>(\frac{2}{7})</td>
<td>(\frac{12}{7})</td>
<td>(-\frac{12}{7})</td>
<td>(\frac{8}{7})</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>(\frac{7}{90})</td>
<td>0</td>
<td>(\frac{32}{90})</td>
<td>(\frac{12}{90})</td>
<td>(\frac{32}{90})</td>
<td>(\frac{7}{90})</td>
</tr>
</tbody>
</table>

- Novel RK(7,5;1) method [AE & JLG 22]

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(\frac{1}{7})</td>
<td>0.1428571428571428</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(\frac{1}{7})</td>
<td>0.0107112392440216</td>
<td>0.27500304647026414</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(\frac{3}{7})</td>
<td>0.4812641640977388</td>
<td>(-0.9634955610240432)</td>
<td></td>
<td></td>
<td>0.9108028254977381</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(\frac{4}{7})</td>
<td>0.3718168921589701</td>
<td>(-0.5615016072648120)</td>
<td>0.5590150320681445</td>
<td></td>
<td></td>
<td>0.202098254662687</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(\frac{5}{7})</td>
<td>0.2210152091353413</td>
<td>0.3526985345185138</td>
<td>(-0.8940286416537777)</td>
<td>0.8097519357352928</td>
<td></td>
<td></td>
<td>0.8097519357352928</td>
<td></td>
</tr>
<tr>
<td>(\frac{6}{7})</td>
<td>0.2038005573304709</td>
<td>(-0.4759394836772968)</td>
<td>1.0938423462712870</td>
<td>(-0.2853403360392873)</td>
<td>0.8097519357352928</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0.0979996468518433</td>
<td>(-0.0044680013474903)</td>
<td>0.3592897484042552</td>
<td>0.0225280828210172</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
All the tests are done by fixing $\text{CFL} \in (0, 1]$ and setting

$$\tau := \text{CFL} \times s \times \tau^*$$

$\Rightarrow$ all the methods perform the same number of flux evaluations and limiting operations independently of $s$

$\Rightarrow$ each method is IDP at least up to $\text{CFL} \leq c_{\text{eff}}$
Methodology for numerical tests

- All the tests are done by fixing CFL $\in (0, 1]$ and setting
  $$\tau := \text{CFL} \times s \times \tau^*$$

  $\implies$ all the methods perform the same number of flux evaluations and
  limiting operations independently of $s$
  $\implies$ each method is IDP at least up to CFL $\leq c_{\text{eff}}$

- Local maximum/minimum principle enforced at every dof

- Global maximum/minimum principle strictly enforced
Methodology for numerical tests

- All the tests are done by fixing $\text{CFL} \in (0, 1]$ and setting

  $$\tau := \text{CFL} \times s \times \tau^*$$

  $\implies$ all the methods perform the same number of flux evaluations and
  limiting operations independently of $s$

  $\implies$ each method is IDP at least up to $\text{CFL} \leq c_{\text{eff}}$

- Local maximum/minimum principle enforced at every dof

- Global maximum/minimum principle strictly enforced

- If the constraints defining $\mathcal{A}$ are affine, use Flux-Corrected Transport (FCT) [Boris & Book 73; Zalesak 79; Kuzmin, Löhner, Turek 12]

- If non-affine constraints, use nonlinear technique [Sanders 88; Coquel & LeFloch 91; Liu & Osher 96; Zhang & Shu 11; Lohman & Kuzmin 16; Guermond, Nazarov, Popov, Tomas 18]
1D linear transport, 4th-order FD (1/3)

- Linear transport, $D := (0, 1)$, periodic BCs

$$\partial_t u + \partial_x u = 0, \quad u_0(x) := \begin{cases} (4 \frac{(x-x_0)(x_1-x)}{(x_1-x_0)^2})^6 & x \in (x_0 := 0.1, 0.4 =: x_1) \\ 0 & \text{otherwise} \end{cases}$$

- 4th order Finite Differences in space
1D linear transport, 4th-order FD (1/3)

- Linear transport, $D := (0, 1)$, periodic BCs

  \[ \partial_t u + \partial_x u = 0, \quad u_0(x) := \begin{cases} 
  (4 \frac{(x-x_0)(x_1-x)}{(x_1-x_0)^2})^6 & x \in (x_0 := 0.1, 0.4 =: x_1) \\
  0 & \text{otherwise} 
\end{cases} \]

- 4th order Finite Differences in space

- In the $L^1$-norm, all the methods achieve the expected convergence order with CFL of the order of 0.5

- Let us look at the more challenging $L^\infty$-error measure
Second-order methods: RK(2,2;1) outperforms SSPRK(2,2;\(\frac{1}{2}\))
1D linear transport, 4th-order FD (2/3)

- **Second-order methods**: RK(2,2;1) outperforms SSPRK(2,2; $\frac{1}{2}$)

<table>
<thead>
<tr>
<th>$I$</th>
<th>RK(2,2;1)</th>
<th>rate</th>
<th>SSPRK(2,2; $\frac{1}{2}$)</th>
<th>rate</th>
<th>CFL = 0.2</th>
<th>RK(2,2;1)</th>
<th>rate</th>
<th>SSPRK(2,2; $\frac{1}{2}$)</th>
<th>rate</th>
<th>CFL = 0.25</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>4.72E-02</td>
<td></td>
<td>1.23E-01</td>
<td></td>
<td>4.91E-02</td>
<td></td>
<td>1.30E-01</td>
<td></td>
<td>4.32E-02</td>
<td>1.60</td>
</tr>
<tr>
<td>100</td>
<td>2.81E-03</td>
<td>4.07</td>
<td>1.50E-02</td>
<td>3.03</td>
<td>4.51E-03</td>
<td>3.44</td>
<td>4.32E-02</td>
<td></td>
<td>2.14E-03</td>
<td>4.34</td>
</tr>
<tr>
<td>200</td>
<td>1.16E-03</td>
<td>1.28</td>
<td>1.24E-03</td>
<td>3.60</td>
<td>2.01E-03</td>
<td>1.17</td>
<td>2.14E-03</td>
<td></td>
<td>5.67E-04</td>
<td>1.91</td>
</tr>
<tr>
<td>400</td>
<td>3.38E-04</td>
<td>1.78</td>
<td>3.47E-04</td>
<td>1.84</td>
<td>5.41E-04</td>
<td>1.89</td>
<td>5.67E-04</td>
<td></td>
<td>1.48E-04</td>
<td>1.94</td>
</tr>
<tr>
<td>800</td>
<td>8.79E-05</td>
<td>1.94</td>
<td>9.28E-05</td>
<td>1.90</td>
<td>1.38E-04</td>
<td>1.97</td>
<td>1.48E-04</td>
<td></td>
<td>1.48E-04</td>
<td>1.94</td>
</tr>
<tr>
<td>1600</td>
<td>2.22E-05</td>
<td>1.98</td>
<td>2.33E-05</td>
<td>1.99</td>
<td>3.47E-05</td>
<td>1.99</td>
<td>3.78E-05</td>
<td></td>
<td>1.97</td>
<td></td>
</tr>
<tr>
<td>3200</td>
<td>5.58E-06</td>
<td>1.99</td>
<td>5.92E-06</td>
<td>1.98</td>
<td>8.73E-06</td>
<td>1.99</td>
<td>5.36E-05</td>
<td></td>
<td>-5.00</td>
<td></td>
</tr>
</tbody>
</table>

- **Third-order methods**: SSPRK(3,3; $\frac{1}{3}$) behaves poorly, RK(4,3;1) performs best

<table>
<thead>
<tr>
<th>$I$</th>
<th>RK(3,3;1)</th>
<th>rate</th>
<th>SSPRK(3,3; $\frac{1}{3}$)</th>
<th>rate</th>
<th>CFL = 0.05</th>
<th>RK(3,3;1)</th>
<th>rate</th>
<th>SSPRK(3,3; $\frac{1}{3}$)</th>
<th>rate</th>
<th>CFL = 0.25</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>5.15E-02</td>
<td></td>
<td>4.76E-02</td>
<td></td>
<td>5.15E-02</td>
<td></td>
<td>1.55E-01</td>
<td></td>
<td>6.08E-02</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>5.41E-03</td>
<td>3.25</td>
<td>5.41E-03</td>
<td>3.14</td>
<td>5.41E-03</td>
<td>3.25</td>
<td>6.12E-02</td>
<td></td>
<td>6.15E-03</td>
<td>3.31</td>
</tr>
<tr>
<td>200</td>
<td>3.79E-04</td>
<td>3.83</td>
<td>3.79E-04</td>
<td>3.83</td>
<td>3.79E-04</td>
<td>3.83</td>
<td>1.07E-03</td>
<td></td>
<td>3.83E-04</td>
<td>4.01</td>
</tr>
<tr>
<td>400</td>
<td>2.27E-05</td>
<td>4.06</td>
<td>2.27E-05</td>
<td>4.06</td>
<td>2.27E-05</td>
<td>4.06</td>
<td>2.18E-04</td>
<td></td>
<td>2.30E-05</td>
<td>4.06</td>
</tr>
<tr>
<td>800</td>
<td>1.58E-06</td>
<td>3.85</td>
<td>1.58E-06</td>
<td>3.85</td>
<td>1.58E-06</td>
<td>3.85</td>
<td>6.41E-05</td>
<td></td>
<td>1.59E-06</td>
<td>3.85</td>
</tr>
<tr>
<td>1600</td>
<td>9.12E-08</td>
<td>4.12</td>
<td>1.22E-07</td>
<td>3.69</td>
<td>8.13E-08</td>
<td>4.28</td>
<td>1.83E-05</td>
<td></td>
<td>8.25E-08</td>
<td>4.27</td>
</tr>
<tr>
<td>3200</td>
<td>1.52E-08</td>
<td>2.58</td>
<td>6.84E-08</td>
<td>0.84</td>
<td>5.31E-09</td>
<td>3.94</td>
<td>5.39E-06</td>
<td></td>
<td>5.39E-09</td>
<td>3.94</td>
</tr>
</tbody>
</table>
Fourth-order methods: RK(5,4;1) outperforms SSPRK(5,4;\( \frac{1}{2} \))

<table>
<thead>
<tr>
<th>$I$</th>
<th>CFL = 0.05</th>
<th>CFL = 0.2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RK(4,4;( \frac{1}{2} )) rate</td>
<td>SSPRK(5,4;( \frac{1}{2} )) rate</td>
</tr>
<tr>
<td>50</td>
<td>4.32E-02</td>
<td>5.37E-02</td>
</tr>
<tr>
<td>100</td>
<td>5.41E-03</td>
<td>5.09E-03</td>
</tr>
<tr>
<td>200</td>
<td>3.79E-04</td>
<td>3.04E-04</td>
</tr>
<tr>
<td>400</td>
<td>2.27E-05</td>
<td>1.91E-05</td>
</tr>
<tr>
<td>800</td>
<td>1.58E-06</td>
<td>1.91E-06</td>
</tr>
<tr>
<td>1600</td>
<td>8.13E-08</td>
<td>7.45E-08</td>
</tr>
<tr>
<td>3200</td>
<td>5.36E-09</td>
<td>4.65E-09</td>
</tr>
</tbody>
</table>
Fourth-order methods: RK(5,4;1) outperforms SSPRK(5,4;1/2)

<table>
<thead>
<tr>
<th>I</th>
<th>CFL = 0.05</th>
<th>CFL = 0.2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RK(4,4;1/2) rate</td>
<td>SSPRK(5,4;1/2) rate</td>
</tr>
<tr>
<td>100</td>
<td>5.41E-03 3.00</td>
<td>5.09E-03 3.40</td>
</tr>
<tr>
<td>400</td>
<td>2.27E-05 4.06</td>
<td>1.91E-05 3.99</td>
</tr>
<tr>
<td>800</td>
<td>1.58E-06 3.85</td>
<td>1.19E-06 4.00</td>
</tr>
<tr>
<td>1600</td>
<td>8.13E-08 4.28</td>
<td>7.45E-08 4.00</td>
</tr>
<tr>
<td>3200</td>
<td>5.36E-09 3.92</td>
<td>4.65E-09 4.00</td>
</tr>
</tbody>
</table>

Fifth-order methods: no SSP competitor!

<table>
<thead>
<tr>
<th>I</th>
<th>CFL = 0.02</th>
<th>CFL = 0.025</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RK(6,5;1/3) rate</td>
<td>RK(7,5;1) rate</td>
</tr>
<tr>
<td>100</td>
<td>5.41E-03 3.26</td>
<td>5.41E-03 3.26</td>
</tr>
<tr>
<td>200</td>
<td>3.79E-04 3.83</td>
<td>3.79E-04 3.83</td>
</tr>
<tr>
<td>400</td>
<td>2.27E-05 4.06</td>
<td>2.27E-05 4.06</td>
</tr>
<tr>
<td>800</td>
<td>1.58E-06 3.85</td>
<td>1.58E-06 3.85</td>
</tr>
<tr>
<td>1600</td>
<td>8.48E-08 4.22</td>
<td>8.13E-08 4.28</td>
</tr>
<tr>
<td>3200</td>
<td>7.10E-09 3.58</td>
<td>5.92E-09 3.78</td>
</tr>
</tbody>
</table>
Linear transport with non-smooth solution

- Three-solid problem with rotating advection field [Zalesak 79]
- Continuous $\mathbb{P}^1$-FEM on unstructured non-nested Delaunay meshes
- Solutions at $T = 1$ using RK(2,2;1) (midpoint rule) at CFL = 0.25
  [From left to right: $I = 6561$; $I = 24917$; $I = 98648$; $I = 389860$ dofs]

Relative error in $L^1$-norm for RK(2,2;1) and RK(4,3;1)

<table>
<thead>
<tr>
<th>$I$</th>
<th>RK(2,2;1)</th>
<th>rate</th>
<th>RK(4,3;1)</th>
<th>rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1605</td>
<td>2.45E-01</td>
<td>–</td>
<td>2.49E-01</td>
<td>–</td>
</tr>
<tr>
<td>6561</td>
<td>1.28E-01</td>
<td>0.93</td>
<td>1.31E-01</td>
<td>0.92</td>
</tr>
<tr>
<td>24917</td>
<td>7.34E-02</td>
<td>0.81</td>
<td>7.49E-02</td>
<td>0.84</td>
</tr>
<tr>
<td>98648</td>
<td>4.26E-02</td>
<td>0.78</td>
<td>4.44E-02</td>
<td>0.76</td>
</tr>
<tr>
<td>389860</td>
<td>2.44E-02</td>
<td>0.81</td>
<td>2.56E-02</td>
<td>0.80</td>
</tr>
</tbody>
</table>
2D Burgers’ equation (1/3)

- 2D Burgers’ equation in $D := (-.25, 1.75)^2$

\[
\partial_t u + \nabla \cdot f(u) = 0, \quad f(u) := \frac{1}{2} (u^2, u^2)^T
\]

with initial data

\[
u_0(x) := \begin{cases} 
1 & \text{if } |x_1 - \frac{1}{2}| \leq 1 \text{ and } |x_2 - \frac{1}{2}| \leq 1 \\
-a & \text{otherwise}
\end{cases}
\]

- This problem exhibits many sonic points, which makes methods with too little low/high-order viscosity to fail [Guermond, Popov 17]

- Solution at $T = 0.65$ computed with RK(4,3;1) at CFL = 0.25 using 801$^2$ grid points
$T = 0.65$, $CFL = 0.25$, relative $L^1$-error for all the methods

<table>
<thead>
<tr>
<th>$I$</th>
<th>RK(2,1;1) rate</th>
<th>SSPRK(2,2; 1/2) rate</th>
<th>RK(4,4; 3/4) rate</th>
<th>SSPRK(5,4; 2/3) rate</th>
<th>RK(6,4; 1) rate</th>
<th>RK(7,5; 1) rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>6.61E-02</td>
<td>6.70E-02</td>
<td>6.60E-02</td>
<td>6.65E-02</td>
<td>6.62E-02</td>
<td>6.60E-02</td>
</tr>
<tr>
<td>100</td>
<td>3.31E-02</td>
<td>3.34E-02</td>
<td>3.31E-02</td>
<td>3.32E-02</td>
<td>3.30E-02</td>
<td>3.30E-02</td>
</tr>
<tr>
<td>200</td>
<td>2.12E-02</td>
<td>2.12E-02</td>
<td>2.12E-02</td>
<td>2.12E-02</td>
<td>2.12E-02</td>
<td>2.12E-02</td>
</tr>
<tr>
<td>400</td>
<td>1.20E-02</td>
<td>1.16E-02</td>
<td>1.18E-02</td>
<td>1.20E-02</td>
<td>1.20E-02</td>
<td>1.20E-02</td>
</tr>
<tr>
<td>800</td>
<td>6.04E-03</td>
<td>5.73E-03</td>
<td>6.06E-03</td>
<td>6.04E-03</td>
<td>6.04E-03</td>
<td>6.04E-03</td>
</tr>
</tbody>
</table>

At moderate CFL, all the methods converge equally well (all at order one)
2D Burgers’ equation (3/3)

- Challenge methods by increasing CFL

- Results for second- and third-order methods (top), fourth-order, fifth-order methods plus a recap for all optimal methods

- \[ \Rightarrow \text{SSPRK (2,2) and SSPRK(3,3) start loosing accuracy at } \text{CFL} \approx 0.5, \text{ whereas IDP-ERK methods behave well over whole CFL range} \]
Conclusions from numerical tests

- All IDP-ERK methods perform as well, and often better, than SSPRK methods of the same order.
Conclusions from numerical tests

- All IDP-ERK methods perform as well, and often better, than SSPRK methods of the same order.
- RK(2,2;1) (midpoint rule) outperforms popular SSPRK(2,2;\frac{1}{2})
- RK(4,3;1) (vastly) outperforms popular SSPRK(3,3;\frac{1}{3})
Conclusions from numerical tests

- All IDP-ERK methods perform as well, and often better, than SSPRK methods of the same order
- RK(2,2;1) (midpoint rule) outperforms popular SSPRK(2,2;\(\frac{1}{2}\))
- RK(4,3;1) (vastly) outperforms popular SSPRK(3,3;\(\frac{1}{3}\))
- The considered fourth-order methods provide comparable results
Conclusions from numerical tests

- All IDP-ERK methods perform as well, and often better, than SSPRK methods of the same order.
- RK(2,2;1) (midpoint rule) outperforms popular SSPRK(2,2;\(\frac{1}{2}\)).
- RK(4,3;1) (vastly) outperforms popular SSPRK(3,3;\(\frac{1}{3}\)).
- The considered fourth-order methods provide comparable results.
- Novel fifth-order IDP-ERK method with no SSP competitor.
New perspective on IMEX schemes
Main ideas

- Consider **low-order and high-order** fluxes for
  - hyperbolic terms
  - parabolic (diffusion/relaxation) terms
Main ideas

- Consider **low-order and high-order** fluxes for
  - **hyperbolic** terms
  - **parabolic (diffusion/relaxation)** terms

- **Quasi-linearization** of **parabolic fluxes** (both low- and high-order)
Main ideas

- Consider low-order and high-order fluxes for
  - hyperbolic terms
  - parabolic (diffusion/relaxation) terms

- Quasi-linearization of parabolic fluxes (both low- and high-order)

- Key assumption: Under CFL condition, both
  - forward Euler with low-order hyperbolic flux
  - backward Euler with low-order quasi-linear parabolic flux
  are IDP
Main ideas

- Consider **low-order and high-order** fluxes for
  - hyperbolic terms
  - parabolic (diffusion/relaxation) terms

- **Quasi-linearization** of parabolic fluxes (both low- and high-order)

- **Key assumption**: Under CFL condition, both
  - forward Euler with low-order hyperbolic flux
  - backward Euler with low-order quasi-linear parabolic flux
  are IDP

- Rewrite IMEX scheme in incremental form
Main ideas

- Consider **low-order and high-order** fluxes for
  - hyperbolic terms
  - parabolic (diffusion/relaxation) terms

- **Quasi-linearization** of **parabolic fluxes** (both low- and high-order)

- **Key assumption:** Under CFL condition, both
  - forward Euler with low-order hyperbolic flux
  - backward Euler with low-order quasi-linear parabolic flux
  are IDP

- Rewrite IMEX scheme in **incremental form**

- Apply (possibly distinct) limiters to **hyperbolic** and **parabolic** substeps
Euler IDP-IMEX scheme

- Gentle introduce ideas on Euler IDP-IMEX scheme
Euler IDP-IMEX scheme

- Gentle introduce ideas on Euler IDP-IMEX scheme

- $F^L$: Low-order approx. of hyperbolic flux $-\nabla \cdot f(u)$

- $G_{L,\text{lin}}^L(W^n; \cdot)$: Low-order quasi-linear approx. of parabolic flux $-\nabla \cdot g(u, \nabla u) + S(u)$
Euler IDP-IMEX scheme

- Gentle introduce ideas on Euler IDP-IMEX scheme
- $F^L$: Low-order approx. of hyperbolic flux $-\nabla \cdot f(u)$
- $G^L,\text{lin}(W^n; \cdot)$: Low-order quasi-linear approx. of parabolic flux $-\nabla \cdot g(u, \nabla u) + S(u)$

Consider low-order quasi-linear update

$$\mathbb{M}^L U^{L,n+1} = \mathbb{M}^L U^n + \tau F^L(U^n) + \tau G^L,\text{lin}(W^L,n; U^{L,n+1})$$

$$= : \mathbb{M}^L W^{L,n}$$
Gentle introduce ideas on Euler IDP-IMEX scheme

- **\( F^L \):** Low-order approx. of hyperbolic flux \(-\nabla \cdot f(u)\)

- **\( G^{L,\text{lin}}(W^n; \cdot) \):** Low-order quasi-linear approx. of parabolic flux 
  \(-\nabla \cdot g(u, \nabla u) + S(u)\)

Consider low-order quasi-linear update

\[
M^L U_{L, n+1} = M^L U^n + \tau F^L(U^n) + \tau G^{L,\text{lin}}(W^n; U_{L, n+1}) =: M^L W^n
\]

This can be decomposed as

- hyperbolic sub-step (explicit update):

  \[
  W^{L, n} := U^n + \tau (M^L)^{-1} F^L(U^n)
  \]

- parabolic sub-step (quasi-linear solve):

  \[
  U_{L, n+1} := (I - \tau (M^L)^{-1} G^{L,\text{lin}}(W^n; \cdot))^{-1}(W^{L, n})
  \]
**Assumption 1.** There exists $\tau^* > 0$ s.t. for all $\tau \in (0, \tau^*],$

- forward Euler with low-order hyperbolic flux is IDP:

$$\{ \mathbf{V} \in \mathcal{A} \} \implies \{ \mathbf{V} + \tau (\mathbf{M}^L)^{-1} \mathbf{F}^L(\mathbf{V}) \in \mathcal{A} \}$$

Notice that quasi-linearization is performed at $\mathbf{V}$. Proposition.

**Low-order Euler IDP-IMEX**

Let Assumption 1 hold. Assume that $\mathbf{U}_n \in \mathcal{A}$ and $\mathbf{g} \in (0, \tau^*].$ Then,

$$\mathbf{U}_{n+1} \in \mathcal{A}$$
Key assumption on low-order fluxes

- **Assumption 1.** There exists $\tau^* > 0$ s.t. for all $\tau \in (0, \tau^*]$, forward Euler with low-order hyperbolic flux is IDP:

\[
\{ V \in \mathcal{A}^I \} \implies \{ V + \tau (M^L)^{-1} F^L(V) \in \mathcal{A}^I \}
\]

- backward Euler with low-order quasi-linear parabolic flux is IDP: For all $W \in \mathcal{A}^I$, $I - \tau (M^L)^{-1} G^L,\text{lin}(W; \cdot) : (\mathbb{R}^m)^I \to (\mathbb{R}^m)^I$ is bijective and

\[
\{ V \in \mathcal{A}^I \} \implies \{ (I - \tau M^L)^{-1} G^L,\text{lin}(V; \cdot))^{-1}(V) \in \mathcal{A}^I \}
\]

Notice that quasi-linearization is performed at $V$
Key assumption on low-order fluxes

- **Assumption 1.** There exists $\tau^* > 0$ s.t. for all $\tau \in (0, \tau^*]$, 
  - forward Euler with low-order hyperbolic flux is IDP:
    \[
    \{V \in A^I\} \implies \{V + \tau(M^L)^{-1}F^L(V) \in A^I\}
    \]
  - backward Euler with low-order quasi-linear parabolic flux is IDP: For all $W \in A^I$, $I - \tau(M^L)^{-1}G^L,\text{lin}(W; \cdot) : (\mathbb{R}^m)^I \rightarrow (\mathbb{R}^m)^I$ is bijective and
    \[
    \{V \in A^I\} \implies \{(I - \tau(M^L)^{-1}G^L,\text{lin}(V; \cdot))^{-1}(V) \in A^I\}
    \]
  Notice that quasi-linearization is performed at $V$

- **Proposition.** [Low-order Euler IDP-IMEX] 
  Let Assumption 1 hold. Assume that $U^n \in A^I$ and $\tau \in (0, \tau^*]$. Then, $U^{L,n+1} \in A^I$
We want to use high-order fluxes in space!
We want to use high-order fluxes in space!

Assumption 2. There exist two nonlinear limiting operators

$$\ell^{\text{hyp}}, \ell^{\text{par}} : \mathcal{A}^I \times (\mathbb{R}^m)^I \times (\mathbb{R}^m)^I \to (\mathbb{R}^m)^I$$

such that

- for all \((V, \Phi^L, \Phi^H) \in \mathcal{A}^I \times (\mathbb{R}^m)^I \times (\mathbb{R}^m)^I\),
  \[
  \{V + \tau(M^L)^{-1}\Phi^L \in \mathcal{A}^I\} \implies \{\ell^{\text{hyp}}(V, \Phi^L, \Phi^H) \in \mathcal{A}^I\}
  \]

- for all \((W, \Psi^L, \Psi^H) \in \mathcal{A}^I \times (\mathbb{R}^m)^I \times (\mathbb{R}^m)^I\),
  \[
  \{W + \tau(M^L)^{-1}\Psi^L \in \mathcal{A}^I\} \implies \{\ell^{\text{par}}(W, \Psi^L, \Psi^H) \in \mathcal{A}^I\}
  \]
We want to use high-order fluxes in space!

Assumption 2. There exist two nonlinear limiting operators

$$\ell^{\text{hyp}}, \ell^{\text{par}} : \mathcal{A}^I \times (\mathbb{R}^m)^I \times (\mathbb{R}^m)^I \rightarrow (\mathbb{R}^m)^I$$

such that

- for all $$(V, \Phi^L, \Phi^H) \in \mathcal{A}^I \times (\mathbb{R}^m)^I \times (\mathbb{R}^m)^I$$,
  $$\{V + \tau(M^L)^{-1}\Phi^L \in \mathcal{A}^I\} \implies \{\ell^{\text{hyp}}(V, \Phi^L, \Phi^H) \in \mathcal{A}^I\}$$

- for all $$(W, \Psi^L, \Psi^H) \in \mathcal{A}^I \times (\mathbb{R}^m)^I \times (\mathbb{R}^m)^I$$,
  $$\{W + \tau(M^L)^{-1}\Psi^L \in \mathcal{A}^I\} \implies \{\ell^{\text{par}}(W, \Psi^L, \Psi^H) \in \mathcal{A}^I\}$$

Important remarks

- the invariant domains enforced by the two limiters can be different
- bounds for limiting are deduced from the low-order updates
Given $U^n \in \mathcal{A}^I$, high-order Euler IDP-IMEX proceeds as follows:

\[
\begin{align*}
U^n & \xrightarrow{(1)} (W_{L,n+1}, W_{H,n+1}) \xrightarrow{(2)} W_{n+1} \xrightarrow{(3)} (U_{L,n+1}, U_{H,n+1}) \xrightarrow{(4)} U_{n+1}
\end{align*}
\]

hyperbolic step  
parabolic step
Given $U^n \in \mathcal{A}$, high-order Euler IDP-IMEX proceeds as follows:

\[
\begin{align*}
U^n &\xrightarrow{(1)} (W^{L,n+1}, W^{H,n+1}) \xrightarrow{(2)} W^{n+1} \xrightarrow{(3)} (U^{L,n+1}, U^{H,n+1}) \xrightarrow{(4)} U^{n+1}
\end{align*}
\]

- **Hyperbolic step**
  - Hyperbolic steps (1) and (2): compute low/high-order updates and limit
    \[
    \begin{align*}
    M^L W^{L,n+1} &= M^L U^n + \tau F^L (U^n), \\
    M^H W^{H,n+1} &= M^H U^n + \tau F^H (U^n), \\
    W^{n+1} &= \ell^{\text{hyp}} (U^n, \Phi^L, \Phi^H)
    \end{align*}
    \]
Given $U^n \in \mathcal{A}^I$, high-order Euler IDP-IMEX proceeds as follows:

\[ U^n \xrightarrow{(1)} (W_{L,n+1}, W_{H,n+1}) \xrightarrow{(2)} W_{n+1} \]

hyperbolic step

\[ U^n \xrightarrow{(3)} (U_{L,n+1}, U_{H,n+1}) \xrightarrow{(4)} U_{n+1} \]

parabolic step

- **Hyperbolic steps (1) and (2):** compute low/high-order updates and limit

  \[
  \begin{align*}
  M^L W_{L,n+1} &:= M^L U^n + \tau F^L(U^n), \\
  M^H W_{H,n+1} &:= M^H U^n + \tau F^H(U^n), \\
  W_{n+1} &= \ell_{\text{hyp}}(U^n, \Phi^L, \Phi^H)
  \end{align*}
  \]

- **Parabolic steps (3) and (4):** compute low/high-order updates (quasi-linear solves) and limit

  \[
  \begin{align*}
  M^L U_{L,n+1} - \tau G^L,\text{lin}(W_{n+1}, U_{L,n+1}) &:= M^L W_{n+1}, \\
  M^H U_{H,n+1} - \tau G^H,\text{lin}(U^n, U_{H,n+1}) &:= M^H W_{n+1}, \\
  U_{n+1} &= \ell_{\text{par}}(W_{n+1}, \Psi^L, \Psi^H)
  \end{align*}
  \]
Given $U^n \in \mathcal{A}^I$, high-order Euler IDP-IMEX proceeds as follows:

$U^n \xrightarrow{(1)} (W^L,n+1, W^H,n+1) \xrightarrow{(2)} W^{n+1} \xrightarrow{(3)} (U^L,n+1, U^H,n+1) \xrightarrow{(4)} U^{n+1}$

- **Hyperbolic steps (1) and (2):** compute low/high-order updates and limit
  
  \[
  M^L W^L,n+1 := M^L U^n + \tau F^L(U^n), \quad W^{n+1} := \ell_{\text{hyp}}(U^n, \Phi^L, \Phi^H),
  \]
  
  \[
  M^H W^H,n+1 := M^H U^n + \tau F^H(U^n),
  \]

- **Parabolic steps (3) and (4):** compute low/high-order updates (quasi-linear solves) and limit
  
  \[
  M^L U^L,n+1 - \tau G^L,\text{lin}(W^{n+1}, U^L,n+1) := M^L W^{n+1}, \quad U^{n+1} := \ell_{\text{par}}(W^{n+1}, \Psi^L, \Psi^H),
  \]
  
  \[
  M^H U^H,n+1 - \tau G^H,\text{lin}(U^n, U^H,n+1) := M^H W^{n+1},
  \]

**Proposition.** [High-order Euler IDP-IMEX]

Let Assumptions 1 and 2 hold. Assume that $U^n \in \mathcal{A}^I$ and $\tau \in (0, \tau^*]$. Then, $U^{n+1} \in \mathcal{A}^I$.
We are ready to go high-order in time!

**Key idea.** Consider the following two ODE systems on \((t^n, t^{n+1})\):

\[
\begin{align*}
\mathbf{M}^L \partial_t \mathbf{U} &= \mathbf{F}^L(\mathbf{U}) + \mathbf{G}^{L,\text{lin}}(\mathbf{W}^{n,l}; \mathbf{U}) \quad \text{(at each stage } l) \\
\mathbf{M}^H \partial_t \mathbf{U} &= \mathbf{F}^H(\mathbf{U}) + \mathbf{G}^H(\mathbf{U}) - \mathbf{G}^{H,\text{lin}}(\mathbf{U}^n; \mathbf{U}) + \mathbf{G}^{H,\text{lin}}(\mathbf{U}^n; \mathbf{U}) \quad \text{explicit} \quad \text{implicit}
\end{align*}
\]
### Explicit Butcher tableau

| \( c_2 \) | \( a_{2,1}^e \) & 0       \\ | \( c_3 \) | \( a_{3,1}^e \) & \( a_{3,2}^e \) & 0 \\ | \( \vdots \) | \( \vdots \) & \( \vdots \) & \( \vdots \) & \( \vdots \) \\ | \( c_s \) | \( a_{s,1}^e \) & \( a_{s,2}^e \) & \( \cdots \) & \( a_{s,s-1}^e \) & 0 \\ \hline 0  & 0       \\ 1  & \( a_{s+1,1}^e \) & \( a_{s+1,2}^e \) & \( \cdots \) & \( a_{s+1,s-1}^e \) & \( a_{s+1,s}^e \) \\
Butcher tableaux

- **Explicit Butcher tableau**

  
  \[
  \begin{array}{c|cccc}
  0 & 0 & & & \\
  c_2 & a_{2,1}^e & a_{2,2}^e & 0 & \\
  c_3 & a_{3,1}^e & a_{3,2}^e & 0 & \\
  \vdots & \vdots & \vdots & \vdots & \vdots \\
  c_s & a_{s,1}^e & a_{s,2}^e & \cdots & a_{s,s-1}^e & 0 \\
  1 & a_{s+1,1}^e & a_{s+1,2}^e & \cdots & a_{s+1,s-1}^e & a_{s+1,s}^e \\
  \end{array}
  \]

- **Implicit Butcher tableau**

  
  \[
  \begin{array}{c|cccc}
  0 & 0 & & & \\
  c_2 & a_{2,1}^i & a_{2,2}^i & & \\
  c_3 & a_{3,1}^i & a_{3,2}^i & a_{3,3}^i & \\
  \vdots & \vdots & \vdots & \vdots & \vdots \\
  c_s & a_{s,1}^i & a_{s,2}^i & \cdots & a_{s,s-1}^i & a_{s,s}^i \\
  1 & a_{s+1,1}^i & a_{s+1,2}^i & \cdots & a_{s+1,s-1}^i & a_{s+1,s}^i \\
  \end{array}
  \]
**Butcher tableaux**

- **Explicit Butcher tableau**

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$c_2$</td>
<td>$a_{2,1}^e$</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>$c_3$</td>
<td>$a_{3,1}^e$</td>
<td>$a_{3,2}^e$</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$c_s$</td>
<td>$a_{s,1}^e$</td>
<td>$a_{s,2}^e$</td>
<td>...</td>
</tr>
<tr>
<td>1</td>
<td>$a_{s+1,1}^e$</td>
<td>$a_{s+1,2}^e$</td>
<td>...</td>
</tr>
</tbody>
</table>

- **Implicit Butcher tableau**

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$c_2$</td>
<td>$a_{2,1}^i$</td>
<td>$a_{2,2}^i$</td>
<td></td>
</tr>
<tr>
<td>$c_3$</td>
<td>$a_{3,1}^i$</td>
<td>$a_{3,2}^i$</td>
<td>$a_{3,3}^i$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$c_s$</td>
<td>$a_{s,1}^i$</td>
<td>$a_{s,2}^i$</td>
<td>...</td>
</tr>
<tr>
<td>1</td>
<td>$a_{s+1,1}^i$</td>
<td>$a_{s+1,2}^i$</td>
<td>...</td>
</tr>
</tbody>
</table>

- Both tableaux share the same coefficients $(c_l)_{l \in \{1:s+1\}}$; recall the notation $l'(l) := \max\{k < l \mid c_k \leq c_l\}$ (think of $l'(l) = l - 1$)
Given $U^n \in \mathcal{A}^l$, set $U^{n,1} := U^n$

At each stage $l \in \{2:s+1\}$, one performs the following steps:

$$U^{n,l'} \xrightarrow{(1)} (W^{L,l}, W^{H,l}) \xrightarrow{(2)} W^{n,l} \xrightarrow{(3)} (U^{L,l}, U^{H,l}) \xrightarrow{(4)} U^{n,l}$$

- Hyperbolic step
- Parabolic step
Details (1/2)

- Given $U^n \in \mathcal{A}^l$, set $U^{n,1} := U^n$

- At each stage $l \in \{2:s + 1\}$, one performs the following steps:

  $U^{n,l'} \xrightarrow{(1)} (W^L,l, W^H,l) \xrightarrow{(2)} W^{n,l} \xrightarrow{(3)} (U^L,l, U^H,l) \xrightarrow{(4)} U^{n,l}$

  hyperbolic step  \hspace{2cm} parabolic step

- Hyperbolic steps (1) and (2): compute low/high-order updates

  $M^L W^L,l := M^L U^{n,l'} + \tau (c_l - c_{l'}) F^L (U^{n,l'})$

  $M^H W^H,l := M^H U^{n,l'} + \tau \sum_{k \in \{1:l-1\}} (a^e_{l,k} - a^e_{l',k}) F^H (U^{n,k})$

  and limit

  $W^{n,l} := \ell^{hyp} (U^{n,l'}, \Phi^L, \Phi^H)$
Recall $W^{n,l}$ just computed from hyperbolic steps (1) and (2)
Recall $W^{n,l}$ just computed from hyperbolic steps (1) and (2)

Parabolic steps (3) and (4): compute low/high-order updates

\[
M^L W^{n,l} = M^L W^{n,l} + \tau (c_l - c_{l'}) G^{L,\text{lin}} (W^{n,l}; U^{L,l}) = M^L W^{n,l}
\]

\[
M^H U^{H,l} = M^H W^{n,l} + \tau a^i_{l,l} G^{H,\text{lin}} (U^n; U^{H,l}) = M^H W^{n,l}
\]

\[
\left( \Delta_l := \sum_{k \in \{1:l-1\}} (a^i_{l,k} - a^i_{l',k}) G^{H,\text{lin}} (U^n; U^{n,k}) + \sum_{k \in \{1:l-1\}} (a^e_{l,k} - a^e_{l',k}) (G^H (U^{n,k}) - G^{H,\text{lin}} (U^n; U^{n,k})) \right)
\]

Notice that $a^i_{l,l} = 0$ for $l = s + 1$ (final high-order stage is explicit)

Limit: $U^{n+1} := \ell^{\text{par}} (W^{n,l}, \Psi^L, \Psi^H)$
Recall $W^{n,l}$ just computed from hyperbolic steps (1) and (2)

Parabolic steps (3) and (4): compute low/high-order updates

$$M_L U^{L,l} - \tau (c_l - c_{l'}) G^{L,\text{lin}}(W^{n,l}; U^{L,l}) := M_L W^{n,l}$$

$$M_H U^{H,l} - \tau a_{l,l}^i G^{H,\text{lin}}(U^n; U^{H,l}) := M_H W^{n,l} + \tau \Delta_l$$

$$\Delta_l := \sum_{k \in \{1:l-1\}} (a_{l,k}^i - a_{l',k}^i) G^{H,\text{lin}}(U^n; U^{n,k}) + \sum_{k \in \{1:l-1\}} (a_{l,k}^e - a_{l',k}^e) (G^H(U^{n,k}) - G^{H,\text{lin}}(U^n; U^{n,k}))$$

Notice that $a_{l,l}^i = 0$ for $l = s + 1$ (final high-order stage is explicit)

Limit: $U^{n+1} := \ell^{\text{par}}(W^{n,l}, \Psi^L, \Psi^H)$

**Theorem.** [High-order IDP-IMEX]

Let Assumptions 1 and 2 hold. Assume that $U^n \in \mathcal{A}^l$. Then, $U^{n+1} \in \mathcal{A}^l$

(as well as all intermediate stages) $\forall \tau \in (0, \tau^*/\max_{l \in \{2:s+1\}} (c_l - c_{l'})]$
The design of low-order linearized parabolic flux $g_{L,\text{lin}}$ is problem-dependent.
The design of low-order linearized parabolic flux $G_{L,\text{lin}}$ is problem-dependent.

The whole scheme can be rewritten using conservative limiters.
The design of low-order linearized parabolic flux $G_{L,\text{lin}}$ is problem-dependent.

The whole scheme can be rewritten using conservative limiters.

The setting allows for the hyperbolic and parabolic problems to be solved each with its own natural set of variables:
- conservative for Euler, primitive for Navier–Stokes.
**Examples: second-order IMEX**

- **Heun + Crank–Nicolson:** efficiency ratio is $\frac{1}{2}$

\[
\begin{array}{c|cc}
0 & 0 & 0 \\
1 & 1 & 0 \\
\hline
1 & \frac{1}{2} & \frac{1}{2} \\
\end{array}
\quad
\begin{array}{c|cc}
0 & 0 & \frac{1}{2} \\
1 & \frac{1}{2} & \frac{1}{2} \\
\hline
1 & \frac{1}{2} & \frac{1}{2} \\
\end{array}
\]
Examples: second-order IMEX

- Heun + Crank–Nicolson: efficiency ratio is $\frac{1}{2}$

$$
\begin{array}{c|ccc}
0 & 0 & & \\
1 & 1 & 0 & \\
\hline
1 & \frac{1}{2} & \frac{1}{2} & \\
\end{array}
\quad
\begin{array}{c|ccc}
0 & 0 & & \\
1 & \frac{1}{2} & \frac{1}{2} & \\
\hline
1 & \frac{1}{2} & \frac{1}{2} & \\
\end{array}
$$

- Explicit + implicit midpoint rules: efficiency ratio is $1$

$$
\begin{array}{c|ccc}
0 & 0 & & \\
\frac{1}{2} & \frac{1}{2} & 0 & \\
\hline
1 & 0 & 1 & \\
\end{array}
\quad
\begin{array}{c|ccc}
0 & 0 & & \\
\frac{1}{2} & 0 & \frac{1}{2} & \\
\hline
1 & 0 & 1 & \\
\end{array}
$$
Examples: second-order IMEX

- Heun + Crank–Nicolson: efficiency ratio is $\frac{1}{2}$

\[
\begin{array}{c|cc}
0 & 0 & 0 \\
1 & 1 & 0 \\
1 & \frac{1}{2} & \frac{1}{2} \\
\end{array}
\quad
\begin{array}{c|cc}
0 & 0 & 0 \\
1 & \frac{1}{2} & \frac{1}{2} \\
1 & \frac{1}{2} & \frac{1}{2} \\
\end{array}
\]

- Explicit + implicit midpoint rules: efficiency ratio is 1

\[
\begin{array}{c|cc}
0 & 0 & 0 \\
\frac{1}{2} & \frac{1}{2} & 0 \\
1 & 0 & 1 \\
\end{array}
\quad
\begin{array}{c|cc}
0 & 0 & 0 \\
\frac{1}{2} & 0 & \frac{1}{2} \\
1 & 0 & 1 \\
\end{array}
\]

- Notice that Strang’s splitting can be rewritten as a four-stage IMEX scheme (efficiency ratio is 1)
Three-stage, third-order method [Nørsett 74, Crouzeix 75]

\( \gamma := \frac{1}{2} + \frac{1}{2\sqrt{3}} \approx 0.78867 \)

\[
\begin{array}{ccc|ccc}
0 & 0 & 0 & 0 & 0 & 0 \\
\gamma & \gamma & 0 & 0 & \gamma & \gamma \\
1 - \gamma & \gamma - 1 & 2 - 2\gamma & 0 & 1 - \gamma & 0 \\
1 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & \frac{1}{2} \frac{1}{2}
\end{array}
\]

Implicit method is A-stable, but efficiency ratio is only \( \frac{1}{3} \gamma \approx 0.26 \)
Examples: third-order IMEX (1/2)

- Three-stage, third-order method [Nørsett 74, Crouzeix 75]
  \[ \gamma := \frac{1}{2} + \frac{1}{2\sqrt{3}} \approx 0.78867 \]

\[
\begin{array}{ccc|ccc}
0 & 0 & 0 & 0 & 0 & 0 \\
\gamma & \gamma - 1 & 2 - 2\gamma & 0 & \gamma & 0 \\
1 - \gamma & 0 & 0 & 1 - \gamma & 0 & 1 - 2\gamma \\
1 & 0 & 1 \frac{1}{2} & 0 & 1 \frac{1}{2} & 1 \frac{1}{2} \\
\end{array}
\]

- Implicit method is A-stable, but efficiency ratio is only \( \frac{1}{3} \gamma \approx 0.26 \)

- New scheme with optimal efficiency 1 [AE & JLG 22]

\[
\begin{array}{ccc|ccc}
0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{3} & \frac{1}{3} & 0 & \frac{1}{3} - \gamma & \gamma & 0 \\
\frac{2}{3} & 0 & \frac{2}{3} & 0 & \frac{2}{3} & \frac{2}{3} - 2\gamma \\
1 & 0 \frac{1}{4} & 0 & \frac{3}{4} & 1 & 0 \frac{3}{4} \\
\end{array}
\]

- Implicit method has the same amplification function as above (and hence is A-stable)
Examples: third-order IMEX (2/2)

- Novel four-stage, third-order IMEX scheme with optimal efficiency 1 and implicit method is L-stable

- Explicit scheme is ERK(4,3;1) (already considered!)

\[
\begin{array}{c|ccc}
0 & 0 \\
\frac{1}{4} & \frac{1}{4} & 0 \\
\frac{1}{2} & 0 & \frac{1}{2} & 0 \\
\frac{3}{4} & 0 & \frac{1}{4} & \frac{1}{2} & 0 \\
1 & 0 & \frac{2}{3} & -\frac{1}{3} & \frac{2}{3}
\end{array}
\]

- Implicit scheme as follows:

\[
\begin{array}{c|cccc}
0 & 0 & 0 & \frac{1}{4} & 0.1858665215084591 & 0.4358665215084591 \\
\frac{1}{4} & -0.1858665215084591 & 0.4358665215084591 & 0.4367256409878701 & 0.5008591194794110 & 0.4358665215084591 \\
\frac{1}{2} & -0.4367256409878701 & 0.5008591194794110 & 0.4358665215084591 & 0.4136426175496265 & 0.4358665215084591 \\
\frac{3}{4} & -0.0423391342724147 & 0.7701152303135821 & -0.4136426175496265 & 0.4358665215084591 \\
1 & 0 & \frac{2}{3} & -\frac{1}{3} & \frac{2}{3}
\end{array}
\]
Examples: fourth-order IMEX

- Five- and six-stage schemes reviewed in [Carpenter & Kennedy 19]

- Novel five-stage scheme devised in [AE & JLG 22]
  - optimal efficiency 1
  - implicit scheme is singly diagonal and L-stable

- Novel six-stage scheme devised in [AE & JLG 22] with similar properties
  - the scheme is of linear order 5
Numerical tests on IDP-IMEX in progress
- Numerical tests on IDP-IMEX in progress
- All methods deliver expected results on stiff ODEs (Kaps, Van der Pol)
Numerical tests on IDP-IMEX in progress
All methods deliver expected results on stiff ODEs (Kaps, Van der Pol)
All methods deliver expected results on viscous Burgers’ equation
Numerical tests on IDP-IMEX in progress

All methods deliver expected results on stiff ODEs (Kaps, Van der Pol)

All methods deliver expected results on viscous Burgers’ equation

Forthcoming: reactive transport, Navier–Stokes, gray radiation hydrodynamics
Every ERK and IMEX method can be made IDP
• Every ERK and IMEX method can be made IDP

• **Recent Finite Element book(s) (Springer, TAM vols. 72-74, 2021)**
  with J.-L. Guermond, 83 chapters of 12/14 pages plus about 500 exercises
  chaps. 79-83 in vol. III discuss $C^0$-FEM for hyperbolic problems
• Every ERK and IMEX method can be made IDP

• **Recent Finite Element book(s)** *(Springer, TAM vols. 72-74, 2021)*
  
  with J.-L. Guermond, 83 chapters of 12/14 pages plus about 500 exercises
  
  chaps. 79-83 in vol. III discuss $C^0$-FEM for hyperbolic problems

---

Thank you for your attention!