A new efficient explicit Deferred Correction framework: analysis and applications to hyperbolic PDEs and adaptivity

Davide Torlo*, Lorenzo Micalizzi

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

Essentially hyperbolic problems: unconventional numerics, and applications
Ascona - October 2022

A new efficient explicit **Deferred Correction** framework: analysis and applications to hyperbolic PDEs and adaptivity

Davide Torlo*, Lorenzo Micalizzi

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

Essentially hyperbolic problems: **unconventional numerics**, and applications
Ascona - October 2022

History of residual distribution and DeC

2016

- PhD in hyperbolic PDE field with Rémi

History of residual distribution and DeC

2016

- PhD in hyperbolic PDE field with Rémi

Spatial Discretizations

- Finite Volume

History of residual distribution and DeC

2016

- PhD in hyperbolic PDE field with Rémi

Spatial Discretizations

- Finite Volume
- Finite Difference

History of residual distribution and DeC

2016

- PhD in hyperbolic PDE field with Rémi

Spatial Discretizations

- Finite Volume
- Finite Difference
- Discontinuous Galerkin

History of residual distribution and DeC

2016

- PhD in hyperbolic PDE field with Rémi

Spatial Discretizations

- Finite Volume
- Finite Difference
- Discontinuous Galerkin

Time Discretizations

- Runge Kutta

History of residual distribution and DeC

2016

- PhD in hyperbolic PDE field with Rémi

Spatial Discretizations

- Finite Volume
- Finite Difference
- Discontinuous Galerkin

Time Discretizations

- Runge Kutta
- Strong Stability Preserving RK

History of residual distribution and DeC

2016

- PhD in hyperbolic PDE field with Rémi

Spatial Discretizations

- Finite Volume
- Finite Difference
- Discontinuous Galerkin

Time Discretizations

- Runge Kutta
- Strong Stability Preserving RK

Residual Distributions

- Originally Finite Volume

History of residual distribution and DeC

2016

- PhD in hyperbolic PDE field with Rémi

Spatial Discretizations

- Finite Volume
- Finite Difference
- Discontinuous Galerkin

Time Discretizations

- Runge Kutta
- Strong Stability Preserving RK

Residual Distributions

- Originally Finite Volume
- Nowadays **Continuous Finite Element**

History of residual distribution and DeC

2016

- PhD in hyperbolic PDE field with Rémi

Spatial Discretizations

- Finite Volume
- Finite Difference
- Discontinuous Galerkin

Time Discretizations

- Runge Kutta
- Strong Stability Preserving RK

Residual Distributions

- Originally Finite Volume
- Nowadays **Continuous Finite Element**
- Stabilizations

2016

- PhD in hyperbolic PDE field with Rémi

Spatial Discretizations

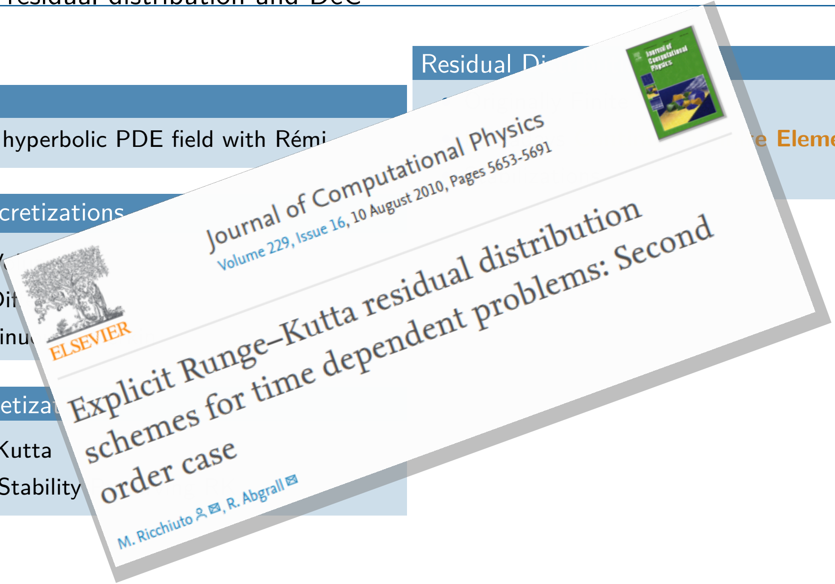
- Finite V
- Finite Di
- Discontinu

Time Discretiza

- Runge Kutta
- Strong Stability

Residual Di

e Element


ELSEVIER
Journal of Computational Physics
Volume 229, Issue 16, 10 August 2010, Pages 5653-5691
Explicit Runge–Kutta residual distribution schemes for time dependent problems: Second order case
M. Ricchiuto, R. Abgrall

History of residual distribution and DeC

2016

- PhD in hyperbolic PDE field with Rémi

Spatial Discretizations

- Finite Volume
- Finite Difference
- Discontinuous Galerkin

Time Discretizations

- Runge Kutta
- Strong Stability Preserving RK

Residual Distributions

- Originally Finite Volume
- Nowadays **Continuous Finite Element**
- Stabilizations

Time Discretization

- Runge-Kutta 2 and Runge-Kutta 3

History of residual distribution and DeC

2016

- PhD in hyperbolic PDE field with Rémi

Spatial Discretizations

- Finite Volume
- Finite Difference
- Discontinuous Galerkin

Time Discretizations

- Runge Kutta
- Strong Stability Preserving RK

Residual Distributions

- Originally Finite Volume
- Nowadays **Continuous Finite Element**
- Stabilizations

Time Discretization

- Runge-Kutta 2 and Runge-Kutta 3
- Mass lumping + Correction

History of residual distribution and DeC

2016

- PhD in hyper

Spatial Discret

- Finite V
- Finite Di
- Discontinu

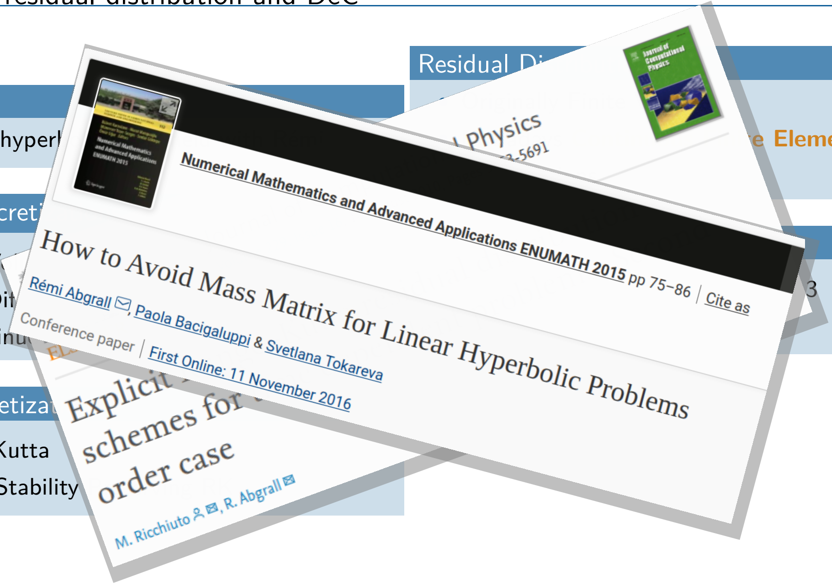
Time Discretiza

- Runge Kutta
- Strong Stability

Residual Di

e Element

3



History of residual distribution and DeC

2016

- PhD in hyper

Spatial Discret

- Finite V
- Finite D
- Disconti

Time Discreti

- Runge Kutt
- Strong Stabi

Residual Di

Element

History of residual distribution and DeC

2016

- PhD in hyperbolic PDE field with Rémi

Spatial Discretizations

- Finite Volume
- Finite Difference
- Discontinuous Galerkin

Time Discretizations

- Runge Kutta
- Strong Stability Preserving RK

Residual Distributions

- Originally Finite Volume
- Nowadays **Continuous Finite Element**
- Stabilizations

Time Discretization

- Runge-Kutta 2 and Runge-Kutta 3
- Mass lumping + Correction

Deferred Correction (January 2017)

- Arbitrarily high order

History of residual distribution and DeC

2016

- PhD in hyperbolic PDE field with Rémi

Spatial Discretizations

- Finite Volume
- Finite Difference
- Discontinuous Galerkin

Time Discretizations

- Runge Kutta
- Strong Stability Preserving RK

Residual Distributions

- Originally Finite Volume
- Nowadays **Continuous Finite Element**
- Stabilizations

Time Discretization

- Runge-Kutta 2 and Runge-Kutta 3
- Mass lumping + Correction

Deferred Correction (January 2017)

- Arbitrarily high order
- Mass lumping + Correction

History of residual distribution and DeC

2016

- PhD in hyperbolic PDE field with Rémi

Spatial Discretizations

- Finite Volume
- Finite Difference
- Discontinuous Galerkin

Time Discretizations

- Runge Kutta
- Strong Stability Preserving RK

Residual Distributions

- Originally Finite Volume
- Nowadays **Continuous Finite Element**
- Stabilizations

Time Discretization

- Runge-Kutta 2 and Runge-Kutta 3
- Mass lumping + Correction

Deferred Correction (January 2017)

- Arbitrarily high order
- Mass lumping + Correction
- General algorithm

History of residual distribution and DeC

2016

- PhD in hyp

Spatial Disc

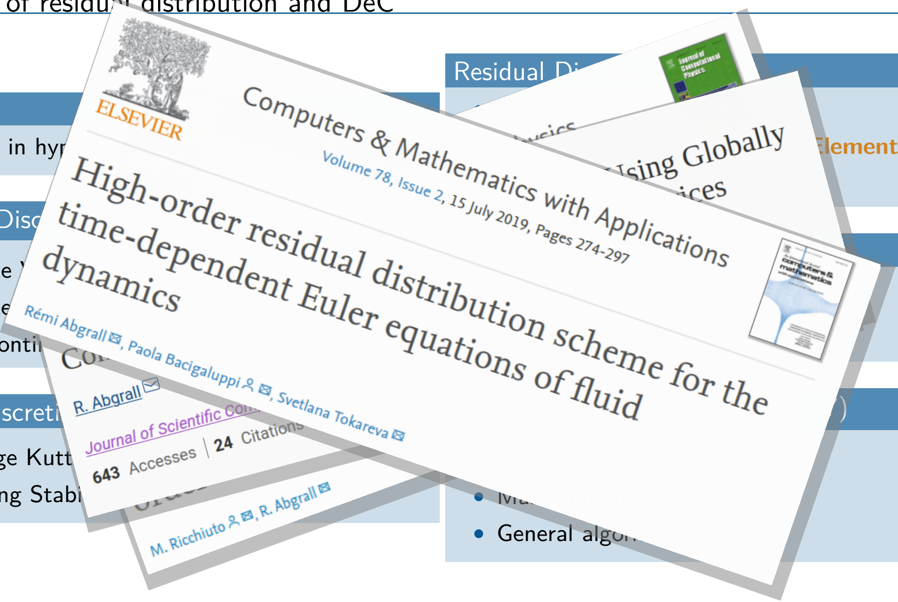
- Finite
- Finite
- Disconti

Time Discreti

- Runge Kutt
- Strong Stabi

Residual Di

- Element
- Ma
- General algo



History of residual distribution and DeC

2016

- PhD in hyp

Spatial Disc

- Finite
- Finite
- Disconti

Time Discreti

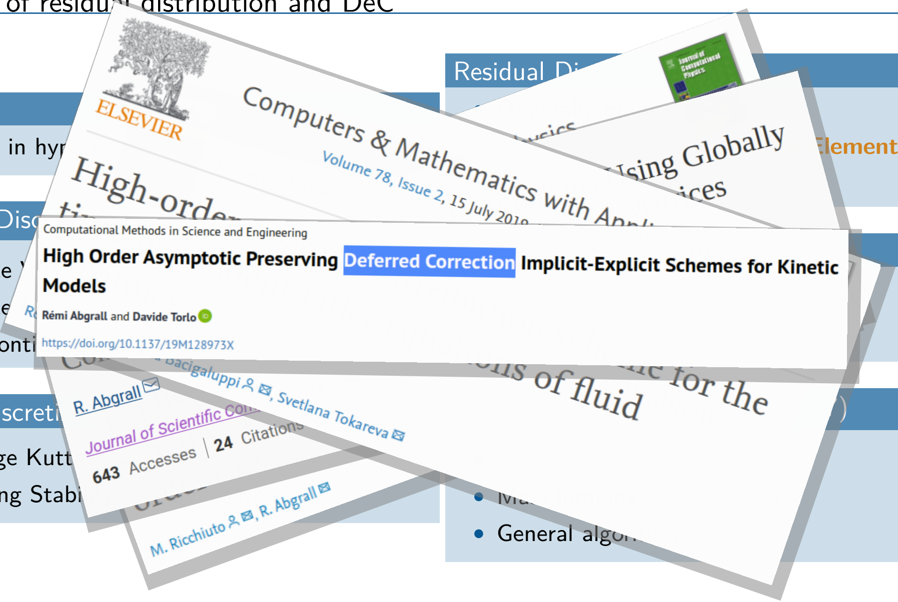
- Runge Kutt
- Strong Stabi

Residual Di

Element

)

- Ma
- General algo



History of residual distribution and DeC

2016
- PhD in hyp

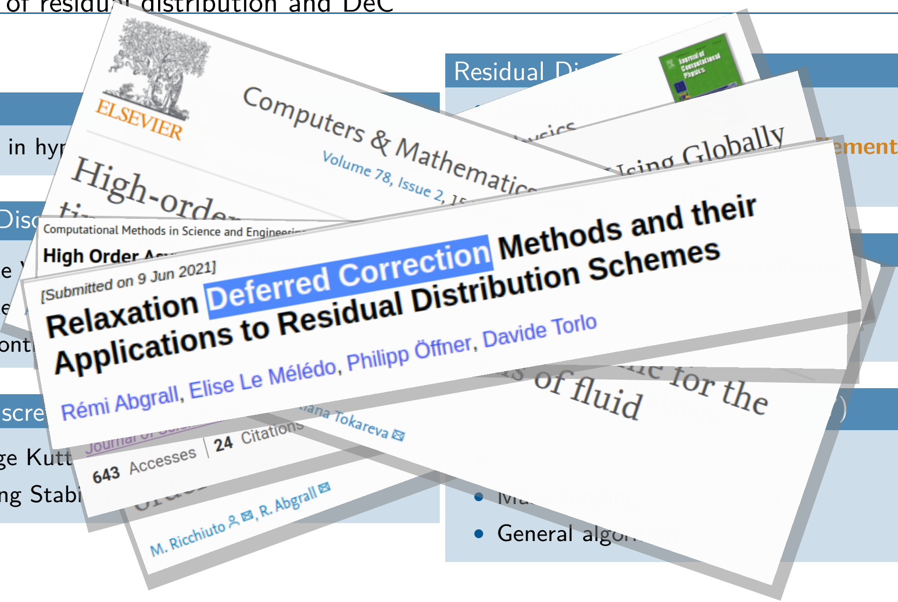
Spatial Disc
- Finite
- Finite
- Discont

Time Discre
- Runge Kutt
- Strong Stabi

Residual Di
ement

)

- General algo



History of residual distribution and DeC

2016
- PhD in h

Spatial Dis
- Finite
- Finite
- Discont

Time Discre
- Runge Kutt
- Strong Stabi

Residual Di

ement

- Ma
- General algo

History of residual distribution and DeC

2016
- PhD in h

Spatial Dis
- Finite
- Finite
- Discon

Time Discr
- Runge K
- Strong Stabi

Residual Di
ment

- Ma
- General algo

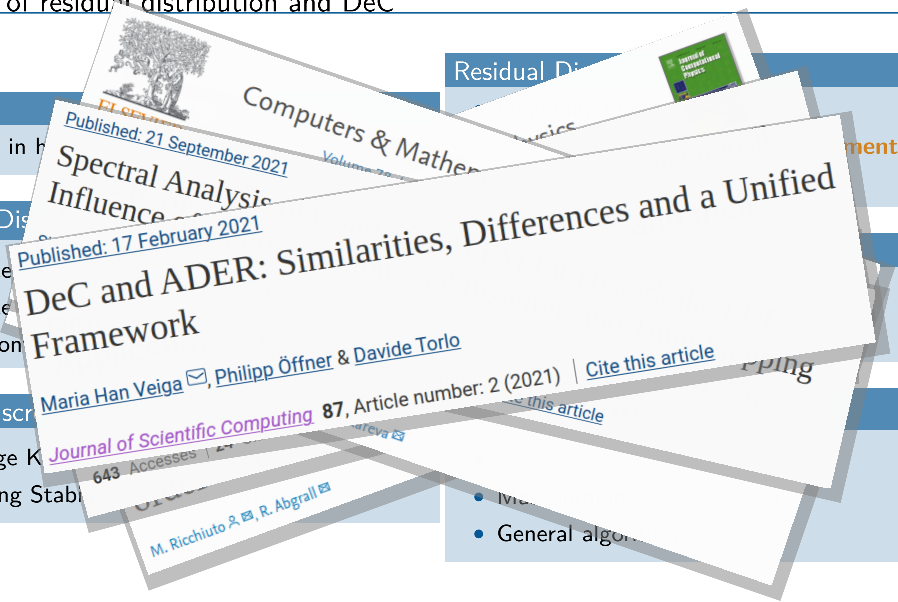


Table of contents

Table of contents

Deferred Correction (DeC)

History of DeC

- Original framework for solution of **nonlinear equations**

Deferred Correction (DeC)

History of DeC

- Original framework for solution of **nonlinear equations**
- Iterative method for **ODEs** with Taylor expansion
  Fox and Goodwin (1949)

Deferred Correction (DeC)

History of DeC

- Original framework for solution of **nonlinear equations**
- Iterative method for **ODEs** with Taylor expansion
  Fox and Goodwin (1949)
- *Pereyra (1968), Frank and Ueberhuber (1977), Stetter (1978), Skeel (1982)*

Deferred Correction (DeC)

History of DeC

- Original framework for solution of **nonlinear equations**
- Iterative method for **ODEs** with Taylor expansion
  Fox and Goodwin (1949)
- *Pereyra (1968), Frank and Ueberhuber (1977), Stetter (1978), Skeel (1982)*
- **Spectral** Deferred Correction (Gauss-Legendre quadrature + Lagrange polynomials) for ODEs.
  Explicit, implicit, stability
  Dutt, Greengard, Rokhlin (2000)

Deferred Correction (DeC)

History of DeC

- Original framework for solution of **nonlinear equations**
- Iterative method for **ODEs** with Taylor expansion
  Fox and Goodwin (1949)
- *Pereyra (1968), Frank and Ueberhuber (1977), Stetter (1978), Skeel (1982)*
- **Spectral** Deferred Correction (Gauss-Legendre quadrature + Lagrange polynomials) for ODEs. Explicit, implicit, stability
  Dutt, Greengard, Rokhlin (2000)
- **IMEX** DeC: *Minion (2003)*

Deferred Correction (DeC)

History of DeC

- Original framework for solution of **nonlinear equations**
- Iterative method for **ODEs** with Taylor expansion
  Fox and Goodwin (1949)
- *Pereyra (1968), Frank and Ueberhuber (1977), Stetter (1978), Skeel (1982)*
- **Spectral** Deferred Correction (Gauss-Legendre quadrature + Lagrange polynomials) for ODEs.
  Explicit, implicit, stability
  Dutt, Greengard, Rokhlin (2000)
- **IMEX** DeC: *Minion (2003)*
- **Operators** based DeC, generalization to many problems: *Abgrall (2017)*

DeC iterations

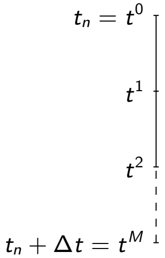
$$\frac{d}{dt}\boldsymbol{u}(t) = \boldsymbol{G}(t, \boldsymbol{u}(t)),$$

DeC iterations

$$\frac{d}{dt}\boldsymbol{u}(t) = \boldsymbol{G}(t, \boldsymbol{u}(t)), \qquad \boldsymbol{u}_n \approx \boldsymbol{u}(t_n)$$

DeC iterations

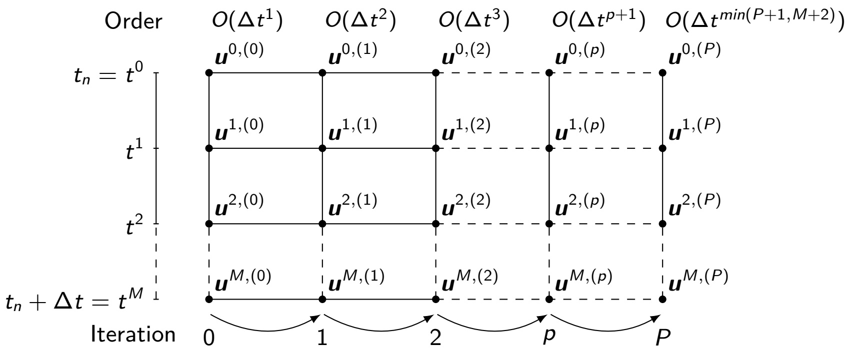
$$\frac{d}{dt}\boldsymbol{u}(t) = \boldsymbol{G}(t, \boldsymbol{u}(t)), \qquad \boldsymbol{u}_n \approx \boldsymbol{u}(t_n), \qquad \boldsymbol{u}^m \approx \boldsymbol{u}(t^m)$$



Which sub time nodes? **Equispaced**, **Gauss-Lobatto**

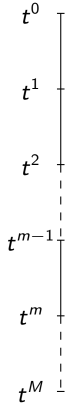
DeC iterations

$$\frac{d}{dt}\boldsymbol{u}(t) = \boldsymbol{G}(t, \boldsymbol{u}(t)), \qquad \boldsymbol{u}_n \approx \boldsymbol{u}(t_n), \qquad \boldsymbol{u}^m \approx \boldsymbol{u}(t^m)$$



Which sub time nodes? **Equispaced**, **Gauss-Lobatto**

DeC operators

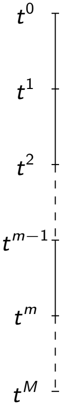


$\mathcal{L}^2_\Delta$ operator

$$\mathcal{L}^2_\Delta(\boldsymbol{u}^0,\dots,\boldsymbol{u}^M)=\mathcal{L}^2_\Delta(\underline{\boldsymbol{u}}):=$$
$$\begin{cases}\boldsymbol{u}^M-\boldsymbol{u}^0-\int_{t^0}^{t^M}\boldsymbol{G}(\boldsymbol{u}(s))ds\\ \vdots\\ \boldsymbol{u}^1-\boldsymbol{u}^0-\int_{t^0}^{t^1}\boldsymbol{G}(\boldsymbol{u}(s))ds\end{cases}$$

- Implicit RK
- Order of accuracy $\geq M+1$
- Difficult to solve directly

DeC operators

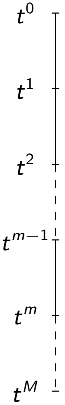


$\mathcal{L}^2_\Delta$ operator

$$\mathcal{L}^2_\Delta(\boldsymbol{u}^0,\dots,\boldsymbol{u}^M) = \mathcal{L}^2_\Delta(\underline{\boldsymbol{u}}) := \begin{cases} \boldsymbol{u}^M - \boldsymbol{u}^0 - \Delta t \sum_{r=0}^{M} \theta_r^M \boldsymbol{G}(\boldsymbol{u}^r) \\ \dots \\ \boldsymbol{u}^1 - \boldsymbol{u}^0 - \Delta t \sum_{r=0}^{M} \theta_r^1 \boldsymbol{G}(\boldsymbol{u}^r) \end{cases}$$

- Implicit RK
- Order of accuracy $\geq M+1$
- Difficult to solve directly

DeC operators



$\mathcal{L}^2_\Delta$ operator

$$\mathcal{L}^2_\Delta(\boldsymbol{u}^0,\dots,\boldsymbol{u}^M)=\mathcal{L}^2_\Delta(\underline{\boldsymbol{u}}):=$$

$$\begin{cases} \boldsymbol{u}^M-\boldsymbol{u}^0-\Delta t\sum_{r=0}^M \theta_r^M \boldsymbol{G}(\boldsymbol{u}^r)\\ \dots\\ \boldsymbol{u}^1-\boldsymbol{u}^0-\Delta t\sum_{r=0}^M \theta_r^1 \boldsymbol{G}(\boldsymbol{u}^r) \end{cases}$$

- Implicit RK
- Order of accuracy $\geq M+1$
- Difficult to solve directly

$\mathcal{L}^1_\Delta$ operator

$$\mathcal{L}^1_\Delta(\boldsymbol{u}^0,\dots,\boldsymbol{u}^M)=\mathcal{L}^1_\Delta(\underline{\boldsymbol{u}}):=$$

$$\begin{cases} \boldsymbol{u}^M-\boldsymbol{u}^0-\Delta t\beta^M \boldsymbol{G}(\boldsymbol{u}^0)\\ \dots\\ \boldsymbol{u}^1-\boldsymbol{u}^0-\Delta t\beta^1 \boldsymbol{G}(\boldsymbol{u}^0) \end{cases}$$

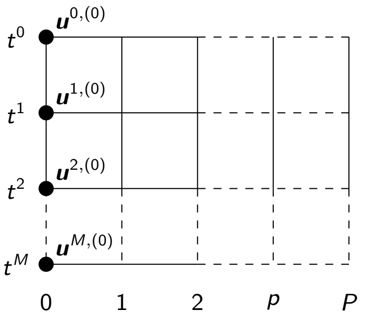
- First order accurate
- Explicit or easy to solve

Deferred Correction

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

$$\boldsymbol{u}^{0,(p)} := \boldsymbol{u}(t_n), \quad p = 0, \dots, P,$$
$$\boldsymbol{u}^{m,(0)} := \boldsymbol{u}(t_n), \quad m = 1, \dots, M$$
$$\mathcal{L}^1_\Delta(\underline{\boldsymbol{u}}^{(p)}) = \mathcal{L}^1_\Delta(\underline{\boldsymbol{u}}^{(p-1)}) - \mathcal{L}^2_\Delta(\underline{\boldsymbol{u}}^{(p-1)}) \text{ with } p = 1, \dots, P.$$

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



DeC Theorem

- $\mathcal{L}^1_\Delta$ coercive
- $\mathcal{L}^1_\Delta - \mathcal{L}^2_\Delta$ Lipschitz

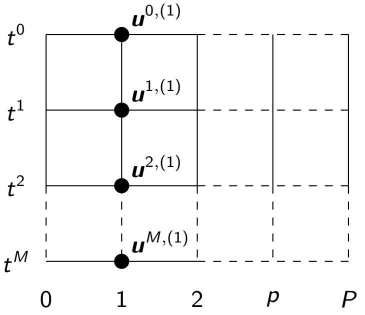
DeC converges and $\min(P, M+1)$ is the order of accuracy.

Deferred Correction

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

$$\boldsymbol{u}^{0,(p)} := \boldsymbol{u}(t_n), \quad p = 0, \dots, P,$$
$$\boldsymbol{u}^{m,(0)} := \boldsymbol{u}(t_n), \quad m = 1, \dots, M$$
$$\mathcal{L}^1_\Delta(\underline{\boldsymbol{u}}^{(p)}) = \mathcal{L}^1_\Delta(\underline{\boldsymbol{u}}^{(p-1)}) - \mathcal{L}^2_\Delta(\underline{\boldsymbol{u}}^{(p-1)}) \text{ with } p = 1, \dots, P.$$

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



DeC Theorem

- $\mathcal{L}^1_\Delta$ coercive
- $\mathcal{L}^1_\Delta - \mathcal{L}^2_\Delta$ Lipschitz

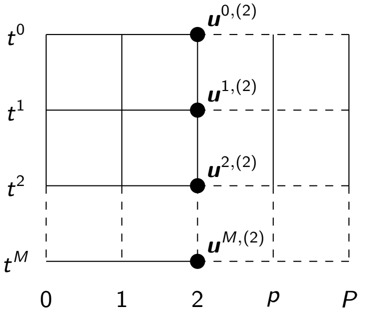
DeC converges and $\min(P, M+1)$ is the order of accuracy.

Deferred Correction

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 $M+1$.

$$\boldsymbol{u}^{0,(p)} := \boldsymbol{u}(t_n), \quad p = 0, \dots, P,$$
$$\boldsymbol{u}^{m,(0)} := \boldsymbol{u}(t_n), \quad m = 1, \dots, M$$
$$\mathcal{L}^1_\Delta(\underline{\boldsymbol{u}}^{(p)}) = \mathcal{L}^1_\Delta(\underline{\boldsymbol{u}}^{(p-1)}) - \mathcal{L}^2_\Delta(\underline{\boldsymbol{u}}^{(p-1)}) \text{ with } p = 1, \dots, P.$$



DeC Theorem

- $\mathcal{L}^1_\Delta$ coercive
- $\mathcal{L}^1_\Delta - \mathcal{L}^2_\Delta$ Lipschitz

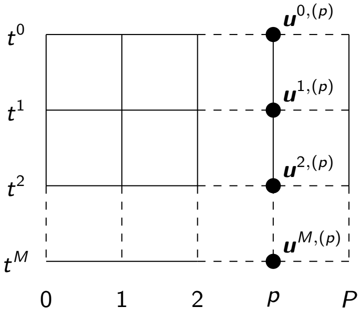
DeC converges and $\min(P, M+1)$ is the order of accuracy.

Deferred Correction

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 $M+1$.

$$\boldsymbol{u}^{0,(p)} := \boldsymbol{u}(t_n), \quad p=0,\dots,P,$$
$$\boldsymbol{u}^{m,(0)} := \boldsymbol{u}(t_n), \quad m=1,\dots,M$$
$$\mathcal{L}^1_\Delta(\underline{\boldsymbol{u}}^{(p)}) = \mathcal{L}^1_\Delta(\underline{\boldsymbol{u}}^{(p-1)}) - \mathcal{L}^2_\Delta(\underline{\boldsymbol{u}}^{(p-1)}) \text{ with } p=1,\dots,P.$$



DeC Theorem

- $\mathcal{L}^1_\Delta$ coercive
- $\mathcal{L}^1_\Delta - \mathcal{L}^2_\Delta$ Lipschitz

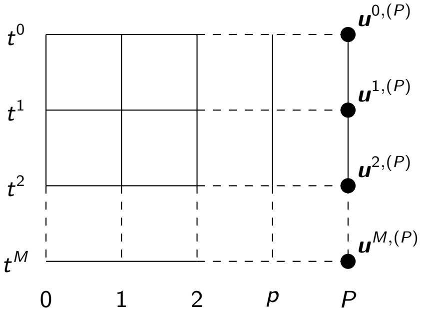
DeC converges and $\min(P, M+1)$ is the order of accuracy.

Deferred Correction

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 $M+1$.

$$\boldsymbol{u}^{0,(p)} := \boldsymbol{u}(t_n), \quad p = 0, \dots, P,$$
$$\boldsymbol{u}^{m,(0)} := \boldsymbol{u}(t_n), \quad m = 1, \dots, M$$
$$\mathcal{L}^1_\Delta(\underline{\boldsymbol{u}}^{(p)}) = \mathcal{L}^1_\Delta(\underline{\boldsymbol{u}}^{(p-1)}) - \mathcal{L}^2_\Delta(\underline{\boldsymbol{u}}^{(p-1)}) \text{ with } p = 1, \dots, P.$$



DeC Theorem

- $\mathcal{L}^1_\Delta$ coercive
- $\mathcal{L}^1_\Delta - \mathcal{L}^2_\Delta$ Lipschitz

DeC converges and $\min(P, M+1)$ is the order of accuracy.

DeC as RK for ODEs

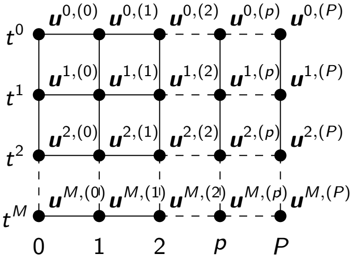
$$\mathcal{L}^1_\Delta(\underline{\boldsymbol{u}}^{(p)}) = \mathcal{L}^1_\Delta(\underline{\boldsymbol{u}}^{(p-1)}) - \mathcal{L}^2_\Delta(\underline{\boldsymbol{u}}^{(p-1)}) \text{ with } p = 1, \dots, P.$$

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

DeC as RK for ODEs

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

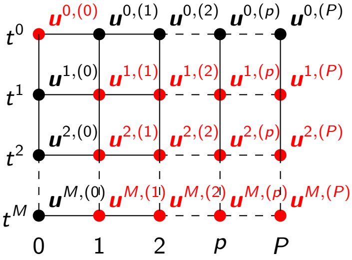
$$\boldsymbol{u}^{m,(p)} = \boldsymbol{u}^0 + \sum_{r=0}^{M} \theta^m_r \boldsymbol{G}(t^r, \boldsymbol{u}^{r,(p-1)}), \qquad \forall m = 1, \dots, M, \; p = 1, \dots, P$$



DeC as RK for ODEs

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

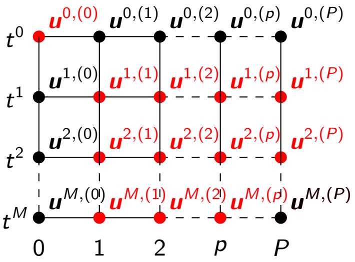
$$\boldsymbol{u}^{m,(p)} = \boldsymbol{u}^0 + \sum_{r=0}^{M} \theta_r^m \boldsymbol{G}(t^r, \boldsymbol{u}^{r,(p-1)}), \qquad \forall m = 1, \dots, M, \; p = 1, \dots, P$$



DeC as RK for ODEs

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

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



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

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

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

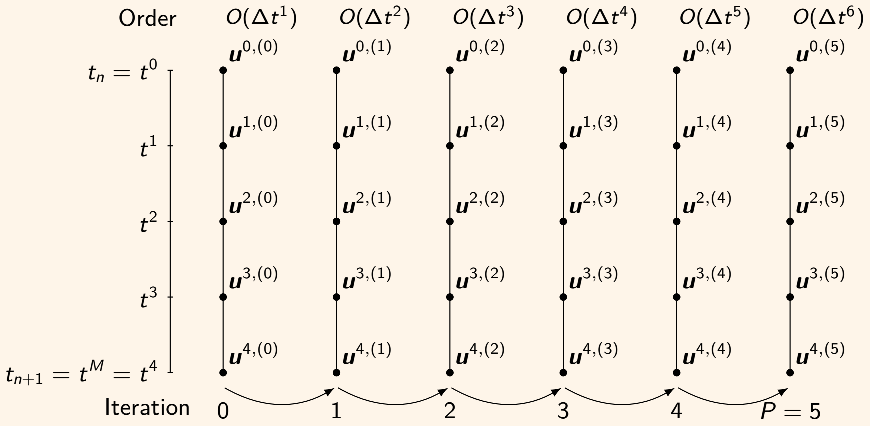
Gauss–Lobatto

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

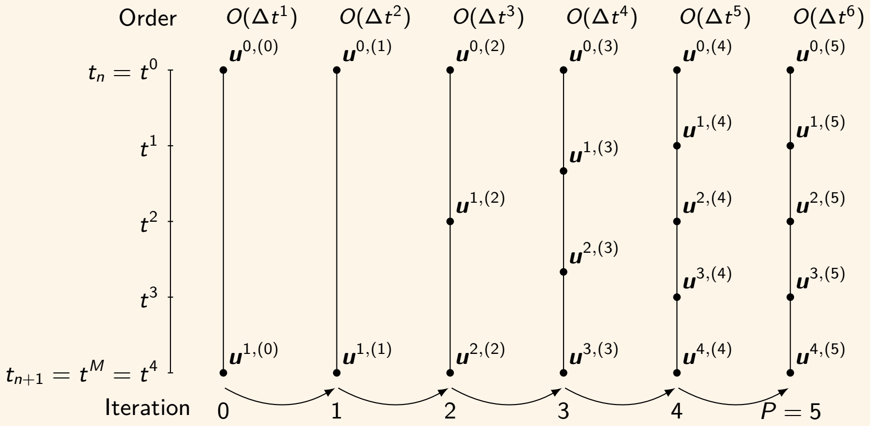
How can we save computational time?

Table of contents

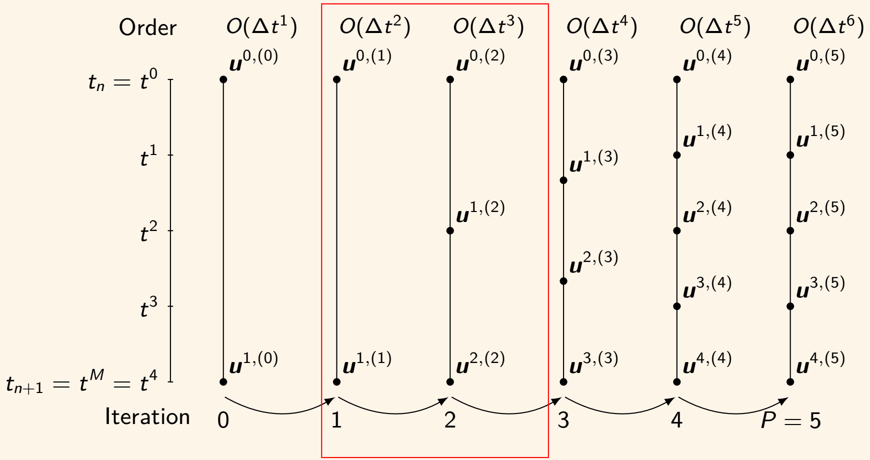
Idea for reduction of stages



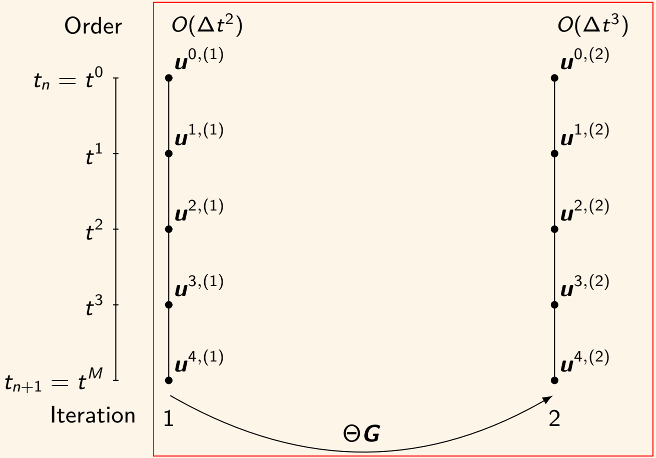
Idea for reduction of stages



Idea for reduction of stages



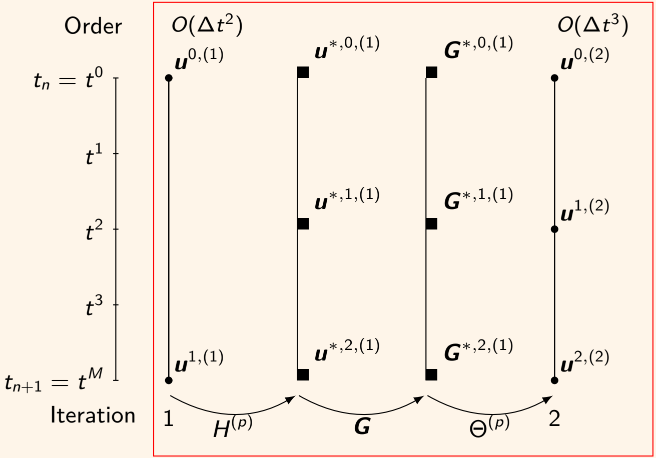
How to communicate between iterations?



DeC

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

How to communicate between iterations?



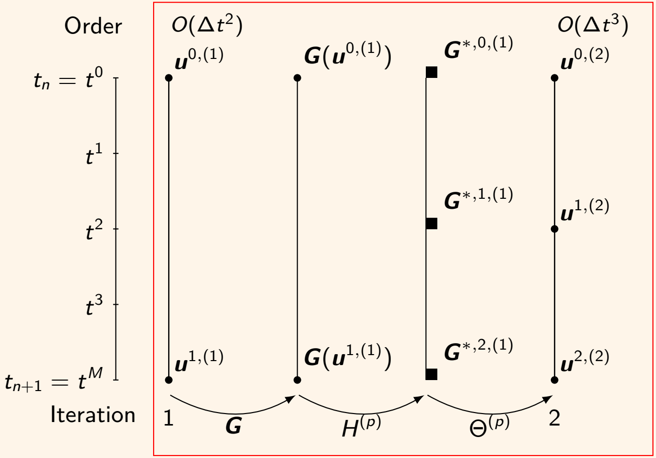
DeC

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

DeCu

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

How to communicate between iterations?



DeC

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

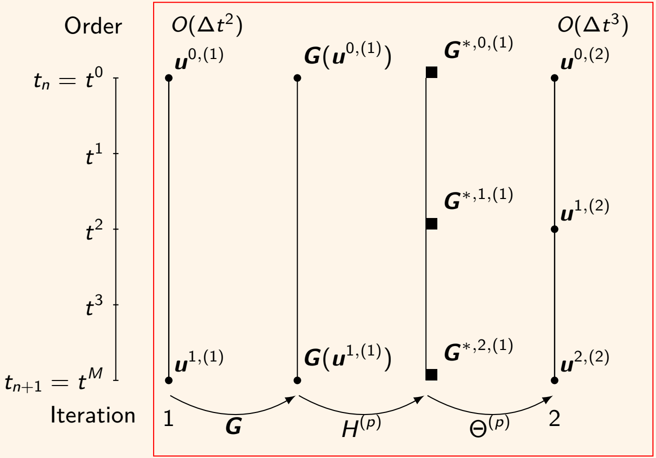
DeCu

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

DeCdu

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

How to communicate between iterations?



DeC

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

DeCu

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

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

DeCdu

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

Efficient DeC into RK framework

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

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

Efficient DeC into RK framework

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

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

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

Efficient DeC into RK framework

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

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

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

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

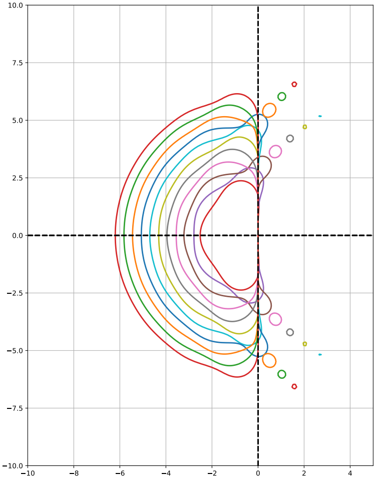
Stability Properties

DeC-DeCu-DeCdu

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

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

DeC, DeCu, DeCdu



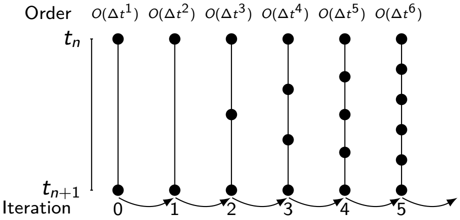
Adaptive DeC

How can we exploit the increasing order of accuracy?

How can we exploit the increasing order of accuracy?

Adaptive order DeC

- Set tolerance ε
- Check at each iteration if $\left\| \underline{\boldsymbol{u}}^{(p)} - \underline{\boldsymbol{u}}^{(p-1)} \right\| < \varepsilon$
- Stop at a certain order when tolerance is reached



Adaptive DeC

How can we exploit the increasing order of accuracy?

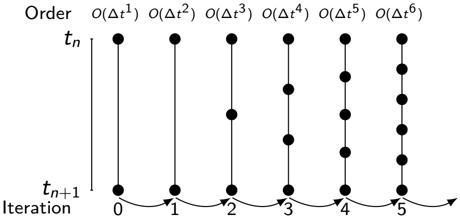
Adaptive order DeC

- Set tolerance ε
- Check at each iteration if $\left\| \underline{\boldsymbol{u}}^{(p)} - \underline{\boldsymbol{u}}^{(p-1)} \right\| < \varepsilon$
- Stop at a certain order when tolerance is reached

☺ Saving on useless iterations

☺ Reach the needed order for tolerance

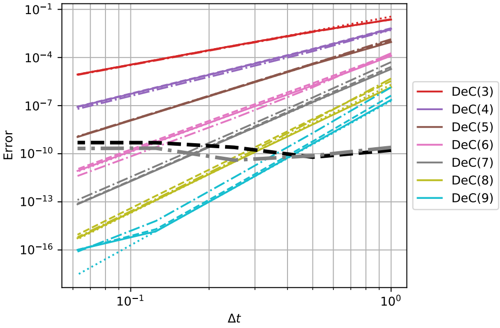
☹ Sub-optimal (waste of few stages)



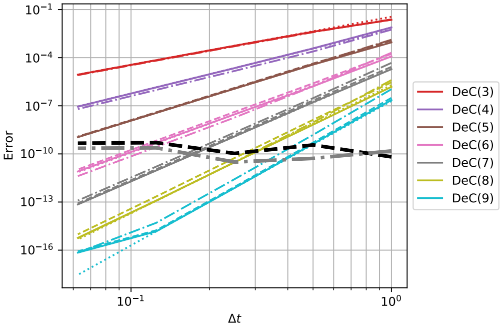
ODE test: Vibrating system

$$my'' + ry' + ky = F\cos(\Omega t + \varphi), \qquad y(0) = A, \qquad y'(0) = B.$$

Equispaced



Gauss-Lobatto

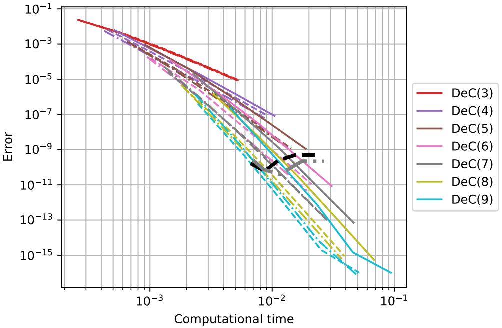


DeC $-$, DeCu $--$, DeCdu $-\cdot-$, adaptive in grey/black

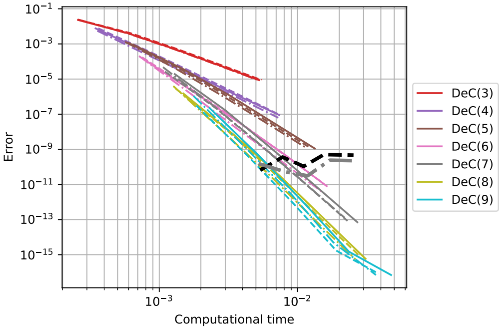
ODE test: Vibrating system

$$my'' + ry' + ky = F\cos(\Omega t + \varphi), \qquad y(0) = A, \qquad y'(0) = B.$$

Equispaced



Gauss-Lobatto

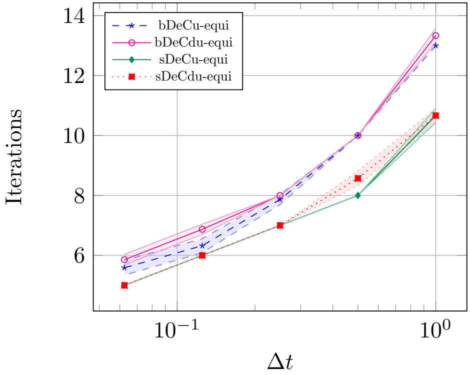


DeC —, DeCu – –, DeCdu – · –, adaptive in grey/black

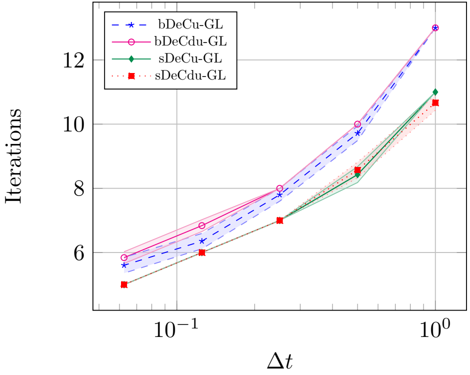
ODE test: Vibrating system

$$my'' + ry' + ky = F\cos(\Omega t + \varphi), \qquad y(0) = A, \qquad y'(0) = B.$$

Equispaced



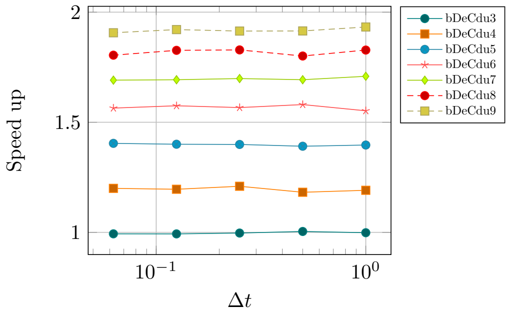
Gauss-Lobatto



ODE test: Vibrating system

$$my'' + ry' + ky = F\cos(\Omega t + \varphi), \qquad y(0) = A, \qquad y'(0) = B.$$

Equispaced



Gauss-Lobatto

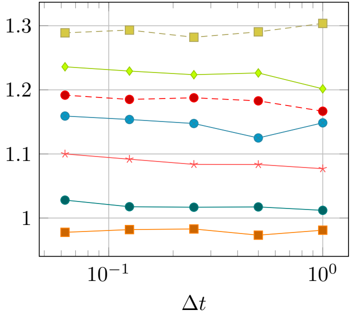


Table of contents

Residual Distributions and DeC

Residual Distribution (RD)

- Originally somehow Finite Volume
- **Finite Element**
- Runge Kutta + Mass matrix correction (Rémi + Mario)
- DeC + RD (Rémi 2017)

$\mathcal{L}^2_\Delta$

$$\mathcal{L}^{2,m}_{\Delta,i}(\boldsymbol{u}) := \int_\Omega \varphi_i \varphi_j dx (u_j^m - u_j^0) + \Delta t \sum_{r=0}^{M} \theta_r^m \sum_K \Phi_K^i(u^r)$$

RD setting

- $\partial_t u + \nabla \cdot F(u) = 0$
- $V_h = \{u \in \mathcal{C}(\Omega) : u|_K \in \mathbb{P}_M\}$
- $\Phi_K(u) = \int_K \nabla \cdot F(u) dx$
- $\Phi_K^i(u) = \int_K \varphi_i(x) \nabla \cdot F(u) dx + \mathrm{ST}_i(u)$
- NOT method of lines

$\mathcal{L}^1_\Delta$

$$\mathcal{L}^{1,m}_{\Delta,i}(\boldsymbol{u}) := \int_\Omega \varphi_i dx (u_i^m - u_i^0) + \Delta t \beta^m \sum_K \Phi_K^i(u^0)$$

DeCu for RD

DeC for RD

$$\underbrace{\int_K \varphi_i dx (u_i^{m,(p)} - u_i^{m,(p-1)})}_{\mathcal{L}^{1,m}_{\Delta,i}(u^{(p)})-\mathcal{L}^{1,m}_{\Delta,i}(u^{(p-1)})} = \underbrace{\int_K \varphi_i \varphi_j dx (u_j^{m,(p)} - u_j^0) + \Delta t \sum_{r=0}^{M} \theta_r^m \sum_K \Phi_K^i (u^{r,(p-1)})}_{\mathcal{L}^{2,m}_{\Delta,i}(u^{(p-1)})}$$

DeCu for RD

DeC for RD

$$\underbrace{\int_K \varphi_i dx (u_i^{m,(p)} - u_i^{m,(p-1)})}_{\mathcal{L}^{1,m}_{\Delta,i}(u^{(p)}) - \mathcal{L}^{1,m}_{\Delta,i}(u^{(p-1)})} = \underbrace{\int_K \varphi_i \varphi_j dx (u_j^{m,(p)} - u_j^0) + \Delta t \sum_{r=0}^M \theta_r^m \sum_K \Phi_K^i (u^{r,(p-1)})}_{\mathcal{L}^{2,m}_{\Delta,i}(u^{(p-1)})}$$

DeCu for RD

$$\underbrace{\int_K \varphi_i dx (u_i^{m,(p)} - u_i^{*,m,(p-1)})}_{\mathcal{L}^{1,m}_{\Delta,i}(u^{(p)}) - \mathcal{L}^{1,m}_{\Delta,i}(u^{*,(p-1)})} = \underbrace{\int_K \varphi_i \varphi_j dx (u_j^{*,m,(p-1)} - u_j^0) + \Delta t \sum_{r=0}^M \theta_r^m \sum_K \Phi_K^i (u^{*,r,(p-1)})}_{\mathcal{L}^{2,m}_{\Delta,i}(u^{*,(p-1)})}$$

DeCu for RD

DeC for RD

$$\underbrace{\int_K \varphi_i dx (u_i^{m,(p)} - u_i^{m,(p-1)})}_{\mathcal{L}^{1,m}_{\Delta,i}(u^{(p)})-\mathcal{L}^{1,m}_{\Delta,i}(u^{(p-1)})} = \underbrace{\int_K \varphi_i \varphi_j dx (u_j^{m,(p)} - u_j^0) + \Delta t \sum_{r=0}^M \theta_r^m \sum_K \Phi_K^i(u^{r,(p-1)})}_{\mathcal{L}^{2,m}_{\Delta,i}(u^{(p-1)})}$$

DeCu for RD

$$\underbrace{\int_K \varphi_i dx (u_i^{m,(p)} - {\color{red}u_i^{*,m,(p-1)}})}_{\mathcal{L}^{1,m}_{\Delta,i}(u^{(p)})-\mathcal{L}^{1,m}_{\Delta,i}({\color{red}u^{*,(p-1)}})} = \underbrace{\int_K \varphi_i \varphi_j dx ({\color{red}u_j^{*,m,(p-1)}} - u_j^0) + \Delta t \sum_{r=0}^M \theta_r^m \sum_K \Phi_K^i({\color{red}u^{*,r,(p-1)}})}_{\mathcal{L}^{2,m}_{\Delta,i}({\color{red}u^{*,(p-1)}})}$$

Computational cost

- Depends on update evaluation, less on flux evaluations

- DeC $C \approx (P-1)M+1$
- DeCu $C \approx (P-1)M+1-\frac{M(M-1)}{2}$

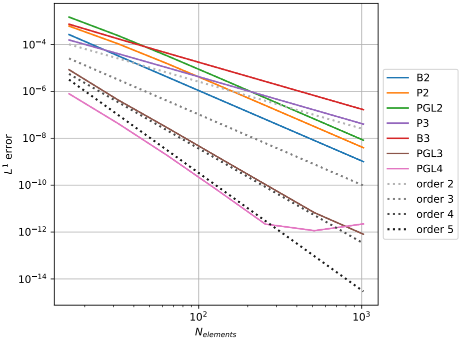
Test PDE: linear advection equation

$$\begin{cases} \partial_t u + \partial_x u = 0 \\ u(0,x) = \cos(2\pi x) \end{cases}$$

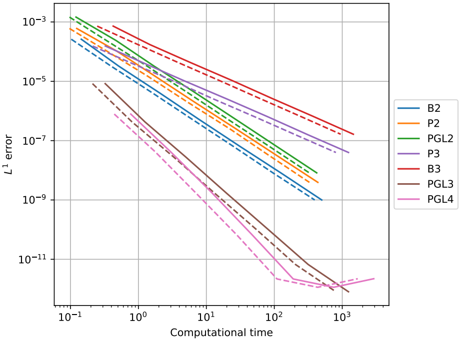
DeC —

DeCu — —

Convergence



Computational Time



Test PDE: linear advection equation

$$\begin{cases} \partial_t u + \partial_x u = 0 \\ u(0,x) = \cos(2\pi x) \end{cases}$$

Speed up

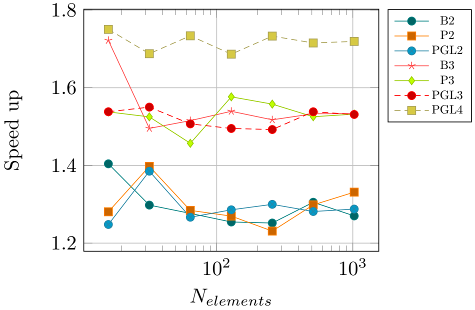


Table of contents

Summary and perspectives

Summary

- DeC
- Efficient DeC
- Increasing order of reconstruction with iterations
- Great Speed up
- Not too big implementation in a DeC code (Remi's birthday present)
- Adaptive with tolerance
- DeC and RD for PDEs

Summary and perspectives

Summary

- DeC
- Efficient DeC
- Increasing order of reconstruction with iterations
- Great Speed up
- Not too big implementation in a DeC code (Remi's birthday present)
- Adaptive with tolerance
- DeC and RD for PDEs

Perspectives

- Increasing spatial discretizations order
- IMEX
- ADER

Summary and perspectives

Summary

- DeC
- Efficient DeC
- Increasing order of reconstruction with iterations
- Great Speed up
- Not too big implementation in a DeC code (Remi's birthday present)
- Adaptive with tolerance
- DeC and RD for PDEs

Perspectives

- Increasing spatial discretizations order
- IMEX
- ADER

THANK YOU!

Preprint

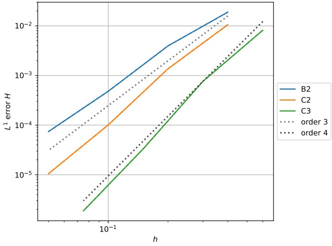
L. Micalizzi, D. Torlo. A new efficient explicit Deferred Correction framework: analysis and applications to hyperbolic PDEs and adaptivity. arXiv:2210.02976.

Test PDE: shallow water equations

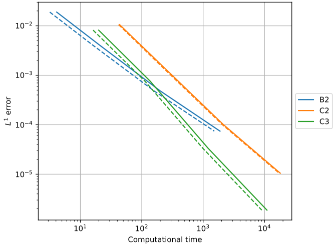
$$\begin{cases} \partial_t \begin{pmatrix} h \\ hu \end{pmatrix} + \partial_x \begin{pmatrix} hu \\ hu^2 + \frac{g}{2} h^2 \end{pmatrix} = 0 \\ \text{IC} = \text{moving vortex} \end{cases}$$

bDeC —
bDeCu – –

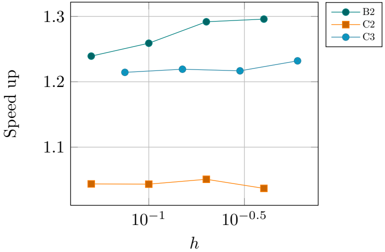
Convergence



Computational Time

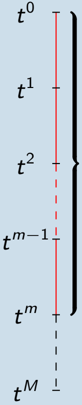


Speed Up



Many DeCs

Which sub-time-interval?



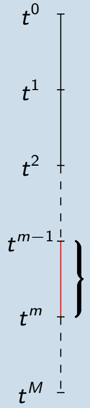
Big DeC (bDeC)

$$\mathcal{L}^2_\Delta(\boldsymbol{u}^0,\dots,\boldsymbol{u}^M)=\mathcal{L}^2_\Delta(\underline{\boldsymbol{u}}):=$$

$$\begin{cases} \boldsymbol{u}^M-\boldsymbol{u}^0-\int_{t^0}^{t^M}\boldsymbol{G}(\boldsymbol{u}(s))ds \\ \vdots \\ \boldsymbol{u}^m-\boldsymbol{u}^0-\int_{t^0}^{t^m}\boldsymbol{G}(\boldsymbol{u}(s))ds \\ \vdots \\ \boldsymbol{u}^1-\boldsymbol{u}^0-\int_{t^0}^{t^1}\boldsymbol{G}(\boldsymbol{u}(s))ds \end{cases}$$

Parallel subtimesteps

Which sub-time-interval?



Small DeC (sDeC)

$$\mathcal{L}^2_\Delta(\boldsymbol{u}^0,\dots,\boldsymbol{u}^M)=\mathcal{L}^2_\Delta(\underline{\boldsymbol{u}}):=$$

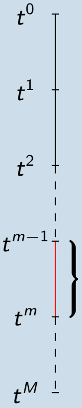
$$\begin{cases} \boldsymbol{u}^M-\boldsymbol{u}^{M-1}-\int_{t^{M-1}}^{t^M}\boldsymbol{G}(\boldsymbol{u}(s))ds\\ \vdots\\ \boldsymbol{u}^m-\boldsymbol{u}^{m-1}-\int_{t^{m-1}}^{t^m}\boldsymbol{G}(\boldsymbol{u}(s))ds\\ \vdots\\ \boldsymbol{u}^1-\boldsymbol{u}^0-\int_{t^0}^{t^1}\boldsymbol{G}(\boldsymbol{u}(s))ds \end{cases}$$

Serial subtimesteps

More accurate

Many DeCs

Which sub-time-interval?

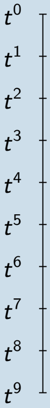


Small DeC (sDeC)

$$\mathcal{L}^2_\Delta(\boldsymbol{u}^0,\dots,\boldsymbol{u}^M)=\mathcal{L}^2_\Delta(\underline{\boldsymbol{u}}):=\begin{cases}\boldsymbol{u}^M-\boldsymbol{u}^{M-1}-\int_{t^{M-1}}^{t^M}\boldsymbol{G}(\boldsymbol{u}(s))ds\\ \vdots\\ \boldsymbol{u}^m-\boldsymbol{u}^{m-1}-\int_{t^{m-1}}^{t^m}\boldsymbol{G}(\boldsymbol{u}(s))ds\\ \vdots\\ \boldsymbol{u}^1-\boldsymbol{u}^0-\int_{t^0}^{t^1}\boldsymbol{G}(\boldsymbol{u}(s))ds\end{cases}$$

- Serial subtimesteps
- More accurate

Which sub-time-nodes?

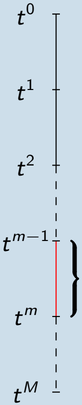


Equispaced

- Easy to implement
- Accuracy $M+1$

Many DeCs

Which sub-time-interval?



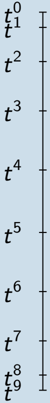
Small DeC (sDeC)

$$\mathcal{L}^2_\Delta(\boldsymbol{u}^0,\dots,\boldsymbol{u}^M) = \mathcal{L}^2_\Delta(\underline{\boldsymbol{u}}) :=$$

$$\begin{cases} \boldsymbol{u}^M - \boldsymbol{u}^{M-1} - \int_{t^{M-1}}^{t^M} \boldsymbol{G}(\boldsymbol{u}(s))ds \\ \vdots \\ \boldsymbol{u}^m - \boldsymbol{u}^{m-1} - \int_{t^{m-1}}^{t^m} \boldsymbol{G}(\boldsymbol{u}(s))ds \\ \vdots \\ \boldsymbol{u}^1 - \boldsymbol{u}^0 - \int_{t^0}^{t^1} \boldsymbol{G}(\boldsymbol{u}(s))ds \end{cases}$$

- ☹ Serial subtimesteps
- ☺ More accurate

Which sub-time-nodes?



Equispaced

- ☺ Easy to implement
- ☹ Accuracy $M+1$

Gauss-Lobatto

- ☺ Accuracy $2M$
- ☹ Less standard polynomials in time