Multirate Time-Integration based on Dynamic ODE Partitioning through Adaptively Refined Meshes for Convection-Dominated Flows

Daniel Doehring[1]

in collaboration with

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

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

2024/07/23

Applied and Computational Mathematics

RWTH AACHEN UNIVERSITY

Motivation & Research Question

Motivation:

- Explicit time-integration on non-uniform meshes with global time-stepping leaves room for efficiency gains

Research Question:

[1] Vermeire. JCP. 2019

[2] Nasab, Vermeire. JCP. 2022

Motivation & Research Question

Motivation:

- Explicit time-integration on non-uniform meshes with global time-stepping leaves room for efficiency gains
- $\Rightarrow$ Local time stepping, but this leads in general to changes in the spatial discretization methodology

Research Question:

[1] Vermeire. JCP. 2019

[2] Nasab, Vermeire. JCP. 2022

Motivation & Research Question

Motivation:

- Explicit time-integration on non-uniform meshes with global time-stepping leaves room for efficiency gains
- $\Rightarrow$ Local time stepping, but this leads in general to changes in the spatial discretization methodology
- $\Rightarrow$ Non-intrusive multirate time-integration offered by Paired-Explicit Runge-Kutta schemes (P-ERK)[1] [2]

Research Question:

[1] Vermeire. JCP. 2019

[2] Nasab, Vermeire. JCP. 2022

Motivation & Research Question

Motivation:

- Explicit time-integration on non-uniform meshes with global time-stepping leaves room for efficiency gains
- $\Rightarrow$ Local time stepping, but this leads in general to changes in the spatial discretization methodology
- $\Rightarrow$ Non-intrusive multirate time-integration offered by Paired-Explicit Runge-Kutta schemes (P-ERK)[1] [2]

Research Question:

- *Can we gain efficiency also for dynamic, adaptively refined meshes which require a dynamic ODE re-partitioning?*

[1] Vermeire. JCP. 2019

[2] Nasab, Vermeire. JCP. 2022

Motivation & Research Question

Motivation:

- Explicit time-integration on non-uniform meshes with global time-stepping leaves room for efficiency gains
- $\Rightarrow$ Local time stepping, but this leads in general to changes in the spatial discretization methodology
- $\Rightarrow$ Non-intrusive multirate time-integration offered by Paired-Explicit Runge-Kutta schemes (P-ERK)[1] [2]

Research Question:

- *Can we gain efficiency also for dynamic, adaptively refined meshes which require a dynamic ODE re-partitioning?*

  $\Rightarrow$ **Yes!** (but there are some limitations)

  see Doehring et al. JCP. 2024. DOI: 10.1016/j.jcp.2024.113223

[1] Vermeire. JCP. 2019

[2] Nasab, Vermeire. JCP. 2022

Contents

Preliminaries: Convection-Dominance

We consider PDEs of the form

$$\partial_t \mathbf{u}(t, \mathbf{x}) + \nabla \cdot \mathbf{f}\big(\mathbf{u}(t, \mathbf{x}), \nabla \mathbf{u}(t, \mathbf{x})\big) = \mathbf{s}\big(\mathbf{u}(t, \mathbf{x}), t, \mathbf{x}\big) \qquad (1)$$

which describe the evolution of compressible flows.

Preliminaries: Convection-Dominance

We consider PDEs of the form

$$\partial_t \mathbf{u}(t, \mathbf{x}) + \nabla \cdot \mathbf{f}\big(\mathbf{u}(t, \mathbf{x}), \nabla \mathbf{u}(t, \mathbf{x})\big) = \mathbf{s}\big(\mathbf{u}(t, \mathbf{x}), t, \mathbf{x}\big) \qquad (1)$$

which describe the evolution of compressible flows.
We employ a method-of-lines approach (via the DGSEM) to obtain

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{U}(t) = \mathbf{F}\big(t, \mathbf{U}(t)\big) \, . \qquad (2)$$

Preliminaries: Convection-Dominance

We consider PDEs of the form

$$\partial_t \mathbf{u}(t, \mathbf{x}) + \nabla \cdot \mathbf{f}\big(\mathbf{u}(t, \mathbf{x}), \nabla \mathbf{u}(t, \mathbf{x})\big) = \mathbf{s}\big(\mathbf{u}(t, \mathbf{x}), t, \mathbf{x}\big) \tag{1}$$

which describe the evolution of compressible flows.
We employ a method-of-lines approach (via the DGSEM) to obtain

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{U}(t) = \mathbf{F}\big(t, \mathbf{U}(t)\big) \, . \tag{2}$$

We say a system is **convection-dominated** if

$$\rho \sim \frac{|a|}{\Delta x} \gg \frac{d}{\Delta x^2} \tag{3}$$

where ρ is the spectral radius of

$$J_{\mathbf{F}}(\mathbf{U}) := \frac{\partial \mathbf{F}}{\partial \mathbf{U}}\bigg|_{\mathbf{U}} \, . \tag{4}$$

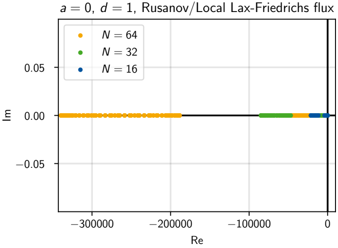
Preliminaries: Spectra

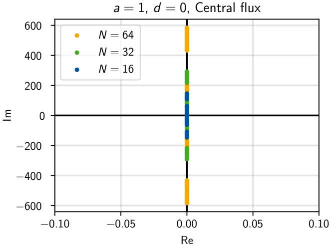
$$u_t + au_x + du_{xx} = 0, \qquad \Omega = (0, 1),\ k = 3$$

Jacobians generated with
`Trixi.jl`

Preliminaries: Spectra

$$u_t + au_x + du_{xx} = 0, \qquad \Omega = (0,1),\ k = 3$$

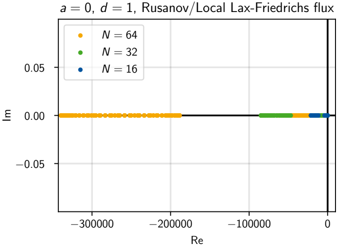


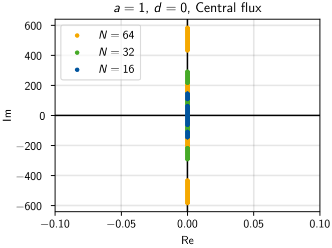


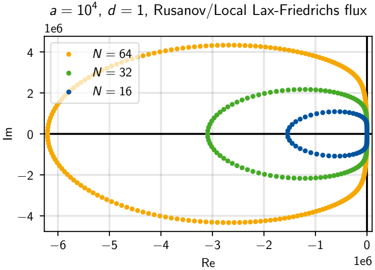
Jacobians generated with `Trixi.jl`

Preliminaries: Spectra

$$u_t + au_x + du_{xx} = 0, \qquad \Omega = (0, 1),\ k = 3$$





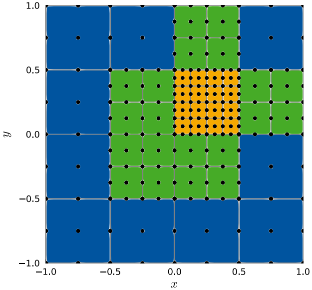


Jacobians generated with
`Trixi.jl`

$\Rightarrow$ Problems within scope

Preliminaries: Optimized Runge-Kutta Schemes

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

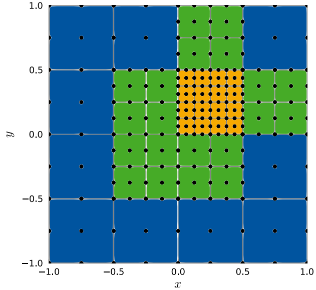


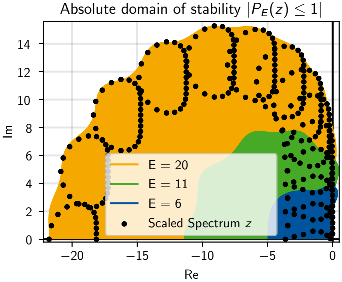
- For convection dominated problems:

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

Preliminaries: Optimized Runge-Kutta Schemes

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





- For convection dominated problems:

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

- For optimized Runge-Kutta schemes:

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

Contents

P-ERK Schemes: ODE Partitioning

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

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

P-ERK Schemes: ODE Partitioning

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

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

and employ different RKMs

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

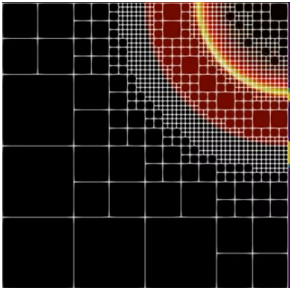
$$\mathbf{U}_{n+1} = \mathbf{U}_n + \Delta t \sum_{i=1}^{S} \sum_{k=1}^{R} b_i^{(r)} \mathbf{K}_i^{(r)}$$

optimized for the spectrum $\sigma\big(J(\mathbf{U}_0)\big) = \sigma\left(\frac{\partial \mathbf{F}}{\partial \mathbf{U}}\Big|_{\mathbf{U}_0}\right)$.

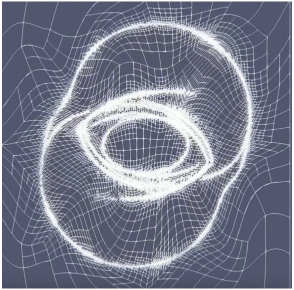
P-ERK Schemes: ODE Partitioning

Dynamic Re-Partitioning

- **Cartesian** Quad/Octree starting from square/cube



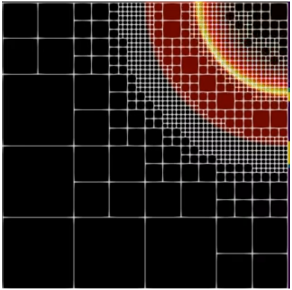
- **Curved** Quad/Octree on (in principle) general domains



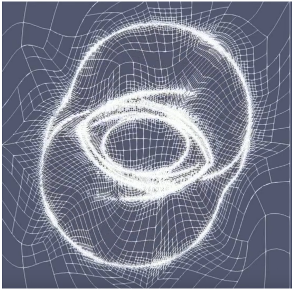
P-ERK Schemes: ODE Partitioning

Dynamic Re-Partitioning

- **Cartesian** Quad/Octree starting from square/cube
  1. Partition r governed by position of cell in the tree
     $\Rightarrow$ Very efficient assignment of r!



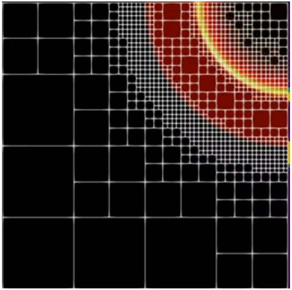
- **Curved** Quad/Octree on (in principle) general domains



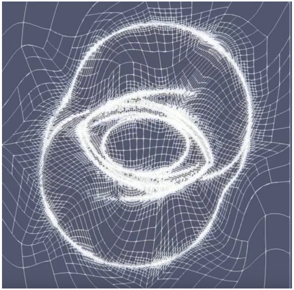
P-ERK Schemes: ODE Partitioning

Dynamic Re-Partitioning

- **Cartesian** Quad/Octree starting from square/cube
  1. Partition r governed by position of cell in the tree
     $\Rightarrow$ Very efficient assignment of r!



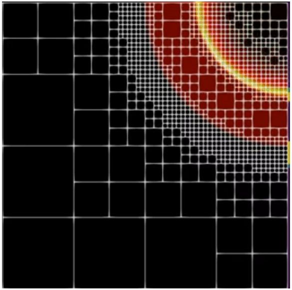
- **Curved** Quad/Octree on (in principle) general domains
  1. Loop over entire mesh to find Δx_{min}, Δx_{max}
     $\Rightarrow$ Determine bins $I^{(r)} := \left[x_-^{(r)}, x_+^{(r)}\right)$



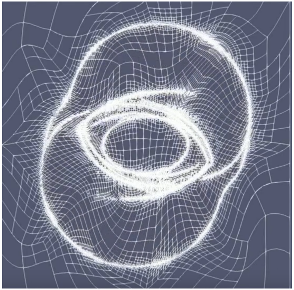
P-ERK Schemes: ODE Partitioning

Dynamic Re-Partitioning

- **Cartesian** Quad/Octree starting from square/cube
  1. Partition r governed by position of cell in the tree
     $\Rightarrow$ Very efficient assignment of r!



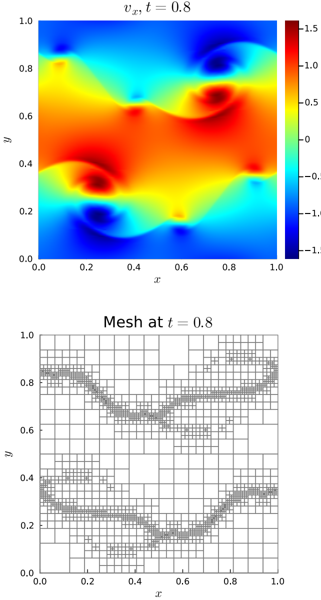
- **Curved** Quad/Octree on (in principle) general domains
  1. Loop over entire mesh to find Δx_{min}, Δx_{max}
     $\Rightarrow$ Determine bins $I^{(r)} := \left[x_-^{(r)}, x_+^{(r)}\right)$
  2. Assign elements i to partition r such that $\Delta x_i \in I^{(r)}$



Contents

Applications: Doubly Periodic Shear Layer (NSF)

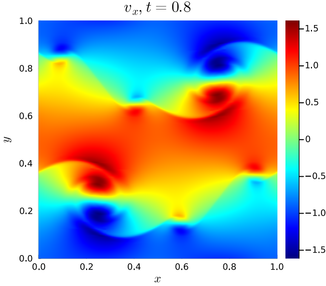
- Flow parameters as in Brown, Minion. JCP. 1995.
- $k = 3$, HLLC surf. flux, EC vol. flux
- AMR based on v_x

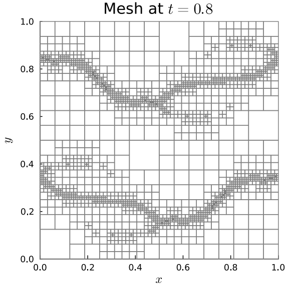


Applications: Doubly Periodic Shear Layer (NSF)

– Flow parameters as in Brown, Minion. JCP. 1995.
– $k = 3$, HLLC surf. flux, EC vol. flux
– AMR based on v_x

- 7 Levels
- $N_{\mathrm{AMR}} = 20$

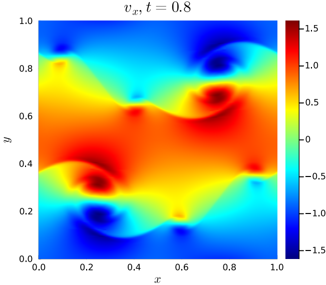


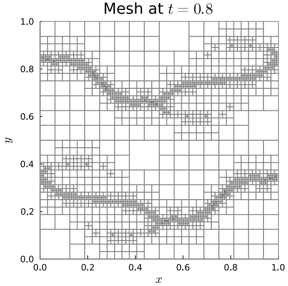


Applications: Doubly Periodic Shear Layer (NSF)

– Flow parameters as in Brown, Minion. JCP. 1995.
– $k = 3$, HLLC surf. flux, EC vol. flux
– AMR based on v_x

- 7 Levels
- $N_{\text{AMR}} = 20$
- Re-partitioning: 2.5% of τ

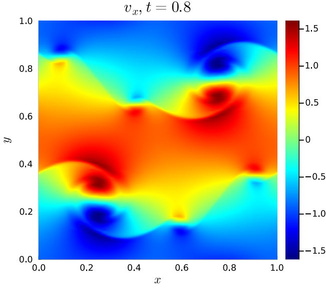


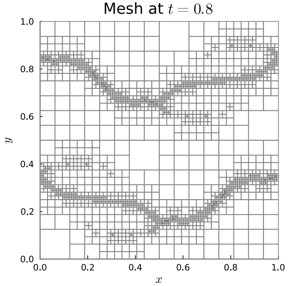


Applications: Doubly Periodic Shear Layer (NSF)

– Flow parameters as in Brown, Minion. JCP. 1995.
– $k = 3$, HLLC surf. flux, EC vol. flux
– AMR based on v_x

- 7 Levels
- $N_{\text{AMR}} = 20$
- Re-partitioning: 2.5% of τ
- Memory footprint increase: 20%

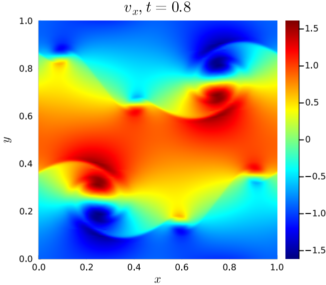




Applications: Doubly Periodic Shear Layer (NSF)

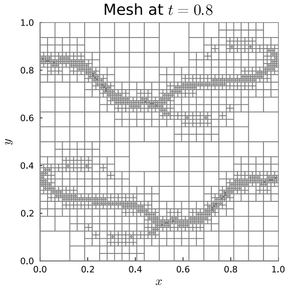
- Flow parameters as in Brown, Minion. JCP. 1995.
- $k = 3$, HLLC surf. flux, EC vol. flux
- AMR based on v_x

- 7 Levels
- $N_{\text{AMR}} = 20$
- Re-partitioning: 2.5% of τ
- Memory footprint increase: 20%



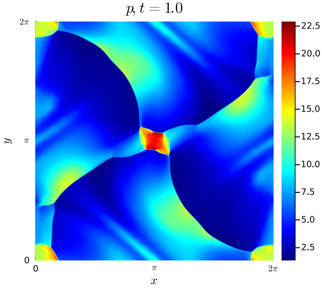
Method	$\tau/\tau_{\text{P-ERK}_{3;\{3,4,7\}}}$	$N/N_{\text{P-ERK}_{3;\{3,4,7\}}}$
P-ERK$_{3;\{3,4,7\}}$	1.0	1.0
P-ERK$_{3;7}$	1.62	1.76
DGLDD$_{3;7}$	1.65	1.62
PKD3S$_{3;5}$	2.16	2.03
RDPKFSAL$_{3;5}$	2.53	2.47
CKLLS$_{3;4}$	2.74	2.76
SSP$_{3;3}$	2.74	2.70

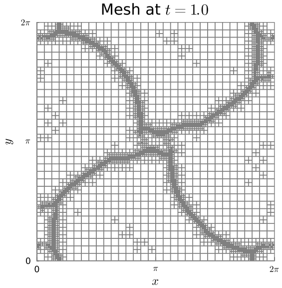
Methods implemented in OrdinaryDiffEq.jl



Applications: Orszag-Tang Vortex (Ideal VR-MHD)

– Flow parameters as in Warburton, Karniadakis. JCP. 1999. with visc. & resis.
– $k = 3$, Shock-Capturing
– AMR based on $\rho \cdot p$

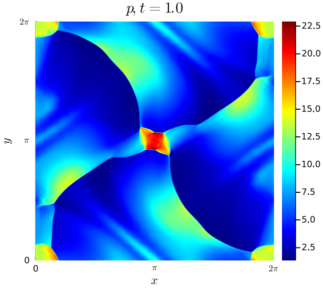


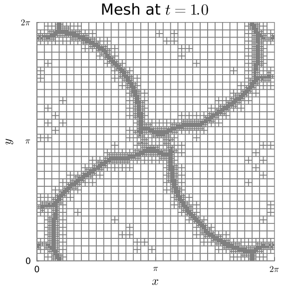


Applications: Orszag-Tang Vortex (Ideal VR-MHD)

– Flow parameters as in Warburton, Karniadakis. JCP. 1999. with visc. & resis.
– $k = 3$, Shock-Capturing
– AMR based on $\rho \cdot p$

- 5 Levels
- $N_{\text{AMR}} = 10$

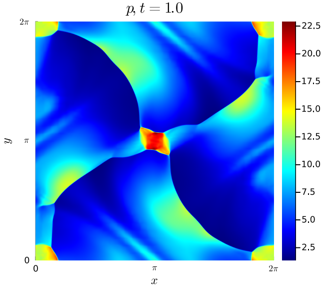


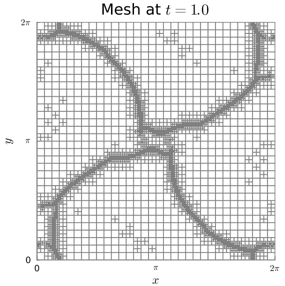


Applications: Orszag-Tang Vortex (Ideal VR-MHD)

– Flow parameters as in Warburton, Karniadakis. JCP. 1999. with visc. & resis.
– $k = 3$, Shock-Capturing
– AMR based on $\rho \cdot p$

- 5 Levels
- $N_{\text{AMR}} = 10$
- Re-partitioning: 1.0% of τ

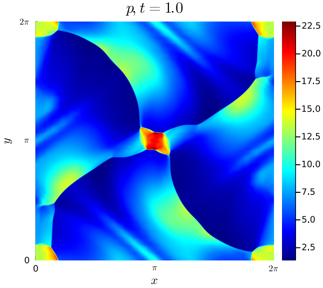


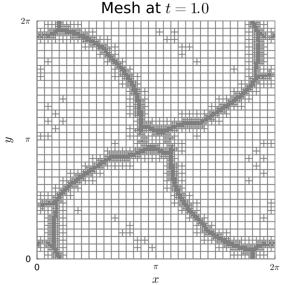


Applications: Orszag-Tang Vortex (Ideal VR-MHD)

– Flow parameters as in Warburton, Karniadakis. JCP. 1999. with visc. & resis.
– $k = 3$, Shock-Capturing
– AMR based on $\rho \cdot p$

- 5 Levels
- $N_{\text{AMR}} = 10$
- Re-partitioning: 1.0% of τ
- Memory footprint increase: 28%



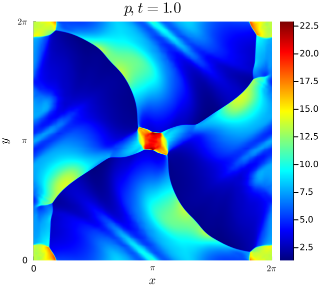


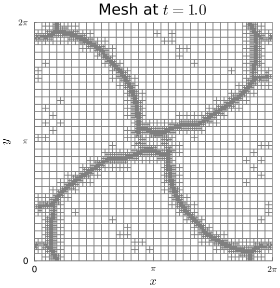
Applications: Orszag-Tang Vortex (Ideal VR-MHD)

– Flow parameters as in Warburton, Karniadakis. JCP. 1999. with visc. & resis.
– $k = 3$, Shock-Capturing
– AMR based on $\rho \cdot p$

- 5 Levels
- $N_{\text{AMR}} = 10$
- Re-partitioning: 1.0% of τ
- Memory footprint increase: 28%

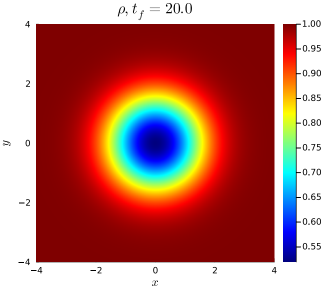
Method	$\tau / \tau_{\text{P-ERK}_{3;\{3,4,6\}}}$	$N / N_{\text{P-ERK}_{3;\{3,4,6\}}}$
P-ERK$_{3;\{3,4,6\}}$	1.0	1.0
P-ERK$_{3;6}$	1.27	1.44
DGLDD$_{3;7}$	1.58	1.77
PKD3S$_{3;5}$	1.72	1.89
RDPKFSAL$_{3;5}$	1.65	1.89
SSP$_{3;3}$	2.01	2.28

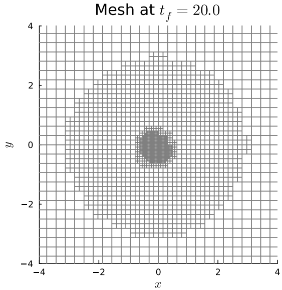




Applications: Isentropic Vortex Advection (CEE)

- Flow parameters as in Vermeire. JCP. 2019.
- $k = 3$, HLLC surf. flux
- Analytic AMR
- $p = 4$ $\rightarrow$Preprint on webpage

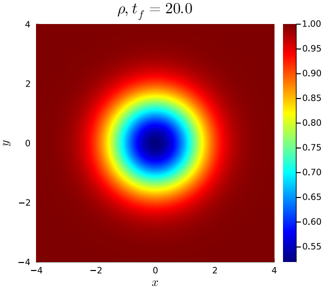


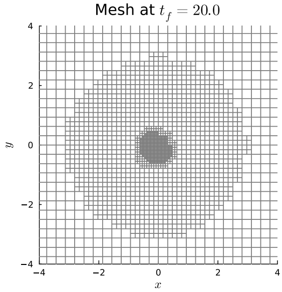


Applications: Isentropic Vortex Advection (CEE)

- Flow parameters as in Vermeire. JCP. 2019.
- $k = 3$, HLLC surf. flux
- Analytic AMR
- $p = 4$ →Preprint on webpage

- 4 Levels
- $N_{\text{AMR}} = 20$

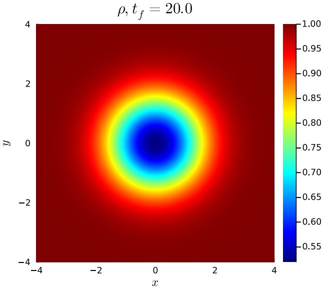


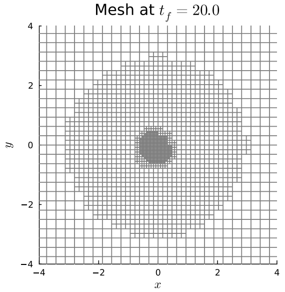


Applications: Isentropic Vortex Advection (CEE)

- Flow parameters as in Vermeire. JCP. 2019.
- $k = 3$, HLLC surf. flux
- Analytic AMR
- $p = 4$ →Preprint on webpage

- 4 Levels
- $N_{\text{AMR}} = 20$
- Re-partitioning: 0.2% of τ

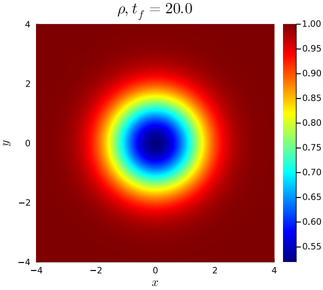


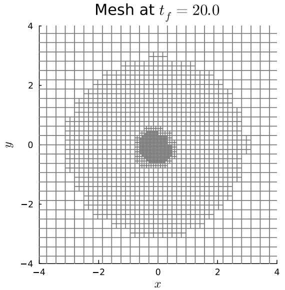


Applications: Isentropic Vortex Advection (CEE)

- Flow parameters as in Vermeire. JCP. 2019.
- $k = 3$, HLLC surf. flux
- Analytic AMR
- $p = 4$ →Preprint on webpage

- 4 Levels
- $N_{\text{AMR}} = 20$
- Re-partitioning: 0.2% of τ
- Memory footprint increase: 10%



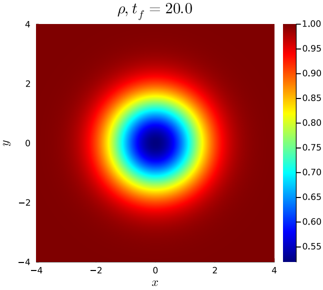


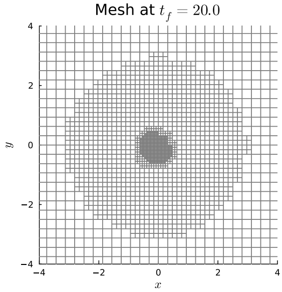
Applications: Isentropic Vortex Advection (CEE)

– Flow parameters as in Vermeire. JCP. 2019.
– $k = 3$, HLLC surf. flux
– Analytic AMR
– $p = 4$ →Preprint on webpage

- 4 Levels
- $N_{\text{AMR}} = 20$
- Re-partitioning: 0.2% of τ
- Memory footprint increase: 10%

Method	$\tau / \tau_{\text{P-ERK}_{4;\{5,7,11,19\}}}$	$N / N_{\text{P-ERK}_{4;\{5,7,11,19\}}}$
P-ERK$_{4;\{5,7,11,19\}}$	1.0	1.0
P-ERK$_{4;19}$	1.20	1.38
NDB$_{4;14}$	1.50	1.82
TD$_{4;8}$	1.52	1.71
RDPK$_{4;9}$	1.40	1.63
RK$_{4;4}$	1.79	2.08





Contents

Limitations: Loss of Monotonicity

Model problem: 1D Advection

$$u_t + u_x = 0, \tag{5a}$$

$$u_0(t_0 = 0, x) = 1 + \frac{1}{2}\sin(\pi x) \tag{5b}$$

on $\Omega = (-1, 1)$ with periodic boundaries and *Godunov (upwind)* scheme on potentially non-uniform mesh:

Limitations: Loss of Monotonicity

Model problem: 1D Advection

$$u_t + u_x = 0, \tag{5a}$$

$$u_0(t_0 = 0, x) = 1 + \frac{1}{2}\sin(\pi x) \tag{5b}$$

on $\Omega = (-1, 1)$ with periodic boundaries and *Godunov (upwind)* scheme on potentially non-uniform mesh:

$$\mathbf{U}_0 = \mathbf{U}(t_0) \tag{6a}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{U}(t) = \underbrace{\begin{pmatrix} -\frac{1}{\Delta x_1} & 0 & \dots & 0 & \frac{1}{\Delta x_1} \\ \frac{1}{\Delta x_2} & -\frac{1}{\Delta x_2} & 0 & \dots & 0 \\ & \ddots & \ddots & & \\ 0 & \dots & 0 & \frac{1}{\Delta x_N} & -\frac{1}{\Delta x_N} \end{pmatrix}}_{=:L} \mathbf{U}(t), \tag{6b}$$

Limitations: Loss of Monotonicity

Applying a two-level ($R = 2$) P-ERK scheme to (6b) leads to the fully discrete system

$$\mathbf{U}_{n+1} = \underbrace{\begin{pmatrix} D^{(1)}(\Delta t, L) & 0 \\ 0 & D^{(2)}(\Delta t, L) \end{pmatrix}}_{=:D} \mathbf{U}_n \tag{7}$$

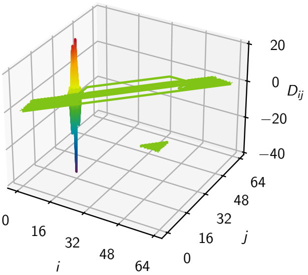
Limitations: Loss of Monotonicity

Applying a two-level ($R = 2$) P-ERK scheme to (6b) leads to the fully discrete system

$$\mathbf{U}_{n+1} = \underbrace{\begin{pmatrix} D^{(1)}(\Delta t, L) & 0 \\ 0 & D^{(2)}(\Delta t, L) \end{pmatrix}}_{=:D} \mathbf{U}_n \tag{7}$$

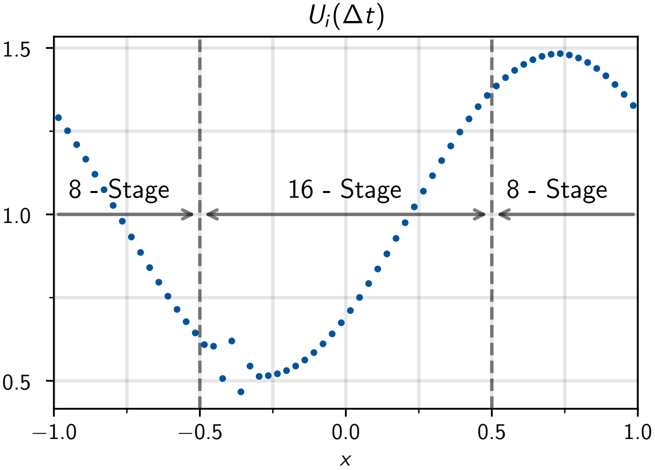
The fully discrete system (7) is *monotonicity-preserving* in the sense of Harten if and only if

$$D_{ij} \geq 0 \quad \forall\, i, j = 1, \ldots, N. \tag{8}$$



Limitations: Loss of Monotonicity

Result $\mathbf{U}_1$ of (7) after one step with $E^{(1)} = 8, E^{(2)} = 16$.



Contents

Conclusion & Outlook

Summary:

- Non-intrusive multirate time-integration for convection-dominated problems

Up next:

Conclusion & Outlook

Summary:

- Non-intrusive multirate time-integration for convection-dominated problems
  $\rightarrow$ MoL and stabilized schemes

Up next:

Conclusion & Outlook

Summary:

- Non-intrusive multirate time-integration for convection-dominated problems
  - $\rightarrow$ MoL and stabilized schemes
  - $\rightarrow$ Efficient re-partitioning for cartesian quad/octree meshes

Up next:

Conclusion & Outlook

Summary:

- Non-intrusive multirate time-integration for convection-dominated problems
  - $\rightarrow$ MoL and stabilized schemes
  - $\rightarrow$ Efficient re-partitioning for cartesian quad/octree meshes
  - $\rightarrow$ Speed-up: 20% to 62% compared to optimized methods
    40% to 274% for off-the-shelf methods

Up next:

Conclusion & Outlook

Summary:

- Non-intrusive multirate time-integration for convection-dominated problems
  - $\rightarrow$ MoL and stabilized schemes
  - $\rightarrow$ Efficient re-partitioning for cartesian quad/octree meshes
  - $\rightarrow$ Speed-up: 20% to 62% compared to optimized methods
    40% to 274% for off-the-shelf methods
- Investigation Nonlinear stability

Up next:

Conclusion & Outlook

Summary:

- Non-intrusive multirate time-integration for convection-dominated problems
  - → MoL and stabilized schemes
  - → Efficient re-partitioning for cartesian quad/octree meshes
  - → Speed-up: 20% to 62% compared to optimized methods
    40% to 274% for off-the-shelf methods
- Investigation Nonlinear stability

Up next:

- Combination of local & multirate time-stepping

Conclusion & Outlook

Summary:

- Non-intrusive multirate time-integration for convection-dominated problems
  - → MoL and stabilized schemes
  - → Efficient re-partitioning for cartesian quad/octree meshes
  - → Speed-up: 20% to 62% compared to optimized methods
    40% to 274% for off-the-shelf methods
- Investigation Nonlinear stability

Up next:

- Combination of local & multirate time-stepping
- Parallelize partition assignment for unstructured meshes

<u>Thank You</u> for your attention!

Questions?

doehring@acom.rwth-aachen.de

https://www.acom.rwth-aachen.de

https://github.com/DanielDoehring

References:

P-ERK + AMR Paper:	Doehring et al. JCP. 2024. DOI: 10.1016/j.jcp.2024.113223
Reproducibility Repository:	https://github.com/trixi-framework/paper-2024-amr-paired-rk
P-ERK $p = 4$ Preprint:	https://www.acom.rwth-aachen.de/media/perk_p4.pdf

Limitations: Loss of Monotonicity

$$e^{\mathsf{TV}} := \frac{\|\mathbf{U}_1\|_{\mathsf{TV}} - \|\mathbf{U}_0\|_{\mathsf{TV}}}{\|\mathbf{U}_0\|_{\mathsf{TV}}}$$

$$\mathrm{CFL} = 1.0$$

$$E^{(1)} = 8, \quad E^{(2)} = \alpha \cdot E^{(1)}$$

$$\Delta x^{(2)} = \alpha \cdot \Delta x^{(1)}, \quad \Delta x^{(1)} = \frac{1}{32}$$

$E^{(2)}$	9	10	11	12	13	14	15	16
α	1.125	1.25	1.375	1.5	1.625	1.75	1.875	2.0
e^{TV}	-0.03	0.11	0.55	1.65	3.71	7.85	15.4	26.0

Table: Increase in relative total variation e_{TV} for non-uniform meshes with grid sizes Δx, $\frac{\Delta x}{\alpha}$. The $E^{(1)} = 8$ method is used in the non-refined region for all α.

Limitations: Loss of Monotonicity

$$e^{\mathrm{TV}} := \frac{\|\mathbf{U}_1\|_{\mathrm{TV}} - \|\mathbf{U}_0\|_{\mathrm{TV}}}{\|\mathbf{U}_0\|_{\mathrm{TV}}}$$

$$E^{(1)} = 8, \quad E^{(2)} = 16$$

$$\Delta x^{(2)} = 2\Delta x^{(1)}, \quad \Delta x^{(1)} = \frac{1}{32}$$

CFL	0.4	0.5	0.6	0.7	0.8	0.9	1.0
e^{TV}	-0.01	0.03	0.28	1.21	4.04	11.5	26.0

Table: Increase in relative total variation e_{TV} for non-uniform meshes with varying base grid size $\Delta x^{(1)} = 2/64$ and two times ($\alpha = 2$) finer grid in $\Omega^{(2)} = [-0.5, 0.5]$ under reduction of timestep $\Delta t = \mathrm{CFL} \cdot \Delta t_8$. The coarse cells are integrated with the $E^{(1)} = 8$ stage evaluation method, while the fine cells are integrated using the $E^{(2)} = 16$ method.

Limitations: Loss of Monotonicity

$$e^{\text{TV}} := \frac{\|\mathbf{U}_1\|_{\text{TV}} - \|\mathbf{U}_0\|_{\text{TV}}}{\|\mathbf{U}_0\|_{\text{TV}}}$$

$$\text{CFL} = 1.0$$

$$E^{(1)} = 8, \quad E^{(2)} = 16$$

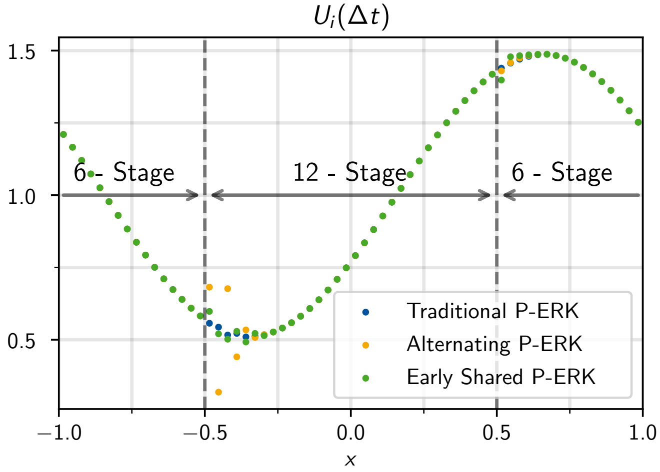
$$\Delta x^{(2)} = 2\Delta x^{(1)}, \qquad \Delta x^{(1)} = \frac{1}{N^{(1)}}$$

$N^{(1)}$	64	128	256	512	1024	2048	4096
e^{TV}	26.0	7.27	1.82	0.45	0.11	0.02	0.01

Table: Increase in relative total variation e_{TV} for non-uniform meshes with varying base grid size $\Delta x_{\text{Base}} = {}^2/_{N^{(1)}}$ and two times ($\alpha = 2$) finer grid in $\Omega^{(2)} = [-0.5, 0.5]$. The coarse cells are integrated with the $E^{(1)} = 8$ stage evaluation method, while the fine cells are integrated using the $E^{(2)} = 16$ method.

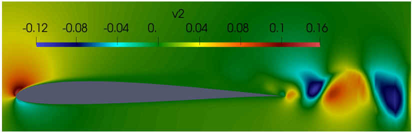
Appendix: Loss of Monotonicity

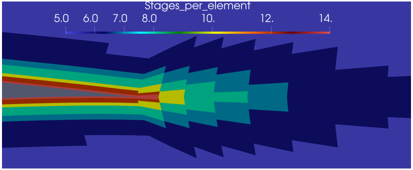
Result $\mathbf{U}^1$ to (7) after one timestep of two different 6-12 P-ERK schemes with different stage evaluation patterns on a uniform mesh.



Application: Laminar flow around 2D SD7003 airfoil

- $\mathrm{Re} = 10^4, \mathrm{Ma} = 0.2$
- $k = 3, p = 4$, 7605 quads
- $\Omega = [-20, 40] \times [-20, 20]$





Source	$\overline{C}_L$	$\overline{C}_D$
P-ERK $p = 4$	0.3827	0.04995
P-ERK $p = 2$[a]	0.3841	0.04990
P-ERK $p = 3$[b]	0.3848	0.04910
Uranga et al.[c]	0.3755	0.04978
López-Morales et al.[d]	0.3719	0.04940

Method	$\tau / \tau^\star$	$N_{\text{RHS}} / N^\star_{\text{RHS}}$
P-ERK$_{4;\{5,6,\dots 14\}}$	1.0	1.0
P-ERK$_{4;12}$	1.37	2.05
RK$_{4;4}$	2.20	2.33

(Lower is better)

[a] Vermeire. JCP. 2019

[b] Nasab, Vermeire. JCP. 2022

[c] Uranga et al. Int. J. Num Methods Eng. 2011

[d] López-Morales et al. 32nd AIAA AAC. 2014

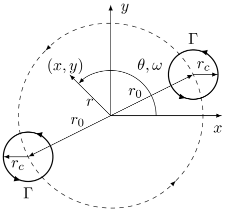
Application: Euler-Acoustic Simulation

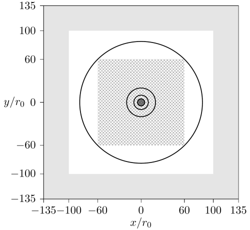
Co-rotating vortices

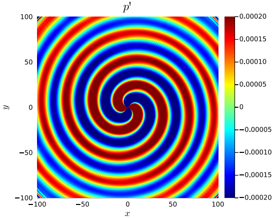
- Acoustic-Purturbation Equations (APE)[3]
- with Lamb-vector source term from CEE
- Hybrid method[4]: CAA + CFD

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$_{4;14}$	1.61	1.87
NDB$_{4;14}$	1.82	2.20
TD$_{4;8}$	2.65	3.02
CFR$_{4;6}$	3.86	4.26
RK$_{4;4}$	4.77	4.52

(Lower is better)





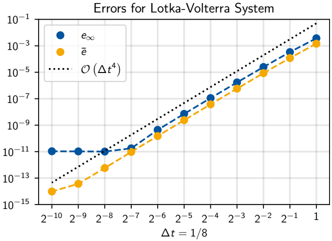


[3] Ewert, Schröder. JCP. 2003

[4] Schlottke-Lakemper et al. Computers & Fluids. 2017

P-ERK: Properties

Paired-Explicit Runge-Kutta (P-ERK) methods[5] are a class of partitioned RKMs that satisfy



- Order conditions: $p = 2$[1], $p = 3$[6] or $p = 4$[7]
- Conservation[8]
- Internal Consistency[4]

	P-ERK$_{4;\{6,10,16\}}$	SSP$_{4;10}$
e_ρ^{Cons}	$3.78 \cdot 10^{-13}$	$2.11 \cdot 10^{-13}$
$e_{\rho v_x}^{\text{Cons}}$	$5.70 \cdot 10^{-14}$	$1.26 \cdot 10^{-13}$
$e_{\rho v_y}^{\text{Cons}}$	$6.53 \cdot 10^{-14}$	$1.29 \cdot 10^{-13}$
$e_{\rho e}^{\text{Cons}}$	$9.40 \cdot 10^{-13}$	$3.66 \cdot 10^{-12}$

Method	$\|\|e_\rho(x,y)\|\|_{L^\infty(\Omega)}$
P-ERK$_{4;6}$	$1.1497 \cdot 10^{-5}$
P-ERK$_{4;10}$	$1.1497 \cdot 10^{-5}$
P-ERK$_{4;16}$	$1.1497 \cdot 10^{-5}$
P-ERK$_{4;\{6,10,16\}}$	$1.1499 \cdot 10^{-5}$

[5] Vermeire. JCP. 2019

[6] Nasab, Vermeire. JCP. 2022

[7] Doehring et al. To Appear. 2024

[8] Hundsdorfer, Ketcheson, Savostianov. J. Sci. Comput. 2015