ADER and DeC: arbitrarily high order (explicit) methods for PDEs and ODEs

Davide Torlo

*MathLab, Mathematics Area, SISSA International School for Advanced Studies, Trieste, Italy
davidetorlo.it

Based on: Han Veiga, M., Öffner, P. & Torlo, D. *DeC and ADER: Similarities, Differences and a Unified Framework.* J Sci Comput 87, 2 (2021).
https://doi.org/10.1007/s10915-020-01397-5

Outline

Outline

Motivation: high order accurate explicit method

Methods used to solve a hyperbolic PDE system for $u : \mathbb{R}^+ \times \Omega \to \mathbb{R}^D$

$$\partial_t u + \nabla_{\mathbf{x}} \mathcal{F}(u) = 0. \tag{1}$$

Or ODE system for $\boldsymbol{u} : \mathbb{R}^+ \to \mathbb{R}^S$

$$\partial_t \boldsymbol{u} = F(\boldsymbol{u}). \tag{2}$$

Applications:

- Fluids/transport
- Chemical/biological processes

How?

- Arbitrarily high order accurate
-

Motivation: high order accurate explicit method

Methods used to solve a hyperbolic PDE system for $u : \mathbb{R}^+ \times \Omega \to \mathbb{R}^D$

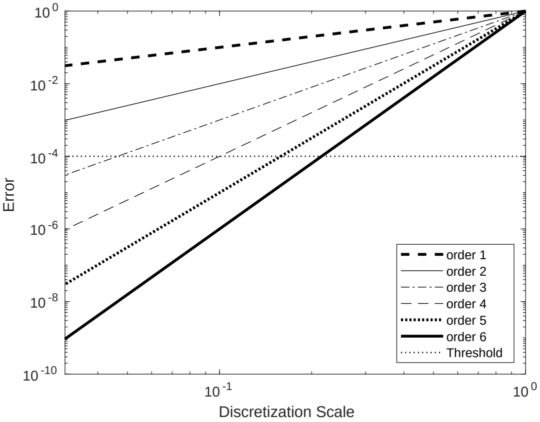
(1)

Or ODE system for $\boldsymbol{u}$:

(2)

Applications:

- Fluids/transport
- Chemical/biologica



How?

- Arbitrarily high orc
-

Motivation: high order accurate explicit method

Methods used to solve a hyperbolic PDE system for $u : \mathbb{R}^+ \times \Omega \to \mathbb{R}^D$

$$\partial_t u + \nabla_{\mathbf{x}} \mathcal{F}(u) = 0. \tag{1}$$

Or ODE system for $\boldsymbol{u} : \mathbb{R}^+ \to \mathbb{R}^S$

$$\partial_t \boldsymbol{u} = F(\boldsymbol{u}). \tag{2}$$

Applications:

- Fluids/transport
- Chemical/biological processes

How?

- Arbitrarily high order accurate
- Explicit (if nonstiff problem)

Classical time integration: Runge–Kutta

$$\boldsymbol{u}^{(1)} := \boldsymbol{u}^n, \tag{3}$$

$$\boldsymbol{u}^{(k)} := \boldsymbol{u}^n + \sum_{s=1}^{K} a_{ks} F\left(t^n + c_s \Delta t, \boldsymbol{u}^{(s)}\right), \quad \text{for } k = 2, \dots, K, \tag{4}$$

$$\boldsymbol{u}^{n+1} := \boldsymbol{u}^n + \sum_{s=1}^{K} b_s F\left(t^n + c_s \Delta t, \boldsymbol{u}^{(s)}\right). \tag{5}$$

Classical time integration: Explicit Runge–Kutta

$$\boldsymbol{u}^{(k)} := \boldsymbol{u}^n + \sum_{s=1}^{k-1} a_{ks} F\left(t^n + c_s \Delta t, \boldsymbol{u}^{(s)}\right), \quad \text{for } k = 2, \dots, K.$$

- Easy to solve
- High orders involved:
  - Order conditions: system of many equations
  - Stages $K \geq d$ order of accuracy (e.g. RK44, RK65)

Classical time integration: Implicit Runge–Kutta

$$\boldsymbol{u}^{(k)} := \boldsymbol{u}^n + \sum_{s=1}^{K} a_{ks} F\left(t^n + c_s \Delta t, \boldsymbol{u}^{(s)}\right), \quad \text{for } k = 2, \dots, K.$$

- More complicated to solve for nonlinear systems
- High orders easily done:
  - Take a high order quadrature rule on $[t^n, t^{n+1}]$
  - Compute the coefficients accordingly, see Gauss–Legendre or Gauss–Lobatto polynomials
  - Order up to $d = 2K$

ADER and DeC

Two iterative explicit arbitrarily high order accurate methods.

- ADER[1] for hyperbolic PDE, after a first analytic more complicated approach.
- Deferred Correction (DeC): introduced for explicit ODE[2], extended to implicit ODE[3] and to hyperbolic PDE[4].

[1] M. Dumbser, D. S. Balsara, E. F. Toro, and C.-D. Munz. A unified framework for the construction of one-step finite volume and discontinuous galerkin schemes on unstructured meshes. Journal of Computational Physics, 227(18):8209–8253, 2008.

[2] A. Dutt, L. Greengard, and V. Rokhlin. Spectral Deferred Correction Methods for Ordinary Differential Equations. BIT Numerical Mathematics, 40(2):241–266, 2000.

[3] M. L. Minion. Semi-implicit spectral deferred correction methods for ordinary differential equations. Commun. Math. Sci., 1(3):471–500, 09 2003.

[4] R. Abgrall. High order schemes for hyperbolic problems using globally continuous approximation and avoiding mass matrices. Journal of Scientific Computing, 73(2):461–494, Dec 2017.

Outline

DeC high order time discretization: $\mathcal{L}^2$

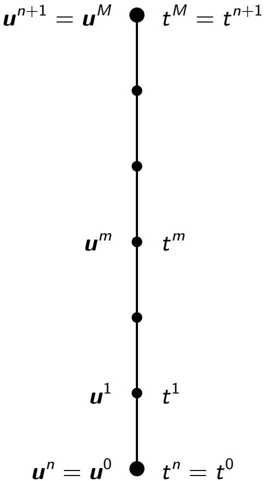
High order in time: we discretize our variable on $[t^n, t^{n+1}]$ in M substeps ($\boldsymbol{u}^m$).

$$\partial_t \boldsymbol{u} = F(\boldsymbol{u}(t)).$$

Thanks to Picard–Lindelöf theorem, we can rewrite

$$\boldsymbol{u}^m = \boldsymbol{u}^0 + \int_{t^0}^{t^m} F(\boldsymbol{u}(t))dt.$$

and if we want to reach order $r+1$ we need $M = r$.

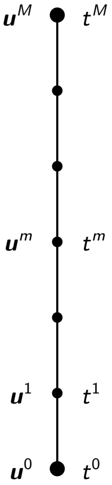


DeC high order time discretization: $\mathcal{L}^2$

More precisely, for each σ we want to solve $\mathcal{L}^2(\boldsymbol{u}^{n,0},\dots,\boldsymbol{u}^{n,M})=0$, where

$$\mathcal{L}^2(\boldsymbol{u}^0,\dots,\boldsymbol{u}^M)=\begin{pmatrix} \boldsymbol{u}^M-\boldsymbol{u}^0+\sum_{r=0}^M\int_{t^0}^{t^M}F(\boldsymbol{u}^r)\varphi_r(s)\mathrm{d}s \\ \vdots \\ \boldsymbol{u}^1-\boldsymbol{u}^0+\sum_{r=0}^M\int_{t^0}^{t^1}F(\boldsymbol{u}^r)\varphi_r(s)\mathrm{d}s \end{pmatrix}$$

- $\mathcal{L}^2=0$ is a system of $M\times S$ coupled (non)linear equations
- $\mathcal{L}^2$ is an implicit method (collocation method: Gauss, LobattoIIIA)
- Not easy to solve directly $\mathcal{L}^2(\underline{\boldsymbol{u}}^*)=0$
- High order (equispaced $M+1$, Gauss–Lobatto $2M$), depending on points distribution

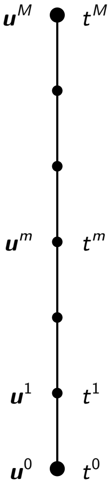


DeC high order time discretization: $\mathcal{L}^2$

More precisely, for each σ we want to solve $\mathcal{L}^2(\boldsymbol{u}^{n,0},\dots,\boldsymbol{u}^{n,M})=0$, where

$$\mathcal{L}^2(\boldsymbol{u}^0,\dots,\boldsymbol{u}^M)=\begin{pmatrix} \boldsymbol{u}^M-\boldsymbol{u}^0+\Delta t\sum_{r=0}^M \theta_r^M F(\boldsymbol{u}^r)\\ \vdots \\ \boldsymbol{u}^1-\boldsymbol{u}^0+\Delta t\sum_{r=0}^M \theta_r^1 F(\boldsymbol{u}^r)\end{pmatrix}$$

- $\mathcal{L}^2=0$ is a system of $M\times S$ coupled (non)linear equations
- $\mathcal{L}^2$ is an implicit method (collocation method: Gauss, LobattoIIIA)
- Not easy to solve directly $\mathcal{L}^2(\underline{\boldsymbol{u}}^*)=0$
- High order (equispaced $M+1$, Gauss–Lobatto $2M$), depending on points distribution

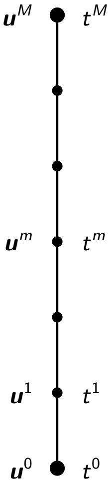


DeC low order time discretization: $\mathcal{L}^1$

Instead of solving the implicit system directly (difficult), we introduce a first order scheme $\mathcal{L}^1(\boldsymbol{u}^{n,0},\dots,\boldsymbol{u}^{n,M})$:

$$\mathcal{L}^1(\boldsymbol{u}^0,\dots,\boldsymbol{u}^M)=\begin{pmatrix}\boldsymbol{u}^M-\boldsymbol{u}^0+\Delta t\beta^M F(\boldsymbol{u}^0)\\ \vdots \\ \boldsymbol{u}^1-\boldsymbol{u}^0+\Delta t\beta^1 F(\boldsymbol{u}^0)\end{pmatrix}$$

- First order approximation
- Explicit Euler
- Easy to solve $\mathcal{L}^1(\underline{\boldsymbol{u}})=0$

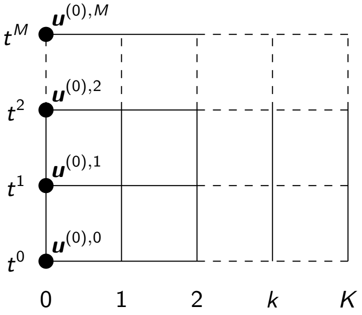


Deferred Correction[5]

How to combine two methods keeping the accuracy of the second and the stability and simplicity of the first one?

- $\mathcal{L}^1(\underline{\boldsymbol{u}})=0$, first order accuracy, easily invertible.
- $\mathcal{L}^2(\underline{\boldsymbol{u}})=0$, high order $M+1$.

$$\boldsymbol{u}^{0,(k)} := \boldsymbol{u}(t^n), \quad k=0,\dots,K,$$
$$\boldsymbol{u}^{m,(0)} := \boldsymbol{u}(t^n), \quad m=1,\dots,M$$
$$\mathcal{L}^1(\underline{\boldsymbol{u}}^{(k)}) = \mathcal{L}^1(\underline{\boldsymbol{u}}^{(k-1)}) - \mathcal{L}^2(\underline{\boldsymbol{u}}^{(k-1)}) \text{ with } k=1,\dots,K.$$



Theorem (Convergence DeC)

- $\mathcal{L}^2(\underline{\boldsymbol{u}}^*)=0$
- *If $\mathcal{L}^1$ coercive with constant C_1*
- *If $\mathcal{L}^1-\mathcal{L}^2$ Lipschitz with constant $C_2\Delta t$*

Then $\|\underline{\boldsymbol{u}}^{(K)}-\underline{\boldsymbol{u}}^*\| \le C(\Delta t)^K$

[5] A. Dutt, L. Greengard, and V. Rokhlin. BIT Numerical Mathematics, 40(2):241–266, 2000.

Deferred Correction[5]

How to combine two methods keeping the accuracy of the second and the stability and simplicity of the first one?

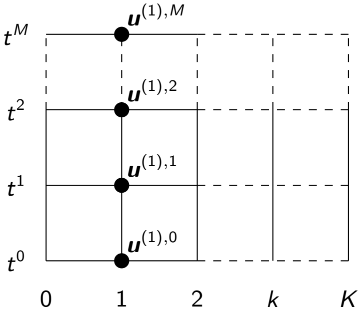
$$\boldsymbol{u}^{0,(k)} := \boldsymbol{u}(t^n), \quad k = 0, \dots, K,$$
$$\boldsymbol{u}^{m,(0)} := \boldsymbol{u}(t^n), \quad m = 1, \dots, M$$
$$\mathcal{L}^1(\underline{\boldsymbol{u}}^{(k)}) = \mathcal{L}^1(\underline{\boldsymbol{u}}^{(k-1)}) - \mathcal{L}^2(\underline{\boldsymbol{u}}^{(k-1)}) \text{ with } k = 1, \dots, K.$$

Theorem (Convergence DeC)

- $\mathcal{L}^2(\underline{\boldsymbol{u}}^*) = 0$
- *If $\mathcal{L}^1$ coercive with constant C_1*
- *If $\mathcal{L}^1 - \mathcal{L}^2$ Lipschitz with constant $C_2 \Delta t$*

Then $\|\underline{\boldsymbol{u}}^{(K)} - \underline{\boldsymbol{u}}^*\| \leq C(\Delta t)^K$

- $\mathcal{L}^1(\underline{\boldsymbol{u}}) = 0$, first order accuracy, easily invertible.
- $\mathcal{L}^2(\underline{\boldsymbol{u}}) = 0$, high order $M+1$.



[5] A. Dutt, L. Greengard, and V. Rokhlin. BIT Numerical Mathematics, 40(2):241–266, 2000.

Deferred Correction[5]

How to combine two methods keeping the accuracy of the second and the stability and simplicity of the first one?

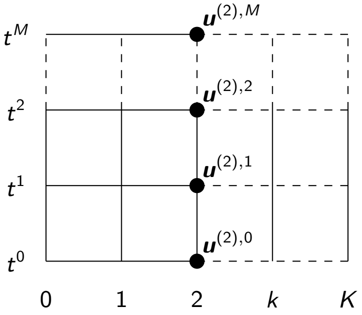
- $\mathcal{L}^1(\underline{\boldsymbol{u}}) = 0$, first order accuracy, easily invertible.
- $\mathcal{L}^2(\underline{\boldsymbol{u}}) = 0$, high order $M+1$.

$$\boldsymbol{u}^{0,(k)} := \boldsymbol{u}(t^n), \quad k = 0, \dots, K,$$
$$\boldsymbol{u}^{m,(0)} := \boldsymbol{u}(t^n), \quad m = 1, \dots, M$$
$$\mathcal{L}^1(\underline{\boldsymbol{u}}^{(k)}) = \mathcal{L}^1(\underline{\boldsymbol{u}}^{(k-1)}) - \mathcal{L}^2(\underline{\boldsymbol{u}}^{(k-1)}) \text{ with } k = 1, \dots, K.$$

Theorem (Convergence DeC)

- $\mathcal{L}^2(\underline{\boldsymbol{u}}^*) = 0$
- *If $\mathcal{L}^1$ coercive with constant C_1*
- *If $\mathcal{L}^1 - \mathcal{L}^2$ Lipschitz with constant $C_2\Delta t$*

Then $\|\underline{\boldsymbol{u}}^{(K)} - \underline{\boldsymbol{u}}^*\| \le C(\Delta t)^K$



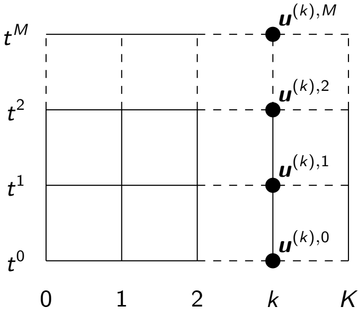
[5] A. Dutt, L. Greengard, and V. Rokhlin. BIT Numerical Mathematics, 40(2):241–266, 2000.

Deferred Correction[5]

How to combine two methods keeping the accuracy of the second and the stability and simplicity of the first one?

- $\mathcal{L}^1(\underline{\boldsymbol{u}})=0$, first order accuracy, easily invertible.
- $\mathcal{L}^2(\underline{\boldsymbol{u}})=0$, high order $M+1$.

$$\boldsymbol{u}^{0,(k)} := \boldsymbol{u}(t^n), \quad k=0,\dots,K,$$
$$\boldsymbol{u}^{m,(0)} := \boldsymbol{u}(t^n), \quad m=1,\dots,M$$
$$\mathcal{L}^1(\underline{\boldsymbol{u}}^{(k)}) = \mathcal{L}^1(\underline{\boldsymbol{u}}^{(k-1)}) - \mathcal{L}^2(\underline{\boldsymbol{u}}^{(k-1)}) \text{ with } k=1,\dots,K.$$



Theorem (Convergence DeC)

- $\mathcal{L}^2(\underline{\boldsymbol{u}}^*)=0$
- *If $\mathcal{L}^1$ coercive with constant C_1*
- *If $\mathcal{L}^1-\mathcal{L}^2$ Lipschitz with constant $C_2\Delta t$*

Then $\|\underline{\boldsymbol{u}}^{(K)}-\underline{\boldsymbol{u}}^*\| \leq C(\Delta t)^K$

[5] A. Dutt, L. Greengard, and V. Rokhlin. BIT Numerical Mathematics, 40(2):241–266, 2000.

Deferred Correction[5]

How to combine two methods keeping the accuracy of the second and the stability and simplicity of the first one?

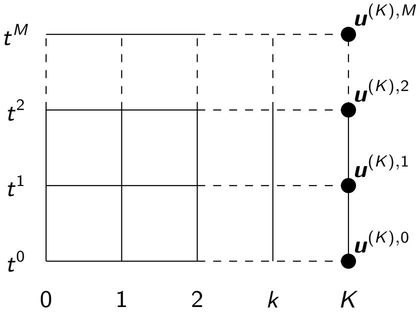
- $\mathcal{L}^1(\underline{\boldsymbol{u}}) = 0$, first order accuracy, easily invertible.
- $\mathcal{L}^2(\underline{\boldsymbol{u}}) = 0$, high order $M+1$.

$$\boldsymbol{u}^{0,(k)} := \boldsymbol{u}(t^n), \quad k = 0, \dots, K,$$
$$\boldsymbol{u}^{m,(0)} := \boldsymbol{u}(t^n), \quad m = 1, \dots, M$$
$$\mathcal{L}^1(\underline{\boldsymbol{u}}^{(k)}) = \mathcal{L}^1(\underline{\boldsymbol{u}}^{(k-1)}) - \mathcal{L}^2(\underline{\boldsymbol{u}}^{(k-1)}) \text{ with } k = 1, \dots, K.$$

Theorem (Convergence DeC)

- $\mathcal{L}^2(\underline{\boldsymbol{u}}^*) = 0$
- *If $\mathcal{L}^1$ coercive with constant C_1*
- *If $\mathcal{L}^1 - \mathcal{L}^2$ Lipschitz with constant $C_2 \Delta t$*

Then $\|\underline{\boldsymbol{u}}^{(K)} - \underline{\boldsymbol{u}}^*\| \leq C(\Delta t)^K$



[5] A. Dutt, L. Greengard, and V. Rokhlin. BIT Numerical Mathematics, 40(2):241–266, 2000.

DeC – Proof

Proof.

Let f^* be the solution of $\mathcal{L}^2(\underline{\boldsymbol{u}}^*) = 0$. We know that $\mathcal{L}^1(\underline{\boldsymbol{u}}^*) = \mathcal{L}^1(\underline{\boldsymbol{u}}^*) - \mathcal{L}^2(\underline{\boldsymbol{u}}^*)$, so that

$\square$

DeC – Proof

Proof.

Let f^* be the solution of $\mathcal{L}^2(\underline{\boldsymbol{u}}^*)=0$. We know that $\mathcal{L}^1(\underline{\boldsymbol{u}}^*)=\mathcal{L}^1(\underline{\boldsymbol{u}}^*)-\mathcal{L}^2(\underline{\boldsymbol{u}}^*)$, so that

$$\mathcal{L}^1(\underline{\boldsymbol{u}}^{(k+1)})-\mathcal{L}^1(\underline{\boldsymbol{u}}^*)=\left(\mathcal{L}^1(\underline{\boldsymbol{u}}^{(k)})-\mathcal{L}^2(\underline{\boldsymbol{u}}^{(k)})\right)-\left(\mathcal{L}^1(\underline{\boldsymbol{u}}^*)-\mathcal{L}^2(\underline{\boldsymbol{u}}^*)\right)$$

$$\begin{aligned}
{\color{red}C_1}||\underline{\boldsymbol{u}}^{(k+1)}-\underline{\boldsymbol{u}}^*|| \leq & ||\mathcal{L}^1(\underline{\boldsymbol{u}}^{(k+1)})-\mathcal{L}^1(\underline{\boldsymbol{u}}^*)||=\\
= & ||\mathcal{L}^1(\underline{\boldsymbol{u}}^{(k)})-\mathcal{L}^2(\underline{\boldsymbol{u}}^{(k)})-(\mathcal{L}^1(\underline{\boldsymbol{u}}^*)-\mathcal{L}^2(\underline{\boldsymbol{u}}^*))|| \leq\\
\leq & {\color{red}C_2\Delta}||\underline{\boldsymbol{u}}^{(k)}-\underline{\boldsymbol{u}}^*||.
\end{aligned}$$

$$||\underline{\boldsymbol{u}}^{(k+1)}-\underline{\boldsymbol{u}}^*|| \leq \left(\frac{C_2}{C_1}\Delta\right)||\underline{\boldsymbol{u}}^{(k)}-\underline{\boldsymbol{u}}^*|| \leq \left(\frac{C_2}{C_1}\Delta\right)^{k+1}||\underline{\boldsymbol{u}}^{(0)}-\underline{\boldsymbol{u}}^*||.$$

After K iteration we have an error at most of $\left(\frac{C_2}{C_1}\Delta\right)^K||\underline{\boldsymbol{u}}^{(0)}-\underline{\boldsymbol{u}}^*||$. □

DeC: Second order example

DeC: Second order example

DeC: Second order example

DeC: Second order example

Simplification of DeC for ODE

In practice

$$\mathcal{L}^1(\underline{\boldsymbol{u}}^{(k)}) = \mathcal{L}^1(\underline{\boldsymbol{u}}^{(k-1)}) - \mathcal{L}^2(\underline{\boldsymbol{u}}^{(k-1)}), \qquad k = 1, \dots, K,$$

For $m = 1, \dots, M$

$$\begin{aligned}&\boldsymbol{u}^{(k),m} - \boldsymbol{u}^0 - \beta^m \Delta t F(\boldsymbol{u}^0) - \boldsymbol{u}^{(k-1),m} + \boldsymbol{u}^0 + \beta^m \Delta t F(\boldsymbol{u}^0)\\ &+ \boldsymbol{u}^{(k-1),m} - \boldsymbol{u}^0 - \Delta t \sum_{r=0}^{M} \theta_r^m F(\boldsymbol{u}^{(k-1),r}) = 0\end{aligned}$$

Simplification of DeC for ODE

In practice

$$\mathcal{L}^1(\underline{\boldsymbol{u}}^{(k)}) = \mathcal{L}^1(\underline{\boldsymbol{u}}^{(k-1)}) - \mathcal{L}^2(\underline{\boldsymbol{u}}^{(k-1)}), \qquad k = 1, \dots, K,$$

For $m = 1, \dots, M$

$$\boldsymbol{u}^{(k),m} \cancel{- \boldsymbol{u}^0 - \beta^m \Delta t F(\boldsymbol{u}^0)} - \boldsymbol{u}^{(k-1),m} \cancel{+ \boldsymbol{u}^0 + \beta^m \Delta t F(\boldsymbol{u}^0)}$$

$$+ \boldsymbol{u}^{(k-1),m} - \boldsymbol{u}^0 - \Delta t \sum_{r=0}^{M} \theta_r^m F(\boldsymbol{u}^{(k-1),r}) = 0$$

Simplification of DeC for ODE

In practice

$$\mathcal{L}^1(\underline{\boldsymbol{u}}^{(k)}) = \mathcal{L}^1(\underline{\boldsymbol{u}}^{(k-1)}) - \mathcal{L}^2(\underline{\boldsymbol{u}}^{(k-1)}), \qquad k = 1, \dots, K,$$

For $m = 1, \dots, M$

$$\boldsymbol{u}^{(k),m} - \cancel{\boldsymbol{u}^0 - \beta^m \Delta t F(\boldsymbol{u}^0)} - \cancel{\boldsymbol{u}^{(k-1),m}} + \cancel{\boldsymbol{u}^0 + \beta^m \Delta t F(\boldsymbol{u}^0)}$$

$$+ \cancel{\boldsymbol{u}^{(k-1),m}} - \boldsymbol{u}^0 - \Delta t \sum_{r=0}^{M} \theta_r^m F(\boldsymbol{u}^{(k-1),r}) = 0$$

Simplification of DeC for ODE

In practice

$$\mathcal{L}^1(\underline{\boldsymbol{u}}^{(k)}) = \mathcal{L}^1(\underline{\boldsymbol{u}}^{(k-1)}) - \mathcal{L}^2(\underline{\boldsymbol{u}}^{(k-1)}), \qquad k = 1, \dots, K,$$

For $m = 1, \dots, M$

$$\boldsymbol{u}^{(k),m} - \cancel{\boldsymbol{u}^0 - \beta^m \Delta t F(\boldsymbol{u}^0)} - \cancel{\boldsymbol{u}^{(k-1),m}} + \cancel{\boldsymbol{u}^0 + \beta^m \Delta t F(\boldsymbol{u}^0)}$$

$$+ \cancel{\boldsymbol{u}^{(k-1),m}} - \boldsymbol{u}^0 - \Delta t \sum_{r=0}^{M} \theta_r^m F(\boldsymbol{u}^{(k-1),r}) = 0$$

$$\boldsymbol{u}^{(k),m} - \boldsymbol{u}^0 - \Delta t \sum_{r=0}^{M} \theta_r^m F(\boldsymbol{u}^{(k-1),r}) = 0.$$

DeC and residual distribution

Deferred Correction + Residual distribution

- Residual distribution (FV $\Rightarrow$ FE) $\Rightarrow$ High order in space
- Prediction/correction/iterations $\Rightarrow$ High order in time
- Subtimesteps $\Rightarrow$ High order in time

$$U_\xi^{m,(k+1)} = U_\xi^{m,(k)} - |C_p|^{-1} \sum_{\mathrm{E}|\xi\in\mathrm{E}} \left(\int_{\mathrm{E}} \Phi_\xi \left(U^{m,(k)} - U^{n,0} \right) \mathrm{d}\mathbf{x} + \Delta t \sum_{r=0}^{M} \theta_r^m \mathcal{R}_\xi^{\mathrm{E}}(U^{r,(k)}) \right),$$

with

$$\sum_{\xi\in\mathrm{E}} \mathcal{R}_\xi^{\mathrm{E}}(u) = \int_{\mathrm{E}} \nabla_{\mathbf{x}} F(u) \mathrm{d}\mathbf{x}.$$

- The $\mathcal{L}^2$ operator contains also the complications of the spatial discretization (e.g. mass matrix)
- $\mathcal{L}^1$ operator further simplified up to a first order approximation (e.g. **mass lumping**)

$\mathcal{L}^1$ with mass lumping

Implicit simple DeC (Rosenbrock)

Define $\mathcal{L}^1$ as

$$\mathcal{L}^1(\boldsymbol{u}^0, \dots, \boldsymbol{u}^M) = \begin{pmatrix} \boldsymbol{u}^M - \boldsymbol{u}^0 - \Delta t \beta^M F(\boldsymbol{u}^0) \\ \vdots \\ \boldsymbol{u}^1 - \boldsymbol{u}^0 - \Delta t \beta^1 F(\boldsymbol{u}^0) \end{pmatrix}$$

Implicit simple DeC (Rosenbrock)

Define $\mathcal{L}^1$ as

$$\mathcal{L}^1(\boldsymbol{u}^0,\dots,\boldsymbol{u}^M) = \begin{pmatrix} \boldsymbol{u}^M - \boldsymbol{u}^0 - \Delta t \beta^M \left(F(\boldsymbol{u}^0) + \partial_{\boldsymbol{u}} F(\boldsymbol{u}^0)(\boldsymbol{u}^M - \boldsymbol{u}^0)\right) \\ \vdots \\ \boldsymbol{u}^1 - \boldsymbol{u}^0 - \Delta t \beta^1 \left(F(\boldsymbol{u}^0) + \partial_{\boldsymbol{u}} F(\boldsymbol{u}^0)(\boldsymbol{u}^1 - \boldsymbol{u}^0)\right) \end{pmatrix}$$

$$= \begin{pmatrix} \boldsymbol{u}^M - \boldsymbol{u}^0 - \Delta t \beta^M \partial_{\boldsymbol{u}} F(\boldsymbol{u}^0)\boldsymbol{u}^M \\ \vdots \\ \boldsymbol{u}^1 - \boldsymbol{u}^0 - \Delta t \beta^1 \partial_{\boldsymbol{u}} F(\boldsymbol{u}^0)\boldsymbol{u}^1 \end{pmatrix}$$

Implicit simple DeC (Rosenbrock)

$$\mathcal{L}^{1,m}(\boldsymbol{u}^0,\dots,\boldsymbol{u}^M) = \boldsymbol{u}^m - \boldsymbol{u}^0 - \Delta t \beta^m \partial_{\boldsymbol{u}} F(\boldsymbol{u}^0)\boldsymbol{u}^m$$
$$\mathcal{L}^{2,m}(\boldsymbol{u}^0,\dots,\boldsymbol{u}^M) = \boldsymbol{u}^m - \boldsymbol{u}^0 - \Delta t \sum_r \theta_r^m F(\boldsymbol{u}^r)$$

Implicit simple DeC (Rosenbrock)

DeC as RK

$$\boldsymbol{u}^{(k),m} - \boldsymbol{u}^0 - \Delta t \sum_{r=0}^{M} \theta_r^m F(\boldsymbol{u}^{(k-1),r}) = 0$$

DeC as RK

DeC as RK

We can write DeC as RK defining $\underline{\theta}_0 = \{\theta_0^m\}_{m=1}^M$, $\underline{\theta}^M = \theta_r^M$ with $r \in 1, \dots, M$, denoting the vector $\underline{\theta}_r^{M,T} = (\theta_1^M, \dots, \theta_M^M)$. The Butcher tableau for an arbitrarily high order DeC approach is given by:

$$
\begin{array}{c|ccccccc}
0 & 0 & & & & & & \\
\underline{\beta} & \underline{\beta} & & & & & & \\
\underline{\beta} & \underline{\theta}_0 & \underline{\underline{\tilde{\theta}}} & & & & & \\
\vdots & \underline{\theta}_0 & \underline{\underline{0}} & \underline{\underline{\tilde{\theta}}} & & & & \\
\vdots & \underline{\theta}_0 & \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{\tilde{\theta}}} & & & \\
\vdots & \vdots & \vdots & \vdots & \ddots & \ddots & & \\
\underline{\beta} & \underline{\theta}_0 & \underline{\underline{0}} & \dots & \dots & \underline{\underline{0}} & \underline{\underline{\tilde{\theta}}} & \\
\hline
 & \theta_0^M & \underline{0}^T & \dots & & \dots & \underline{0}^T & \underline{\theta}_r^{M,T}
\end{array}
. \quad (6)
$$

Stability of (explicit) DeC

Idea: study the RK version!

$$u' = \lambda u \qquad \Re(\lambda) < 0. \tag{7}$$

$$u_{n+1} = R(\lambda \Delta t) u_n, \qquad R(z) = 1 + z b^T (I - zA)^{-1} \mathbf{1}, \qquad z = \lambda \Delta t \tag{8}$$

Goal: find $z \in \mathbb{C}$ such that $|R(z)| < 1$.
Recall: stability function for explicit RK methods is a polynomial, indeed the inverse of $(I - zA)$ can be written in Taylor expansion as

$$(I - zA)^{-1} = \sum_{r=0}^{\infty} z^r A^s = I + zA + z^2 A^2 + \dots, \tag{9}$$

and, since A is strictly lower triangular, it is nilpotent. Hence, $R(z)$ is a polynomial in z with degree at most equal to S.

Stability of (explicit) DeC

Theorem

If the RK method is of order P, then

$$R(z) = 1 + z + \frac{z^2}{2!} + \dots + \frac{z^P}{P!} + O(z^{P+1}). \tag{10}$$

The first $P+1$ terms of the stability functions $R(\cdot)$ for explicit DeCs of order P are known.

Theorem

The stability function of any explicit DeC of order P (with P iterations) is

$$R(z) = \sum_{r=0}^{P} \frac{z^r}{r!} = 1 + z + \frac{z^2}{2!} + \dots + \frac{z^P}{P!} \tag{11}$$

and does not depend on the distribution of the subtimenodes.

Stability of (explicit) DeC

Proof (1/3)

$$A = \begin{pmatrix} 0 & 0 & 0 & \dots & 0 & 0 \\ \star & 0 & 0 & \dots & 0 & 0 \\ \star & \star & 0 & \dots & 0 & 0 \\ \star & 0 & \star & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \star & 0 & 0 & \cdots & \star & 0 \end{pmatrix},$$

Block structure of the matrix A
$\star$ are some non-zero block matrices and the 0 are some zero block matrices.
The number of blocks in each line and row of these matrices is P, the order of the scheme.

Stability of (explicit) DeC

Proof (2/3)

By induction, A^k has zeros in the upper triangular part, in the main block diagonal, and in all the $k-1$ block diagonals below the main diagonal, i.e.,

$$(A^k)_{i,j} = 0 \quad , \text{if } i < j+k,$$

where the indexes here refer to the blocks. Indeed, it is true that $A_{i,j} = 0$ if $i < j+1$. Now, let us consider the entry $(A^{k+1})_{i,j}$ with $i < j+k+1$, i.e., $i-k < j+1$. It is defined as

$$(A^{k+1})_{i,j} = \sum_w (A^k)_{i,w} A_{w,j}. \tag{12}$$

Now, we can prove that all the terms of the sum are 0. Let $w < j+1$, then $A_{w,j} = 0$ because of the structure of A; while, if $w \geq j+1 > i-k$, we have that $i < w+k$, so $(A^k)_{i,w} = 0$ by induction.

Proof (3/3)

In particular, this means that $A^P = \underline{\underline{0}}$, because i is always smaller than $j+P$ as P is the number of the block matrices that we have. Hence,

$$(I - zA)^{-1} = \sum_{r=0}^{\infty} z^r A^s = \sum_{r=0}^{P-1} z^r A^s = I + zA + z^2A^2 + \dots + z^{P-1}A^{P-1}. \tag{13}$$

Plugging this result into $R(z) = 1 + zb^T(I - zA)^{-1}\mathbf{1}$, the stability function $R(z)$ is a polynomial of degree P, the order of the scheme. All terms of order lower or equal to P must agree with the expansion of the exponential function, so it must be

$$R(z) = \sum_{r=0}^{P} \frac{z^r}{r!} = 1 + z + \frac{z^2}{2!} + \dots + \frac{z^P}{P!}. \tag{14}$$

Note: no assumption on the distribution of the subtimenodes.

CODE

- Choice of iterations (P) and order
- Choice of point distributions $t^0, \dots, t^M$
- Computation of θ
- Loop for timesteps
- Loop for correction
- Loop for subtimesteps

Outline

ADER

- Cauchy–Kovalevskaya theorem
- Modern automatic version
- Space/time DG
- Prediction/Correction
- Fixed-point iteration process

Modern approach is DG in space time for hyperbolic problem

$$\partial_t u(x,t) + \nabla \cdot F(u(x,t)) = 0,\ x \in \Omega \subset \mathbb{R}^d,\ \ t > 0. \quad (15)$$

Prediction: iterative procedure

$$\int_{T^n \times V_i} \theta_{rs}(x,t) \partial_t \theta_{pq}(x,t) z^{pq} \mathrm{d}x \mathrm{d}t + \int_{T^n \times V_i} \theta_{rs}(x,t) \nabla_{\mathbf{x}} \cdot F(\theta_{pq}(x,t) z^{pq}) \mathrm{d}x \mathrm{d}t = 0.$$

Correction step: communication between cells

$$\int_{V_i} \Phi_r \left(u(t^{n+1}) - u(t^n) \right) \mathrm{d}x + \int_{T^n \times \partial V_i} \Phi_r(x) \mathcal{G}(z^-, z^+) \cdot \mathbf{n} \, \mathrm{d}S \, \mathrm{d}t - \int_{T^n \times V_i} \nabla_{\mathbf{x}} \Phi_r \cdot F(z) \, \mathrm{d}x \, \mathrm{d}t = 0,$$

ADER: space-time discretization

Defining $\theta_{rs}(x,t) = \Phi_r(x)\phi_s(t)$ basis functions in space and time

$$\int_{T^n \times V_i} \theta_{rs}(x,t)\partial_t \theta_{pq}(x,t) u^{pq} \mathrm{d}x\mathrm{d}t + \int_{T^n \times V_i} \theta_{rs}(x,t) \nabla \cdot F(\theta_{pq}(x,t)u^{pq}) \mathrm{d}x\mathrm{d}t = 0. \tag{16}$$

ADER: space-time discretization

Defining $\theta_{rs}(x,t) = \Phi_r(x)\phi_s(t)$ basis functions in space and time

$$\int_{T^n \times V_i} \theta_{rs}(x,t)\partial_t \theta_{pq}(x,t)u^{pq}\mathrm{d}x\mathrm{d}t + \int_{T^n \times V_i} \theta_{rs}(x,t)\nabla \cdot F(\theta_{pq}(x,t)u^{pq})\mathrm{d}x\mathrm{d}t = 0. \tag{16}$$

This leads to

$$\underline{\underline{\underline{\underline{\mathrm{M}}}}}_{rspq} u^{pq} = \underline{\underline{r}}(\underline{\underline{\mathbf{u}}})_{rs}, \tag{17}$$

solved with fixed point iteration method.
$+$ Correction step where cells communication is allowed (derived from (16)).

ADER: time integration method

Simplify! Take $\boldsymbol{u}(t) = \sum_{m=0}^M \phi_m(t)\boldsymbol{u}^m = \underline{\phi}(t)^T \underline{\boldsymbol{u}}$

$$\int_{T^n} \psi(t)\partial_t \boldsymbol{u}(t)dt - \int_{T^n} \psi(t)F(\boldsymbol{u}(t))dt = 0, \quad \forall \psi : T^n = [t^n, t^{n+1}] \to \mathbb{R}.$$

$$\mathcal{L}^2(\underline{\boldsymbol{u}}) := \int_{T^n} \underline{\phi}(t)\partial_t \underline{\phi}(t)^T \underline{\boldsymbol{u}} dt - \int_{T^n} \underline{\phi}(t) F(\underline{\phi}(t)^T \underline{\boldsymbol{u}})dt = 0$$

$$\underline{\phi}(t) = (\phi_0(t), \dots, \phi_M(t))^T$$

Quadrature...

$$\mathcal{L}^2(\underline{\boldsymbol{u}}) := \underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}} - \underline{r}(\underline{\boldsymbol{u}}) = 0 \iff \underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}} = \underline{r}(\underline{\boldsymbol{u}}). \tag{18}$$

Nonlinear system of $M \times S$ equations

ADER: Mass matrix

What goes into the mass matrix? Use of the integration by parts

$$
\begin{aligned}
\mathcal{L}^2(\underline{\boldsymbol{u}}) := &\int_{T^n} \underline{\phi}(t)\partial_t\underline{\phi}(t)^T\underline{\boldsymbol{u}}dt + \int_{T^n}\underline{\phi}(t)F(\underline{\phi}(t)^T\underline{\boldsymbol{u}})dt = \\
&\underline{\phi}(t^{n+1})\underline{\phi}(t^{n+1})^T\underline{\boldsymbol{u}} - \underline{\phi}(t^n)\boldsymbol{u}^n - \int_{T^n}\partial_t\underline{\phi}(t)\underline{\phi}(t)^T\underline{\boldsymbol{u}} - \int_{T^n}\underline{\phi}(t)F(\underline{\phi}(t)^T\underline{\boldsymbol{u}})dt
\end{aligned}
$$

$$\underline{\underline{\mathrm{M}}} = \underline{\phi}(t^{n+1})\underline{\phi}(t^{n+1})^T - \int_{T^n}\partial_t\underline{\phi}(t)\underline{\phi}(t)^T$$

$$\underline{r}(\underline{\boldsymbol{u}}) = \underline{\phi}(t^n)\boldsymbol{u}^n + \int_{T^n}\underline{\phi}(t)F(\underline{\phi}(t)^T\underline{\boldsymbol{u}})dt$$

$$\underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}} = \underline{r}(\underline{\boldsymbol{u}})$$

ADER: Fixed point iteration

Iterative procedure to solve the problem for each time step

$$\underline{\boldsymbol{u}}^{(k)} = \underline{\underline{\mathrm{M}}}^{-1} \underline{r}(\underline{\boldsymbol{u}}^{(k-1)}), \quad k = 1, \dots, \text{convergence} \tag{19}$$

with $\underline{\boldsymbol{u}}^{(0)} = \boldsymbol{u}(t^n)$.
Reconstruction step

$$\boldsymbol{u}(t^{n+1}) = \boldsymbol{u}(t^n) - \int_{T^n} F(\boldsymbol{u}^{(K)}(t))dt.$$

- Convergence?
- How many steps K?
- Accuracy $\mathcal{L}^2$?

ADER 2nd order

Example with 2 Gauss Legendre points, Lagrange polynomials and 2 iterations
Let us consider the timestep interval $[t^n, t^{n+1}]$, rescaled to $[0,1]$.
Gauss-Legendre points quadrature and interpolation (in the interval $[0,1]$)

$$\underline{t}_q = \left(t_q^0, t_q^1\right) = \left(t^0, t^1\right) = \left(\frac{\sqrt{3}-1}{2\sqrt{3}}, \frac{\sqrt{3}+1}{2\sqrt{3}}\right), \quad \underline{w} = (1/2, 1/2)\,.$$

$$\underline{\phi}(t) = (\phi_0(t), \phi_1(t)) = \left(\frac{t-t^1}{t^0-t^1}, \frac{t-t^0}{t^1-t^0}\right).$$

Then, the mass matrix is given by

$$\underline{\underline{\mathrm{M}}}_{m,l} = \phi_m(1)\phi_l(1) - \phi_m'(t^l)w_l, \quad m, l = 0, 1,$$

$$\underline{\underline{\mathrm{M}}} = \begin{pmatrix} 1 & \frac{\sqrt{3}-1}{2} \\ -\frac{\sqrt{3}+1}{2} & 1 \end{pmatrix}.$$

ADER 2nd order

The right hand side is given

$$r(\underline{\boldsymbol{u}})_m = \alpha(0)\phi_m(0) + \Delta t F(\alpha(t^m))w_m, \quad m = 0, 1.$$

$$\underline{r}(\underline{\boldsymbol{u}}) = \alpha(0)\underline{\phi}(0) + \Delta t \begin{pmatrix} F(\alpha(t^1))w_1 \\ F(\alpha(t^2))w_2. \end{pmatrix}.$$

Then, the coefficients $\underline{\boldsymbol{u}}$ are given by

$$\underline{\boldsymbol{u}}^{(k+1)} = \underline{\underline{\mathrm{M}}}^{-1}\underline{r}(\underline{\boldsymbol{u}}^{(k)}).$$

Finally, use $\underline{\boldsymbol{u}}^{(k+1)}$ to reconstruct the solution at the time step t^{n+1}:

$$\boldsymbol{u}^{n+1} = \underline{\phi}(1)^T \underline{\boldsymbol{u}}^{(k+1)} = \boldsymbol{u}^n + \int_{T^n} \underline{\phi}(t)^T dt\, F(\underline{\boldsymbol{u}}^{(k)}).$$

CODE

- Choice: ϕ Lagrangian basis functions
- Different subtimesteps: Gauss-Legendre, Gauss–Lobatto, equispaced
- Precompute $\underline{\underline{\mathrm{M}}}$
- Precompute the rhs vector part using quadratures after a further approximation

$$\underline{r}(\underline{\boldsymbol{u}}) = \underline{\phi}(t^n)\boldsymbol{u}^n + \int_{T^n} \underline{\phi}(t)F(\underline{\phi}(t)^T\underline{\boldsymbol{u}})dt \approx \underline{\phi}(t^n)\boldsymbol{u}^n + \underbrace{\int_{T^n} \underline{\phi}(t)\underline{\phi}(t)^T dt}_{\text{Can be stored}} F(\underline{\boldsymbol{u}})$$

- Precompute the reconstruction coefficients $\underline{\phi}(1)^T$

Outline

ADER[6] and DeC[7]: immediate similarities

- High order time-space discretization
- Start from a well known space discretization (FE/DG/FV)
- FE reconstruction in time
- System in time, with M equations
- Iterative method / K corrections

[6] M. Dumbser, D. S. Balsara, E. F. Toro, and C.-D. Munz. A unified framework for the construction of one-step finite volume and discontinuous galerkin schemes on unstructured meshes. Journal of Computational Physics, 227(18):8209–8253, 2008.

[7] R. Abgrall. High order schemes for hyperbolic problems using globally continuous approximation and avoiding mass matrices. Journal of Scientific Computing, 73(2):461–494, Dec 2017.

ADER[6] and DeC[7]: immediate similarities

- High order time-space discretization
- Start from a well known space discretization (FE/DG/FV)
- FE reconstruction in time
- System in time, with M equations
- Iterative method / K corrections
- Both high order explicit time integration methods (neglecting spatial discretization)

[6] M. Dumbser, D. S. Balsara, E. F. Toro, and C.-D. Munz. A unified framework for the construction of one-step finite volume and discontinuous galerkin schemes on unstructured meshes. Journal of Computational Physics, 227(18):8209–8253, 2008.

[7] R. Abgrall. High order schemes for hyperbolic problems using globally continuous approximation and avoiding mass matrices. Journal of Scientific Computing, 73(2):461–494, Dec 2017.

ADER as DeC

ADER as DeC

ADER as DeC

$$\mathcal{L}^2(\underline{\boldsymbol{u}}) := \underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}} - r(\underline{\boldsymbol{u}}),$$
$$\mathcal{L}^1(\underline{\boldsymbol{u}}) := \underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}} - r(\boldsymbol{u}(t^n)).$$

$$\mathcal{L}^1(\underline{\boldsymbol{u}}^{(k)}) = \mathcal{L}^1(\underline{\boldsymbol{u}}^{(k-1)}) - \mathcal{L}^2(\underline{\boldsymbol{u}}^{(k-1)}), \qquad k = 1, \dots, K,$$

$$\underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}}^{(k)} - r(\boldsymbol{u}^{(k),0}) - \underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}}^{(k-1)} + r(\boldsymbol{u}^{(k-1),0}) + \underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}}^{(k-1)} - r(\underline{\boldsymbol{u}}^{(k-1)}) = 0$$

ADER as DeC

$$\mathcal{L}^2(\underline{\boldsymbol{u}}) := \underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}} - r(\underline{\boldsymbol{u}}),$$
$$\mathcal{L}^1(\underline{\boldsymbol{u}}) := \underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}} - r(\boldsymbol{u}(t^n)).$$

$$\mathcal{L}^1(\underline{\boldsymbol{u}}^{(k)}) = \mathcal{L}^1(\underline{\boldsymbol{u}}^{(k-1)}) - \mathcal{L}^2(\underline{\boldsymbol{u}}^{(k-1)}), \qquad k = 1, \dots, K,$$

$$\underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}}^{(k)} - \cancel{r(\boldsymbol{u}^{(k),0})} - \underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}}^{(k-1)} + \cancel{r(\boldsymbol{u}^{(k-1),0})} + \underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}}^{(k-1)} - r(\underline{\boldsymbol{u}}^{(k-1)}) = 0$$

ADER as DeC

$$\mathcal{L}^2(\underline{\boldsymbol{u}}) := \underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}} - r(\underline{\boldsymbol{u}}),$$
$$\mathcal{L}^1(\underline{\boldsymbol{u}}) := \underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}} - r(\boldsymbol{u}(t^n)).$$

$$\mathcal{L}^1(\underline{\boldsymbol{u}}^{(k)}) = \mathcal{L}^1(\underline{\boldsymbol{u}}^{(k-1)}) - \mathcal{L}^2(\underline{\boldsymbol{u}}^{(k-1)}), \qquad k = 1, \dots, K,$$

$$\underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}}^{(k)} - \cancel{r(\boldsymbol{u}^{(k),0})} - \cancel{\underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}}^{(k-1)}} + \cancel{r(\boldsymbol{u}^{(k-1),0})} + \cancel{\underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}}^{(k-1)}} - r(\underline{\boldsymbol{u}}^{(k-1)}) = 0$$

ADER as DeC

$$\mathcal{L}^2(\underline{\boldsymbol{u}}) := \underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}} - r(\underline{\boldsymbol{u}}),$$
$$\mathcal{L}^1(\underline{\boldsymbol{u}}) := \underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}} - r(\boldsymbol{u}(t^n)).$$

$$\mathcal{L}^1(\underline{\boldsymbol{u}}^{(k)}) = \mathcal{L}^1(\underline{\boldsymbol{u}}^{(k-1)}) - \mathcal{L}^2(\underline{\boldsymbol{u}}^{(k-1)}), \qquad k = 1, \dots, K,$$

$$\underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}}^{(k)} - \cancel{r(\boldsymbol{u}^{(k),0})} - \cancel{\underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}}^{(k-1)}} + \cancel{r(\boldsymbol{u}^{(k-1),0})} + \cancel{\underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}}^{(k-1)}} - r(\underline{\boldsymbol{u}}^{(k-1)}) = 0$$
$$\underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}}^{(k)} - r(\underline{\boldsymbol{u}}^{(k-1)}) = 0.$$

ADER as DeC

$$\mathcal{L}^2(\underline{\boldsymbol{u}}) := \underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}} - r(\underline{\boldsymbol{u}}),$$
$$\mathcal{L}^1(\underline{\boldsymbol{u}}) := \underline{\underline{\mathrm{M}}}\underline{\boldsymbol{u}} - r(\boldsymbol{u}(t^n)).$$

Apply the DeC Convergence theorem!

- $\mathcal{L}^1$ is coercive because $\underline{\underline{\mathrm{M}}}$ is always invertible
- $\mathcal{L}^1 - \mathcal{L}^2$ is Lipschitz with constant $C\Delta t$ because they are consistent approx of the same problem
- Hence, after K iterations we obtain a Kth order accurate approximation of $\underline{\boldsymbol{u}}^*$

DeC as ADER

$$\mathcal{L}^2(\boldsymbol{u}^0,\dots,\boldsymbol{u}^M) := \begin{cases} \boldsymbol{u}^M - \boldsymbol{u}^0 - \sum_{r=0}^M \int_{t^0}^{t^M} F(\boldsymbol{u}^r)\varphi_r(s)\mathrm{d}s \\ \dots \\ \boldsymbol{u}^1 - \boldsymbol{u}^0 - \sum_{r=0}^M \int_{t^0}^{t^1} F(\boldsymbol{u}^r)\varphi_r(s)\mathrm{d}s \end{cases}.$$

DeC as ADER

DeC as ADER

DeC as ADER

$$\mathcal{L}^2(\boldsymbol{u}^0,\dots,\boldsymbol{u}^M) := \begin{cases} \boldsymbol{u}^M - \boldsymbol{u}^0 - \sum_{r=0}^M \int_{t^0}^{t^M} F(\boldsymbol{u}^r)\varphi_r(s)\mathrm{d}s \\ \dots \\ \boldsymbol{u}^1 - \boldsymbol{u}^0 - \sum_{r=0}^M \int_{t^0}^{t^1} F(\boldsymbol{u}^r)\varphi_r(s)\mathrm{d}s \end{cases}.$$

DeC as ADER

$$\mathcal{L}^2(\boldsymbol{u}^0,\dots,\boldsymbol{u}^M) := \begin{cases} \boldsymbol{u}^M - \boldsymbol{u}^0 - \sum_{r=0}^M \int_{t^0}^{t^M} F(\boldsymbol{u}^r)\varphi_r(s)\mathrm{d}s \\ \dots \\ \boldsymbol{u}^1 - \boldsymbol{u}^0 - \sum_{r=0}^M \int_{t^0}^{t^1} F(\boldsymbol{u}^r)\varphi_r(s)\mathrm{d}s \end{cases} .$$

$$\chi_{[t^0,t^m]}(t^m)\boldsymbol{u}^m - \chi_{[t^0,t^m]}(t_0)\boldsymbol{u}^0 - \int_{t^0}^{t^m} \chi_{[t^0,t^m]}(t)\sum_{r=0}^M F(\boldsymbol{u}^r)\varphi_r(t)\mathrm{d}t = 0$$

$$\int_{t^0}^{t^M} \chi_{[t^0,t^m]}(t)\partial_t\left(\boldsymbol{u}(t)\right)\mathrm{d}t - \int_{t^0}^{t^M} \chi_{[t^0,t^m]}(t)\sum_{r=0}^M F(\boldsymbol{u}^r)\varphi_r(t)\mathrm{d}t = 0,$$

$$\int_{T^n} \psi_m(t)\partial_t \boldsymbol{u}(t)\mathrm{d}t - \int_{T^n} \psi_m(t) F(\boldsymbol{u}(t))\mathrm{d}t = 0.$$

Runge Kutta vs DeC–ADER

Classical Runge Kutta (RK)

- One step method
- Internal stages

Explicit Runge Kutta

+ Simple to code
- Not easily generalizable to arbitrary order
- Stages $>$ order

Implicit Runge Kutta

+ Arbitrarily high order
- Require nonlinear solvers for nonlinear systems
- May not converge

DeC – ADER

- One step method
- Internal subtimesteps
- Can be rewritten as explicit RK (for ODE)

+ Explicit
+ Simple to code
+ Iterations $=$ order
+ Arbitrarily high order
- Large memory storage

Outline

Stability

Since ADER can be written as a DeC, the stability functions are given by the same formula as for DeC and the stability regions are the following.

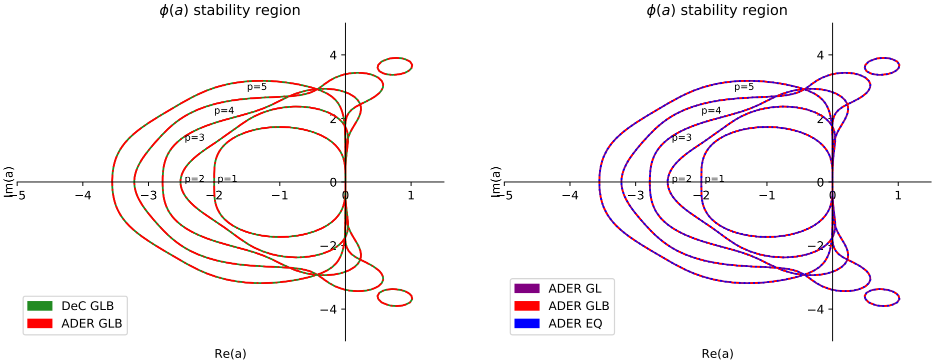


Figure: Stability region

Accuracy of ADER $\mathcal{L}^2$ operators

The two things that determine the accuracy of the ADER method are the iterations P and the accuracy of $\mathcal{L}^2$.

Accuracy of ADER $\mathcal{L}^2$ for different distributions

- Equispaced: boring, minimum accuracy possible $M+1$ nodes $p=M+1$
- Guass–Lobatto: this generates the LobattoIIIC methods, $M+1$ nodes $p=2M$
- Gauss–Legendre: this does not generate Gauss methods, $M+1$ nodes $p=2M+1$

$\mathcal{L}^2$ ADER as RK

Here, we see $\mathcal{L}^2$ as an implicit RK

$$\mathcal{L}^{2,m}(\underline{\boldsymbol{u}}) = \underline{\underline{\mathrm{M}}}^m_j \boldsymbol{u}^{(j)} - \underline{\phi}^m(t^n)\boldsymbol{u}^n - \underbrace{\int_{T^n} \underline{\phi}^m(t)\underline{\phi}(t)_j dt}_{\Delta t \underline{\underline{\mathrm{R}}}^m_j} F(\boldsymbol{u}^{(j)}) = 0$$

$$\tilde{\mathcal{L}}^{2,z}(\underline{\boldsymbol{u}}) = \boldsymbol{u}^{(z)} - (\underline{\underline{\mathrm{M}}}^{-1})^z_m \underline{\phi}^m(t^n)\boldsymbol{u}^n - \Delta t (\underline{\underline{\mathrm{M}}}^{-1})^z_m \underline{\underline{\mathrm{R}}}^m_j F(\boldsymbol{u}^{(j)}) = 0$$

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

- $a_{mj} = (\underline{\underline{\mathrm{M}}}^{-1})^z_m \underline{\underline{\mathrm{R}}}^m_j$
- Prove that $(\underline{\underline{\mathrm{M}}}^{-1})^z_m \underline{\phi}^m(t^n) = 1$ for every z
- $c^m = \sum_r a_{mr} = t^m$
- $b_r = \frac{1}{\Delta t}\int_{T^m} \phi_r(t)dt = w_r$ quadrature weights

$\mathcal{L}^2$ ADER as RK

$\mathcal{L}^2$ ADER as RK

BCD conditions (Butcher 1964)

Define the conditions

$$B(p): \quad \sum_{i=1}^{s} b_i c_i^{z-1} = \frac{1}{z}, \qquad z = 1, \dots, p; \tag{20}$$

$$C(\eta): \quad \sum_{j=1}^{s} a_{ij} c_j^{z-1} = \frac{c_i^z}{z}, \qquad i = 1, \dots, s,\ z = 1, \dots, \eta; \tag{21}$$

$$D(\zeta): \quad \sum_{i=1}^{s} b_i c_i^{z-1} a_{ij} = \frac{b_j}{z}(1 - c_j^z), \qquad j = 1, \dots, s,\ z = 1, \dots, \zeta. \tag{22}$$

Theorem (Butcher 1964)

If the coefficients b_i, c_i, a_{ij} of a RK scheme satisfy $B(p)$, $C(\eta)$ and $D(\zeta)$ with $p \le \eta + \zeta + 1$ and $p \le 2\eta + 2$, then the method is of order p.

$C(s-1)$ $D(s-1)$

Lemma

$\mathcal{L}^2$ operator of ADER defined by Gauss–Lobatto or Gauss–Legendre points and quadrature (they coincide) with $s=M+1$ stages satisfies $C(s-1)$ and $D(s-1)$.

Proof (1/4).

Interpolation with ϕ^j is exact for polynomials of degree $s-1$.
The quadrature is exact for polynomials of degree $2s-3$.
Recall that $\underline{\underline{A}} = \underline{\underline{\mathrm{MR}}}$, Condition $C(s-1)$ reads

$$\underline{\underline{A}}\,\underline{c^{z-1}} = \frac{1}{z}\underline{c^z} \iff \underline{\underline{\mathrm{R}}}\,\underline{c^{z-1}} = \frac{1}{z}\underline{\underline{\mathrm{M}}}\,\underline{c^z} \iff \underline{\mathcal{X}} := \underline{\underline{\mathrm{R}}}\,\underline{c^{z-1}} - \frac{1}{z}\underline{\underline{\mathrm{M}}}\,\underline{c^z} = \underline{0}, \qquad z=1,\dots,s-1.$$

Recall $b_m = t^m$, $c_m = w_m$, $\underline{\underline{\mathrm{R}}}_{i,j} = \delta_{i,j} w_i$ and the definition of $\underline{\underline{\mathrm{M}}}$

$$\mathcal{X}_m := w_m (t^m)^{z-1} - \frac{1}{z}\left(\phi^m(1)\phi^j(1)(t^j)^z - \int_0^1 \frac{d}{d\xi}\phi^m(\xi)\phi^j(\xi)(t^j)^z d\xi \right).$$

$C(s-1)$ $D(s-1)$

Proof (2/4).

Now, the interpolation of t^z with $z \leq s-1$ with basis functions ϕ^j is exact. Hence, we can substitute $\phi^j(\xi)(t^j)^z = \xi^z$ for all $z = 1, \dots, s-1$, obtaining

$$\mathcal{X}_m = w_m (t^m)^{z-1} - \frac{1}{z}\left(\phi^m(1) 1^z - \int_0^1 \frac{d}{d\xi}\phi^m(\xi)\xi^z d\xi \right).$$

Using the exactness of the quadrature for polynomials of degree $2s-3$, both true for Gauss–Lobatto and Gauss–Legendre, we know that the previous integral is exactly computed as $\frac{d}{d\xi}\phi^m(\xi)$ is of degree at most $s-2$ and ξ^z is at most $s-1$. So, we can use integration by parts and obtain

$$\mathcal{X}_m = w_m (t^m)^{z-1} - \frac{1}{z}\left(\phi^m(0) 0^z + \int_0^1 \phi^m(\xi)\frac{d}{d\xi}\xi^z d\xi \right) = w_m (t^m)^{z-1} - \int_0^1 \phi^m(\xi)\xi^{z-1} d\xi = 0$$

by the exactness of the quadrature rule and the definition of w_m. Note that the condition is sharp, since the interpolation is not anymore exact for $z = s$, hence $C(s)$ is not satisfied.

Proof (3/4).

To prove $D(s-1)$, we write explicitly the condition in matricial form, for all $z = 1, \dots, s-1$

$$\underline{bc^{z-1}}\,\underline{\underline{A}} = \frac{1}{z}\underline{b(1-c^z)} \iff \underline{bc^{z-1}}\,\underline{\underline{\mathrm{M}}}^{-1}\,\underline{\underline{\mathrm{R}}} = \frac{1}{z}\underline{b(1-c^z)} \iff \underline{bc^{z-1}} = \frac{1}{z}\underline{b(1-c^z)}\,\underline{\underline{\mathrm{R}}}^{-1}\,\underline{\underline{\mathrm{M}}}.$$

Note that $b^m = w_m$ and $\underline{\underline{\mathrm{R}}}_r^m = w_m\delta_r^m$, so $\underline{b(1-c^z)}\,\underline{\underline{\mathrm{R}}}^{-1} = \underline{(1-c^z)}$. It is left to prove that

$$\mathcal{Y} := \underline{bc^{z-1}} - \frac{1}{z}\underline{(1-c^z)}\underline{\underline{\mathrm{M}}} = \underline{0}.$$

$$\mathcal{Y}_m = w_m(t^m)^{z-1} - \frac{1}{z}\sum_{j=1}^{s}\left(1-(t^j)^z\right)\left(\phi^j(1)\phi^m(1) - \int_0^1 \frac{d}{d\xi}\phi^j(\xi)\phi^m(\xi)d\xi\right).$$

Proof (4/4).

Let us observe that, since $z \leq s-1$, the polynomial is exactly represented by the Lagrangian interpolation $t^z = \sum_{j=1}^{s} \phi(t)(t^m)^z$. Hence, using the exactness of the quadrature for polynomials of degree at most $2s-3$, we have

$$\begin{aligned}\mathcal{Y}_m =& w_m(t^m)^{z-1} - \frac{1}{z}\left(1-(1)^z\right)\phi^m(1) + \frac{1}{z}\int_0^1 \frac{d}{d\xi}\left(1-(\xi)^z\right)\phi^m(\xi)d\xi \\ =& w_m(t^m)^{z-1} - \frac{1}{z}\int_0^1 z\,\xi^{z-1}\phi^m(\xi)d\xi = w_m(t^m)^{z-1} - w_m(t^m)^{z-1} = 0.\end{aligned}$$

Hence, ADER-Legendre and ADER-Lobatto satisfy $D(s-1)$. Note that the condition is sharp, since the interpolation is not anymore exact for $z=s$, hence $D(s)$ is not satisfied.

ADER Gauss–Legendre $\mathcal{L}^2$

Remark (ADER-Legendre is no collocation method)

From the proof of previous Lemma, we can observe that ADER-Legendre methods do not satisfy $C(s)$, hence, the methods are not collocation methods and they do not coincide with Gauss–Legendre implicit RK methods.

Theorem

$\mathcal{L}^2$ of ADER with Gauss–Legendre is of order $2s-1$.

Proof.

ADER-Legendre with $s = M+1$ stages satisfies $B(2s)$ for the quadrature rule and, hence, it satisfies $B(2s-1)$. For previous Lemma it also satisfies $C(s-1)$ and $D(s-1)$. Hence, Butcher's (1964) Theorem ($p \leq \eta + \zeta + 1$ and $p \leq 2\eta + 2$) guarantees that the method is of order $2s-1$, since it is satisfied with $p = 2s-1$ and $\eta = \zeta = s-1$. $\square$

ADER Gauss–Lobatto $\mathcal{L}^2$

Theorem

$\mathcal{L}^2$ of ADER with Gauss-Lobatto is of order $2s-2$.

Proof.

The condition for $B(2s-2)$ is satisfied as (c,b) is the Gauss–Lobatto quadrature with order $2s-2$. Previous Lemma guarantees that ADER-Lobatto satisfies $B(2s-2)$, $C(s-1)$ and $D(s-1)$, so Butcher's (1964) Theorem ($p \leq \eta + \zeta + 1$ and $p \leq 2\eta + 2$) is satisfied for order $p = 2s-2$ and $\eta = \zeta = s-1$. □

ADER Gauss–Lobatto $\mathcal{L}^2$

Theorem

$\mathcal{L}^2$ of ADER with Gauss-Lobatto is LobattoIIIC.

The Lobatto IIIC method is defined using the condition

$$a_{i1} = b_1 \qquad \text{for } i = 1, \dots, s. \tag{23}$$

Lemma

$\mathcal{L}^2$ of ADER with Gauss-Lobatto satisfies (23).

Theorem (Chipman 1971)

Lobatto IIIC schemes (in particular RK a_{ij}) are uniquely determined by Gauss–Lobatto quadrature rule (c, b), condition (23) *and by $C(s-1)$.*

ADER Gauss–Lobatto $\mathcal{L}^2$

Lemma

$\mathcal{L}^2$ of ADER with Gauss-Lobatto satisfies (23).

Proof.

$$a_{i1} = \sum_j (\underline{\underline{\mathrm{M}}}^{-1})_{ij}\mathbb{R}_{j1} = b_1 = w_1 \iff$$

$$\sum_{i,j} \underline{\underline{\mathrm{M}}}_{ki}(\underline{\underline{\mathrm{M}}}^{-1})_{ij}\mathbb{R}_{j1} = \sum_i \underline{\underline{\mathrm{M}}}_{ki} w_1 \iff$$

$$\delta_{k1} w_1 = \mathbb{R}_{k1} = \sum_i \underline{\underline{\mathrm{M}}}_{ki} w_1$$

$$\sum_i \underline{\underline{\mathrm{M}}}_{ki} w_1 = \phi^m(1) w_1 - \int_0^1 \frac{d}{dt}\phi^m(\xi) w_1 dt = w_1 \phi^m(0) = w_1 \delta_{m,1}.$$

$\square$

Outline

Applications

Usages

- Hyperbolic PDEs as explicit iterative methods (ADER: Toro, Dumbser, Klingenberg, Boscheri; DeC: Abgrall, Ricchiuto)
- IMEX solvers for hyperbolic with stiff sources (ADER: Dumbser, Boscheri; DeC: Abgrall, Torlo)
- IMEX solvers for hyperbolic with viscosity (treated implicitly) as compressible Navier Stokes (DeC: Minion, Dumbser, Zeifang)

IMEX

$\partial_t u = F(u) + S(u)$
$S(u)$ stiff to be treated implicitly

Advantages

- Arbitrary high order
- Unique framework to have matching between implicit and explicit terms
- Easy to code
- Iterative solver automatically included

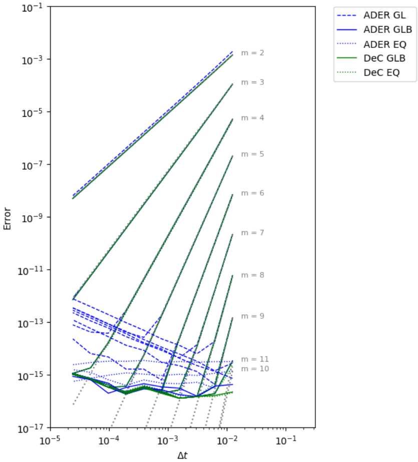
Disadvantages

- Explicit solver: many many stages
- Implicit: many stages
- Explicit: not amazing stability property (wrt SSP RK e.g.)

Convergence

$$
\begin{aligned}
&y'(t) = -|y(t)|y(t), \\
&y(0) = 1, \qquad (24)\\
&t \in [0, 0.1].
\end{aligned}
$$

Convergence curves for ADER and DeC, varying the approximation order and collocation of nodes for the subtimesteps for a scalar nonlinear ODE



Lotka–Volterra

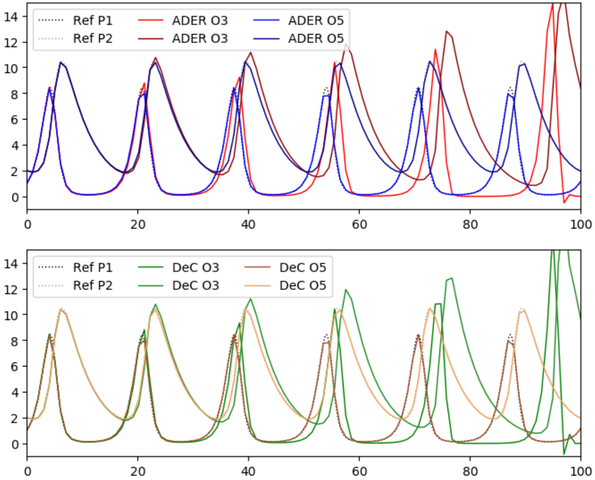
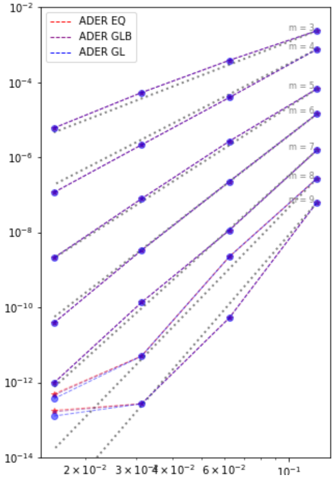
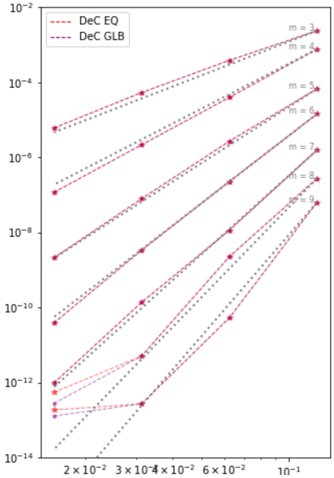


Figure: Numerical solution of the Lotka-Volterra system using ADER (top) and DeC (bottom) with Gauss-Lobatto nodes with timestep $\Delta T = 1$.

PDE: Burgers with spectral difference





Convergence error for Burgers equations: Left ADER right DeC. Space discretization with spectral difference

Outline

Reduce computational cost for explicit DeC

Literature

- *L. Micalizzi and D. Torlo. "A new efficient explicit Deferred Correction framework: analysis and applications to hyperbolic PDEs and adaptivity. " Commun. Appl. Math. Comput. (2023).* arxiv.org/abs/2210.02976
- *L. Micalizzi, D. Torlo and W. Boscheri. "Efficient iterative arbitrary high order methods: an adaptive bridge between low and high order." Commun. Appl. Math. Comput. (2023)* arxiv.org/abs/2212.07783
- *M. Han Veiga, L. Micalizzi and D. Torlo. "On improving the efficiency of ADER methods." Applied Mathematics and Computation, 466, page 128426, 2024.* arxiv.org/abs/2305.13065

Goal

Reduce computational costs of explicit DeC/ADER.

DeC as RK for ODEs

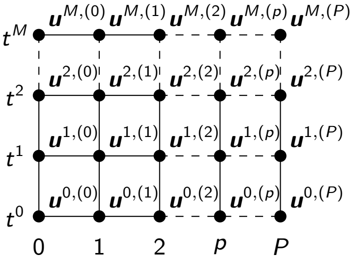
$$\mathcal{L}^1(\underline{\boldsymbol{u}}^{(p)}) = \mathcal{L}^1(\underline{\boldsymbol{u}}^{(p-1)}) - \mathcal{L}^2(\underline{\boldsymbol{u}}^{(p-1)}) \text{ with } p = 1, \dots, P.$$

$$\boldsymbol{u}^{m,(p)} = \boldsymbol{u}^0 + \sum_{r=0}^{M} \theta_r^m F(t^r, \boldsymbol{u}^{r,(p-1)}), \qquad \forall m = 1, \dots, M, \; p = 1, \dots, P$$

DeC as RK for ODEs

$$\mathcal{L}^1(\underline{\boldsymbol{u}}^{(p)}) = \mathcal{L}^1(\underline{\boldsymbol{u}}^{(p-1)}) - \mathcal{L}^2(\underline{\boldsymbol{u}}^{(p-1)}) \text{ with } p = 1, \dots, P.$$

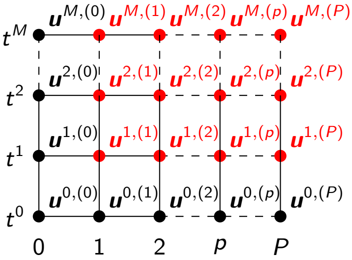
$$\boldsymbol{u}^{m,(p)} = \boldsymbol{u}^0 + \sum_{r=0}^{M} \theta_r^m F(t^r, \boldsymbol{u}^{r,(p-1)}), \qquad \forall m = 1, \dots, M, \; p = 1, \dots, P$$



DeC as RK for ODEs

$$\mathcal{L}^1(\underline{\boldsymbol{u}}^{(p)}) = \mathcal{L}^1(\underline{\boldsymbol{u}}^{(p-1)}) - \mathcal{L}^2(\underline{\boldsymbol{u}}^{(p-1)}) \text{ with } p = 1, \dots, P.$$

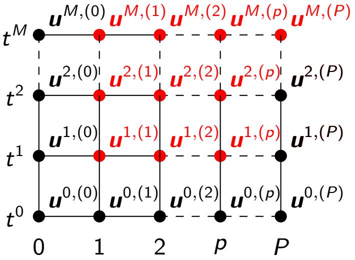
$$\boldsymbol{u}^{m,(p)} = \boldsymbol{u}^0 + \sum_{r=0}^{M} \theta_r^m F(t^r, \boldsymbol{u}^{r,(p-1)}), \qquad \forall m = 1, \dots, M, \; p = 1, \dots, P$$



DeC as RK for ODEs

$$\mathcal{L}^1(\underline{\boldsymbol{u}}^{(p)}) = \mathcal{L}^1(\underline{\boldsymbol{u}}^{(p-1)}) - \mathcal{L}^2(\underline{\boldsymbol{u}}^{(p-1)}) \text{ with } p = 1, \dots, P.$$

$$\boldsymbol{u}^{m,(p)} = \boldsymbol{u}^0 + \sum_{r=0}^{M} \theta_r^m F(t^r, \boldsymbol{u}^{r,(p-1)}), \qquad \forall m = 1, \dots, M, \; p = 1, \dots, P$$



$\underline{c}$	$\mathbf{u}^0$	$\mathbf{u}^{(1)}$	$\mathbf{u}^{(2)}$	$\mathbf{u}^{(3)}$	$\cdots$	$\mathbf{u}^{(M-1)}$	$\mathbf{u}^{(M)}$	A
0	0							$\mathbf{u}^0$
$\underline{\beta}_{1:}$	$\underline{\beta}_{1:}$	$\underline{\underline{0}}$						$\mathbf{u}^{(1)}$
$\underline{\beta}_{1:}$	$\Theta_{1:,0}$	$\Theta_{1:,1:}$	$\underline{\underline{0}}$					$\mathbf{u}^{(2)}$
$\underline{\beta}_{1:}$	$\Theta_{1:,0}$	$\underline{\underline{0}}$	$\Theta_{1:,1:}$	$\underline{\underline{0}}$				$\mathbf{u}^{(3)}$
	$\vdots$	$\vdots$		$\ddots$	$\ddots$			$\vdots$
	$\vdots$	$\vdots$			$\ddots$	$\ddots$		$\vdots$
$\underline{\beta}_{1:}$	$\Theta_{1:,0}$	$\underline{\underline{0}}$	$\cdots$	$\cdots$	$\underline{\underline{0}}$	$\Theta_{1:,1:}$	$\underline{\underline{0}}$	$\mathbf{u}^{(M)}$
$\underline{b}$	$\Theta_{M,0}$	$\underline{0}$	$\cdots$	$\cdots$	$\cdots$	$\underline{0}$	$\Theta_{M,1:}$	$\mathbf{u}^{M,(M+1)}$

Costs

Large costs!

Costs

Large costs!

- DeC $S = M \cdot (P-1)+1$
  - DeC equi $S = (P-1)^2+1$
  - DeC GLB $S = \left\lceil \frac{P}{2} \right\rceil (P-1)+1$

Equispaced

P	M	DeC
2	1	2
3	2	5
4	3	10
5	4	17
6	5	26
7	6	37
8	7	50
9	8	65
10	9	82

Gauss–Lobatto

P	M	DeC
2	1	2
3	2	5
4	2	7
5	3	13
6	3	16
7	4	25
8	4	29
9	5	41
10	5	46

Large costs!

- DeC $S = M \cdot (P-1) + 1$
  - DeC equi $S = (P-1)^2 + 1$
  - DeC GLB $S = \left\lceil \frac{P}{2} \right\rceil (P-1) + 1$

Equispaced

P	M	DeC
2	1	2
3	2	5
4	3	10
5	4	17
6	5	26
7	6	37
8	7	50
9	8	65
10	9	82

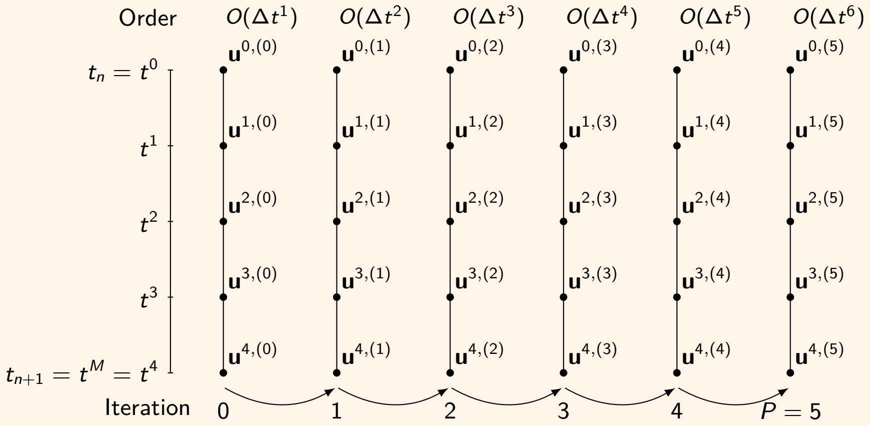
Gauss–Lobatto

P	M	DeC
2	1	2
3	2	5
4	2	7
5	3	13
6	3	16
7	4	25
8	4	29
9	5	41
10	5	46

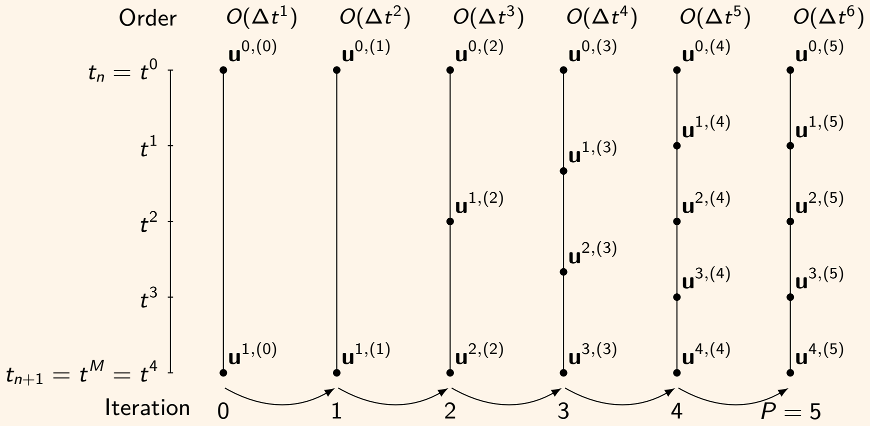
How can we save computational time?

Outline

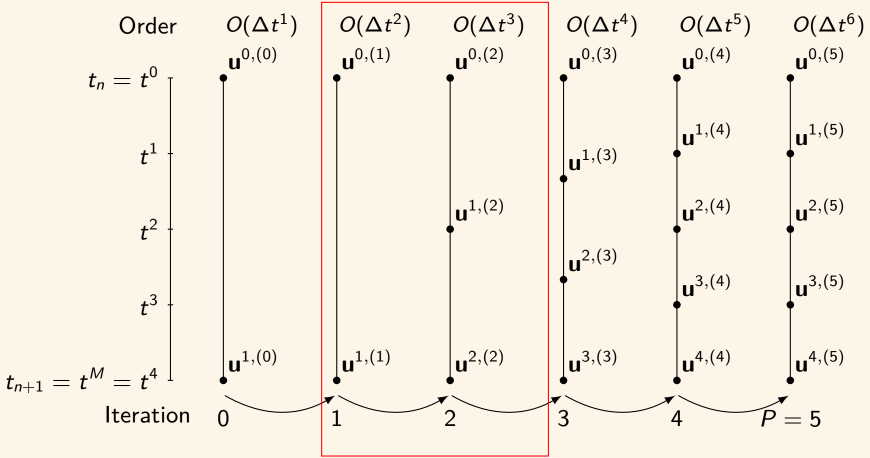
Idea for reduction of stages



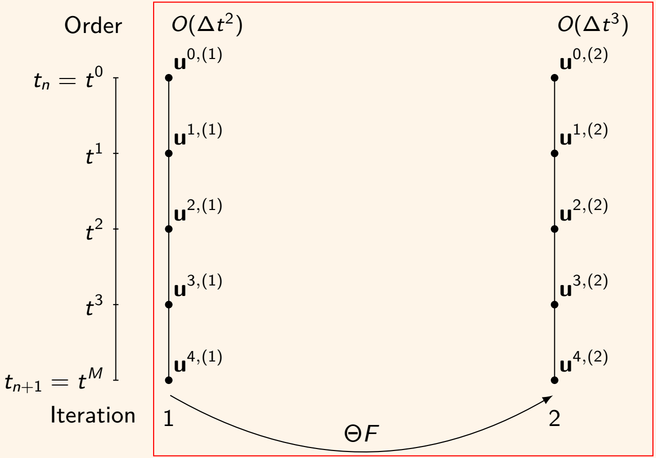
Idea for reduction of stages



Idea for reduction of stages



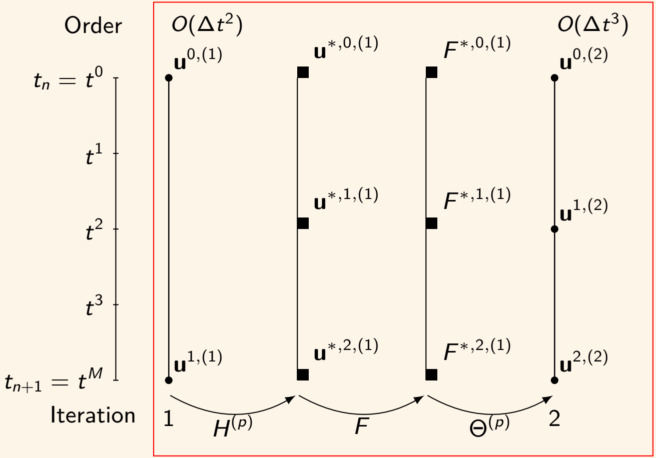
How to communicate between iterations?



DeC

$$\underline{\boldsymbol{u}}^{(p)} = \underline{\boldsymbol{u}}^{0} + \Delta t \Theta F(\underline{\boldsymbol{u}}^{(p-1)})$$

How to communicate between iterations?



DeC

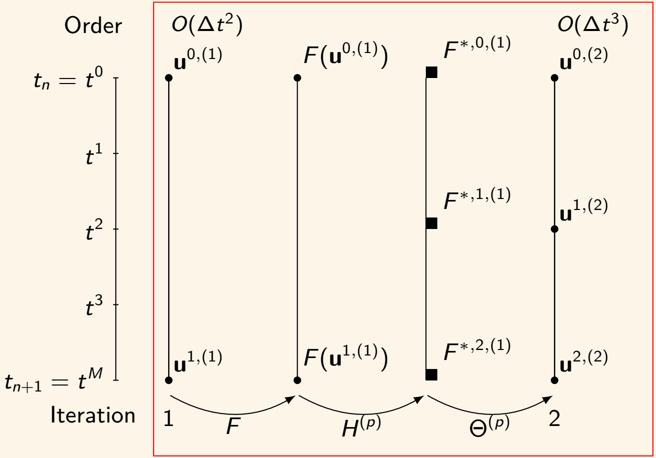
$$\underline{\boldsymbol{u}}^{(p)} = \underline{\boldsymbol{u}}^0 + \Delta t \Theta F(\underline{\boldsymbol{u}}^{(p-1)})$$

DeCu

$$\underline{\boldsymbol{u}}^{(p)} = \underline{\boldsymbol{u}}^0 + \Delta t \Theta^{(p)} F(H^{(p)} \underline{\boldsymbol{u}}^{(p-1)})$$

$$H^{(p)}_{ij} = \phi_j^{(p-1)}(t^{i,(p)})$$

How to communicate between iterations?



DeC

$$\underline{\boldsymbol{u}}^{(p)} = \underline{\boldsymbol{u}}^{0} + \Delta t \Theta F(\underline{\boldsymbol{u}}^{(p-1)})$$

DeCu

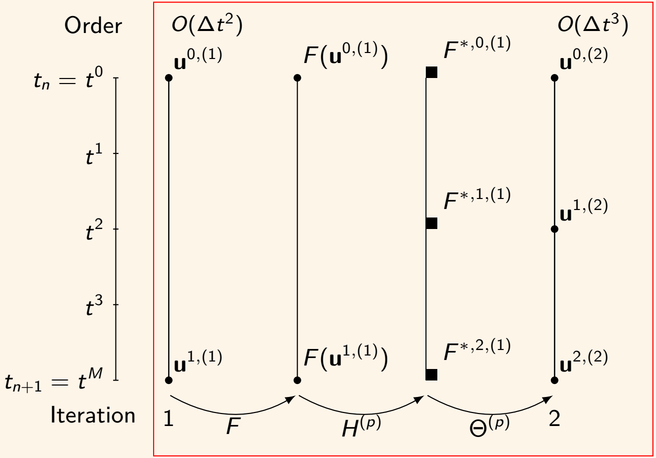
$$\underline{\boldsymbol{u}}^{(p)} = \underline{\boldsymbol{u}}^{0} + \Delta t \Theta^{(p)} F(H^{(p)} \underline{\boldsymbol{u}}^{(p-1)})$$

DeCdu

$$\underline{\boldsymbol{u}}^{(p)} = \underline{\boldsymbol{u}}^{0} + \Delta t \Theta^{(p)} H^{(p)} F(\underline{\boldsymbol{u}}^{(p-1)})$$

$$H^{(p)}_{ij} = \phi_j^{(p-1)}(t^{i,(p)})$$

How to communicate between iterations?



DeC

$$\underline{\boldsymbol{u}}^{(p)} = \underline{\boldsymbol{u}}^{0} + \Delta t \Theta F(\underline{\boldsymbol{u}}^{(p-1)})$$

DeCu

$$\underline{\boldsymbol{u}}^{(p)} = \underline{\boldsymbol{u}}^{0} + \Delta t \Theta^{(p)} F(H^{(p)} \underline{\boldsymbol{u}}^{(p-1)})$$

$$\underline{\boldsymbol{u}}^{*(p)} = \underline{\boldsymbol{u}}^{0} + \Delta t H^{(p)} \Theta^{*(p-1)} F(\underline{\boldsymbol{u}}^{*(p-1)})$$

DeCdu

$$\underline{\boldsymbol{u}}^{(p)} = \underline{\boldsymbol{u}}^{0} + \Delta t \Theta^{(p)} H^{(p)} F(\underline{\boldsymbol{u}}^{(p-1)})$$

$$H^{(p)}_{ij} = \phi_j^{(p-1)}(t^{i,(p)})$$

Efficient DeC into RK framework

DeC $\quad S = M \cdot (P-1) + 1$

$\underline{c}$	$\mathbf{u}^0$	$\mathbf{u}^{(1)}$	$\mathbf{u}^{(2)}$	$\mathbf{u}^{(3)}$	$\cdots$	$\mathbf{u}^{(M-1)}$	$\mathbf{u}^{(M)}$	A	dim
0	0							$\mathbf{u}^0$	1
$\underline{\beta}_{1:}$	$\underline{\beta}_{1:}$	$\underline{\underline{0}}$						$\mathbf{u}^{(1)}$	M
$\underline{\beta}_{1:}$	$\Theta_{1:,0}$	$\Theta_{1:,1:}$	$\underline{\underline{0}}$					$\mathbf{u}^{(2)}$	M
$\underline{\beta}_{1:}$	$\Theta_{1:,0}$	$\underline{\underline{0}}$	$\Theta_{1:,1:}$	$\underline{\underline{0}}$				$\mathbf{u}^{(3)}$	M
	$\vdots$	$\vdots$		$\ddots$	$\ddots$			$\vdots$	M
	$\vdots$	$\vdots$			$\ddots$	$\ddots$		$\vdots$	M
$\underline{\beta}_{1:}$	$\Theta_{1:,0}$	$\underline{\underline{0}}$	$\cdots$	$\cdots$	$\underline{\underline{0}}$	$\Theta_{1:,1:}$	$\underline{\underline{0}}$	$\mathbf{u}^{(M)}$	M
$\underline{b}$	$\Theta_{M,0}$	$\underline{0}$	$\cdots$	$\cdots$	$\cdots$	$\underline{0}$	$\Theta_{M,1:}$	$\mathbf{u}^{M,(M+1)}$	

Efficient DeC into RK framework

DeCu $S = M \cdot (P-1) + 1 - \frac{(M-1)(M-2)}{2}$

$\underline{c}$	$\mathbf{u}^0$	$\mathbf{u}^{*(1)}$	$\mathbf{u}^{*(2)}$	$\mathbf{u}^{*(3)}$	$\dots$	$\mathbf{u}^{*(M-2)}$	$\mathbf{u}^{*(M-1)}$	$\mathbf{u}^{(M)}$	A	dim
0	0								$\mathbf{u}^0$	1
$\underline{\beta}^{(2)}_{1:}$	$\underline{\beta}^{(2)}_{1:}$	$\underline{\underline{0}}$							$\mathbf{u}^{*(1)}$	2
$\underline{\beta}^{(3)}_{1:}$	$W^{(2)}_{1:,0}$	$W^{(2)}_{1:,1:}$	$\underline{\underline{0}}$						$\mathbf{u}^{*(2)}$	3
$\underline{\beta}^{(4)}_{1:}$	$W^{(3)}_{1:,0}$	$\underline{\underline{0}}$	$W^{(3)}_{1:,1:}$	$\underline{\underline{0}}$					$\mathbf{u}^{*(3)}$	4
	$\vdots$	$\vdots$		$\ddots$	$\ddots$				$\vdots$	$\vdots$
	$\vdots$	$\vdots$			$\ddots$	$\ddots$			$\vdots$	$\vdots$
$\underline{\beta}^{(M)}_{1:}$	$W^{(M-1)}_{1:,0}$	$\underline{\underline{0}}$	$\cdots$	$\cdots$	$\underline{\underline{0}}$	$W^{(M-1)}_{1:,1:}$	$\underline{\underline{0}}$	$\underline{\underline{0}}$	$\mathbf{u}^{*(M-1)}$	M
$\underline{\beta}^{(M)}_{1:}$	$W^{(M)}_{1:,0}$	$\underline{\underline{0}}$	$\cdots$	$\cdots$	$\cdots$	$\underline{\underline{0}}$	$W^{(M)}_{1:,1:}$	$\underline{\underline{0}}$	$\mathbf{u}^{(M)}$	M
$\underline{b}$	$W^{(M+1)}_{M,0}$	$\underline{0}$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\underline{0}$	$W^{(M+1)}_{M,1:}$	$\mathbf{u}^{M,(M+1)}$	

$$W^{(p)} := \begin{cases} H^{(p)}\Theta^{(p)} \in \mathbb{R}^{(p+2)\times(p+1)}, & \text{if } p = 2, \dots, M-1, \\ \Theta^{(M)} \in \mathbb{R}^{(M+1)\times(M+1)}, & \text{if } p \geq M. \end{cases}$$

Efficient DeC into RK framework

$$\textbf{DeCdu} \qquad S = M\cdot(P-1)+1 - \frac{M(M-1)}{2}$$

$\underline{c}$	$\mathbf{u}^0$	$\mathbf{u}^{(1)}$	$\mathbf{u}^{(2)}$	$\mathbf{u}^{(3)}$	$\dots$	$\mathbf{u}^{(M-2)}$	$\mathbf{u}^{(M-1)}$	$\mathbf{u}^{(M)}$	A	dim
0	0								$\mathbf{u}^0$	1
$\underline{\beta}^{(1)}_{1:}$	$\underline{\beta}^{(1)}_{1:}$	$\underline{\underline{0}}$							$\mathbf{u}^{(1)}$	1
$\underline{\beta}^{(2)}_{1:}$	$Z^{(2)}_{1:,0}$	$Z^{(2)}_{1:,1:}$	$\underline{\underline{0}}$						$\mathbf{u}^{(2)}$	2
$\underline{\beta}^{(3)}_{1:}$	$Z^{(3)}_{1:,0}$	$\underline{\underline{0}}$	$Z^{(3)}_{1:,1:}$	$\underline{\underline{0}}$					$\mathbf{u}^{(3)}$	3
	$\vdots$	$\vdots$		$\ddots$	$\ddots$				$\vdots$	$\vdots$
	$\vdots$	$\vdots$			$\ddots$	$\ddots$			$\vdots$	$\vdots$
$\underline{\beta}^{(M-1)}_{1:}$	$Z^{(M-1)}_{1:,0}$	$\underline{\underline{0}}$	$\dots$	$\dots$	$\underline{\underline{0}}$	$Z^{(M-1)}_{1:,1:}$	$\underline{\underline{0}}$	$\underline{\underline{0}}$	$\mathbf{u}^{(M-1)}$	$M-1$
$\underline{\beta}^{(M)}_{1:}$	$Z^{(M)}_{1:,0}$	$\underline{\underline{0}}$	$\dots$	$\dots$	$\dots$	$\underline{\underline{0}}$	$Z^{(M)}_{1:,1:}$	$\underline{\underline{0}}$	$\mathbf{u}^{(M)}$	M
$\underline{b}$	$Z^{(M+1)}_{M,0}$	$\underline{0}$	$\dots$	$\dots$	$\dots$	$\dots$	$\underline{0}$	$Z^{(M+1)}_{M,1:}$	$\mathbf{u}^{M,(M+1)}$	

$$Z^{(p)} := \begin{cases} \Theta^{(p)} H^{(p-1)} \in \mathbb{R}^{(p+1)\times p}, & \text{if } p = 1,\dots,M, \\ \Theta^{(M)} \in \mathbb{R}^{(M+1)\times(M+1)}, & \text{if } p > M. \end{cases}$$

Computational costs reduction: RK stages

Equispaced

P	M	DeC	DeCu	DeCdu
2	1	2	2	2
3	2	5	5	4
4	3	10	9	7
5	4	17	14	11
6	5	26	20	16
7	6	37	27	22
8	7	50	35	29
9	8	65	44	37
10	9	82	54	46
11	10	101	65	56
12	11	122	77	67
13	12	145	90	79

Gauss-Lobatto

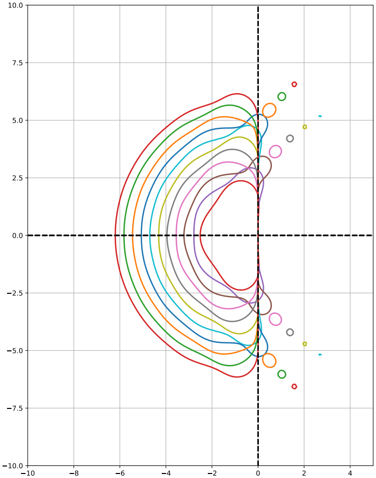
P	M	DeC	DeCu	DeCdu
2	1	2	2	2
3	2	5	5	4
4	2	7	7	6
5	3	13	12	10
6	3	16	15	13
7	4	25	22	19
8	4	29	26	23
9	5	41	35	31
10	5	46	40	36
11	6	61	51	46
12	6	67	57	52
13	7	85	70	64

DeC, DeCu, DeCdu

DeC-DeCu-DeCdu

The **stability function** of DeC, DeCu, DeCdu of order P for any nodes distribution is

$$R(z) = 1 + z + \frac{z^2}{2!} + \cdots + \frac{z^P}{P!}.$$



Exercise

Efficient DeC

- Code DeCu or DeCdu
- Check order of accuracy
- Write a code to obtain its RK matrix
- Check the stability function with nodepy
- Compare computational costs with original DeC

Outline

Summary: ADER/DeC $\mathcal{L}^2$

DeC

- Integral form
- Collocation methods
- Order ($M+1$ equispaced, $2M$ GLB)
- Stability (A-stability for GLB: Lobatto IIIC)

ADER

- Weak form
- Not collocation methods
- Order ($M+1$ equispaced, $2M$ GLB, $2M+1$ GLG)
- Stability (A-stability for GLB GLG, I don't know for equi)

Summary: ADER/DeC iterative