An adaptive DG finite element method in space and time

M. Neumüller[1] O. Steinbach[1]

[1]Institute of Computational Mathematics
Graz University of Technology

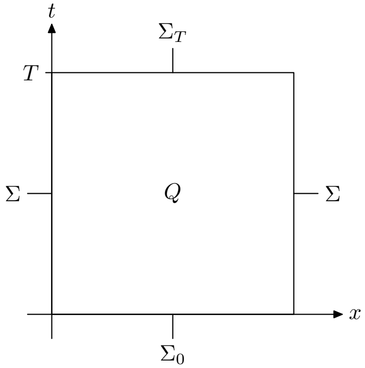
Efficient Solvers in Biomedical Applications
July 4, 2012

Outline

Motivation

Bounded domain $\Omega(t) \subset \mathbb{R}^d$ for $d = 1, 2, 3$ with $t \in [0, T]$.

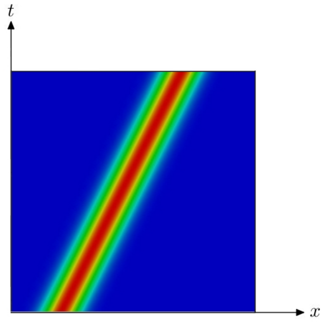
$$\begin{aligned} \partial_t u(x,t) - \Delta u(x,t) &= f(x,t) && \text{for } (x,t) \in Q := \Omega \times (0,T), \\ u(x,t) &= 0 && \text{for } (x,t) \in \Sigma := \partial\Omega \times (0,T), \\ u(x,0) &= u_0(x) && \text{for } (x,t) \in \Sigma_0 := \Omega(0) \times \{0\}. \end{aligned}$$



Motivation

Bounded domain $\Omega(t) \subset \mathbb{R}^d$ for $d = 1, 2, 3$ with $t \in [0, T]$.

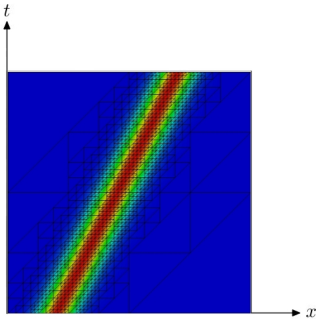
$$\begin{aligned}
\partial_t u(x,t) - \Delta u(x,t) &= f(x,t) && \text{for } (x,t) \in Q := \Omega \times (0,T), \\
u(x,t) &= 0 && \text{for } (x,t) \in \Sigma := \partial\Omega \times (0,T), \\
u(x,0) &= u_0(x) && \text{for } (x,t) \in \Sigma_0 := \Omega(0) \times \{0\}.
\end{aligned}$$

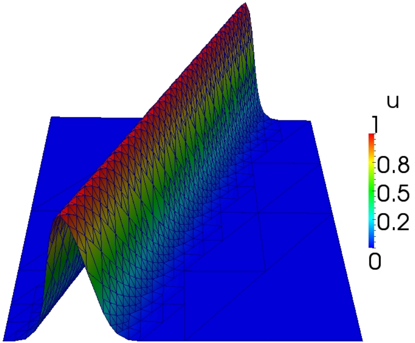


Motivation

Bounded domain $\Omega(t) \subset \mathbb{R}^d$ for $d = 1, 2, 3$ with $t \in [0, T]$.

$$\begin{aligned}
\partial_t u(x,t) - \Delta u(x,t) &= f(x,t) && \text{for } (x,t) \in Q := \Omega \times (0,T),\\
u(x,t) &= 0 && \text{for } (x,t) \in \Sigma := \partial\Omega \times (0,T),\\
u(x,0) &= u_0(x) && \text{for } (x,t) \in \Sigma_0 := \Omega(0) \times \{0\}.
\end{aligned}$$





Discretization

$$v \in S_{h,0}^p(\mathcal{T}_N) := \{v \in L_2(Q) : v_{|\tau_k} \in \mathbb{P}^p(\tau_k) \text{ for all } \tau_k \in \mathcal{T}_N \text{ and } v_{|\Sigma} = 0\}.$$

Discretization

$$v \in S^p_{h,0}(\mathcal{T}_N) := \{v \in L_2(Q) : v_{|\tau_k} \in \mathbb{P}^p(\tau_k) \text{ for all } \tau_k \in \mathcal{T}_N \text{ and } v_{|\Sigma} = 0\}.$$

$$\begin{aligned}
&-\sum_{k=1}^{N}\int_{\tau_k} u\frac{\partial v}{\partial t}\mathrm{d}q + \int_{\Sigma_T} uv\mathrm{d}s_q + \sum_{e\in\mathcal{I}_N}\int_e n_t\{u\}_e^{\mathrm{up}}\,[v]_e\,\mathrm{d}s_q \\
&+\sum_{k=1}^{N}\int_{\tau_k}\nabla u\cdot\nabla v\mathrm{d}q - \sum_{e\in\mathcal{I}_N}\int_e\Big[\langle \underline{n}_x\cdot\nabla u\rangle_e\,[v]_e + [u]_e\,\langle \underline{n}_x\cdot\nabla v\rangle_e\Big]\mathrm{d}s_q \\
&+\sum_{e\in\mathcal{I}_N}\frac{\sigma p_e^2}{\overline{h}_e}\int_e |\underline{n}_x|^2\,[u]_e\,[v]_e\,\mathrm{d}s_q \\
&=\int_Q fv\mathrm{d}q + \int_{\Sigma_0} u_0 v\mathrm{d}s_q
\end{aligned}$$

Discretization

$$v \in S^p_{h,0}(\mathcal{T}_N) := \{v \in L_2(Q) : v_{|\tau_k} \in \mathbb{P}^p(\tau_k) \text{ for all } \tau_k \in \mathcal{T}_N \text{ and } v_{|\Sigma} = 0\}.$$

$$\begin{aligned}
&-\sum_{k=1}^{N} \int_{\tau_k} u \frac{\partial v}{\partial t} \mathrm{d}q + \int_{\Sigma_T} uv \mathrm{d}s_q + \sum_{e \in \mathcal{I}_N} \int_e n_t \{u\}_e^{\mathrm{up}} [v]_e \, \mathrm{d}s_q \\
&+ \sum_{k=1}^{N} \int_{\tau_k} \nabla u \cdot \nabla v \mathrm{d}q - \sum_{e \in \mathcal{I}_N} \int_e \left[\langle \underline{n}_x \cdot \nabla u \rangle_e [v]_e + [u]_e \langle \underline{n}_x \cdot \nabla v \rangle_e \right] \mathrm{d}s_q \\
&+ \sum_{e \in \mathcal{I}_N} \frac{\sigma p_e^2}{\overline{h}_e} \int_e |\underline{n}_x|^2 [u]_e [v]_e \, \mathrm{d}s_q \\
&= \int_Q f v \mathrm{d}q + \int_{\Sigma_0} u_0 v \mathrm{d}s_q
\end{aligned}$$

Find $u \in S^p_{h,0}(\mathcal{T}_N)$ $\qquad a_{\mathrm{DG}}(u, v) = F(v) \qquad$ for all $v \in S^p_{h,0}(\mathcal{T}_N)$.

Discretization

Variational formulation:

$$\text{Find } u \in S^p_{h,0}(\mathcal{T}_N) \qquad a_{\mathrm{DG}}(u,v) = F(v) \qquad \text{for all } v \in S^p_{h,0}(\mathcal{T}_N).$$

Discretization

Variational formulation:

$$\text{Find } u \in S^p_{h,0}(\mathcal{T}_N) \qquad a_{\mathrm{DG}}(u,v) = F(v) \qquad \text{for all } v \in S^p_{h,0}(\mathcal{T}_N).$$

Stability:

$$\sup_{0 \neq v \in S^p_{h,0}(\mathcal{T}_N)} \frac{a_{\mathrm{DG}}(u,v)}{\|v\|_{\mathrm{DG}}} \geq c_S \|u\|_{\mathrm{DG}} \quad \text{for all } v \in S^p_{h,0}(\mathcal{T}_N).$$

Energy norm:

$$\|u\|^2_{\mathrm{DG}} := \sum_{k=1}^{N} \|\nabla_x u\|^2_{L_2(\tau_k)} + \sum_{e \in \mathcal{I}_N} \frac{\sigma p_e^2}{\overline{h}_e} |\underline{n}_x|^2 \, \|[u]_e\|^2_{L_2(e)}$$

$$+ \sum_{k=1}^{N} h_k \|\partial_t u\|^2_{L_2(\tau_k)} + \frac{1}{2}\|u\|^2_{L_2(\Sigma_0 \cup \Sigma_T)} + \frac{1}{2} \sum_{e \in \mathcal{I}_N} |n_t| \, \|[u]_e\|^2_{L_2(e)}$$

Discretization

Variational formulation:

$$\text{Find } u \in S^p_{h,0}(\mathcal{T}_N) \qquad a_{\mathrm{DG}}(u,v) = F(v) \qquad \text{for all } v \in S^p_{h,0}(\mathcal{T}_N).$$

Stability:

$$\sup_{0 \neq v \in S^p_{h,0}(\mathcal{T}_N)} \frac{a_{\mathrm{DG}}(u,v)}{\|v\|_{\mathrm{DG}}} \geq c_S \|u\|_{\mathrm{DG}} \quad \text{for all } v \in S^p_{h,0}(\mathcal{T}_N).$$

Energy norm:

$$\|u\|^2_{\mathrm{DG}} := \sum_{k=1}^{N} \|\nabla_x u\|^2_{L_2(\tau_k)} + \sum_{e \in \mathcal{I}_N} \frac{\sigma p_e^2}{\overline{h}_e} |\underline{n}_x|^2 \, \|[u]_e\|^2_{L_2(e)}$$
$$+ \sum_{k=1}^{N} h_k \|\partial_t u\|^2_{L_2(\tau_k)} + \frac{1}{2}\|u\|^2_{L_2(\Sigma_0 \cup \Sigma_T)} + \frac{1}{2} \sum_{e \in \mathcal{I}_N} |n_t| \, \|[u]_e\|^2_{L_2(e)}$$

A priori error estimate:

$$\|u - u_h\|_{\mathrm{DG}} \leq c h^{\min\{s,p\}} |u|_{H^s(Q)} \qquad \text{for } s > 3/2.$$

Numerical example

Consider the domain $\Omega = (0,1)^3$ and the time $T = 1$. Given exact solution

$$u(x,t) = \sin(\pi x)\sin(\pi y)\sin(\pi z)(1 - t^2).$$

Numerical example

Consider the domain $\Omega = (0,1)^3$ and the time $T = 1$. Given exact solution

$$u(x,t) = \sin(\pi x)\sin(\pi y)\sin(\pi z)(1-t^2).$$

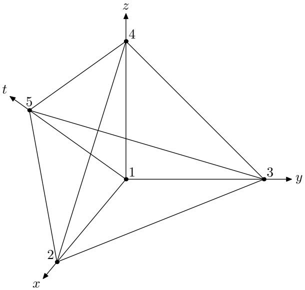


Figure: Pentatope.

level	elements	dof	$\|u-u_h\|_{L_2(Q)}$	eoc
0	96	120	6.3903 − 2	–
1	1536	3546	4.6208 − 2	0.47
2	24576	83155	1.2104 − 2	1.93
3	393216	1620793	2.8615 − 3	2.08
4	6291456	28587985	6.7072 − 4	2.09

Table: Numerical results for $\sigma = 10$, $p = 1$.

Adaptivity

DG residual error estimator for heat equation:

$$\eta_k^2 := \frac{h_k^2}{p_k^2}\|f - \partial_t u + \Delta u\|^2_{L_2(\tau_k)} + \sum_{e\in\partial\tau_k\cap\Sigma} \frac{\sigma p_e^2}{h_e}\|u - g\|^2_{L_2(e)}$$
$$+ \frac{1}{2}\sum_{e\in\partial\tau_k\setminus\Sigma}\left[\frac{\overline{h}_e}{p_e}\|[\nabla_x u]_e\|^2_{L_2(e)} + \frac{\sigma p_e^2}{\overline{h}_e}\left|\underline{n}_x\right|^2\|[u]_e\|^2_{L_2(e)} + \left|\underline{n}_t\right|\frac{\overline{h}_e}{p_e}\|[u]_e\|^2_{L_2(e)}\right]$$

Motivated by convection–diffusion estimators [D. Schötzau, L. Zhu, 2009]

Adaptivity

Numerical example

Consider the domain $\Omega = (0,1)$ and $T = 1$ with the exact solution

$$u(x,t) = \left[x^2 + (t - \frac{1}{2})^2\right]^{1/4} \in H^{3/2-\varepsilon}(Q).$$

Error in the energy norm: $\quad \|u - u_h\|_{DG} \leq c\, h^{1/2-\varepsilon}.$

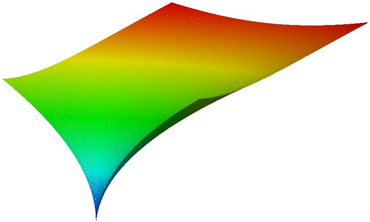


Figure: Solution u.

Adaptivity

Numerical example

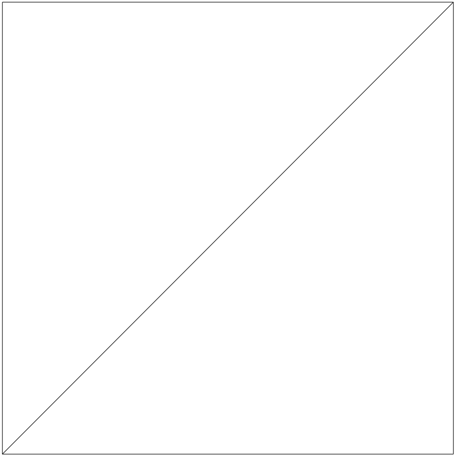


Figure: Results for $p = 1$.

Adaptivity

Numerical example

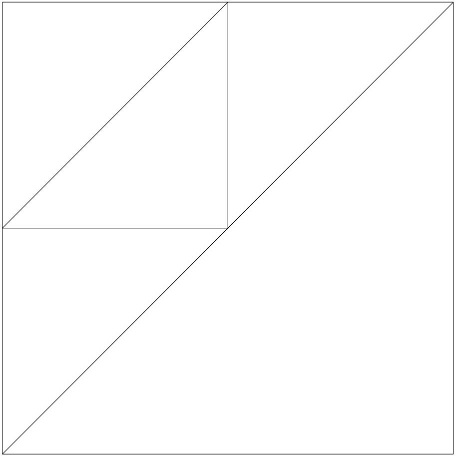


Figure: Results for $p = 1$.

Adaptivity

Numerical example

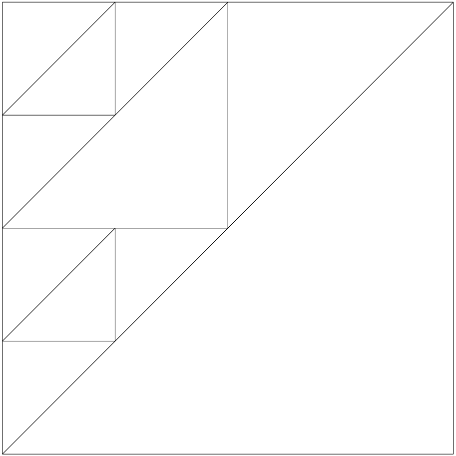


Figure: Results for $p = 1$.

Adaptivity

Numerical example

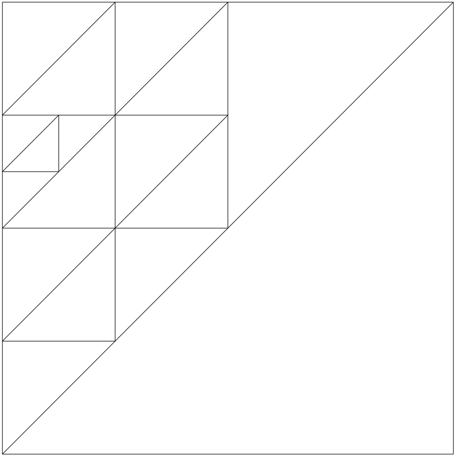


Figure: Results for $p = 1$.

Adaptivity

Numerical example

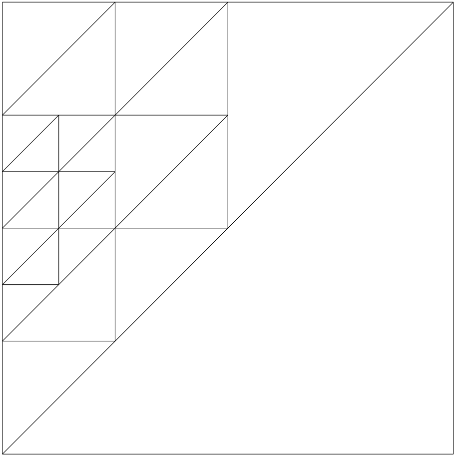


Figure: Results for $p = 1$.

Adaptivity

Numerical example

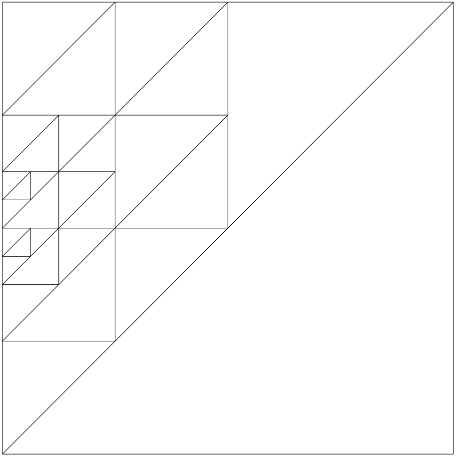


Figure: Results for $p = 1$.

Adaptivity

Numerical example

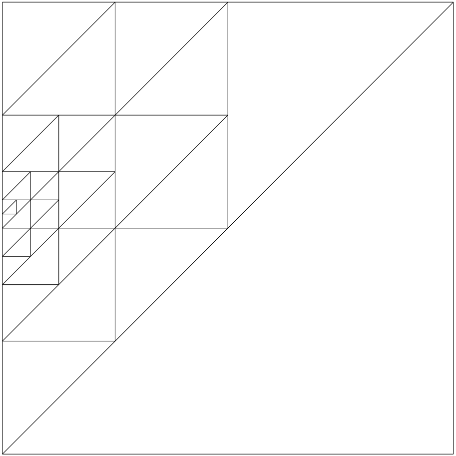


Figure: Results for $p = 1$.

Adaptivity

Numerical example

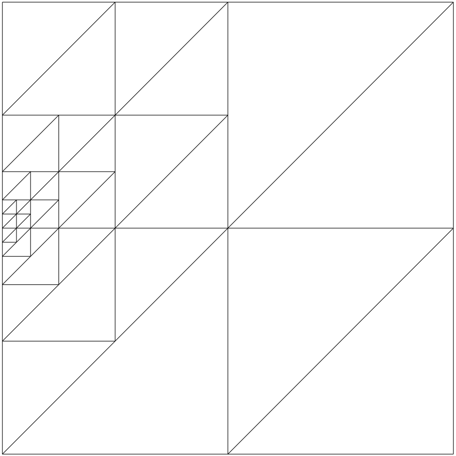


Figure: Results for $p = 1$.

Adaptivity

Numerical example

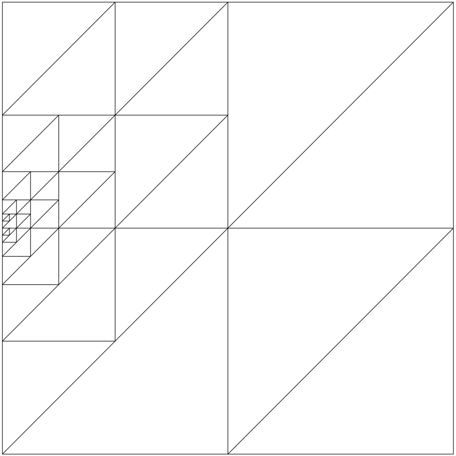


Figure: Results for $p = 1$.

Adaptivity

Numerical example

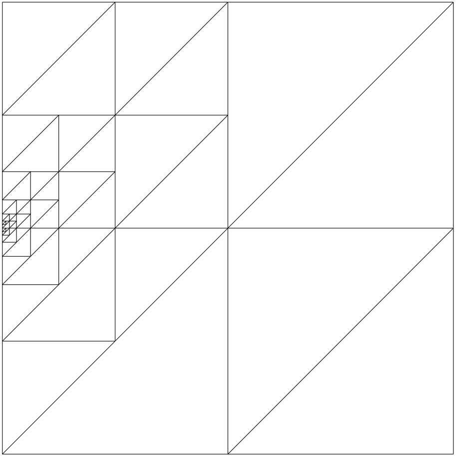


Figure: Results for $p = 1$.

Adaptivity

Numerical example

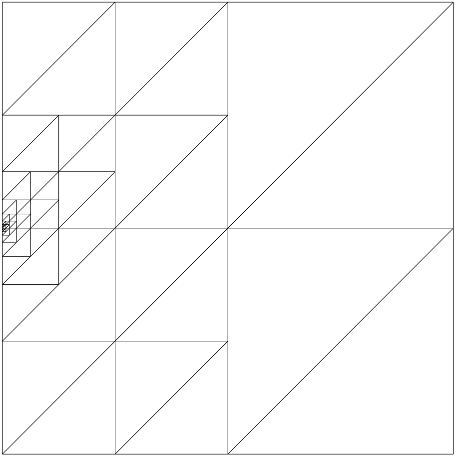


Figure: Results for $p = 1$.

Adaptivity

Numerical example

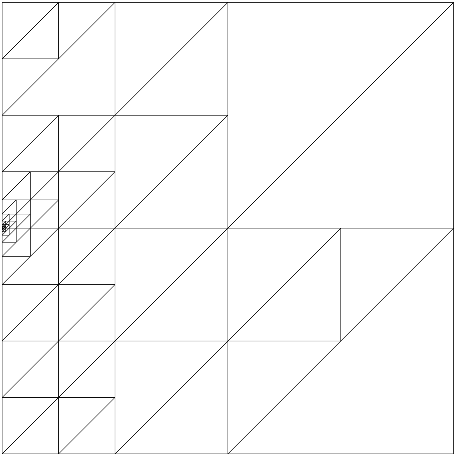


Figure: Results for $p = 1$.

Adaptivity

Numerical example

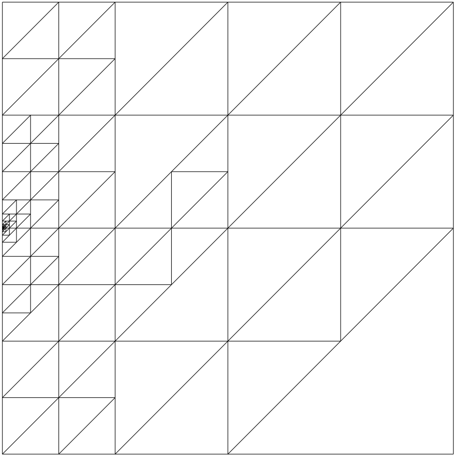


Figure: Results for $p = 1$.

Adaptivity

Numerical example

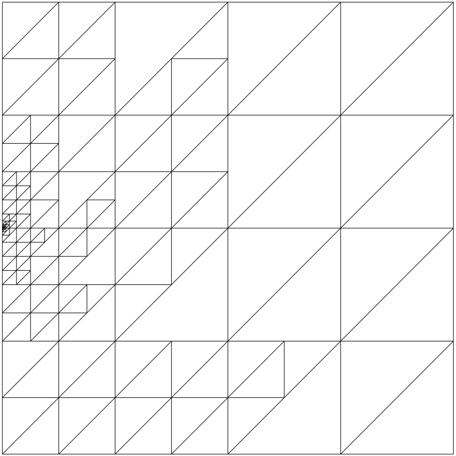


Figure: Results for $p = 1$.

Adaptivity

Numerical example

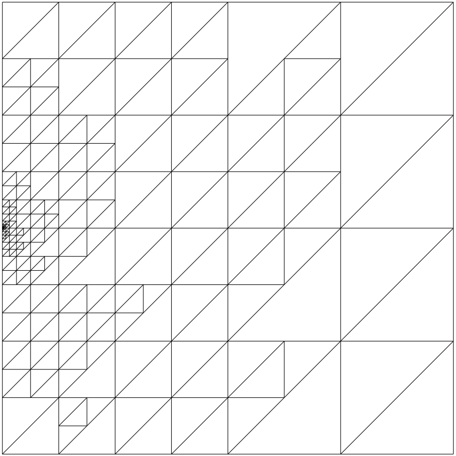


Figure: Results for $p = 1$.

Adaptivity

Numerical example

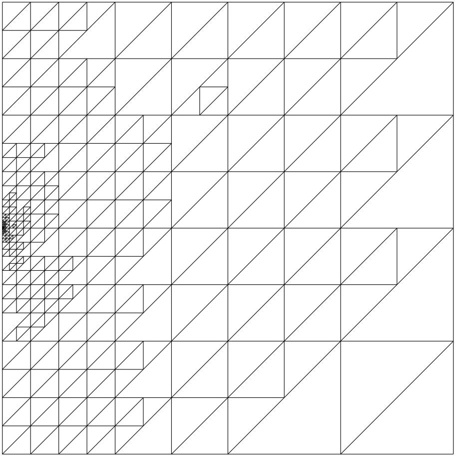


Figure: Results for $p = 1$.

Adaptivity

Numerical example

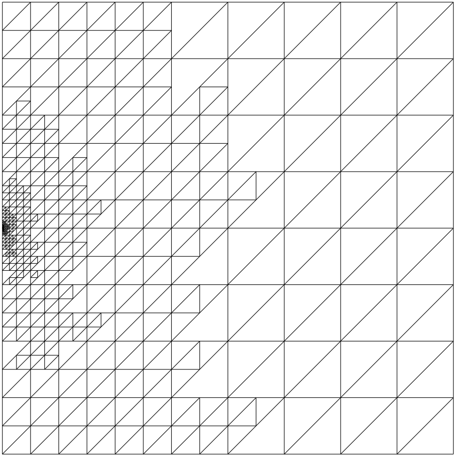


Figure: Results for $p = 1$.

Adaptivity

Numerical example

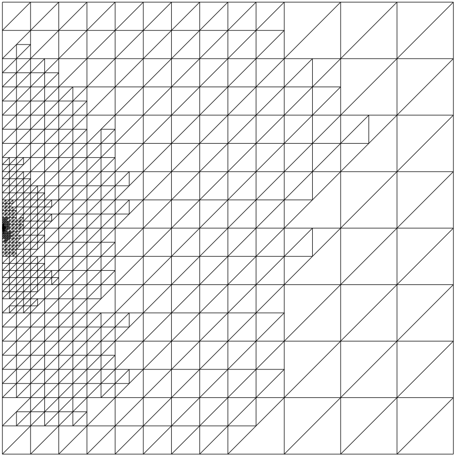


Figure: Results for $p = 1$.

Adaptivity

Numerical example

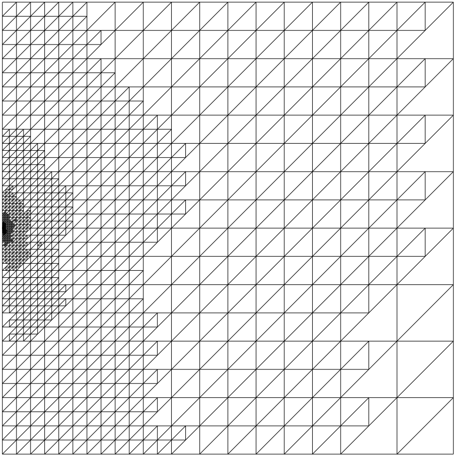


Figure: Results for $p = 1$.

Adaptivity

Numerical example

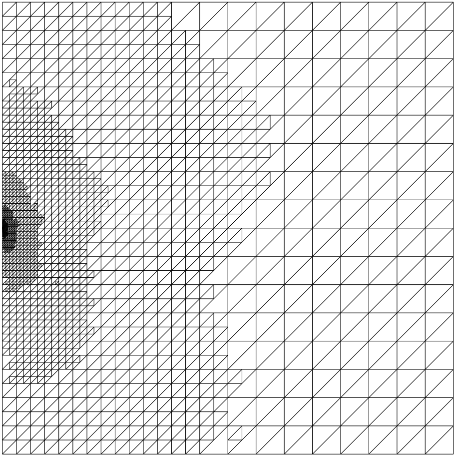


Figure: Results for $p = 1$.

Adaptivity

Numerical example

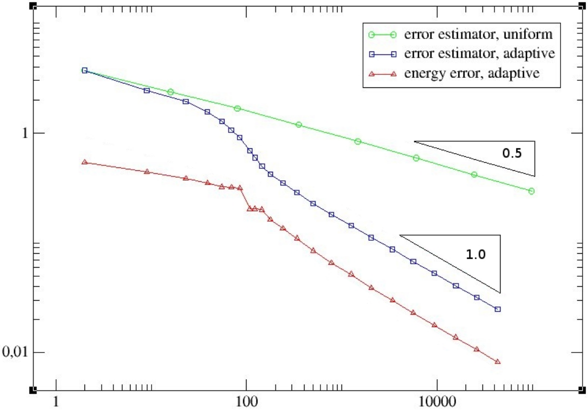


Figure: Results for $p = 1$.

Adaptivity

Numerical example

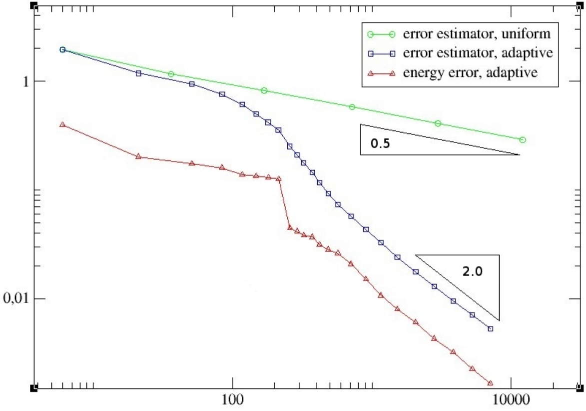


Figure: Results for $p = 2$.

Error estimators

DG residual error estimator:

$$\eta_k^2 := \frac{h_k^2}{p_k^2}\|f - \partial_t u + \Delta u\|^2_{L_2(\tau_k)} + \sum_{e\in\partial\tau_k\cap\Sigma} \frac{\sigma p_e^2}{h_e}\|u - g\|^2_{L_2(e)}$$
$$+ \frac{1}{2}\sum_{e\in\partial\tau_k\setminus\Sigma}\left[\frac{\overline{h}_e}{p_e}\|[\nabla_x u]_e\|^2_{L_2(e)} + \frac{\sigma p_e^2}{\overline{h}_e}\left|\underline{n}_x\right|^2\|[u]_e\|^2_{L_2(e)} + \left|\underline{n}_t\right|\frac{\overline{h}_e}{p_e}\|[u]_e\|^2_{L_2(e)}\right]$$

Error estimators

DG residual error estimator:

$$\eta_k^2 := \frac{h_k^2}{p_k^2}\|f - \partial_t u + \Delta u\|^2_{L_2(\tau_k)} + \sum_{e\in\partial\tau_k\cap\Sigma} \frac{\sigma p_e^2}{h_e}\|u-g\|^2_{L_2(e)}$$

$$+ \frac{1}{2}\sum_{e\in\partial\tau_k\setminus\Sigma}\left[\frac{\overline{h}_e}{p_e}\|[\nabla_x u]_e\|^2_{L_2(e)} + \frac{\sigma p_e^2}{\overline{h}_e}\left|\underline{n}_x\right|^2\|[u]_e\|^2_{L_2(e)} + \left|\underline{n}_t\right|\frac{\overline{h}_e}{p_e}\|[u]_e\|^2_{L_2(e)}\right]$$

DG jump estimator:

$$\tilde{\eta}_k^2 := \frac{1}{2}\sum_{e\in\partial\tau_k\setminus\Sigma}\left[\frac{\sigma p_e^2}{\overline{h}_e}\left|\underline{n}_x\right|^2\|[u]_e\|^2_{L_2(e)} + \left|\underline{n}_t\right|\frac{\overline{h}_e}{p_e}\|[u]_e\|^2_{L_2(e)}\right]$$

$$+ \sum_{e\in\partial\tau_k\cap\Sigma}\frac{\sigma p_e^2}{h_e}\|u-g\|^2_{L_2(e)}$$

[H. Egger, C. Waluga, 2011]

Navier-Stokes

Navier-Stokes equations:

$$\begin{aligned} \partial_t \underline{u} - \nu \Delta \underline{u} + \left(\underline{u} \cdot \nabla\right) \underline{u} + \nabla p &= \underline{f} && \text{in} \quad Q, \\ \operatorname{div}(\underline{u}) &= 0 && \text{in} \quad Q, \\ \underline{u} &= \underline{g}_D && \text{on} \quad \Sigma_D, \\ \nu \left(\nabla \underline{u}\right) \underline{n}_x - p \underline{n}_x &= \underline{0} && \text{on} \quad \Sigma_N, \\ \underline{u}_{|t=0} &= \underline{u}_0 && \text{on} \quad \Sigma_0. \end{aligned}$$

Navier-Stokes

Navier-Stokes equations:

$$\begin{aligned} \partial_t \underline{u} - \nu \Delta \underline{u} + (\underline{u} \cdot \nabla)\, \underline{u} + \nabla p &= \underline{f} && \text{in} \quad Q, \\ \operatorname{div}(\underline{u}) &= 0 && \text{in} \quad Q, \\ \underline{u} &= \underline{g}_D && \text{on} \quad \Sigma_D, \\ \nu\, (\nabla \underline{u})\, \underline{n}_x - p \underline{n}_x &= \underline{0} && \text{on} \quad \Sigma_N, \\ \underline{u}_{|t=0} &= \underline{u}_0 && \text{on} \quad \Sigma_0. \end{aligned}$$

Variational formulation:

Find $\underline{u} \in V_h$ with $\underline{u} = \underline{g}_D$ on Σ_D and $p \in Q_h$:

$$\begin{aligned} a_{\mathrm{DG}}(\underline{u}, \underline{v}) + ((\underline{u} \cdot \nabla)\, \underline{u}, \underline{v})_Q + b(\underline{v}, p) &= \langle \underline{F}, \underline{v} \rangle_Q, \\ b(\underline{u}, q) &= 0, \end{aligned}$$

for all $\underline{v} \in V_h$ with $\underline{v} = \underline{0}$ on Σ_D, $q \in Q_h$.

Bilinearform: $b(\underline{v}, p) := -\sum_{k=1}^{N} \int_{\tau_k} p \operatorname{div}(\underline{v}) \mathrm{d}q + \sum_{e \in \mathcal{I}_N} \int_{e} \langle p \rangle_e \left[\underline{n}_x \cdot \underline{v} \right]_e \mathrm{d}s_q$

Navier-Stokes

Adaptive example

Flow over gap with $T = 0.5$ and $\underline{u}_0 = \underline{0}$:

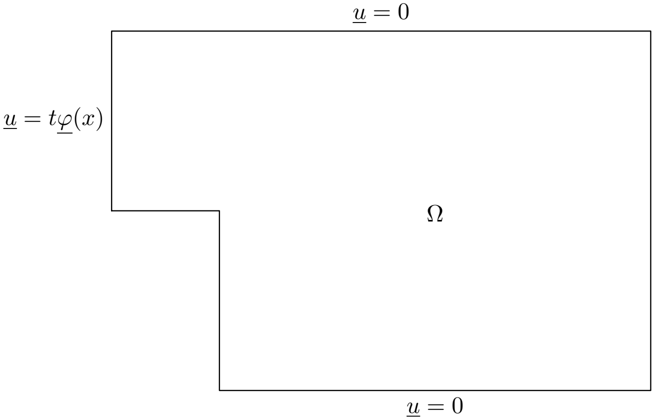


Figure: Domain Ω.

Navier-Stokes

Adaptive example

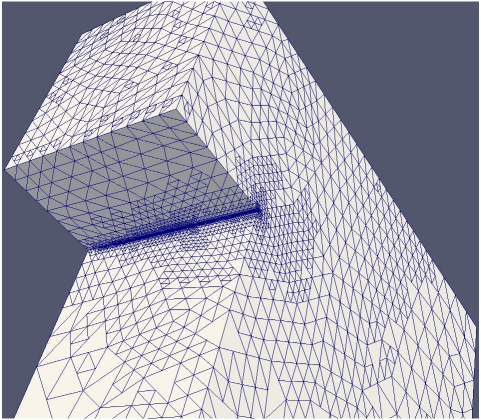


Figure: Adaptive space-time mesh for $\mathbb{P}_2 - \mathbb{P}_1$ elements.

Navier-Stokes

Adaptive example

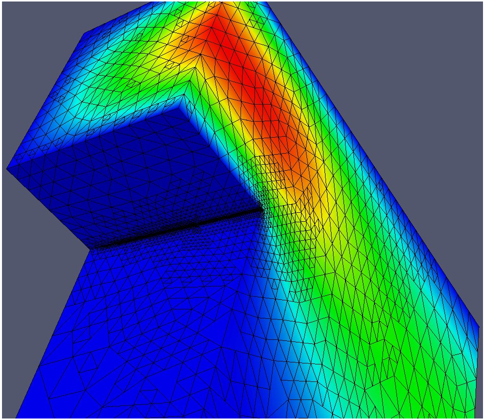


Figure: Adaptive space-time mesh for $\mathbb{P}_2 - \mathbb{P}_1$ elements.

Navier-Stokes

Adaptive example

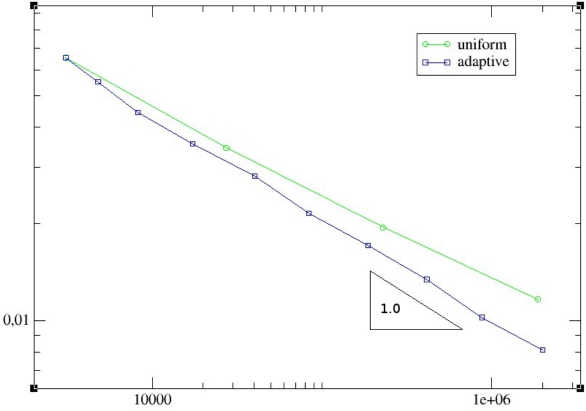


Figure: Results for $\mathbb{P}_1 - \mathbb{P}_0$.

Navier-Stokes

Adaptive example

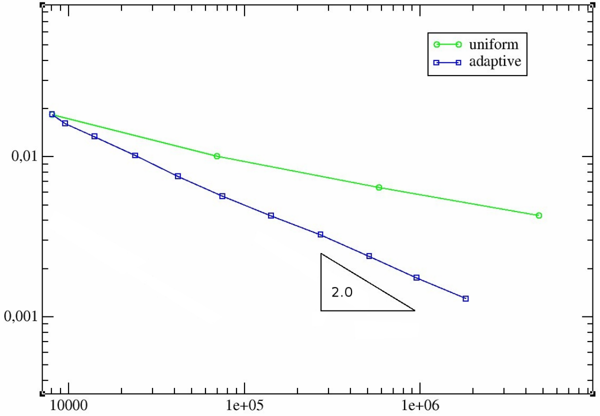
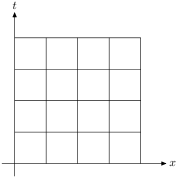
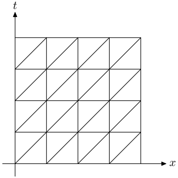


Figure: Results for $\mathbb{P}_2 - \mathbb{P}_1$.

Space-time multigrid method

Linear system

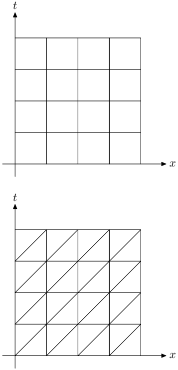




$$\begin{pmatrix} A_{\tau,h} & & & & \\ B_{\tau,h} & A_{\tau,h} & & & \\ & B_{\tau,h} & A_{\tau,h} & & \\ & & \ddots & \ddots & \\ & & & B_{\tau,h} & A_{\tau,h} \end{pmatrix} \begin{pmatrix} \underline{u}_1 \\ \underline{u}_2 \\ \underline{u}_3 \\ \vdots \\ \underline{u}_N \end{pmatrix} = \begin{pmatrix} \underline{f}_1 \\ \underline{f}_2 \\ \underline{f}_3 \\ \vdots \\ \underline{f}_N \end{pmatrix}$$

Space-time multigrid method

Linear system

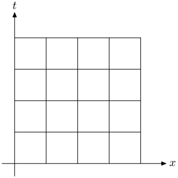


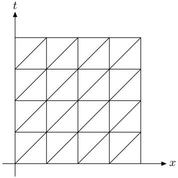
$$\begin{pmatrix} A_{\tau,h} & & & & \\ B_{\tau,h} & A_{\tau,h} & & & \\ & B_{\tau,h} & A_{\tau,h} & & \\ & & \ddots & \ddots & \\ & & & B_{\tau,h} & A_{\tau,h} \end{pmatrix} \begin{pmatrix} \underline{u}_1 \\ \underline{u}_2 \\ \underline{u}_3 \\ \vdots \\ \underline{u}_N \end{pmatrix} = \begin{pmatrix} \underline{f}_1 \\ \underline{f}_2 \\ \underline{f}_3 \\ \vdots \\ \underline{f}_N \end{pmatrix}$$

- Forward solve

Space-time multigrid method

Linear system





$$\begin{pmatrix} A_{\tau,h} & & & & \\ B_{\tau,h} & A_{\tau,h} & & & \\ & B_{\tau,h} & A_{\tau,h} & & \\ & & \ddots & \ddots & \\ & & & B_{\tau,h} & A_{\tau,h} \end{pmatrix} \begin{pmatrix} \underline{u}_1 \\ \underline{u}_2 \\ \underline{u}_3 \\ \vdots \\ \underline{u}_N \end{pmatrix} = \begin{pmatrix} \underline{f}_1 \\ \underline{f}_2 \\ \underline{f}_3 \\ \vdots \\ \underline{f}_N \end{pmatrix}$$

- Forward solve
- Multigrid method $\rightarrow$ smoother, restriction/prolongation

Smoother

Damped Jacobi–Smoother $\underline{x}^{k+1} = (1-\omega_t)\underline{x}^k + \omega_t D^{-1}\left[\underline{f} - L\underline{x}^k\right]$

$$\begin{pmatrix} A_{\tau,h} & & & & \\ B_{\tau,h} & A_{\tau,h} & & & \\ & B_{\tau,h} & A_{\tau,h} & & \\ & & \ddots & \ddots & \\ & & & B_{\tau,h} & A_{\tau,h} \end{pmatrix} \begin{pmatrix} \underline{u}_1 \\ \underline{u}_2 \\ \underline{u}_3 \\ \vdots \\ \underline{u}_N \end{pmatrix} = \begin{pmatrix} \underline{f}_1 \\ \underline{f}_2 \\ \underline{f}_3 \\ \vdots \\ \underline{f}_N \end{pmatrix}$$

Smoother

Damped Jacobi–Smoother $\qquad \underline{x}^{k+1} = (1-\omega_t)\underline{x}^k + \omega_t D^{-1}\left[\underline{f} - L\underline{x}^k\right]$

D^{-1} approximated by 1 space MG iteration with $\omega_t = \frac{1}{2}$:

(Jacobi-GS smoother)

Restriction/Prolongation

Restriction/Prolongation

- Mesh hierarchy in space

Restriction/Prolongation

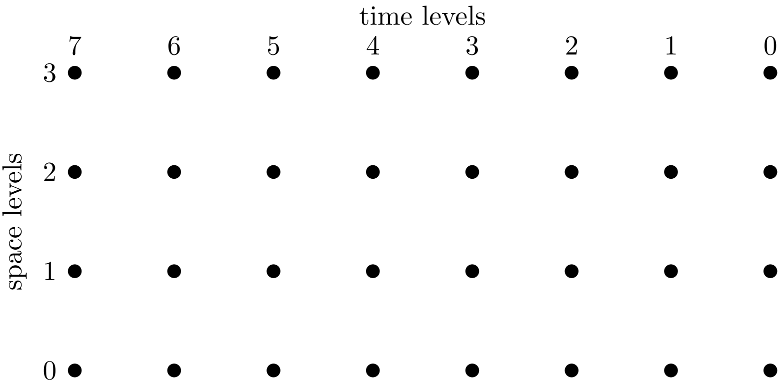
Restriction/Prolongation

- Mesh hierarchy in space
- Mesh hierarchy in time

Restriction/Prolongation

Restriction/Prolongation

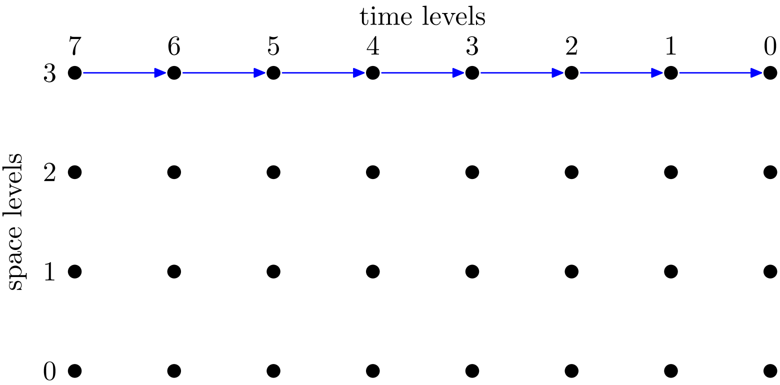
- Mesh hierarchy in space
- Mesh hierarchy in time



Restriction/Prolongation

Restriction/Prolongation

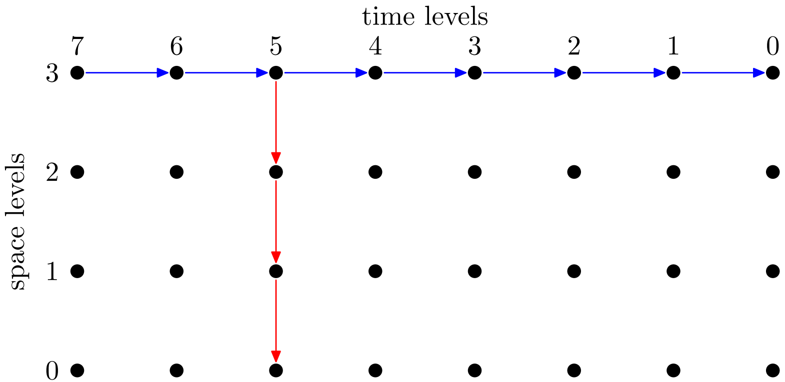
- Mesh hierarchy in space
- Mesh hierarchy in time



Restriction/Prolongation

Restriction/Prolongation

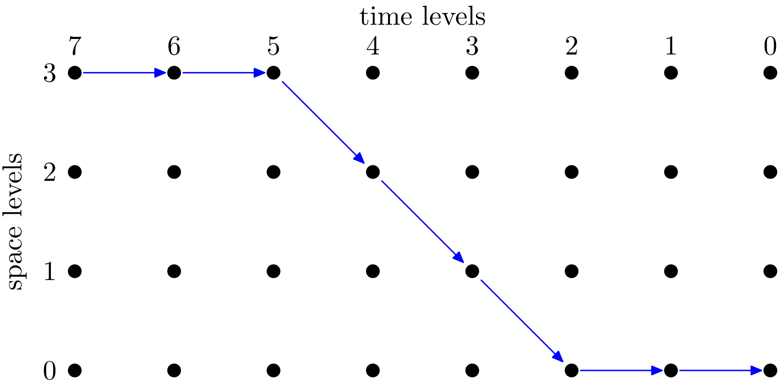
- Mesh hierarchy in space
- Mesh hierarchy in time



Restriction/Prolongation

Restriction/Prolongation

- Mesh hierarchy in space
- Mesh hierarchy in time



Multigrid iterations

- $\Omega = (0,1)^3$, $T = 1$ and exact solution $u(x,t) = \sin(\pi x_1)\sin(\pi x_2)\sin(\pi x_3)\sin(t)$
- space multigrid settings: $\omega_x = \frac{2}{3}$, $s_{x_1} = s_{x_2} = 1$, $\gamma_x = 1$
- time multigrid settings: $\omega_t = \frac{1}{2}$, $s_{t_1} = s_{t_2} = 2$, $\gamma_t = 1$
- relative tolerance $\varepsilon = 10^{-8}$

Multigrid iterations

- $\Omega = (0,1)^3$, $T = 1$ and exact solution $u(x,t) = \sin(\pi x_1)\sin(\pi x_2)\sin(\pi x_3)\sin(t)$
- space multigrid settings: $\omega_x = \frac{2}{3}$, $s_{x_1} = s_{x_2} = 1$, $\gamma_x = 1$
- time multigrid settings: $\omega_t = \frac{1}{2}$, $s_{t_1} = s_{t_2} = 2$, $\gamma_t = 1$
- relative tolerance $\varepsilon = 10^{-8}$

		time levels											
		0	1	2	3	4	5	6	7	8	9	10	11
space levels	0	1	6	7	6	6	7	8	9	9	10	10	10
	1	1	6	7	7	7	7	8	9	9	10	10	10
	2	1	11	11	11	10	9	8	9	9	10	10	10
	3	1	15	15	14	13	12	11	9	9	10	10	10
	4	1	20	19	18	17	15	14	13	11	10	10	10
	5	1	23	23	21	19	18	16	15	13	12	11	10

Table: Multigrid iterations with no space coarsening.

Degree of freedom at level $(5,7)$: $24.022.656 \to 1060.81s$

Degree of freedom at level $(5,11)$: $384.362.496 \to 9059.61s$

Multigrid iterations

- $\Omega = (0,1)^3$, $T = 1$ and exact solution $u(x,t) = \sin(\pi x_1)\sin(\pi x_2)\sin(\pi x_3)\sin(t)$
- space multigrid settings: $\omega_x = \frac{2}{3}$, $s_{x_1} = s_{x_2} = 1$, $\gamma_x = 1$
- time multigrid settings: $\omega_t = \frac{1}{2}$, $s_{t_1} = s_{t_2} = 2$, $\gamma_t = 1$
- relative tolerance $\varepsilon = 10^{-8}$

		time levels											
		0	1	2	3	4	5	6	7	8	9	10	11
space levels	0	1	6	7	6	6	7	8	9	9	10	10	10
	1	1	8	7	7	7	7	8	9	9	10	10	10
	2	1	14	14	11	10	9	8	9	9	10	10	10
	3	1	17	17	17	14	12	11	10	9	9	10	10
	4	1	20	20	20	20	16	14	12	11	10	10	10
	5	1	23	23	21	20	19	17	15	13	12	11	10

Table: Multigrid iterations with moderate space coarsening.

Degree of freedom at level $(5,7)$: $24.022.656 \rightarrow 671.38s$

Degree of freedom at level $(5,11)$: $384.362.496 \rightarrow 8553.27s$

Forward solve vs. multigrid method

- $\Omega = (0,1)^2$, $T = 1$ and exact solution $u(x,t) = \sin(\pi x_1)\sin(\pi x_2)\sin(t)$
- space multigrid settings: $\omega_x = \frac{2}{3}$, $s_{x_1} = s_{x_2} = 1$, $\gamma_x = 1$
- time multigrid settings: $\omega_t = 1$, $s_{t_1} = s_{t_2} = 1$, $\gamma_t = 1$

Forward solve vs. multigrid method

- $\Omega = (0,1)^2$, $T = 1$ and exact solution $u(x,t) = \sin(\pi x_1)\sin(\pi x_2)\sin(t)$
- space multigrid settings: $\omega_x = \frac{2}{3}$, $s_{x_1} = s_{x_2} = 1$, $\gamma_x = 1$
- time multigrid settings: $\omega_t = 1$, $s_{t_1} = s_{t_2} = 1$, $\gamma_t = 1$

		forward		ST multigrid		
L_x	L_t	time	iter	time	iter	dof
0	6	0.79	1.00	0.01	13	256
1	6	0.75	14.66	0.05	13	1.280
2	6	0.83	15.59	0.23	12	6.400
3	6	1.21	15.80	1.16	13	28.928
4	6	3.65	22.84	5.13	13	123.136
5	6	13.19	23.06	22.05	13	508.160
6	6	52.97	23.14	93.55	13	2.064.640
7	6	217.31	22.90	388.41	13	8.323.328
8	6	875.57	22.40	1513.31	13	33.423.616

Table: Solving times in [s].

Forward solve vs. multigrid method

- $\Omega = (0,1)^3$, $T = 1$ and exact solution $u(x,t) = \sin(\pi x_1)\sin(\pi x_2)\sin(\pi x_3)\sin(t)$
- space multigrid settings: $\omega_x = \frac{2}{3}$, $s_{x_1} = s_{x_2} = 1$, $\gamma_x = 1$
- time multigrid settings: $\omega_t = 1$, $s_{t_1} = s_{t_2} = 1$, $\gamma_t = 1$

		forward		ST multigrid		
L_x	L_t	time	iter	time	iter	dof
0	6	3.25	1	0.01	10	256
1	6	3.30	11.25	0.06	10	2.304
2	6	3.69	15.60	1.02	13	23.296
3	6	9.80	21.30	13.19	16	218.880
4	6	95.27	30.70	136.99	18	1.912.576
5	6	1031.43	37.05	1155.12	17	16.015.104
6	6	9970.89	39.53	10416.90	17	131.120.896

Table: Solving times in [s].

Forward solve vs. multigrid method

- $\Omega = (0,1)^3$, $T = 1$ and exact solution $u(x,t) = \sin(\pi x_1)\sin(\pi x_2)\sin(\pi x_3)\sin(t)$
- space multigrid settings: $\omega_x = \frac{2}{3}$, $s_{x_1} = s_{x_2} = 1$, $\gamma_x = 1$
- time multigrid settings: $\omega_t = 1$, $s_{t_1} = s_{t_2} = 1$, $\gamma_t = 1$

		forward		ST multigrid		
L_x	L_t	time	iter	time	iter	dof
0	6	3.25	1	0.01	10	256
1	6	3.30	11.25	0.06	10	2.304
2	6	3.69	15.60	1.02	13	23.296
3	6	9.80	21.30	13.19	16	218.880
4	6	95.27	30.70	136.99	18	1.912.576
5	6	1031.43	37.05	1155.12	17	16.015.104
6	6	9970.89	39.53	10416.90	17	131.120.896

Table: Solving times in [s].

→ Parallelization in time:

- [G. Horton, S. Vandewalle, 1995]
- [M.J. Gander, S. Vandewalle, 2007]
- [T. Weinzierl, T. Köppl, 2012]

Parallelization

- space multigrid settings: $\omega_x = \frac{2}{3}$, $s_{x_1} = s_{x_2} = 1$, $\gamma_x = 1$
- time multigrid settings: $\omega_t = 1$, $s_{t_1} = s_{t_2} = 1$, $\gamma_t = 1$

L_x	L_t	forw. solve	$P=1$	$P=2$		$P=4$		$P=8$		dof
0	6	0.79	0.01	0.01	78.26%	0.01	61.98%	0.00	49.71%	256
1	6	0.75	0.05	0.03	89.97%	0.01	82.81%	0.01	76.20%	1.280
2	6	0.83	0.23	0.12	92.30%	0.06	90.94%	0.03	85.27%	6.400
3	6	1.21	1.16	0.62	92.83%	0.31	93.60%	0.16	90.54%	28.928
4	6	3.65	5.13	2.69	95.21%	1.36	94.43%	0.69	93.32%	123.136
5	6	13.19	22.05	11.25	98.01%	5.68	97.14%	3.01	91.70%	508.160
6	6	52.97	93.55	49.02	95.42%	24.13	96.91%	13.33	87.72%	2.064.640
7	6	217.31	388.41	204.12	95.14%	101.09	96.05%	55.79	87.02%	8.323.328
8	6	875.57	1513.31	790.87	95.67%	404.96	93.42%	225.90	83.74%	33.423.616

Table: Solving times in [s] for $\Omega = (0,1)^2$.

L_x	L_t	forw. solve	$P=1$	$P=2$		$P=4$		$P=8$		dof
0	6	3.25	0.01	0.00	68.41%	0.00	47.70%	0.00	28.48%	256
1	6	3.30	0.06	0.04	88.52%	0.02	83.66%	0.01	72.81%	2.304
2	6	3.69	1.02	0.56	90.36%	0.29	89.04%	0.15	85.78%	23.296
3	6	9.80	13.19	6.78	97.24%	3.93	83.92%	1.86	88.68%	218.880
4	6	95.27	136.99	74.16	92.36%	36.69	93.36%	20.10	85.19%	1.912.576
5	6	1031.43	1155.12	605.07	95.45%	316.18	91.33%	168.90	85.49%	16.015.104
6	6	9970.89	10416.90	5251.48	99.18%	2800.79	92.98%	1503.15	86.63%	131.120.896

Table: Solving times in [s] for $\Omega = (0,1)^3$.

Summary and outlook

Summary:

- Space-time discretization

Summary and outlook

Summary:

- Space-time discretization
- Adaptivity

Summary and outlook

Summary:

- Space-time discretization
- Adaptivity
- Space-time multigrid method

Summary and outlook

Summary:

- Space-time discretization
- Adaptivity
- Space-time multigrid method

Outlook:

- Theory

Summary and outlook

Summary:

- Space-time discretization
- Adaptivity
- Space-time multigrid method

Outlook:

- Theory
- Adaptive solvers

Summary and outlook

Summary:

- Space-time discretization
- Adaptivity
- Space-time multigrid method

Outlook:

- Theory
- Adaptive solvers
- Space-time multigrid $\rightarrow$ Navier Stokes