Stabilized High-Order Multirate Time-Integration for Multiphysics via Paired-Explicit Runge-Kutta Methods

SIAM CSE 2025

Daniel Doehring[1]

in collaboration with

Michael Schlottke-Lakemper[2], Manuel Torrilhon[1], and Gregor Gassner[3]

[1]**ACoM, RWTH Aachen University** [2]HPSC, University of Augsburg [3]CDS, University of Cologne

2025/03/05

Applied and Computational Mathematics

Outline

Stabilized High-Order Multirate Time-Integration for Multiphysics via Paired-Explicit Runge-Kutta Methods

Scope

Consider PDEs of the form

$$\partial_t \mathbf{u}(t, \mathbf{x}) + \nabla \cdot \mathbf{f}\big(\mathbf{u}, \nabla \mathbf{u}\big) + \mathbf{g}(\mathbf{u}, \nabla \mathbf{u}) = \mathbf{s}\big(t, \mathbf{x}; \mathbf{u}\big)$$

which describe e.g. compressible flows and related systems.

Scope

Consider PDEs of the form

$$\partial_t \mathbf{u}(t, \mathbf{x}) + \nabla \cdot \mathbf{f}\big(\mathbf{u}, \nabla \mathbf{u}\big) + \mathbf{g}(\mathbf{u}, \nabla \mathbf{u}) = \mathbf{s}\big(t, \mathbf{x}; \mathbf{u}\big)$$

which describe e.g. compressible flows and related systems.
We employ a method-of-lines approach (via the DGSEM) to obtain

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{U}(t) = \mathbf{F}\big(t, \mathbf{U}(t)\big)$$

Scope

Consider PDEs of the form

$$\partial_t \mathbf{u}(t, \mathbf{x}) + \nabla \cdot \mathbf{f}\big(\mathbf{u}, \nabla \mathbf{u}\big) + \mathbf{g}(\mathbf{u}, \nabla \mathbf{u}) = \mathbf{s}\big(t, \mathbf{x}; \mathbf{u}\big)$$

which describe e.g. compressible flows and related systems.
We employ a method-of-lines approach (via the DGSEM) to obtain

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{U}(t) = \mathbf{F}\big(t, \mathbf{U}(t)\big)$$

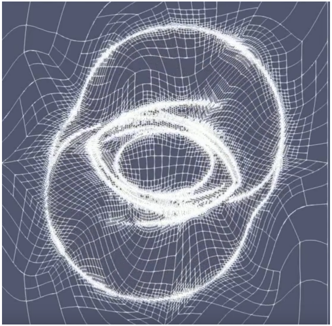
and target **convection-dominated** PDEs, i.e.,

$$\rho \sim \frac{|a|}{\Delta x} \gg \frac{d}{\Delta x^2}$$

where ρ is the spectral radius of

$$J_{\mathbf{F}}(\mathbf{U}) := \frac{\partial \mathbf{F}}{\partial \mathbf{U}}\bigg|_{\mathbf{U}}.$$

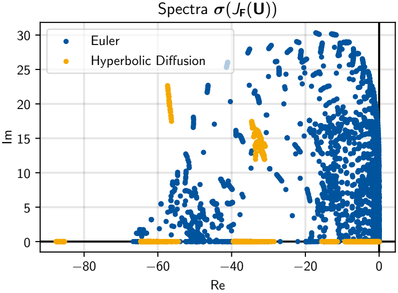
Motivation



Explicit time-integration for

1. non-uniform meshes

2. coupled systems



with **global** time-stepping leaves room for efficiency gains!

Ingredient #1: Optimized Stability Polynomials

Objective:[1]

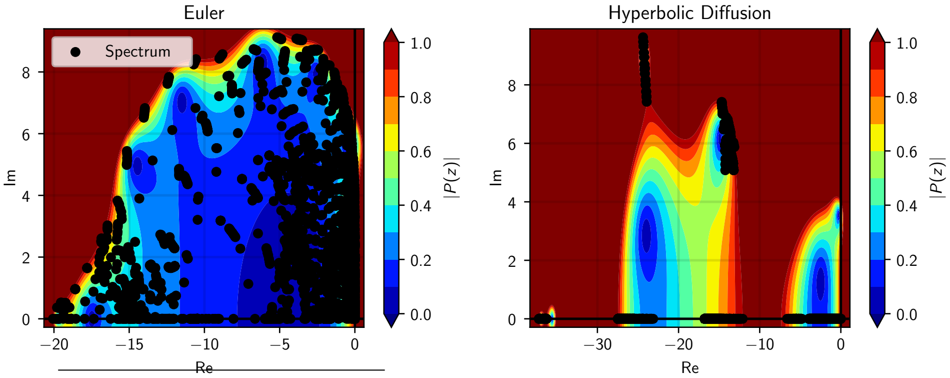
$$\max_{P_{p;S} \in \mathcal{P}_{p;S}} \Delta t \text{ such that } \left| P_{p;S}(\Delta t \lambda_m) \right| \leq 1, \quad \lambda_m \in \boldsymbol{\sigma}\left(J_{\mathbf{F}}(\mathbf{U}_0)\right)$$

[1] Ketcheson & Ahmadia; CAMCoS; 2012

Ingredient #1: Optimized Stability Polynomials

Objective:[1]

$$\max_{P_{p;S} \in \mathcal{P}_{p;S}} \Delta t \text{ such that } \left|P_{p;S}(\Delta t \lambda_m)\right| \leq 1, \quad \lambda_m \in \boldsymbol{\sigma}\left(J_{\mathbf{F}}(\mathbf{U}_0)\right)$$

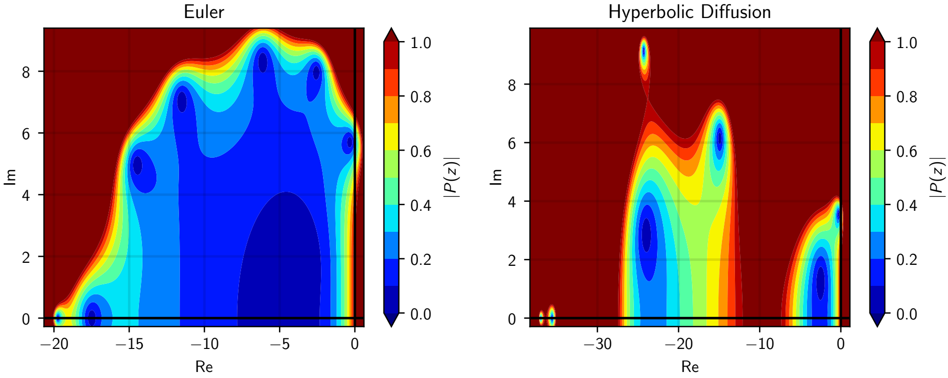


[1] Ketcheson & Ahmadia; CAMCoS; 2012

Ingredient #1: Optimized Stability Polynomials

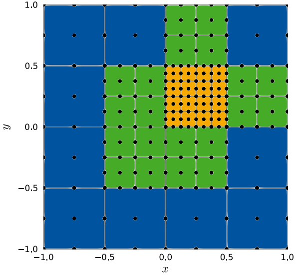
Objective:

$$\max_{P_{p;S} \in \mathcal{P}_{p;S}} \Delta t \text{ such that } \left| P_{p;S}(\Delta t \lambda_m) \right| \leq 1, \quad \lambda_m \in \boldsymbol{\sigma}\left(J_{\mathbf{F}}(\mathbf{U}_0)\right)$$



Ingredient #2: Partitioned RKMs

Multirate time-integration by using **stabilized/optimized** schemes in regions with higher characteristic speeds (stricter CFL)

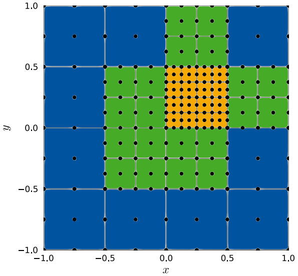


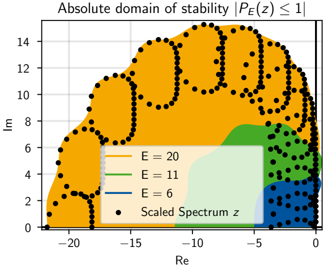
- For *convection-dominated* problems:

$$\Delta t \overset{!}{\leq} C_t(S) \cdot C_x(k) \min_{i,j} \frac{\Delta x_i}{|\mu_j(\mathbf{u}_i)|}, \qquad \mu_j \in \boldsymbol{\sigma}\left(\frac{\partial \mathbf{f}_j}{\partial \mathbf{u}}\right)$$

Ingredient #2: Partitioned RKMs

Multirate time-integration by using **stabilized/optimized** schemes in regions with higher characteristic speeds (stricter CFL)





- For *convection-dominated* problems:

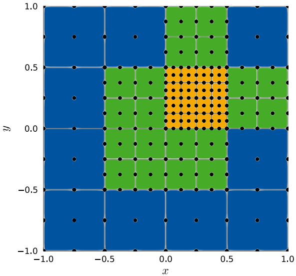
$$\Delta t \overset{!}{\leq} C_t(S) \cdot C_x(k) \min_{i,j} \frac{\Delta x_i}{|\mu_j(\mathbf{u}_i)|}, \qquad \mu_j \in \boldsymbol{\sigma}\left(\frac{\partial \mathbf{f}_j}{\partial \mathbf{u}}\right)$$

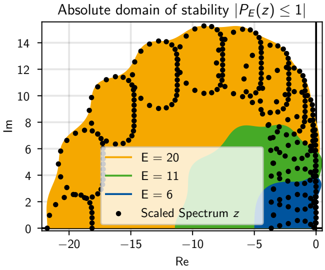
- For *optimized* Runge-Kutta schemes:

$$\Delta t_{p;S} \overset{S \to \infty}{\simeq} \Delta t_{p;S_0} \frac{S}{S_0}$$

Ingredient #2: Partitioned RKMs

Multirate time-integration by using **stabilized/optimized** schemes in regions with higher characteristic speeds (stricter CFL)





- For *convection-dominated* problems:

$$\Delta t \overset{!}{\leq} C_t(S) \cdot C_x(k) \min_{i,j} \frac{\Delta x_i}{|\mu_j(\mathbf{u}_i)|}, \qquad \mu_j \in \boldsymbol{\sigma}\left(\frac{\partial \mathbf{f}_j}{\partial \mathbf{u}}\right)$$

- For *optimized* Runge-Kutta schemes:

$$\Delta t_{p;S} \overset{S\to\infty}{\simeq} \Delta t_{p;S_0} \frac{S}{S_0}$$

Ingredient #2: Partitioned RKMs

Partition ODE according to characteristic speeds $a_i := \min_j \frac{\Delta x_i}{|\mu_j(\mathbf{u}_i)|}$

$$\mathbf{U}'(t) = \begin{pmatrix} \mathbf{U}^{(1)}(t) \\ \vdots \\ \mathbf{U}^{(R)}(t) \end{pmatrix}' = \begin{pmatrix} \mathbf{F}^{(1)}\left(t, \mathbf{U}^{(1)}(t), \dots, \mathbf{U}^{(R)}(t)\right) \\ \vdots \\ \mathbf{F}^{(R)}\left(t, \mathbf{U}^{(1)}(t), \dots, \mathbf{U}^{(R)}(t)\right) \end{pmatrix} = \mathbf{F}\big(t, \mathbf{U}(t)\big)$$

Ingredient #2: Partitioned RKMs

Partition ODE according to characteristic speeds $a_i := \min_j \frac{\Delta x_i}{|\mu_j(\mathbf{u}_i)|}$

$$\mathbf{U}'(t) = \begin{pmatrix} \mathbf{U}^{(1)}(t) \\ \vdots \\ \mathbf{U}^{(R)}(t) \end{pmatrix}' = \begin{pmatrix} \mathbf{F}^{(1)}\left(t, \mathbf{U}^{(1)}(t), \dots, \mathbf{U}^{(R)}(t)\right) \\ \vdots \\ \mathbf{F}^{(R)}\left(t, \mathbf{U}^{(1)}(t), \dots, \mathbf{U}^{(R)}(t)\right) \end{pmatrix} = \mathbf{F}\left(t, \mathbf{U}(t)\right)$$

and employ different RKMs

$$\mathbf{K}_i^{(r)} = \mathbf{F}^{(r)}\left(t_n + c_i \Delta t, \mathbf{U}_n + \Delta t \sum_{j=1}^{S} \sum_{k=1}^{R} a_{i,j}^{(k)} \mathbf{K}_j^{(k)}\right),$$

$$\mathbf{U}_{n+1} = \mathbf{U}_n + \Delta t \sum_{i=1}^{S} b_i \sum_{k=1}^{R} \mathbf{K}_i^{(r)}$$

optimized for the spectrum $\boldsymbol{\sigma}\left(\frac{\partial \mathbf{F}}{\partial \mathbf{U}}\Big|_{\mathbf{U}_0}\right)$. *(Ingredient #1)*

Ingredient #2: Partitioned/Paired RKMs

- $S = 6,\ p = 2$ **Paired**[1] $E = \{3, 6\}$ RK

i	$\mathbf{c}$	$A^{(1)}$						$A^{(2)}$					
1	0												
2	$1/10$	$1/10$						$1/10$					
3	$2/10$	$2/10$	0					$2/10 - a_{3,2}^{(2)}$	$a_{3,2}^{(2)}$				
4	$3/10$	$3/10$	0	0				$3/10 - a_{4,3}^{(2)}$	0	$a_{4,3}^{(2)}$			
5	$4/10$	$4/10$	0	0	0			$4/10 - a_{5,4}^{(2)}$	0	0	$a_{5,4}^{(2)}$		
6	$5/10$	$5/10 - a_{6,5}^{(1)}$	0	0	0	$a_{6,5}^{(1)}$		$5/10 - a_{6,4}^{(2)}$	0	0	0	$a_{6,4}^{(2)}$	
	$\mathbf{b}^T$	0	0	0	0	0	1	0	0	0	0	0	1

[1] Vermeire; JCP; 2019

Ingredient #2: Partitioned/Paired RKMs

- $S = 6,\ p = 2$ **Paired**[1] $E = \{3, 6\}$ RK

i	$\mathbf{c}$	$A^{(1)}$						$A^{(2)}$					
1	0												
2	$1/10$	$1/10$						$1/10$					
3	$2/10$	$2/10$	0					$2/10 - a^{(2)}_{3,2}$	$a^{(2)}_{3,2}$				
4	$3/10$	$3/10$	0	0				$3/10 - a^{(2)}_{4,3}$	0	$a^{(2)}_{4,3}$			
5	$4/10$	$4/10$	0	0	0			$4/10 - a^{(2)}_{5,4}$	0	0	$a^{(2)}_{5,4}$		
6	$5/10$	$5/10 - a^{(1)}_{6,5}$	0	0	0	$a^{(1)}_{6,5}$		$5/10 - a^{(2)}_{6,4}$	0	0	0	$a^{(2)}_{6,4}$	
	$\mathbf{b}^T$	0	0	0	0	0	1	0	0	0	0	0	1

- Method $r = 1$ requires $\mathbf{F}^{(1)}$ *evaluation only for* $i = \mathbf{1, 5, 6}$

[1] Vermeire; JCP; 2019

Ingredient #2: Partitioned/Paired RKMs

- $S = 6,\ p = 2$ **Paired**[1] $E = \{3, 6\}$ RK

i	$\mathbf{c}$	$A^{(1)}$						$A^{(2)}$					
1	0												
2	$1/10$	$1/10$						$1/10$					
3	$2/10$	$2/10$	0					$2/10 - a^{(2)}_{3,2}$	$a^{(2)}_{3,2}$				
4	$3/10$	$3/10$	0	0				$3/10 - a^{(2)}_{4,3}$	0	$a^{(2)}_{4,3}$			
5	$4/10$	$4/10$	0	0	0			$4/10 - a^{(2)}_{5,4}$	0	0	$a^{(2)}_{5,4}$		
6	$5/10$	$5/10 - a^{(1)}_{6,5}$	0	0	0	$a^{(1)}_{6,5}$		$5/10 - a^{(2)}_{6,4}$	0	0	0	$a^{(2)}_{6,4}$	
	$\mathbf{b}^T$	0	0	0	0	0	1	0	0	0	0	0	1

- Method $r = 1$ requires $\mathbf{F}^{(1)}$ *evaluation only for* $i = \mathbf{1}, \mathbf{5}, \mathbf{6}$
- Method $r = 2$ requires $\mathbf{F}^{(2)}$ *evaluation for all* $i = \mathbf{1}, \dots, \mathbf{6}$

[1] Vermeire; JCP; 2019

Application: Euler-Gravity

Euler (2D)

$$\partial_t \begin{pmatrix} \rho \\ \rho v_x \\ \rho v_y \\ E \end{pmatrix} + \nabla \cdot \begin{pmatrix} \rho v_x & \rho v_y \\ \rho v_x^2 + p & \rho v_x v_y \\ \rho v_x v_y & \rho v_y^2 + p \\ (E+p) v_x & (E+p) v_y \end{pmatrix}$$

[1] Schlottke-Lakemper et al.; JCP; 2021

Application: Euler-Gravity

Euler (2D)

$$\partial_t \begin{pmatrix} \rho \\ \rho v_x \\ \rho v_y \\ E \end{pmatrix} + \nabla \cdot \begin{pmatrix} \rho v_x & \rho v_y \\ \rho v_x^2 + p & \rho v_x v_y \\ \rho v_x v_y & \rho v_y^2 + p \\ (E+p) v_x & (E+p) v_y \end{pmatrix}$$

\+ a **self-gravitational potential** ϕ

$$-\Delta \phi = -4\pi G \rho$$

[1] Schlottke-Lakemper et al.; JCP; 2021

Application: Euler-Gravity

Euler (2D)

$$\partial_t \begin{pmatrix} \rho \\ \rho v_x \\ \rho v_y \\ E \end{pmatrix} + \nabla \cdot \begin{pmatrix} \rho v_x & \rho v_y \\ \rho v_x^2 + p & \rho v_x v_y \\ \rho v_x v_y & \rho v_y^2 + p \\ (E+p) v_x & (E+p) v_y \end{pmatrix} = \begin{pmatrix} 0 \\ -\rho \partial_x \phi \\ -\rho \partial_y \phi \\ -\rho \mathbf{v} \cdot \nabla \phi \end{pmatrix}$$

+ a **self-gravitational potential** ϕ

$$-\Delta \phi = -4\pi G \rho$$

which *couples* to the momentum and energy equations[1].

[1] Schlottke-Lakemper et al.; JCP; 2021

Application: Euler-Gravity

Euler (2D) (hyperbolic)

$$\partial_t \begin{pmatrix} \rho \\ \rho v_x \\ \rho v_y \\ E \end{pmatrix} + \nabla \cdot \begin{pmatrix} \rho v_x & \rho v_y \\ \rho v_x^2 + p & \rho v_x v_y \\ \rho v_x v_y & \rho v_y^2 + p \\ (E+p) v_x & (E+p) v_y \end{pmatrix} = \begin{pmatrix} 0 \\ -\rho \partial_x \phi \\ -\rho \partial_y \phi \\ -\rho \mathbf{v} \cdot \nabla \phi \end{pmatrix}$$

\+ a **self-gravitational potential** ϕ (elliptic)

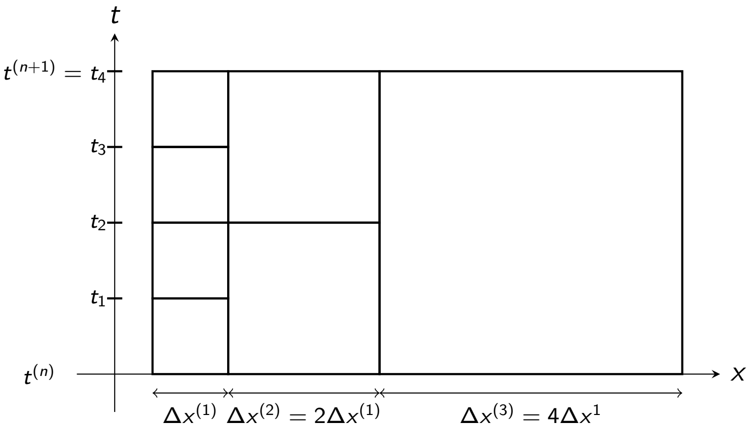
$$-\Delta \phi = -4\pi G \rho$$

which *couples* to the momentum and energy equations[1].

[1] Schlottke-Lakemper et al.; JCP; 2021

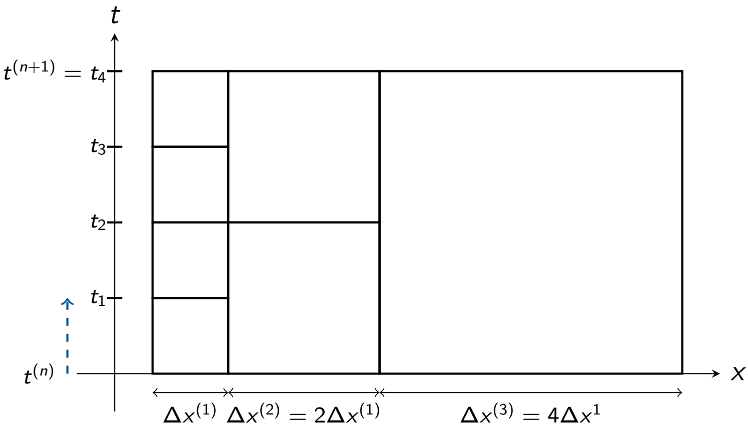
Application: Euler-Gravity

- **Hyperbolic – Elliptic coupling** for *Local Time-Stepping*



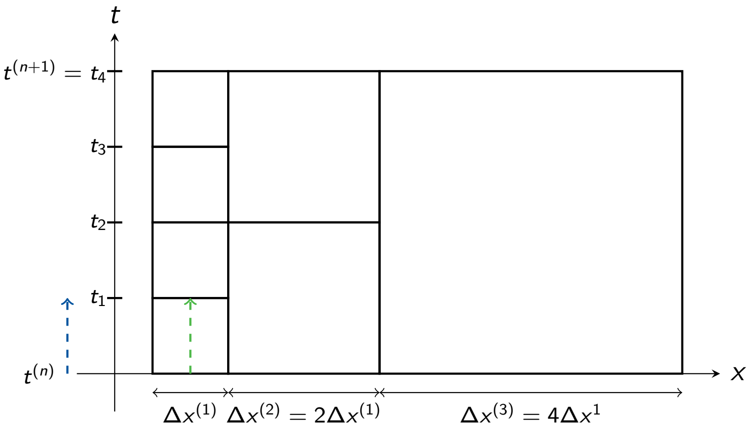
Application: Euler-Gravity

- **Hyperbolic – Elliptic coupling** for *Local Time-Stepping*



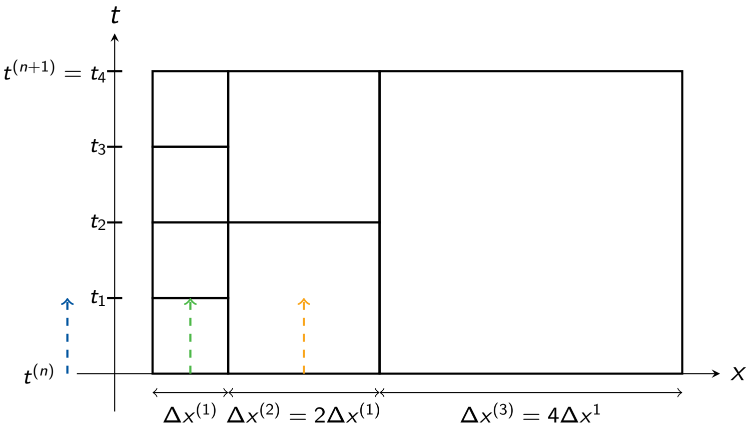
Application: Euler-Gravity

- **Hyperbolic – Elliptic coupling** for *Local Time-Stepping*



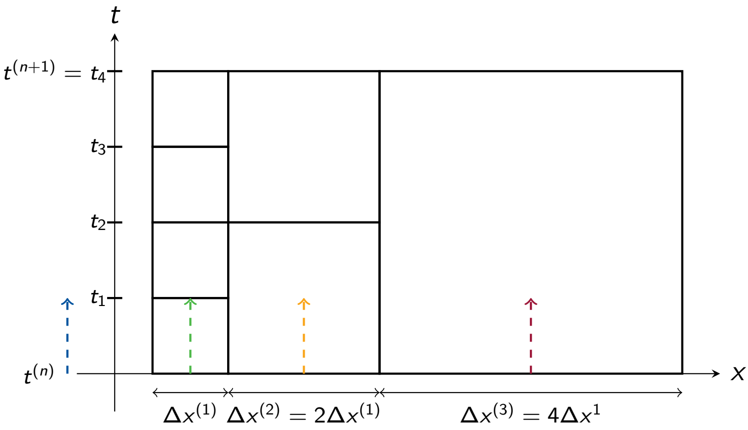
Application: Euler-Gravity

- **Hyperbolic – Elliptic coupling** for *Local Time-Stepping*



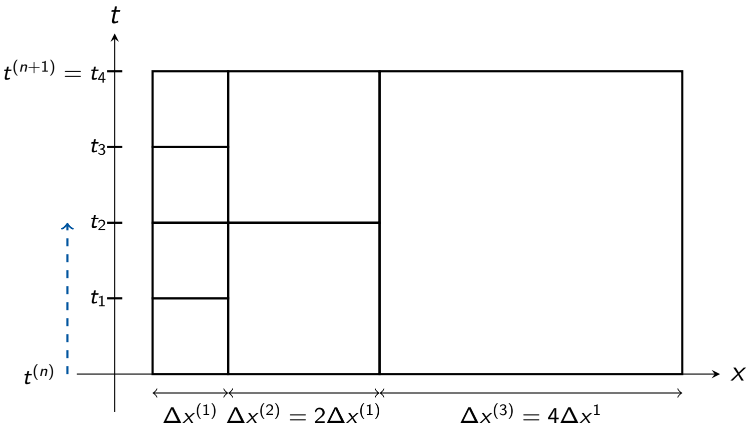
Application: Euler-Gravity

- **Hyperbolic – Elliptic coupling** for *Local Time-Stepping*



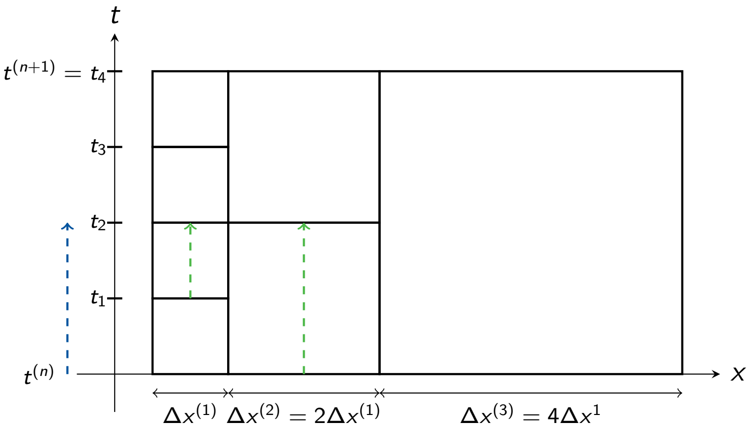
Application: Euler-Gravity

- **Hyperbolic – Elliptic coupling** for *Local Time-Stepping*



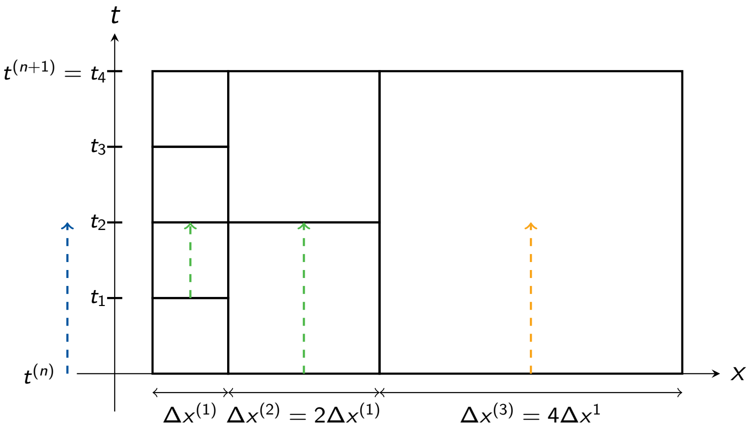
Application: Euler-Gravity

- **Hyperbolic – Elliptic coupling** for *Local Time-Stepping*



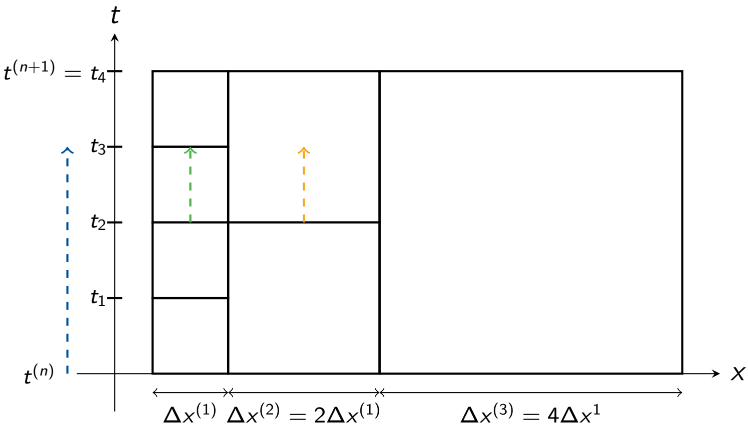
Application: Euler-Gravity

- **Hyperbolic – Elliptic coupling** for *Local Time-Stepping*



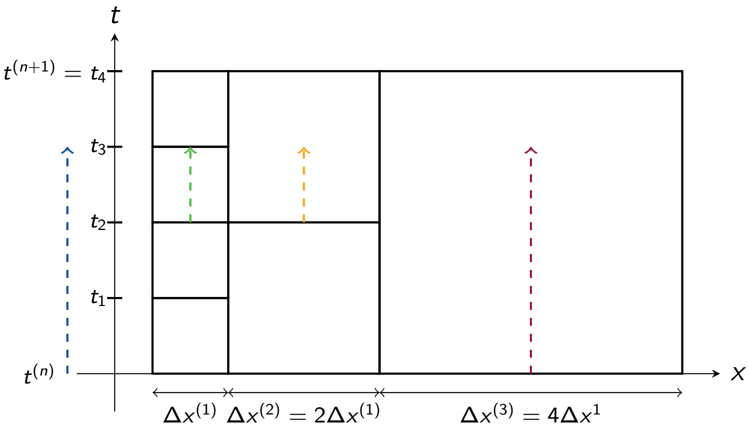
Application: Euler-Gravity

- **Hyperbolic – Elliptic coupling** for *Local Time-Stepping*



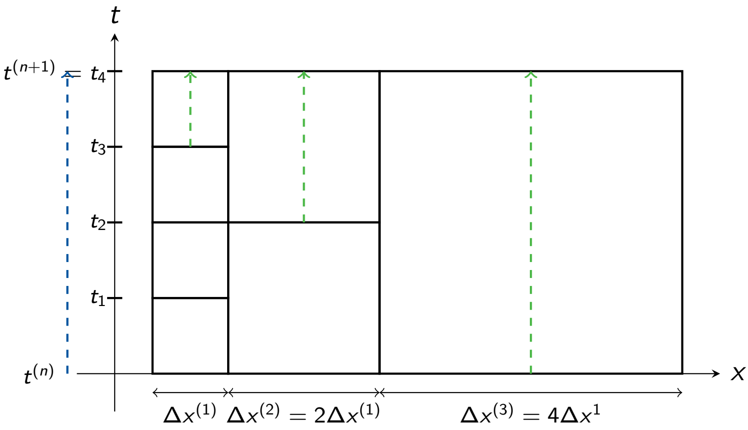
Application: Euler-Gravity

- **Hyperbolic – Elliptic coupling** for *Local Time-Stepping*



Application: Euler-Gravity

- **Hyperbolic – Elliptic coupling** for *Local Time-Stepping*



Application: Euler-Gravity

Need to solve

$$-\Delta\phi = -4\pi G\rho$$

after every update to ρ.

Application: Euler-Gravity

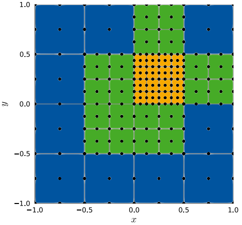
Need to solve

$$-\Delta\phi = -4\pi G\rho$$

$$\rightsquigarrow \underbrace{\partial_\tau \begin{pmatrix} \phi \\ q_1 \\ q_2 \end{pmatrix} + \nabla \cdot \begin{pmatrix} -q_1 & -q_2 \\ -\phi/T_r & 0 \\ 0 & -\phi/T_r \end{pmatrix} = \begin{pmatrix} -4\pi G\rho \\ -q_1/T_r \\ -q_2/T_r \end{pmatrix}}_{\textit{Hyperbolic Diffusion Equations}}, \qquad \begin{matrix} \mathbf{q} = \nabla\phi \\ T_r = \text{const} \end{matrix}$$

after every update to ρ.

Application: Euler-Gravity



Need to solve

$$-\Delta \phi = -4\pi G \rho$$

$$\rightsquigarrow \underbrace{\partial_\tau \begin{pmatrix} \phi \\ q_1 \\ q_2 \end{pmatrix} + \nabla \cdot \begin{pmatrix} -q_1 & -q_2 \\ -\phi/T_r & 0 \\ 0 & -\phi/T_r \end{pmatrix} = \begin{pmatrix} -4\pi G \rho \\ -q_1/T_r \\ -q_2/T_r \end{pmatrix}}_{\textit{Hyperbolic Diffusion Equations}}, \qquad \mathbf{q} = \nabla \phi, \quad T_r = \text{const}$$

after every update to ρ.

- P-ERK schemes: Always at same time-level $t = t_n + c_i \Delta t$!

$$\mathbf{c} = \begin{array}{c|c|c|c} c_1 & & & \\ \vdots & A^{(1)} & A^{(2)} & A^{(3)} \\ c_S & & & \\ \hline & & \mathbf{b}^T & \end{array}$$

Application: Euler-Gravity

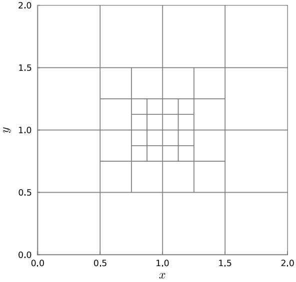
- **Convergence Study**

$$\rho = 2 + \frac{1}{10}\sin\left(\pi(x+y-t)\right)$$
$$v_x = v_y = 1$$
$$p = \frac{\rho^2}{\pi^2}$$
$$\phi = \frac{2}{\pi}(2-\rho)$$

$+$ source terms (manufactured solution)

3-Level Grid

$\Rightarrow$ 3-Level methods for Euler & Hyp.–Diff.



Application: Euler-Gravity

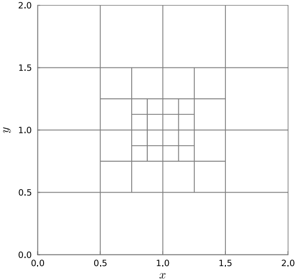
- **Convergence Study**

$$\rho = 2 + \frac{1}{10}\sin\left(\pi(x+y-t)\right)$$
$$v_x = v_y = 1$$
$$p = \frac{\rho^2}{\pi^2}$$
$$\phi = \frac{2}{\pi}(2-\rho)$$

$+$ source terms (manufactured solution)

3-Level Grid

$\Rightarrow$ 3-Level methods for Euler & Hyp.–Diff.



N	$L^\infty(\rho)$	$L^\infty(v_{x,y})$	$L^\infty(p)$	$L^\infty(\phi)$	$L^\infty(q_{1,2})$
40	$2.5 \cdot 10^{-4}$	$5.2 \cdot 10^{-4}$	$3.0 \cdot 10^{-3}$	$1.5 \cdot 10^{-3}$	$3.8 \cdot 10^{-3}$
160	$1.4 \cdot 10^{-4}$	$4.4 \cdot 10^{-5}$	$1.6 \cdot 10^{-4}$	$7.4 \cdot 10^{-5}$	$2.8 \cdot 10^{-4}$
640	$6.6 \cdot 10^{-6}$	$3.0 \cdot 10^{-6}$	$1.0 \cdot 10^{-5}$	$7.8 \cdot 10^{-6}$	$3.2 \cdot 10^{-5}$
2560	$4.2 \cdot 10^{-7}$	$2.1 \cdot 10^{-7}$	$5.9 \cdot 10^{-7}$	$4.4 \cdot 10^{-6}$	$3.8 \cdot 10^{-6}$
Avg. EOC	**4.2**	**3.75**	**4.1**	**2.8**	**3.3**

Application: Euler-Gravity

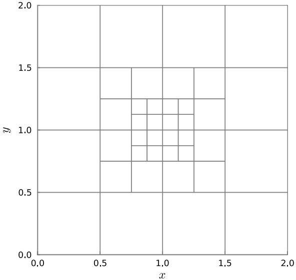
- **Convergence Study**

$$\rho = 2 + \frac{1}{10}\sin\left(\pi(x+y-t)\right)$$
$$v_x = v_y = 1$$
$$p = \frac{\rho^2}{\pi^2}$$
$$\phi = \frac{2}{\pi}(2-\rho)$$

$+$ source terms (manufactured solution)

3-Level Grid

$\Rightarrow$ 3-Level methods for Euler & Hyp.–Diff.



N	$L^1(\rho)$	$L^1(v_{x,y})$	$L^1(p)$	$L^2(\phi)$	$L^2(q_{1,2})$
40	$2.8\cdot10^{-4}$	$7.8\cdot10^{-5}$	$3.5\cdot10^{-4}$	$4.4\cdot10^{-4}$	$7.1\cdot10^{-4}$
160	$1.5\cdot10^{-5}$	$5.6\cdot10^{-6}$	$1.9\cdot10^{-5}$	$1.4\cdot10^{-5}$	$5.0\cdot10^{-5}$
640	$7.3\cdot10^{-7}$	$3.2\cdot10^{-7}$	$9.2\cdot10^{-7}$	$3.2\cdot10^{-6}$	$3.6\cdot10^{-6}$
2560	$4.4\cdot10^{-8}$	$1.9\cdot10^{-8}$	$5.6\cdot10^{-8}$	$2.9\cdot10^{-6}$	$2.5\cdot10^{-7}$
Avg. EOC	**4.2**	**4.0**	**4.2**	**2.4**	**3.8**

Application: Euler-Gravity

Jeans Gravitational Instability[1]

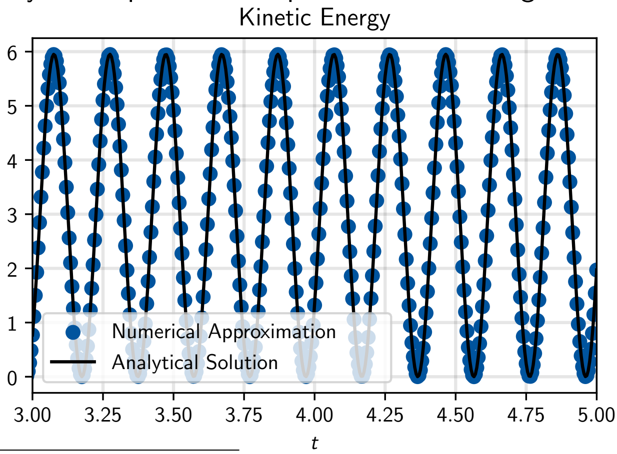
- Stable oscillation of a self-gravitating gas cloud
- Analytical expressions for pot., kin., int. energies

[1] Binney, Tremaine; Galactic Dynamics; 2011.

Application: Euler-Gravity

Jeans Gravitational Instability[1]

- Stable oscillation of a self-gravitating gas cloud
- Analytical expressions for pot., kin., int. energies

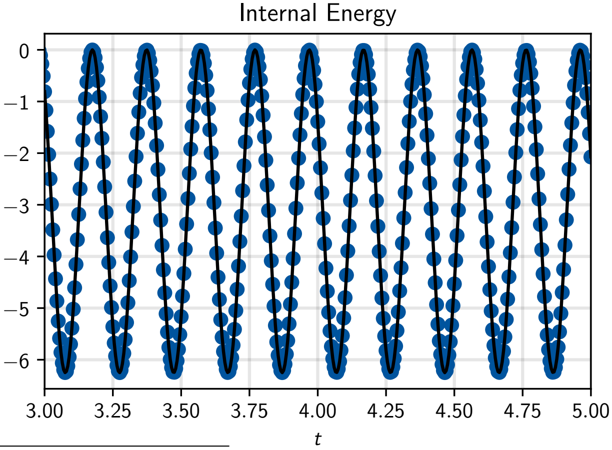


[1] Binney, Tremaine; Galactic Dynamics; 2011.

Application: Euler-Gravity

Jeans Gravitational Instability[1]

- Stable oscillation of a self-gravitating gas cloud
- Analytical expressions for pot., kin., int. energies

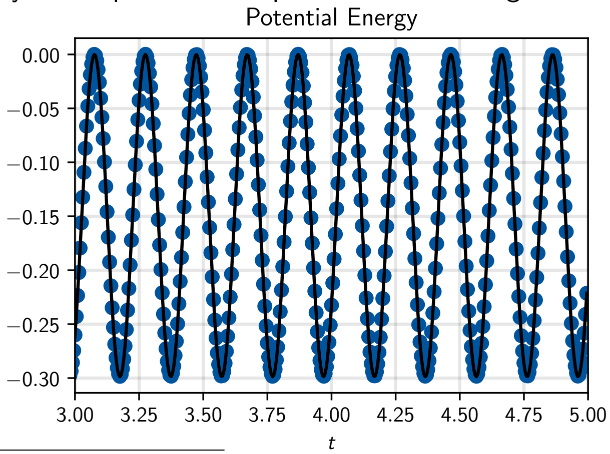


[1] Binney, Tremaine; Galactic Dynamics; 2011.

Application: Euler-Gravity

Jeans Gravitational Instability[1]

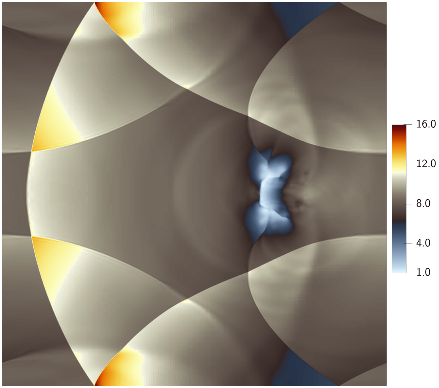
- Stable oscillation of a self-gravitating gas cloud
- Analytical expressions for pot., kin., int. energies



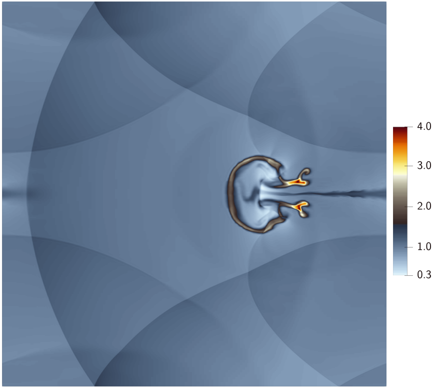
[1] Binney, Tremaine; Galactic Dynamics; 2011.

Application: Euler-Gravity

Blob Test[1] + Self-Gravity



Pressure p

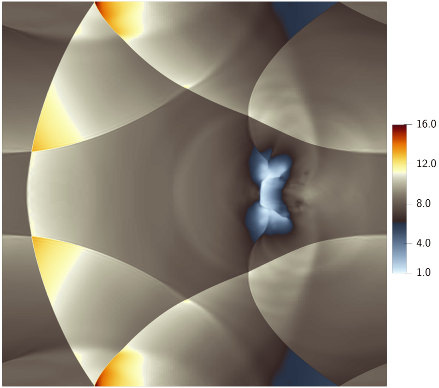


Density ρ

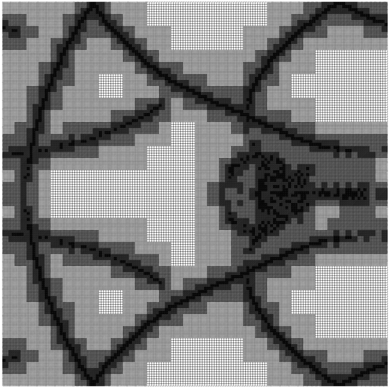
[1] Aretz et al.; Mon. Not. R. Astron. Soc; 2006.

Application: Euler-Gravity

Blob Test[1] + Self-Gravity



Pressure p



Mesh

[1] Aretz et. al; Mon. Not. R. Astron. Soc; 2006.

Conclusion & Outlook

Advantages of Multirate RK Schemes for Multiphysics:

Conclusion & Outlook

Advantages of Multirate RK Schemes for Multiphysics:

- Ease of Implementation

Conclusion & Outlook

Advantages of Multirate RK Schemes for Multiphysics:

- Ease of Implementation
- Optimality of Composing Schemes

Conclusion & Outlook

Advantages of Multirate RK Schemes for Multiphysics:

- Ease of Implementation
- Optimality of Composing Schemes
- Reduced Operations for Hyperbolic-Elliptic Problems compared to Local Time-Stepping

Conclusion & Outlook

Advantages of Multirate RK Schemes for Multiphysics:

- Ease of Implementation
- Optimality of Composing Schemes
- Reduced Operations for Hyperbolic-Elliptic Problems compared to Local Time-Stepping

Up next:

- Two-Fluid Euler-Poisson

Conclusion & Outlook

Advantages of Multirate RK Schemes for Multiphysics:

- Ease of Implementation
- Optimality of Composing Schemes
- Reduced Operations for Hyperbolic-Elliptic Problems compared to Local Time-Stepping

Up next:

- Two-Fluid Euler-Poisson
- Shallow-Water with non-hydrostatic Pressure

Thank You for your attention!

Questions?

doehring@acom.rwth-aachen.de
https://www.acom.rwth-aachen.de
https://github.com/DanielDoehring

Application: Euler-Acoustic

Sound by rotating vortices

- Acoustic-Purturbation Equations (APE)

Application: Euler-Acoustic

Sound by rotating vortices

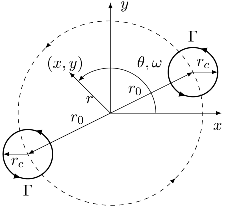
- Acoustic-Purturbation Equations (APE)
- with Lamb-vector source term from CEE

Application: Euler-Acoustic

Sound by rotating vortices

- Acoustic-Purturbation Equations (APE)
- with Lamb-vector source term from CEE
- Hybrid method: CAA + CFD

Application: Euler-Acoustic



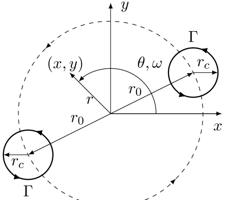
Sound by rotating vortices

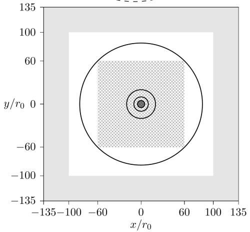
- Acoustic-Purturbation Equations (APE)
- with Lamb-vector source term from CEE
- Hybrid method: CAA + CFD

Application: Euler-Acoustic

Sound by rotating vortices

- Acoustic-Purturbation Equations (APE)
- with Lamb-vector source term from CEE
- Hybrid method: CAA + CFD

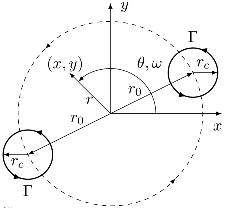


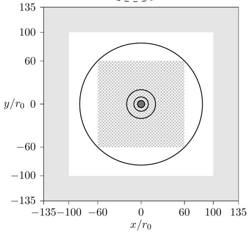


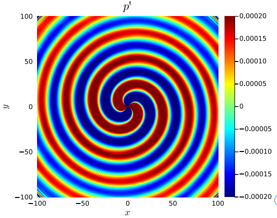
Application: Euler-Acoustic

Sound by rotating vortices

- Acoustic-Purturbation Equations (APE)
- with Lamb-vector source term from CEE
- Hybrid method: CAA + CFD







Application: Euler-Acoustic

Sound by rotating vortices

- Acoustic-Purturbation Equations (APE)
- with Lamb-vector source term from CEE
- Hybrid method: CAA + CFD

Method	$\tau / \tau^\star$	$N_{\text{RHS}} / N^\star_{\text{RHS}}$
P-ERK$_{4;\{5,6,8,13\}}$, P-ERK$_{4;\{5,6,9,14\}}$	1.0	1.0
P-ERK$_{4;13}$, P-ERK$_{4;14}$	1.82	2.08
NDB$_{4;14}$	2.28	2.71
TD$_{4;8}$	3.32	3.73
CFR$_{4;6}$	4.72	5.29
RK$_{4;4}$	6.13	5.45