Universität
Münster

Parareal with Spectral Coarse Solvers

Martin J. Gander[1], Mario Ohlberger[2], Stephan Rave[2]
[1]Université de Genève, Switzerland
[2]University of Münster, Germany

DD29

Milan, June 24, 2025

living.knowledge

MM
Mathematics
Münster
Cluster of Excellence

Projection-Based Model Order Reduction
(Reduced Basis Methods)

Full order model (basic example)

For given parameter $\mu \in \mathcal{P}$, find $u_h(\mu) \in V_h$ s.t. (dim $V_h > 10^5$)

$$a(u_h(\mu), v_h; \mu) = f(v_h) \qquad \forall v_h \in V_h$$

Projection-Based Model Order Reduction
(Reduced Basis Methods)

Full order model (basic example)

For given parameter $\mu \in \mathcal{P}$, find $u_h(\mu) \in V_h$ s.t. (dim $V_h > 10^5$)

$$a(u_h(\mu), v_h; \mu) = f(v_h) \qquad \forall v_h \in V_h$$

Reduced order model (via Galerkin projection)

For given $V_N \subset V_h$, find $u_N(\mu) \in V_N$ s.t. (dim $V_N \approx 10 - 100$)

$$a(u_N(\mu), v_N; \mu) = f(v_N) \qquad \forall v_N \in V_N$$

How to find V_N?

Weak greedy basis generation

function WEAK-GREEDY($\mathcal{S}_{train} \subset \mathcal{P}, \varepsilon$)
    $V_N \leftarrow \{0\}$
    while $\max_{\mu \in \mathcal{S}_{train}}$ ERR-EST(ROM-SOLVE(μ), μ) $> \varepsilon$ **do**
        $\mu^* \leftarrow \text{arg-max}_{\mu \in \mathcal{S}_{train}}$ ERR-EST(ROM-SOLVE(μ), μ)
        $V_N \leftarrow \text{span}(V_N \cup \{\text{FOM-SOLVE}(\mu^*)\})$
    end while
    return V_N
end function

ERR-EST

Use residual-based error estimate w.r.t. FOM (finite dimensional $\rightsquigarrow$ can compute dual norms).

- Use parameter separability / hyperreduction to gain online efficiency.

Example: MOR for Li-Ion Battery Models

ulm university universität ulm

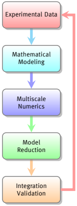
DLR Institute of Technical Thermodynamics

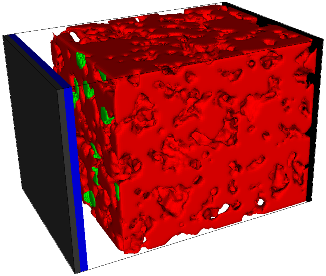
MULTIBAT

SPONSORED BY THE Federal Ministry of Education and Research

Fraunhofer ITWM

Westfälische Wilhelms-Universität Münster





MULTIBAT: Gain understanding of degradation processes in rechargeable Li-Ion Batteries through mathematical modeling and simulation at the pore scale.

FOM:

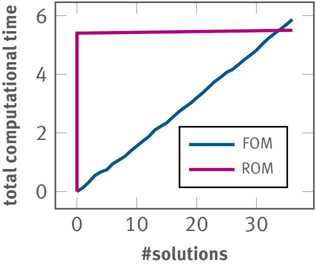
- 2.920.000 DOFs
- Simulation time: ≈ 15.5h

ROM:

- Snapshots: 3
- dim V_N = 245
- Rel. err.: < $4.5 \cdot 10^{-3}$
- Reduction time: ≈ 14h
- Simulation time: ≈ 8m
- Speedup: 120

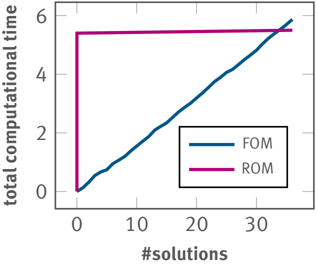
Caveats

- Potentially high offline time
- Especially when dim $\mathcal{P}$ large?

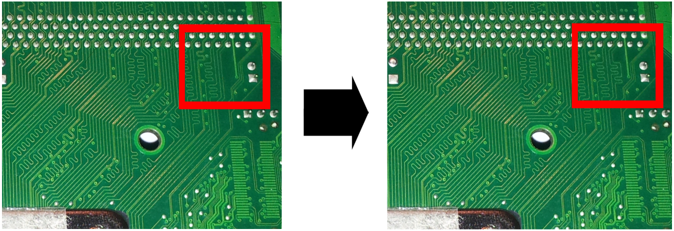


Caveats

- Potentially high offline time
- Especially when dim $\mathcal{P}$ large?

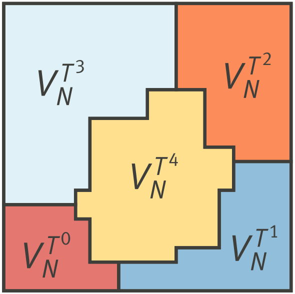


Scenario: Many parameters with only local influence / local non-parametric changes.



Localized MOR

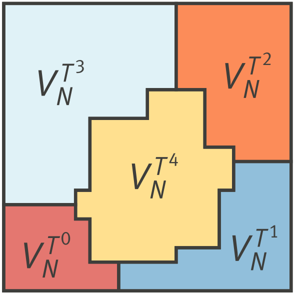
- coarse triangulation $\mathcal{T}_H$ of Ω
- build local reduced spaces V_N^T, $T \in \mathcal{T}_H$
- global reduced space $V_N = \bigoplus_{T \in \mathcal{T}_H} V_N^T$
- Various approaches:
  - overlapping / non-overlapping
  - different coupling approaches
  - interface spaces
  - ...



Localized MOR

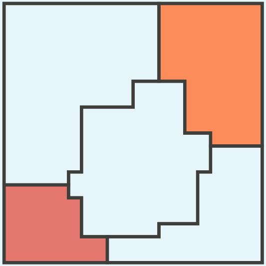
- coarse triangulation $\mathcal{T}_H$ of Ω
- build local reduced spaces V_N^T, $T \in \mathcal{T}_H$
- global reduced space $V_N = \bigoplus_{T\in\mathcal{T}_H} V_N^T$
- Various approaches:
  - overlapping / non-overlapping
  - different coupling approaches
  - interface spaces
  - ...

How to construct V_N^T?



Online-Adaptive Enrichment of V_N

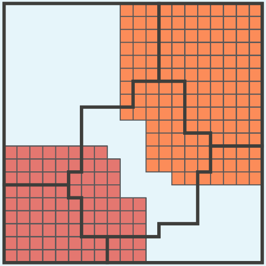
Enrichment algorithm — for some $\mu \in \mathcal{P}$



- compute reduced solution $u_N(\mu)$
- estimate error $\eta_{h,N}(\mu)$
- if $\eta_{h,N}(\mu) > \Delta$, start intermediate local enrichment phase:
  - compute local error indicators
  - mark subdomains for enrichment: $\mathcal{X} = \mathtt{mark}(\mathcal{T}_H)$

Online-Adaptive Enrichment of V_N

Enrichment algorithm for some $\mu \in \mathcal{P}$



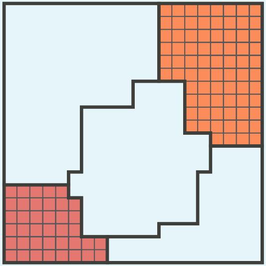
- compute reduced solution $u_N(\mu)$
- estimate error $\eta_{h,N}(\mu)$
- if $\eta_{h,N}(\mu) > \Delta$, start intermediate local enrichment phase:
  - compute local error indicators
  - mark subdomains for enrichment: $\mathcal{X} = \mathtt{mark}(\mathcal{T}_H)$
  - solve corrector problem on oversampling subdomain $T^\delta \supset T$ for all $T \in \mathcal{X}$:

$$a(\varphi_h(\mu), v_h; \mu) = f(v_h) \qquad \text{in } T^\delta$$

$$\varphi_h(\mu) = u_N(\mu) \qquad \text{on } \partial T^\delta$$

Online-Adaptive Enrichment of V_N

Enrichment algorithm — for some $\mu \in \mathcal{P}$



- compute reduced solution $u_N(\mu)$
- estimate error $\eta_{h,N}(\mu)$
- if $\eta_{h,N}(\mu) > \Delta$, start intermediate local enrichment phase:
  - compute local error indicators
  - mark subdomains for enrichment: $\mathcal{X} = \mathtt{mark}(\mathcal{T}_H)$
  - solve corrector problem on oversampling subdomain $T^\delta \supset T$ for all $T \in \mathcal{X}$:

$$a(\varphi_h(\mu), v_h; \mu) = f(v_h) \qquad \text{in } T^\delta$$

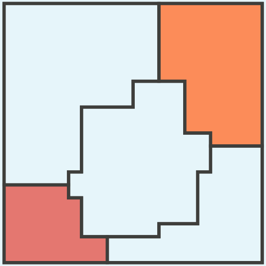
$$\varphi_h(\mu) = u_N(\mu) \qquad \text{on } \partial T^\delta$$

  - extend local reduced basis for all $T \in \mathcal{X}$:

$$V_N^T := \operatorname{span} V_N^T \cup \{ \varphi_h(\mu)|_T \}$$

Online-Adaptive Enrichment of V_N

Enrichment algorithm — for some $\mu \in \mathcal{P}$



- compute reduced solution $u_N(\mu)$
- estimate error $\eta_{h,N}(\mu)$
- if $\eta_{h,N}(\mu) > \Delta$, start intermediate local enrichment phase:
  - compute local error indicators
  - mark subdomains for enrichment: $\mathcal{X} = \mathtt{mark}(\mathcal{T}_H)$
  - solve corrector problem on oversampling subdomain $T^\delta \supset T$ for all $T \in \mathcal{X}$:

$$a(\varphi_h(\mu), v_h; \mu) = f(v_h) \qquad \text{in } T^\delta$$

$$\varphi_h(\mu) = u_N(\mu) \qquad \text{on } \partial T^\delta$$

  - extend local reduced basis for all $T \in \mathcal{X}$:

$$V_N^T := \operatorname{span} V_N^T \cup \{\varphi_h(\mu)|_T\}$$

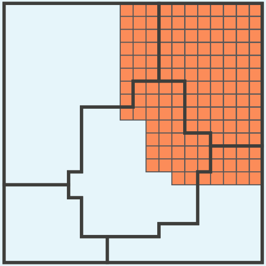
  - update reduced quantities
  - compute updated reduced solution $u_N(\mu)$ and $\eta_{h,N}(\mu)$
- iterate until $\eta_{h,N}(u_{\mu,N}) \leq \Delta$, return $u_N(\mu)$

Offline Initialization of V_N

Training algorithm for all $T \in \mathcal{T}_H$

Offline Initialization of V_N

Training algorithm — for all $T \in \mathcal{T}_H$



- For every $\mu \in \mathcal{S}_{train} \subset \mathcal{P}$:
  - Solve training problem on oversampling subdomain $T^\delta \supset T$:

$$a(\varphi_{h,0}(\mu), v_h; \mu) = f(v_h) \quad \text{in } T^\delta$$
$$\varphi_{h,0}(\mu) = 0 \quad \text{on } \partial T^\delta$$

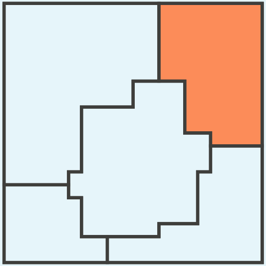
  - For $1 \leq k \leq K$, solve training problem:

$$a(\varphi_{h,k}(\mu), v_h; \mu) = 0 \quad \text{in } T^\delta$$
$$\varphi_{h,k}(\mu) = g_k \quad \text{on } \partial T^\delta$$

  for K random Dirichlet data functions g_k on ∂T^δ.

Offline Initialization of V_N

Training algorithm — for all $T \in \mathcal{T}_H$



- For every $\mu \in \mathcal{S}_{train} \subset \mathcal{P}$:
  - Solve training problem on oversampling subdomain $T^\delta \supset T$:
    $$a(\varphi_{h,0}(\mu), v_h; \mu) = f(v_h) \quad \text{in } T^\delta$$
    $$\varphi_{h,0}(\mu) = 0 \quad \text{on } \partial T^\delta$$
  - For $1 \leq k \leq K$, solve training problem:
    $$a(\varphi_{h,k}(\mu), v_h; \mu) = 0 \quad \text{in } T^\delta$$
    $$\varphi_{h,k}(\mu) = g_k \quad \text{on } \partial T^\delta$$
    for K random Dirichlet data functions g_k on ∂T^δ.
- Initialize local RB space on T as

$$V_N^T := \operatorname{span} \bigcup_{\mu \in \mathcal{S}_{train}} \left\{ \varphi_{h,0}(\mu)\big|_T, \ldots, \varphi_{h,K}(\mu)\big|_T \right\}.$$

More on Training

Transfer operator

$$\mathcal{T}_T \colon H^{1/2}(\partial T^\delta) \to H^1(T), \qquad \text{boundary values on } \partial T^\delta \mapsto \text{solution inside } T$$

- $\mathcal{T}_T$ is compact!
- "Optimal" V_N^T spanned by right-singular vectors of $\mathcal{T}_T$.
- Randomized training $\rightsquigarrow$ Radomized SVD of $\mathcal{T}_T$ [Buhr, Smetana, 2018]

More on Training

Transfer operator

$$\mathcal{T}_T \colon H^{1/2}(\partial T^\delta) \to H^1(T), \qquad \text{boundary values on } \partial T^\delta \mapsto \text{solution inside } T$$

- $\mathcal{T}_T$ is compact!
- "Optimal" V_N^T spanned by right-singular vectors of $\mathcal{T}_T$.
- Randomized training $\rightsquigarrow$ Radomized SVD of $\mathcal{T}_T$ [Buhr, Smetana, 2018]

Related ideas:

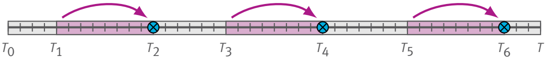
- Cell-problems in multiscale methods (HMM, (G)MsFEM, LOD, etc.)
- GFEM [Babuska, Lipton, 2011]
- Spectral coarse spaces

Localized MOR in Time

Transfer operator in time

$$\mathcal{T}_{T_n \to T_{n+1}} : L^2(\Omega) \to L^2(\Omega), \qquad \text{initial values at } T_n \mapsto \text{solution at } T_{n+1}$$

- For parabolic problems, $\mathcal{T}_{T_n \to T_{n+1}}$ is compact.
- [Schleuß, Smetana, 2023]:
  - $V_N := \{\text{right-singular vectors of } \mathcal{T}_{T_n \to T_{n+1}} \,|\, n = 1, \ldots N-1\}$
  - Use randomized SVD.
  - Select T_n based on PDE coefficients.

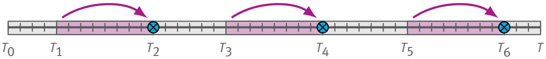


Localized MOR in Time

Transfer operator in time

$$\mathcal{T}_{T_n \to T_{n+1}} : L^2(\Omega) \to L^2(\Omega), \qquad \text{initial values at } T_n \mapsto \text{solution at } T_{n+1}$$

- For parabolic problems, $\mathcal{T}_{T_n \to T_{n+1}}$ is compact.
- [Schleuß, Smetana, 2023]:
  - $V_N := \{\text{right-singular vectors of } \mathcal{T}_{T_n \to T_{n+1}} \mid n = 1, \dots N-1\}$
  - Use randomized SVD.
  - Select T_n based on PDE coefficients.



- Iterative scheme to converge to arbitrary precision?

Parareal algorithm

Solve $\partial_t u(t) = f(t, u(t))$ using:

$F_n u := F(u, T_{n-1}, T_n)$ fine solver (accurate, but slow)

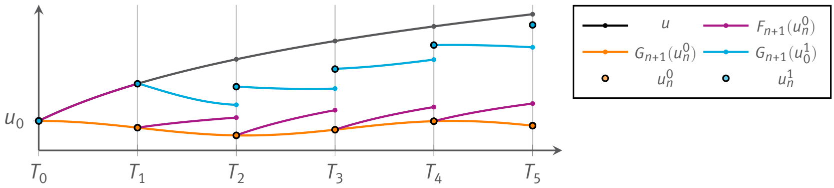
$G_n u := G(u, T_{n-1}, T_n)$ coarse solver (fast, but inaccurate)

Parareal iteration

$u_0^0 := u_0, \quad u_{n+1}^0 := G_{n+1} u_n^0$ $\qquad 0 \le n < N$

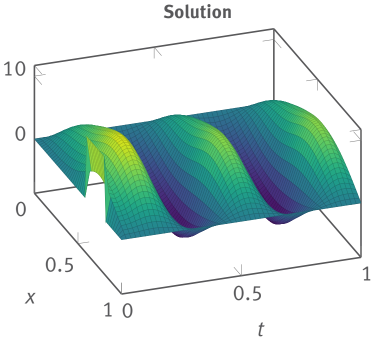
$u_{n+1}^{k+1} := F_{n+1} u_n^k + G_{n+1} u_n^{k+1} - G_{n+1} u_n^k$ $\qquad 0 \le n < N, k \in \mathbb{N}_0$

F_n can be computed in parallel!



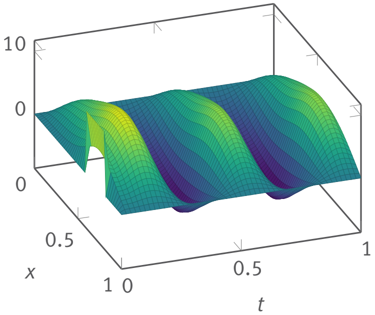
Example: Heat Equation

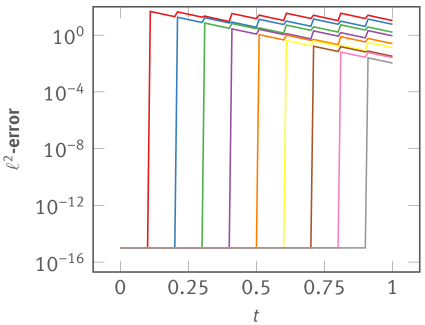
$$u_t(t,x) - u_{xx}(t,x) = 100 \cdot \sin(5\pi t)(1 + \cos(3\pi x)) \qquad x \in (0,1)$$
$$u(0,x) = u_0(x) = 10\chi_{[0.6,0.8]} \qquad t \in [0,T]$$
$$u(0,t) = u(1,t) = 0$$



Example: Heat Equation

$$u_t(t,x) - u_{xx}(t,x) = 100 \cdot \sin(5\pi t)(1 + \cos(3\pi x)) \qquad x \in (0,1)$$
$$u(0,x) = u_0(x) = 10\chi_{[0.6,0.8]} \qquad t \in [0,T]$$
$$u(0,t) = u(1,t) = 0$$





Example: Heat Equation

Exact solution:

$$u(x,t) = \sum_{m=1}^{\infty} \hat{u}_m(t)\sqrt{2}\sin(m\pi x)$$

$$\hat{u}_m(t) = \hat{u}_{0,m} e^{-m^2\pi^2 t} + \int_0^t \hat{f}_m(\tau) e^{-m^2\pi^2(t-\tau)} \mathrm{d}\tau,$$

Example: Heat Equation

Exact solution:

$$u(x,t) = \sum_{m=1}^{\infty} \hat{u}_m(t)\sqrt{2}\sin(m\pi x)$$

$$\hat{u}_m(t) = \hat{u}_{0,m} e^{-m^2\pi^2 t} + \int_0^t \hat{f}_m(\tau) e^{-m^2\pi^2(t-\tau)} \mathrm{d}\tau,$$

Coarse solver:

$$G_n u := \sum_{m=1}^{R} \hat{u}_m(T_n)\sqrt{2}\sin(m\pi x)$$

$$\hat{u}_m(T_n) := \hat{u}_m e^{-m^2\pi^2(T_n - T_{n-1})}$$

Example: Heat Equation

Exact solution:

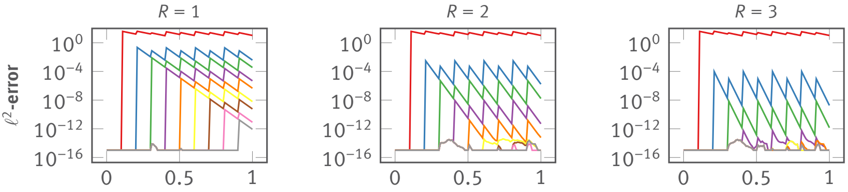
$$u(x,t) = \sum_{m=1}^{\infty} \hat{u}_m(t)\sqrt{2}\sin(m\pi x)$$

$$\hat{u}_m(t) = \hat{u}_{0,m}e^{-m^2\pi^2 t} + \int_0^t \hat{f}_m(\tau)e^{-m^2\pi^2(t-\tau)}\mathrm{d}\tau,$$

Coarse solver:

$$G_n u := \sum_{m=1}^{R} \hat{u}_m(T_n)\sqrt{2}\sin(m\pi x)$$

$$\hat{u}_m(T_n) := \hat{u}_m e^{-m^2\pi^2(T_n - T_{n-1})}$$



Example: Heat Equation

Exact solution:

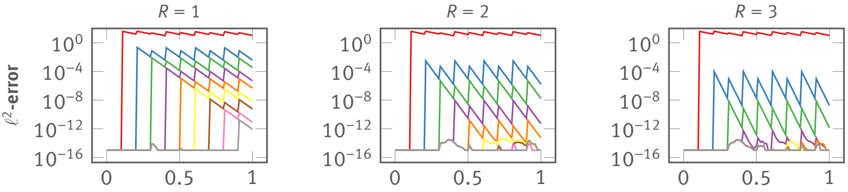
$$u(x,t) = \sum_{m=1}^{\infty} \hat{u}_m(t)\sqrt{2}\sin(m\pi x)$$

$$\hat{u}_m(t) = \hat{u}_{0,m}e^{-m^2\pi^2 t} + \int_0^t \hat{f}_m(\tau)e^{-m^2\pi^2(t-\tau)}\mathrm{d}\tau,$$

Coarse solver:

$$G_n u := \sum_{m=1}^{R} \hat{u}_m(T_n)\sqrt{2}\sin(m\pi x)$$

$$\hat{u}_m(T_n) := \hat{u}_m e^{-m^2\pi^2(T_n - T_{n-1})}$$



A priori error bound (time-invariant, self-adjoint case)

$$\max_{1\leq n\leq N} \|e_n^k\| \leq \left(\sup_{\lambda\in\sigma(F')\setminus\Gamma} |\lambda|\right)^k \cdot \max_{1\leq m\leq N-k} \|e_m^0\|$$

Parareal with Spectral Coarse Solver

- V Hilbert space. $F_n\colon V \to V$ be compact and affine linear:

$$F_n v = F_n' v + b_n, \qquad b_n := F_n 0$$

  (linear parabolic PDE with time-varying coefficients)

- SVD of F_n':

$$F_n' v = \sum_{r=1}^{\operatorname{rank} F_n'} \sigma_{n,r} \cdot (\varphi_{n,r}, v)_V \cdot \psi_{n,r}.$$

Parareal with Spectral Coarse Solver

- V Hilbert space. $F_n\colon V \to V$ be compact and affine linear:

$$F_n v = F_n' v + b_n, \qquad b_n := F_n 0$$

(linear parabolic PDE with time-varying coefficients)

- SVD of F_n':

$$F_n' v = \sum_{r=1}^{\operatorname{rank} F_n'} \sigma_{n,r} \cdot (\varphi_{n,r}, v)_V \cdot \psi_{n,r}.$$

Spectral coarse solver

$$G_n v := \sum_{r=1}^{R_n} \sigma_{n,r} \cdot (\phi_{n,r}, v) \cdot \psi_{n,r} + b_n.$$

Approximation error

$$\|F_n - G_n\| = \sigma_{n,R_n+1}.$$

Computing G_n via Randomized SVD

1. $W := \operatorname{span}\{F_n'\omega_1, \dots, F_n'\omega_{R_n+p}\}$, ω_i randomly chosen

Computing G_n via Randomized SVD

1. $W := \operatorname{span}\{F_n'\omega_1, \dots, F_n'\omega_{R_n+p}\}$, ω_i randomly chosen
2. $w_1, \dots, w_{R_n+p}$ ONB of W

Computing G_n via Randomized SVD

1. $W := \operatorname{span}\{F_n'\omega_1, \dots, F_n'\omega_{R_n+p}\}$, ω_i randomly chosen
2. $w_1, \dots, w_{R_n+p}$ ONB of W
3. $X := \operatorname{span}\{F_n'^* w_1, \dots, F_n'^* w_{R_n+p}\}$

Computing G_n via Randomized SVD

1. $W := \operatorname{span}\{F_n'\omega_1, \dots, F_n'\omega_{R_n+p}\}$, ω_i randomly chosen
2. $w_1, \dots, w_{R_n+p}$ ONB of W
3. $X := \operatorname{span}\{F_n'^* w_1, \dots, F_n'^* w_{R_n+p}\}$
4. $v_1, \dots, v_{R_n+p}$ ONB for X

$$F_n' v \approx P_W F_n' v = \sum_{i=1}^{R_n+p} w_i \cdot (w_i, F_n' v)_V = \sum_{i=1}^{R_n+p} w_i \cdot (F_n'^* w_i, v)_V = \sum_{i,j=1}^{R_n+p} w_i \cdot (F_n'^* w_i, v_j)_V \cdot (v_j, v)_V.$$

Computing G_n via Randomized SVD

1. $W := \operatorname{span}\{F_n' \omega_1, \dots, F_n' \omega_{R_n+p}\}$, ω_i randomly chosen
2. $w_1, \dots, w_{R_n+p}$ ONB of W
3. $X := \operatorname{span}\{F_n'^* w_1, \dots, F_n'^* w_{R_n+p}\}$
4. $v_1, \dots, v_{R_n+p}$ ONB for X

$$F_n' v \approx P_W F_n' v = \sum_{i=1}^{R_n+p} w_i \cdot (w_i, F_n' v)_V = \sum_{i=1}^{R_n+p} w_i \cdot (F_n'^* w_i, v)_V = \sum_{i,j=1}^{R_n+p} w_i \cdot (F_n'^* w_i, v_j)_V \cdot (v_j, v)_V.$$

5. SVD of $M \in R^{(R_n+p)\times(R_n+p)}$, $M_{i,j} := (F_n'^* w_i, v_j)_V$ with singular values/vectors $\sigma_r, \underline{\psi}_r, \underline{\varphi}_r$.

Computing G_n via Randomized SVD

1. $W := \operatorname{span}\{F_n'\omega_1, \dots, F_n'\omega_{R_n+p}\}$, ω_i randomly chosen
2. $w_1, \dots, w_{R_n+p}$ ONB of W
3. $X := \operatorname{span}\{F_n'^* w_1, \dots, F_n'^* w_{R_n+p}\}$
4. $v_1, \dots, v_{R_n+p}$ ONB for X

$$F_n' v \approx P_W F_n' v = \sum_{i=1}^{R_n+p} w_i \cdot (w_i, F_n' v)_V = \sum_{i=1}^{R_n+p} w_i \cdot (F_n'^* w_i, v)_V = \sum_{i,j=1}^{R_n+p} w_i \cdot (F_n'^* w_i, v_j)_V \cdot (v_j, v)_V.$$

5. SVD of $M \in R^{(R_n+p)\times(R_n+p)}$, $M_{i,j} := (F_n'^* w_i, v_j)_V$ with singular values/vectors $\sigma_r, \underline{\psi}_r, \underline{\varphi}_r$.
6. Return

$$\sigma_r, \quad \varphi_r := \sum_{i=1}^{R_n+p} \underline{\varphi}_{r,i} \cdot w_i, \quad \psi_r := \sum_{i=1}^{R_n+p} \underline{\psi}_{r,i} \cdot v_i \qquad 1 \le r \le R_n.$$

Computing G_n via Randomized SVD

1. $W := \operatorname{span}\{F_n'\omega_1, \dots, F_n'\omega_{R_n+p}\}$, ω_i randomly chosen
2. $w_1, \dots, w_{R_n+p}$ ONB of W
3. $X := \operatorname{span}\{F_n'^* w_1, \dots, F_n'^* w_{R_n+p}\}$
4. $v_1, \dots, v_{R_n+p}$ ONB for X

$$F_n' v \approx P_W F_n' v = \sum_{i=1}^{R_n+p} w_i \cdot (w_i, F_n' v)_V = \sum_{i=1}^{R_n+p} w_i \cdot (F_n'^* w_i, v)_V = \sum_{i,j=1}^{R_n+p} w_i \cdot (F_n'^* w_i, v_j)_V \cdot (v_j, v)_V.$$

5. SVD of $M \in R^{(R_n+p)\times(R_n+p)}$, $M_{i,j} := (F_n'^* w_i, v_j)_V$ with singular values/vectors $\sigma_r, \underline{\psi}_r, \underline{\varphi}_r$.
6. Return

$$\sigma_r, \quad \varphi_r := \sum_{i=1}^{R_n+p} \underline{\varphi}_{r,i} \cdot w_i, \quad \psi_r := \sum_{i=1}^{R_n+p} \underline{\psi}_{r,i} \cdot v_i \qquad 1 \le r \le R_n.$$

Computational effort

$R_n + p + 1$ eval. of F_n (embarrasingly parallel) and $R_n + p$ eval. of F_n^* (embarassingly parallel)

A Priori Error Bounds

Superlinear convergence

Let

$$\delta = \max_{1 \le n \le N} \sigma_{n,1} \qquad \varepsilon = \max_{1 \le n \le N} \sigma_{n,R_n+1}$$

Then:

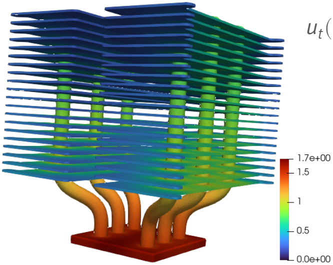
$$\begin{aligned} \|e_n^k\| &\le \varepsilon^k \sum_{m=1}^{n-k} \binom{n-m}{k-1} \delta^{n-m-k} \|e_m^0\| \\ &\le 2\varepsilon^k \sum_{m=1}^{n-k-1} \binom{n-m}{k} \delta^{n-m-k} \|b_m\| \end{aligned}$$

Linear convergence (long time)

If $\delta < 1$, we have for $k \in \mathbb{N}$:

$$\max_{1 \le n \le N} \|e_n^k\| \le \left(\frac{\varepsilon}{1-\delta}\right)^k \max_{1 \le n \le N-k} \|e_n^0\|$$

Example: Heat conduction with time-varying Robin boundary



$$
\begin{aligned}
u_t(t,\mathbf{x}) - \nabla \cdot [d(\mathbf{x})\nabla_{\mathbf{x}} u(\mathbf{x})] &= 0 && \mathbf{x} \in \Omega,\ t \in [0,1] \\
-d(\mathbf{x})\nabla_{\mathbf{x}} u(t,\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) &= (\tfrac{1}{2} + t) \cdot u(t,\mathbf{x}) && \mathbf{x} \in \Gamma_{\text{fin}} \\
-d(\mathbf{x})\nabla_{\mathbf{x}} u(t,\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) &= g(t) && \mathbf{x} \in \Gamma_{\text{bot}} \\
-d(\mathbf{x})\nabla_{\mathbf{x}} u(t,\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) &= 0 && \mathbf{x} \in \Gamma \setminus (\Gamma_{\text{fin}} \cup \Gamma_{\text{bot}}) \\
u(0,\mathbf{x}) &= 0.
\end{aligned}
$$

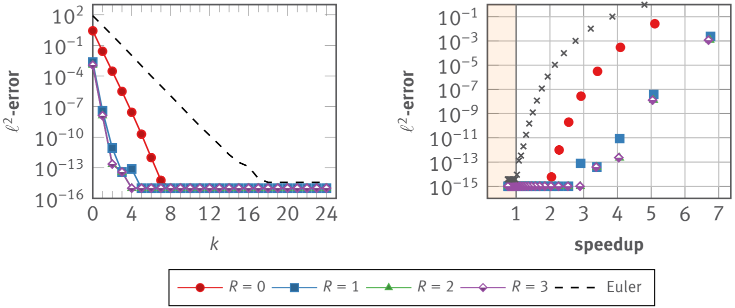
$$
d(\mathbf{x}) = \begin{cases} 10 & \mathbf{x} \in \Omega_{\text{fin}} \\ 100 & \mathbf{x} \in \Omega_{\text{base}} \\ 1000 & \mathbf{x} \in \Omega_{\text{pipe}}, \end{cases}
$$

- P1 simplicial FEs
- 444,693 DOFs
- $N = 25$

$$
g(t) = \begin{cases} 50 \cdot \frac{t}{0.3} & t \leq 0.3 \\ 50 \cdot \left(1 + \operatorname{sign}\left(\sin\left(\frac{t-0.3}{0.3} \cdot 8 \cdot \pi\right)\right)\right) & 0.3 < t \leq 0.6 \\ 50 \cdot \left(1 + \cos\left(\frac{t-0.6}{0.4} \cdot 20 \cdot \pi\right)\right) & 0.6 < t. \end{cases}
$$

Example: Heat conduction with time-varying Robin boundary

- Spectral G_n with $p = 1$ compared to G_n = single backward Euler step.
- $R = 2$: Error of 10^{13} at $k = 2$ iterations.
- Choose R to tune parallelism vs. computational work.
- **Less** F_n evaluations needed as for Euler.



A Posteriori Error Bounds

Error bound

$$\|e_n^k\| \leq \varepsilon \sum_{m=1}^{n-1} \delta^{n-m-1} \|u_m^k - u_m^{k-1}\|$$

- Easily computable (δ, ε known from SVDs).
- Rigorous when randomized SVD error taken into account.

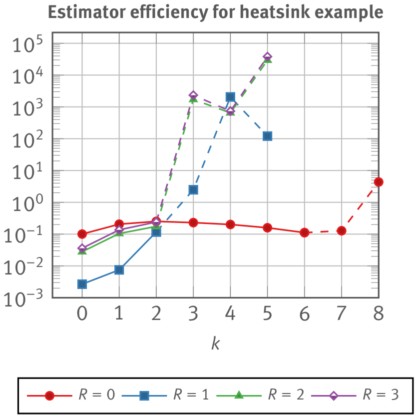


Figure: dashed plot when error below 10^{-13}

Thank you for your attention!

Gander, Ohlberger, R. A Parareal algorithm without Coarse Propagator?
arXiv:2409.02673

Gander, Ohlberger, R. A Parareal algorithm with Spectral Coarse Solver.
in preparation

Slides: `https://stephanrave.de/talks/dd29.pdf`