A Space-Time Multigrid Method for Space-Time Finite-Element Discretizations

SIAM CSE 2025, Fort Worth, Texas, U.S.

Peter Munch[†], Nils Margenberg[‡]

[†]Institute of Mathematics, Technical University of Berlin, Germany
[‡]Chair of Numerical Mathematics, Helmut Schmidt University Hamburg, Germany

March 7, 2025

Part 1:

Introduction

(Tensor-product) space-time FEM

Idea: time-dependent PDE

- space: standard continuous Lagrange finite element
- time: use DG (dG(k)) or FEM (cGP(k))

Advantages:

- variational time discretization $\to$ natural integration with the variational space discretization and natural capture of coupled problems and nonlinearities
- advantageous for duality and goal-oriented adaptivity in space and time [Schmich and Vexler '08, Bause et al. '21, Besier & Rannacher 2012, Roth et al. 2023]
- unified approach to stability and error analysis [Matthies and Schieweck '11]
- solves multiple time steps at once $\to$ relation to "parallel in time" algorithms [Gander '15, Ong and Schroder '20, Falgout et al. '14, '17]

Considered equations

- heat equation

$$\partial_t u - \nabla \cdot (\rho \nabla u) = f$$

- wave equations

$$\partial_t u - v = 0, \quad \partial_t v - \nabla \cdot (\rho \nabla u) = f$$

- convection-diffusion-reaction equation

$$\partial_t u - \nabla \cdot (\varepsilon \nabla u) + b \cdot \nabla u + \alpha u = f$$

- Stokes equations

$$\partial_t \boldsymbol{v} - \nu \Delta \boldsymbol{v} + \nabla p = \boldsymbol{f}, \quad \nabla \cdot \boldsymbol{v} = 0$$

- Navier–Stokes equations (WIP)

$$\partial_t \boldsymbol{u} - \nu \Delta \boldsymbol{u} + (\boldsymbol{u} \cdot \nabla) \boldsymbol{u} + \nabla p = \boldsymbol{f}, \quad \nabla \cdot \boldsymbol{u} = 0$$

Considered equations

- heat equation

$$\partial_t u - \nabla \cdot (\rho \nabla u) = f$$

- wave equations

$$\partial_t u - v = 0, \quad \partial_t v - \nabla \cdot (\rho \nabla u) = f$$

- convection-diffusion-reaction equation

$$\partial_t u - \nabla \cdot (\varepsilon \nabla u) + b \cdot \nabla u + \alpha u = f$$

- Stokes equations

$$\partial_t \boldsymbol{v} - \nu \Delta \boldsymbol{v} + \nabla p = \boldsymbol{f}, \quad \nabla \cdot \boldsymbol{v} = 0$$

- Navier–Stokes equations (WIP)

$$\partial_t \boldsymbol{u} - \nu \Delta \boldsymbol{u} + (\boldsymbol{u} \cdot \nabla) \boldsymbol{u} + \nabla p = \boldsymbol{f}, \quad \nabla \cdot \boldsymbol{u} = 0$$

Considered equations

- heat equation

$$\partial_t u - \nabla \cdot (\rho \nabla u) = f$$

- convection-diffusion-reaction equation

$$\partial_t u - \nabla \cdot (\varepsilon \nabla u) + b \cdot \nabla u + \alpha u = f$$

- Stokes equations

$$\partial_t \boldsymbol{v} - \nu \Delta \boldsymbol{v} + \nabla p = \boldsymbol{f}, \quad \nabla \cdot \boldsymbol{v} = 0$$

- Navier–Stokes equations (WIP)

$$\partial_t \boldsymbol{u} - \nu \Delta \boldsymbol{u} + (\boldsymbol{u} \cdot \nabla)\boldsymbol{u} + \nabla p = \boldsymbol{f}, \quad \nabla \cdot \boldsymbol{u} = 0$$

- wave equations

$$\partial_t u - v = 0, \quad \partial_t v - \nabla \cdot (\rho \nabla u) = f$$

Table of content:

1. space-time multigrid
2. block preconditioning

Considered equations

- heat equation

$$\partial_t u - \nabla \cdot (\rho \nabla u) = f$$

- convection-diffusion-reaction equation

$$\partial_t u - \nabla \cdot (\varepsilon \nabla u) + b \cdot \nabla u + \alpha u = f$$

- Stokes equations

$$\partial_t \boldsymbol{v} - \nu \Delta \boldsymbol{v} + \nabla p = \boldsymbol{f}, \quad \nabla \cdot \boldsymbol{v} = 0$$

- Navier–Stokes equations (WIP)

$$\partial_t \boldsymbol{u} - \nu \Delta \boldsymbol{u} + (\boldsymbol{u} \cdot \nabla) \boldsymbol{u} + \nabla p = \boldsymbol{f}, \quad \nabla \cdot \boldsymbol{u} =$$

- wave equations

$$\partial_t u - v = 0, \quad \partial_t v - \nabla \cdot (\rho \nabla u) = f$$

Table of content:

1. space-time multigrid
2. block preconditioning

N. Margenberg and PM, “A space-time multigrid method for space-time finite element discretizations of parabolic and hyperbolic PDEs”, submitted, 2024.

N. Margenberg, M. Bause, and PM, “An *hp* multigrid approach for tensor-product space-time finite element discretizations of the Stokes equations”, submitted, 2025.

Part 2:

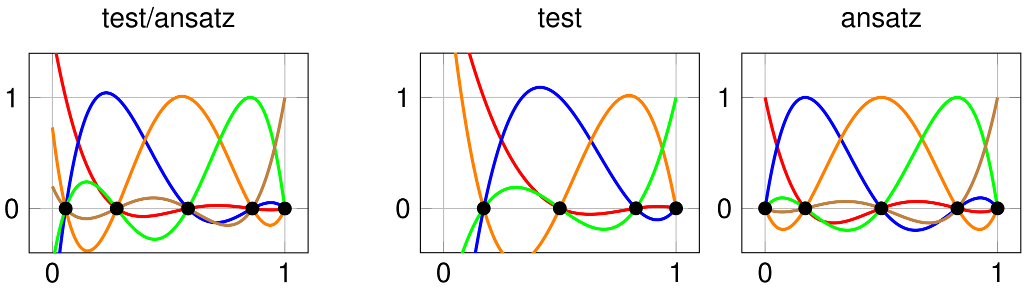
Solution procedures

Tensor-product space-time FEM

Idea: time-dependent PDE

- space: standard continuous Lagrange finite element
- time: use DG (dG(k)) or FEM (cGP(k))

dG(k) **cGP(k)**



jump: $[w_{\tau,h}] = w_n^+ - w_n^-$

continuity condition: $w_n^+ = w_n^-$

Algebraic system for dG(k) discretization of the heat equation

Local algebraic system at n-th time step

$$\underbrace{(\boldsymbol{M}_\tau \otimes \boldsymbol{A}_h + \boldsymbol{A}_\tau \otimes \boldsymbol{M}_h)}_{:=\boldsymbol{S}} \boldsymbol{u}_n = \boldsymbol{M}_\tau \otimes \boldsymbol{M}_h \boldsymbol{f}_n + \alpha \otimes \boldsymbol{M}_h \boldsymbol{u}_{n-1}^{N_t}$$

with $(\boldsymbol{M}_\tau)_{i,j} := \tau \int_{\hat{I}} \hat{\xi}_j(\hat{t})\hat{\xi}_i(\hat{t})\,\mathrm{d}\hat{t}$, $\quad (\boldsymbol{A}_\tau)_{i,j} := \int_{\hat{I}} \hat{\xi}_j'(\hat{t})\hat{\xi}_i(\hat{t})\,\mathrm{d}\hat{t} + \hat{\xi}_j(0)\hat{\xi}_i(0)$, $\quad \alpha_i := \hat{\xi}_i(0)$.

Multiple-time-steps system

Let $\boldsymbol{B} := \boldsymbol{1}_{k+1} \otimes \alpha \otimes \boldsymbol{M}_h$, then we collect consecutive time steps $n_1, \ldots, n_c$

$$\begin{pmatrix} \boldsymbol{S} & & & & \\ -\boldsymbol{B} & \boldsymbol{S} & & & \\ & \ddots & \ddots & & \\ & & -\boldsymbol{B} & \boldsymbol{S} & \\ & & & -\boldsymbol{B} & \boldsymbol{S} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}_{n_1} \\ \boldsymbol{u}_{n_2} \\ \vdots \\ \boldsymbol{u}_{n_c-1} \\ \boldsymbol{u}_{n_c} \end{pmatrix} = \begin{pmatrix} \boldsymbol{M}_\tau \otimes \boldsymbol{M}_h \boldsymbol{f}_{n_1} + \alpha \otimes \boldsymbol{M}_h \boldsymbol{u}_{n_1-1}^{N_t} \\ \boldsymbol{M}_\tau \otimes \boldsymbol{M}_h \boldsymbol{f}_{n_2} \\ \vdots \\ \boldsymbol{M}_\tau \otimes \boldsymbol{M}_h \boldsymbol{f}_{n_c-1} \\ \boldsymbol{M}_\tau \otimes \boldsymbol{M}_h \boldsymbol{f}_{n_c} \end{pmatrix}.$$

Algebraic system for cGP(k) discretization of the heat equation

Local algebraic system at n-th time step

$$\underbrace{\left(\boldsymbol{M}_\tau \otimes \boldsymbol{A}_h + \boldsymbol{A}_\tau \otimes \boldsymbol{M}_h\right)}_{:=\boldsymbol{S}} \boldsymbol{u}_n = \boldsymbol{M}_\tau \otimes \boldsymbol{M}_h \boldsymbol{f}_n - \beta \otimes \boldsymbol{M}_h \boldsymbol{f}_{n-1}^{N_t} + \underbrace{\left(\beta \otimes \boldsymbol{A}_h + \alpha \otimes \boldsymbol{M}_h\right)}_{:=\boldsymbol{b}} \boldsymbol{u}_{n-1}^{N_t}$$

with $(\boldsymbol{M}_\tau)_{i,j-1} := \tau \int_{\hat{I}} \hat{\xi}_j(\hat{t}) \hat{\psi}_i(\hat{t})\,\mathrm{d}\hat{t}$, $\quad (\boldsymbol{A}_\tau)_{i,j-1} := \int_{\hat{I}} \hat{\xi}_j'(\hat{t}) \hat{\psi}_i(\hat{t})\,\mathrm{d}\hat{t}$,

$$\beta_i := \tau \int_{\hat{I}} \hat{\xi}_1(\hat{t}) \hat{\psi}_i(\hat{t})\,\mathrm{d}\hat{t}, \quad \alpha_i := \int_{\hat{I}} \hat{\xi}_1'(\hat{t}) \hat{\psi}_i(\hat{t})\,\mathrm{d}\hat{t}, \quad i = 1, \ldots, k,\ j = 2, \ldots, k+1$$

Multiple-time-steps system

Let $\boldsymbol{B} := \boldsymbol{1}_k \otimes \boldsymbol{b}$, then we collect consecutive time steps $n_1, \ldots, n_c$

$$\begin{pmatrix} \boldsymbol{S} & & & & \\ -\boldsymbol{B} & \boldsymbol{S} & & & \\ & \ddots & \ddots & & \\ & & -\boldsymbol{B} & \boldsymbol{S} & \\ & & & -\boldsymbol{B} & \boldsymbol{S} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}_{n_1} \\ \boldsymbol{u}_{n_2} \\ \vdots \\ \boldsymbol{u}_{n_c-1} \\ \boldsymbol{u}_{n_c} \end{pmatrix} = \begin{pmatrix} \boldsymbol{M}_\tau \otimes \boldsymbol{M}_h \boldsymbol{f}_{n_1} - \beta \otimes \boldsymbol{M}_h \boldsymbol{f}_{n_1-1}^{N_t} + \boldsymbol{b} \otimes u_{n_1-1}^{N_t} \\ \boldsymbol{M}_\tau \otimes \boldsymbol{M}_h \boldsymbol{f}_{n_2} - \beta \otimes \boldsymbol{M}_h \boldsymbol{f}_{n_1}^{N_t} \\ \vdots \\ \boldsymbol{M}_\tau \otimes \boldsymbol{M}_h \boldsymbol{f}_{n_c-1} - \beta \otimes \boldsymbol{M}_h \boldsymbol{f}_{n_c-2}^{N_t} \\ \boldsymbol{M}_\tau \otimes \boldsymbol{M}_h \boldsymbol{f}_{n_c} - \beta \otimes \boldsymbol{M}_h \boldsymbol{f}_{n_c-1}^{N_t} \end{pmatrix}.$$

Space-time multigrid

To solve

$$\begin{pmatrix} \boldsymbol{S} & & & & \\ -\boldsymbol{B} & \boldsymbol{S} & & & \\ & \ddots & \ddots & & \\ & & -\boldsymbol{B} & \boldsymbol{S} & \\ & & & -\boldsymbol{B} & \boldsymbol{S} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}_{n_1} \\ \boldsymbol{u}_{n_2} \\ \vdots \\ \boldsymbol{u}_{n_c-1} \\ \boldsymbol{u}_{n_c} \end{pmatrix} = \begin{pmatrix} \cdots \\ \cdots \\ \vdots \\ \cdots \\ \cdots \end{pmatrix}.$$

we use GMRES with space-time multigrid [Hackbusch ’85, Gander and Neumüller ’16]:

- h- and p-multigrid both in space and time
- first coarsen p and then h
- simultaneously coarsen in space and time
- smoother: additive Schwarz (element-centric patches)
  $\rightarrow$ full matrices with $O(kp^d)$ rows/columns

Space-time multigrid

Software: deal.II

To solve

$$\begin{pmatrix} \boldsymbol{S} & & & & \\ -\boldsymbol{B} & \boldsymbol{S} & & & \\ & \ddots & \ddots & & \\ & & -\boldsymbol{B} & \boldsymbol{S} & \\ & & & -\boldsymbol{B} & \boldsymbol{S} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}_{n_1} \\ \boldsymbol{u}_{n_2} \\ \vdots \\ \boldsymbol{u}_{n_c-1} \\ \boldsymbol{u}_{n_c} \end{pmatrix} = \begin{pmatrix} \cdots \\ \cdots \\ \vdots \\ \cdots \\ \cdots \end{pmatrix}.$$

we use GMRES with space-time multigrid [Hackbusch '85, Gander and Neumüller '16]:

- h- and p-multigrid both in space and time
- first coarsen p and then h
- simultaneously coarsen in space and time
- smoother: additive Schwarz (element-centric patches)
  $\rightarrow$ full matrices with $O(kp^d)$ rows/columns

Space-time multigrid

Software: deal.II

To solve

$$\begin{pmatrix} \boldsymbol{S} & & & & \\ -\boldsymbol{B} & \boldsymbol{S} & & & \\ & \ddots & \ddots & & \\ & & -\boldsymbol{B} & \boldsymbol{S} & \\ & & & -\boldsymbol{B} & \boldsymbol{S} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}_{n_1} \\ \boldsymbol{u}_{n_2} \\ \vdots \\ \boldsymbol{u}_{n_c-1} \\ \boldsymbol{u}_{n_c} \end{pmatrix} = \begin{pmatrix} \cdots \\ \cdots \\ \vdots \\ \cdots \\ \cdots \end{pmatrix}.$$

we use GMRES with space-time multigrid [Hackbusch ’85, Gander and Neumüller ’16]:

- h- and p-multigrid both in space and time
- first coarsen p and then h
- simultaneously coarsen in space and time
- smoother: additive Schwarz (element-centric patches)
  $\rightarrow$ full matrices with $O(kp^d)$ rows/columns

Next:

1. evaluation of $\boldsymbol{S}$
2. transfer operator

Space-time multigrid: Matrix-free operator evaluation

Operator $\boldsymbol{S} = (\boldsymbol{M}_\tau \otimes \boldsymbol{A}_h + \boldsymbol{A}_\tau \otimes \boldsymbol{M}_h)$ is never assembled but directly applied to $\boldsymbol{u}_n$:

$$\boldsymbol{v} = \boldsymbol{S}\boldsymbol{u} = (\boldsymbol{M}_\tau \otimes \boldsymbol{I}_h)(\boldsymbol{I}_\tau \otimes \boldsymbol{A}_h)\boldsymbol{u} + (\boldsymbol{A}_\tau \otimes \boldsymbol{I}_h)(\boldsymbol{I}_\tau \otimes \boldsymbol{M}_h)\boldsymbol{u}$$

implying two steps:

1. apply $\boldsymbol{A}_h/\boldsymbol{M}_h$ to each block $\triangleright$ $(\boldsymbol{I}_\tau \otimes \boldsymbol{A}_h)$, $(\boldsymbol{I}_\tau \otimes \boldsymbol{M}_h)$
2. compute linear combination using $\boldsymbol{A}_\tau/\boldsymbol{M}_\tau$. $\triangleright$ $(\boldsymbol{M}_\tau \otimes \boldsymbol{I}_h)$, $(\boldsymbol{A}_\tau \otimes \boldsymbol{I}_h)$

Furthermore: application of $\boldsymbol{A}_h/\boldsymbol{M}_h$ is efficiently implemented in a matrix-free way [Kronbichler & Kormann, ’12].

Space-time multigrid: transfer operators

Heart of deal.II's multigrid infrastructure: transfer operators

base transfer class

geometric transfer — Clevenger et al. '20, PM et al. '23

polynomial transfer — PM et al. '23

non-nested transfer — Feder et al. '24

time transfer

also working for multivectors: $\boldsymbol{u}^{(f)} = (\boldsymbol{I}_\tau \otimes \boldsymbol{P}_h)\boldsymbol{u}^{(c)}$

Time transfer:

- prolongation as operation on multivectors

$$\boldsymbol{u}^{(f)} = (\boldsymbol{P}_\tau \otimes \boldsymbol{I}_h)\boldsymbol{u}^{(c)}$$

... L^2 projection; $\boldsymbol{P}_\tau$: different for geometric/polynomial coarsening

- restriction as adjoint of prolongation operator

Part 3:

Application: heat equation

Numerical experiments

Test setup

- cG(p)-cGP(k) and cG(p)-dG(k) methods, $p = k$, $k \in \{2, 3, 4, 5\}$
- heat equation with thermal diffusivity $\rho = 1$
- prescribed solution with $f = 2$

$$u(\boldsymbol{x}, t) = \sin(2\pi f t)\sin(2\pi f x)\sin(2\pi f y)\sin(2\pi f z)$$

Study the errors $e_u = u(\boldsymbol{x}, t) - u_{\tau,h}(\boldsymbol{x}, t)$ in the norms given by

$$\|e_u\|_{L^\infty(L^\infty)} = \max_{t\in I}\left(\sup_{\Omega}\|e_u\|_\infty\right), \quad \|e_u\|_{L^2(L^2)} = \left(\int_I\int_\Omega |e_u|^2 \,\mathrm{d}\boldsymbol{x}\,\mathrm{d}t\right)^{\frac{1}{2}}.$$

Numerical experiments (cont.)

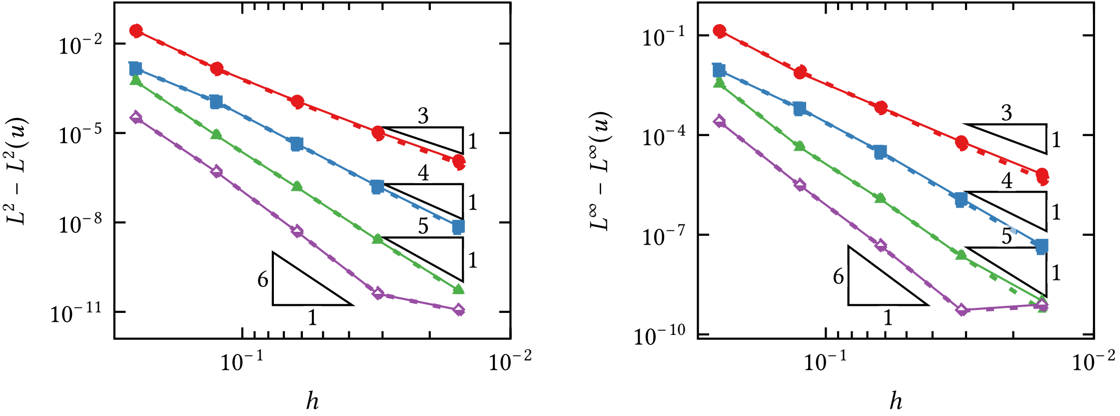


Figure: Computed errors for the displacement u for different polynomial orders $p = k$ for $CG(p) - DG(k)$ discretizations of the heat equation. The expected orders of convergence $k+1$, represented by the triangles, match with the experimental orders.

Numerical experiments (cont.)

$cG(p) - dG(k)$ single time step

$k \setminus r$	2	3	4	5	6
2	9.0	9.75	9.00	8.875	8.656
3	12.0	11.75	10.88	10.188	10.563
4	14.5	14.00	12.88	11.813	11.781

$cG(p) - cGP(k)$ single time step

$k \setminus r$	2	3	4	5	6
2	9.0	9.75	9.25	8.875	8.688
3	12.0	12.00	10.88	10.188	10.594
4	14.5	14.00	12.88	11.875	11.781

Numerical experiments (cont.)

$cG(p) - dG(k)$ single time step

$k \setminus r$	2	3	4	5	6
2	9.0	9.75	9.00	8.875	8.656
3	12.0	11.75	10.88	10.188	10.563
4	14.5	14.00	12.88	11.813	11.781

$cG(p) - cGP(k)$ single time step

$k \setminus r$	2	3	4	5	6
2	9.0	9.75	9.25	8.875	8.688
3	12.0	12.00	10.88	10.188	10.594
4	14.5	14.00	12.88	11.875	11.781

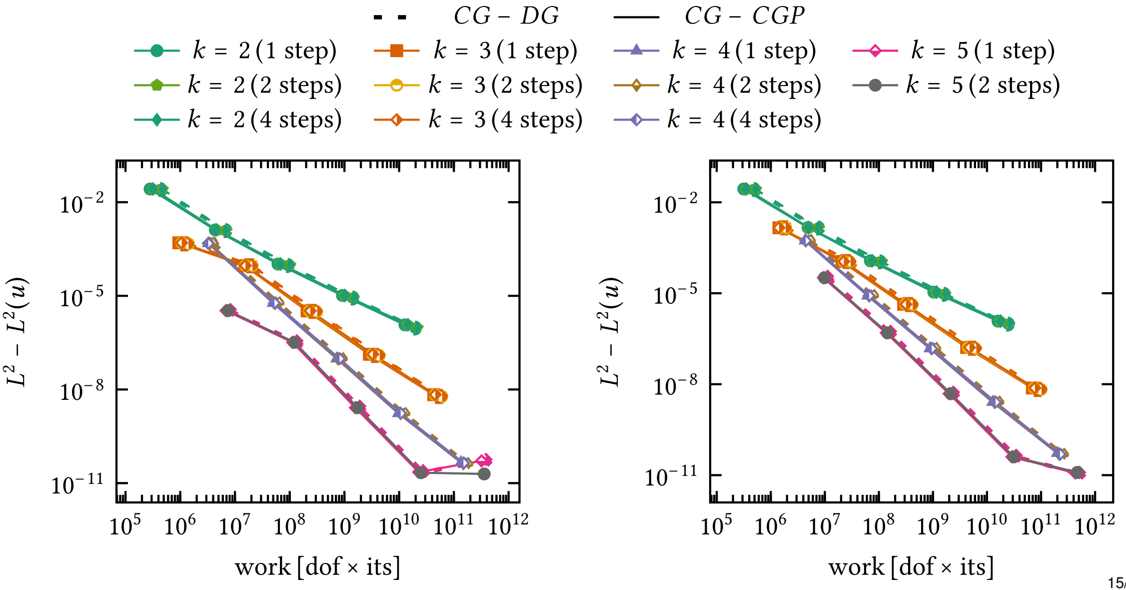
$cG(p) - dG(k)$ 2 time steps at once

$k \setminus r$	2	3	4	5	6
2	10.0	10.0	10.0	9.60	9.234
3	12.0	12.38	11.75	10.88	11.484
4	15.0	15.0	13.75	12.88	12.75

$cG(p) - dGP(k)$ 2 time steps at once

$k \setminus r$	2	3	4	5	6
2	10.0	10.0	10.0	9.75	9.484
3	12.8	13.00	11.75	10.875	11.484
4	15.0	15.00	13.75	12.875	12.75

Numerical experiments (cont.)



Outlook

N. Margenberg and PM, “A space-time multigrid method for space-time finite element discretizations of parabolic and hyperbolic PDEs”, submitted, 2024.

- deformed meshes and heterogeneous coefficients
- wave equation:

$$\begin{aligned} \boldsymbol{v}_n &= \boldsymbol{M}_\tau^{-1}\boldsymbol{A}_\tau \boldsymbol{u}_n - \boldsymbol{M}_\tau^{-1}\alpha \boldsymbol{u}_{n-1}^{N_t}, \\ \underbrace{(\boldsymbol{M}_\tau \otimes \boldsymbol{A}_h + \boldsymbol{A}_\tau \boldsymbol{M}_\tau^{-1}\boldsymbol{A}_\tau \otimes \boldsymbol{M}_h)}_{:=\boldsymbol{S}} \boldsymbol{u}_n &= \boldsymbol{M}_\tau \otimes \boldsymbol{M}_h \boldsymbol{f} + \alpha \otimes \boldsymbol{M}_h \boldsymbol{v}_{n-1}^{N_t} + \underbrace{\boldsymbol{A}_\tau \boldsymbol{M}_\tau^{-1}\alpha \otimes \boldsymbol{M}_h}_{:=\boldsymbol{D}} \boldsymbol{u}_{n-1}^{N_t}. \end{aligned}$$

$$\begin{aligned} \boldsymbol{v}_n &= \boldsymbol{M}_\tau^{-1}\boldsymbol{A}_\tau \boldsymbol{u}_n - \boldsymbol{M}_\tau^{-1}\alpha \boldsymbol{u}_{n-1}^{N_t} + \boldsymbol{M}_\tau^{-1}\beta \boldsymbol{v}_{n-1}^{N_t} \\ \underbrace{(\boldsymbol{M}_\tau \otimes \boldsymbol{A}_h + \boldsymbol{A}_\tau \boldsymbol{M}_\tau^{-1}\boldsymbol{A}_\tau \otimes \boldsymbol{M}_h)}_{:=\boldsymbol{S}} \boldsymbol{u}_n &= \boldsymbol{M}_\tau \otimes \boldsymbol{M}_h \boldsymbol{f} - \beta \otimes \boldsymbol{M}_h \boldsymbol{f}_{n-1}^{N_t} \\ &\quad + (\beta \otimes \boldsymbol{A}_h + \boldsymbol{A}_\tau \boldsymbol{M}_\tau^{-1}\alpha \otimes \boldsymbol{M}_h)\boldsymbol{u}_{n-1}^{N_t} + (\alpha - \boldsymbol{A}_\tau \boldsymbol{M}_\tau^{-1}\beta) \otimes \boldsymbol{M}_h \boldsymbol{v}_{n-1}^{N_t} \end{aligned}$$

- scaling studies with 20,556 MPI ranks

Part 4:

Application: Stokes equations

Solution procedure

dG(k) space-time formulation: Find $(\boldsymbol{V}_n, \boldsymbol{P}_n) \in \mathbf{R}^{(k+1)(M^{\boldsymbol{v}}+M^p)}$ such that

$$\begin{pmatrix} \boldsymbol{K}_n^\tau \otimes \boldsymbol{M}_h + \boldsymbol{M}_n^\tau \otimes \boldsymbol{A}_h & \boldsymbol{M}_n^\tau \otimes \boldsymbol{B}_h^\top \\ \boldsymbol{M}_n^\tau \otimes \boldsymbol{B}_h & \boldsymbol{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{V}_n \\ \boldsymbol{P}_n \end{pmatrix} = \begin{pmatrix} \boldsymbol{F}_n \\ \boldsymbol{0} \end{pmatrix} + \boldsymbol{C}_n^\tau \otimes \begin{pmatrix} \boldsymbol{M}_h \\ \boldsymbol{0} \end{pmatrix} \boldsymbol{V}_{n-1} \,.$$

The global discrete solution spaces are defined by the tensor products

$$\boldsymbol{H}_{\tau,h}^{\boldsymbol{v}} = Y_\tau^k(I) \otimes \boldsymbol{V}_h^{r+1}(\Omega)\,, \quad H_{\tau,h}^p = Y_\tau^k(I) \otimes Q_h^r(\Omega)\,,$$

with

$$\begin{aligned} \boldsymbol{V}_h^{r+1}(\Omega) &:= \{\boldsymbol{v}_h \in \boldsymbol{V} \,:\, \boldsymbol{v}_{h|K} \in \mathbb{Q}_{k+1}^d(K) \text{ for all } K \in T_h\} \cap \boldsymbol{H}_0^1(\Omega)\,, \\ Q_h^k(\Omega) &:= \{q_h \in Q \,:\, q_{h|K} \in \mathbb{P}_r^{\text{disc}}(K) \text{ for all } K \in T_h\}\,. \end{aligned}$$

Preconditioner: space-time multigrid with additive Vanka smoother (element-centric patches consisting of v and p)

Numerical experiments

Model problem on the space-time domain $\Omega \times I = [0,1]^2 \times [0,1]$ with prescribed solution given for velocity $\boldsymbol{v}\colon \Omega \times I \to \mathbb{R}^2$ and pressure $p\colon \Omega \times I \to \mathbb{R}$ by

$$\boldsymbol{v}(\mathbf{x}, t) = \sin(t) \begin{pmatrix} \sin^2(\pi x)\sin(\pi y)\cos(\pi y) \\ \sin(\pi x)\cos(\pi x)\sin^2(\pi y) \end{pmatrix},$$

$$p(\mathbf{x}, t) = \sin(t)\sin(\pi x)\cos(\pi x)\sin(\pi y)\cos(\pi y).$$

We set the kinematic viscosity to $\nu = 0.1$ and choose the external force $\mathbf{f}$ appropriately.

Numerical experiments (cont.)

Table: Number of GMRES iterations until convergence for different polynomial degrees r and numbers of refinements r with $\mathbb{Q}^2_{k+1}/\mathbb{P}^{\text{disc}}_k$ discretization of the Stokes system.

h-multigrid in space

$r \backslash k$	1	2	3	4	5	6
2	14.0	15.0	15.0	14.0	13.0	10.6
3	19.0	17.9	18.9	18.3	16.4	14.0
4	24.0	26.8	24.7	24.6	21.4	18.4
5	26.0	26.4	28.8	27.7	24.7	21.9
6	35.0	33.9	34.6	30.9	29.6	26.9
7	40.0	38.8	39.6	36.7	34.5	31.9

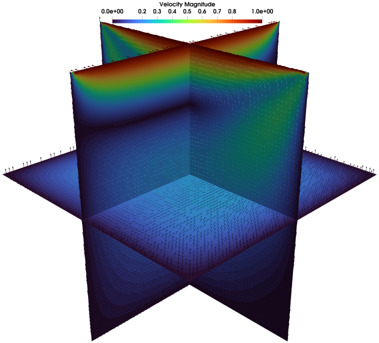
hp STMG

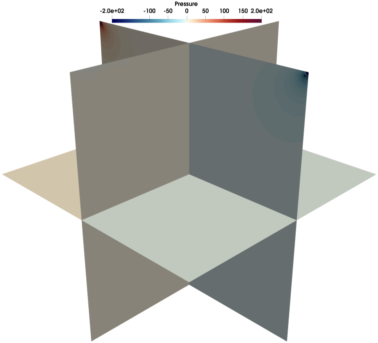
$r \backslash k$	1	2	3	4	5	6
2	14.0	15.0	15.0	14.0	13.0	10.6
3	19.8	15.9	16.0	15.0	13.7	11.0
4	27.8	23.0	22.9	21.9	19.0	15.5
5	31.0	26.4	26.6	22.8	18.7	14.9
6	45.0	36.1	36.7	29.0	23.1	17.2
7	50.8	43.8	42.8	32.8	25.6	19.6

Outlook

N. Margenberg, M. Bause, and PM, "An hp multigrid approach for tensor-product space-time finite element discretizations of the Stokes equations", submitted, 2025.

- lid-driven cavity





- scaling studies with 13,824 MPI ranks

Part 5:

Block preconditioners

Motivation

- space-time multigrid is a monolithic and robust approach, however, needs expensive smoothers (here: element-centric additive patch smoothers)
- efficient implementation of patch smoothers: still open research; examples:
  - Pazner and Persson '17 $\rightarrow$ SVD-based tensor-product preconditioner
  - Brubeck and Farrell '21 $\rightarrow$ vertex-star relaxation
- alternative: block preconditioning $\rightarrow$ use cheaper smoothers on blocks; examples:
  - for space-time FEM: Danieli et al. '22
  - for IRK: Southworth et al. '22, $\underbrace{\text{Axelsson et al. '20, '24, Dravis et al.'24, PM et al.'24}}_{\text{stage-parallel IRK}}$

Stage-parallel implicit Runge–Kutta preconditioning (cont.)

PM, I. Dravins, M. Kronbichler, and M. Neytcheva, “Stage-parallel fully implicit Runge-Kutta implementations with optimal multilevel preconditioners at the scaling limit”, in SISC, 2022.

For a linear system of equations, IRK has the form: ... Butcher tableau: $\begin{array}{c|c} \boldsymbol{c}_Q & A_Q \\ \hline & \boldsymbol{b}_Q^\top \end{array}$

$$\mathbf{u}_{m+1} = \mathbf{u}_m + \tau \sum_{q=1}^{Q} b_q \mathbf{k}_q \quad \text{w.} \quad \underbrace{(A_Q^{-1} \otimes M + \tau \mathbb{I}_Q \otimes K)}_{A} \mathbf{k} = (A_Q^{-1} \otimes \mathbb{I}_n)\overline{\mathbf{g}} - (A_Q^{-1} \otimes K)(\mathbf{e}_Q \otimes \mathbf{u}_0)$$

Following Butcher [1976], A can be factorized, using $A_Q^{-1} = S\Lambda S^{-1}$, and explicitly inverted:

$$A = (S \otimes \mathbb{I}_n)(\Lambda \otimes M + \tau \mathbb{I}_Q \otimes K)(S^{-1} \otimes \mathbb{I}_n) \quad A^{-1} = (S \otimes \mathbb{I}_n)(\Lambda \otimes M + \tau \mathbb{I}_Q \otimes K)^{-1}(S^{-1} \otimes \mathbb{I}_n).$$

Axelsson, Neytcheva [2020] proposed real-value preconditioner ($LU = A_Q^{-1} \rightarrow L = \tilde{S}\tilde{\Lambda}\tilde{S}^{-1}$):

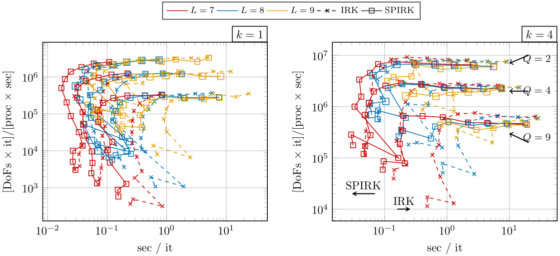
$$P^{-1} = (\tilde{S} \otimes \mathbb{I}_n)(\tilde{\Lambda} \otimes M + \tau \mathbb{I}_Q \otimes K)^{-1}(\tilde{S}^{-1} \otimes \mathbb{I}_n).$$

... Q stages can be solved in parallel! Helmholtz operator $\rightarrow$ multigrid

Stage-parallel implicit Runge–Kutta preconditioning (cont.)

Main results:

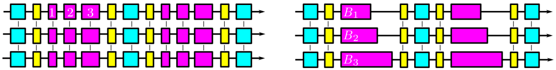
- for the first time shown: stage parallelism shifts the scaling limit



... clear speedup for $\leq$ 10k DoFs per process!

- performance model: minimize lin. iterations performed in serial, speedup limited by Q

$$\sum_{1\leq q\leq Q} N_Q^{\mathrm{IT}} \quad \text{vs.} \quad \max_{1\leq q\leq Q} N_Q^{\mathrm{IT}}$$



(b) parallel IRK with 3 processes (c) stage-parallel IRK with 3 processes

- application also to advection/diffusion; extension to nonlinear equations?

Implicit Runge–Kutta methods vs. space-time FEM

- implicit Runge–Kutta method: $\mathbf{u}_{m+1} = \mathbf{u}_m + \tau \sum_{q=1}^{Q} b_q \mathbf{k}_q$ with

$$(A_Q^{-1} \otimes M + \tau \mathbb{I}_Q \otimes K)\mathbf{k} = (A_Q^{-1} \otimes I_n)(\overline{\mathbf{g}} - \mathbf{e}_Q \otimes (K\mathbf{u}_0))$$

- space-time FEM with dG(k): $\mathbf{u}_{m+1} = \mathbf{k}_Q$ with

$$\boxed{(\tilde{A}_Q^{-1} \otimes M + \tau \mathbb{I}_Q \otimes K)\mathbf{k} = \tau \overline{\mathbf{g}} + \tilde{\alpha} \otimes (M\mathbf{u}_0)}$$

and $\tilde{A}_Q^{-1} = M_\tau^{-1} A_\tau, \tilde{\alpha} = M_\tau^{-1} \alpha$

- space-time FEM with cGP(k): $\mathbf{u}_{m+1} = \mathbf{k}_Q$ with

$$\boxed{(\tilde{A}_Q^{-1} \otimes M + \tau \mathbb{I}_Q \otimes K)\mathbf{k} = \tau \overline{\mathbf{g}} - \tau \tilde{\beta} \otimes \overline{\mathbf{g}}_0 + (\tau \tilde{\beta} \otimes K + \tilde{\alpha} \otimes M)\mathbf{u}_0}$$

and $\tilde{A}_Q^{-1} = M_\tau^{-1} A_\tau$, $\tilde{\alpha} = M_\tau^{-1} \alpha$, and $\tilde{\beta} = M_\tau^{-1} \beta$

Observations:

- system matrix: same structure
- different coefficients
- different rhs

Numerical results

2D setup (similar as above):

$$u(\boldsymbol{x}, t) = \sin(2\pi f t)\sin(2\pi f x)\sin(2\pi f y)$$

We set $f = 1$, $\tau = 0.1$, and run 10 time steps. Preconditioner on block: 1 V-cycle of geometric multigrid with Chebyshev smoother around point Jacobi.

Preliminary results:

r	IRK(Q=2)	dG(k=2)	cGP(k=3)
4	4	4	4
5	4	4	4
6	4	4	4

r	IRK(Q=5)	dG(k=5)	cGP(k=6)
3	7.9	7.9	7.9
4	8	8	8
5	8	8	8

Same number of iterations!

For complex variant, see: Werder et al. ['01], Banks et al. ['14]

Part 6:

Conclusions & Outlook

Conclusions

- space-time multigrid for space-time finite-element computations
  - matrix-free implementation
  - simple implementation with deal.II
  - smoother: additive Schwarz (element-centric patches)
  - robust but expensive
  - application: heat equation and Stokes equation
- alternative: block preconditioners $\rightarrow$ preliminary results

Outlook

- efficient patch smoothers
- block preconditioning
- application: Navier–Stokes equations [Anselmann and Bause '23]

References

- Schmich, M. and Vexler, B., 2008. Adaptivity with dynamic meshes for space-time finite element discretizations of parabolic equations. SIAM Journal on Scientific Computing, 30(1), pp.369-393.
- Besier, M. and Rannacher, R., 2012. Goal-oriented space–time adaptivity in the finite element Galerkin method for the computation of nonstationary incompressible flow. International Journal for Numerical Methods in Fluids, 70(9), pp.1139-1166.
- Roth, J., Thiele, J.P., Köcher, U. and Wick, T., 2024. Tensor-product space-time goal-oriented error control and adaptivity with partition-of-unity dual-weighted residuals for nonstationary flow problems. Computational Methods in Applied Mathematics, 24(1), pp.185-214.
- Bause, M., Bruchhäuser, M.P. and Köcher, U., 2021. Flexible goal-oriented adaptivity for higher-order space–time discretizations of transport problems with coupled flow. Computers & Mathematics with Applications, 91, pp.17-35.
- Matthies, G. and Schieweck, F., 2011. Higher order variational time discretizations for nonlinear systems of ordinary differential equations.
- Gander, M.J., 2015. 50 years of time parallel time integration. In Multiple Shooting and Time Domain Decomposition Methods: MuS-TDD, Heidelberg, May 6-8, 2013 (pp. 69-113). Cham: Springer International Publishing.
- Ong, B.W. and Schroder, J.B., 2020. Applications of time parallelization. Computing and Visualization in Science, 23, pp.1-15.

References (cont.)

- Falgout, R.D., Friedhoff, S., Kolev, T.V., MacLachlan, S.P. and Schroder, J.B., 2014. Parallel time integration with multigrid. SIAM Journal on Scientific Computing, 36(6), pp.C635-C661.
- Falgout, R.D., Friedhoff, S., Kolev, T.V., MacLachlan, S.P., Schroder, J.B. and Vandewalle, S., 2017. Multigrid methods with space–time concurrency. Computing and Visualization in Science, 18, pp.123-143.
- Hackbusch, W., 2013. Multi-grid methods and applications (Vol. 4). Springer Science & Business Media.
- Gander, M.J. and Neumuller, M., 2016. Analysis of a new space-time parallel multigrid algorithm for parabolic problems. SIAM Journal on Scientific Computing, 38(4), pp.A2173-A2208.
- Clevenger, T.C., Heister, T., Kanschat, G. and Kronbichler, M., 2020. A flexible, parallel, adaptive geometric multigrid method for FEM. ACM Transactions on Mathematical Software (TOMS), 47(1), pp.1-27.
- Munch, P., Heister, T., Prieto Saavedra, L. and Kronbichler, M., 2023. Efficient distributed matrix-free multigrid methods on locally refined meshes for FEM computations. ACM transactions on parallel computing, 10(1), pp.1-38.
- Feder, M., Heltai, L., Kronbichler, M. and Munch, P., 2024. Matrix-free implementation of the non-nested multigrid method. arXiv preprint arXiv:2412.10910.
- Pazner, W. and Persson, P.O., 2017. Stage-parallel fully implicit Runge–Kutta solvers for discontinuous Galerkin fluid simulations. Journal of Computational Physics, 335, pp.700-717. Brubeck, P.D. and Farrell, P.E., 2021. A scalable and robust vertex-star relaxation for high-order FEM. arXiv preprint arXiv:2107.14758.

References (cont.)

- Danieli, F., Southworth, B.S. and Schroder, J.B., 2023. Space-Time Block Preconditioning for Incompressible Resistive Magnetohydrodynamics. arXiv preprint arXiv:2309.00768.
- Southworth, B.S., Krzysik, O., Pazner, W. and Sterck, H.D., 2022. Fast Solution of Fully Implicit Runge–Kutta and Discontinuous Galerkin in Time for Numerical PDEs, Part I: the Linear Setting. SIAM Journal on Scientific Computing, 44(1), pp.A416-A443.
- Axelsson, O. and Neytcheva, M., 2020. Numerical solution methods for implicit Runge-Kutta methods of arbitrarily high order. In Proceedings of the conference'Algoritmy (pp. 11-20). Axelsson, O., Dravins, I. and Neytcheva, M., 2024. Stage-parallel preconditioners for implicit Runge–Kutta methods of arbitrarily high order, linear problems. Numerical Linear Algebra with Applications, 31(1), p.e2532.
- Dravins, I., Serra-Capizzano, S. and Neytcheva, M., 2024. Spectral Analysis of Preconditioned Matrices Arising from Stage-Parallel Implicit Runge–Kutta Methods of Arbitrarily High Order. SIAM Journal on Matrix Analysis and Applications, 45(2), pp.1007-1034.
- Munch, P., Dravins, I., Kronbichler, M. and Neytcheva, M., 2024. Stage-parallel fully implicit Runge–Kutta implementations with optimal multilevel preconditioners at the scaling limit. SIAM Journal on Scientific Computing, 46(2), pp.S71-S96.
- Butcher, J.C., 1976. On the implementation of implicit Runge-Kutta methods. BIT Numerical Mathematics, 16(3), pp.237-240.

References (cont.)

- Anselmann, M. and Bause, M., 2023. A geometric multigrid method for space-time finite element discretizations of the Navier–Stokes equations and its application to 3d flow simulation. ACM Transactions on Mathematical Software, 49(1), pp.1-25.