technische universität dortmund

A Space-Time Multigrid Solver for Optimal Distributed Control of Incompressible Fluid Flow

Michael Köster, Stefan Turek

(supported by M.Hinze, Uni Hamburg)

Michael Köster

Institute for Applied Mathematics
TU Dortmund

Isola d’Ischia, 19.-23. Sep. 2010

Flow Control Model Problem

Distributed Control of the nonstationary Navier-Stokes equation with tracking-type functional for a given z:

$$J(y,u) = \frac{1}{2}||y-z||_Q^2 + \frac{\alpha}{2}||u||_Q^2 + \frac{\gamma}{2}||y(T)-z(T)||_\Omega^2 \quad \rightarrow \quad \text{min!}$$

on $Q = \Omega \times [0,T]$ such that

$$\begin{array}{rcll} y_t - \nu\Delta y + (y\nabla)y + \nabla p & = & u & \text{in } Q \\ -\nabla\cdot y & = & 0 & \text{in } Q \end{array} \qquad + \text{BC}$$

No constraints (for simplicity).

Design goals for optimal control tools

Aim:

- Discretisation with modern CFD techniques
- Solver with optimal complexity:

$$\text{costs for optimisation} = O(N)$$

$$\frac{\text{costs for optimisation}}{\text{costs for simulation}} \le C \approx 10 - 50$$

In the following:

1.) Discretisation
2.) Solver design
3.) Numerical examples

Design goals for optimal control tools

Aim:

- Discretisation with modern CFD techniques
- Solver with optimal complexity:

$$\text{costs for optimisation} = O(N)$$

$$\frac{\text{costs for optimisation}}{\text{costs for simulation}} \leq C \approx 10 - 50$$

In the following:

1.) Discretisation
2.) Solver design
3.) Numerical examples

Corresponding KKT-System (unconstrained case):

$$
\begin{aligned}
y_t - \nu\Delta y + (y\nabla)y + \nabla p &= u && \text{in } Q\\
-\nabla \cdot y &= 0 && \text{in } Q\\
\\
-\lambda_t - \nu\Delta\lambda - (y\nabla)\lambda + (\nabla y)^t\lambda + \nabla\xi &= (y - z) && \text{in } Q\\
-\nabla \cdot \lambda &= 0 && \text{in } Q\\
\\
u &= -\tfrac{1}{\alpha}\lambda && \text{in } Q
\end{aligned}
$$

+ boundary conditions
+ initial condition
+ terminal condition $\lambda(T) = \gamma(y(T) - z(T))$ in Ω

'Reduced' KKT-System (unconstrained case):

$$
\begin{aligned}
y_t + N(y)y + \nabla p &= -\tfrac{1}{\alpha}\lambda && \text{in } Q \\
-\lambda_t + N^*(y)\lambda + \nabla \xi &= (y - z) && \text{in } Q
\end{aligned}
$$

+ incompressibility ($-\nabla \cdot y = -\nabla \cdot \lambda = 0$ in Q)
+ initial/boundary conditions
+ terminal condition $\lambda(T) = \gamma(y(T) - z(T))$ in Ω

Properties of the KKT system

Observation:

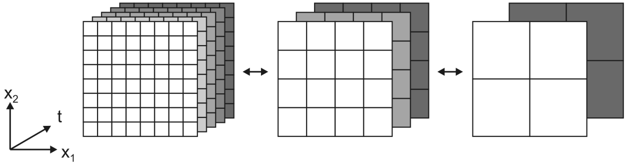
- KKT-system $\rightarrow$ *elliptic* BVP in space/time $(-y_{tt} + \Delta^2 y + \ldots)$

Idea:

- Apply 'optimal' $O(N)$ ingredients from CFD to this BVP!

  $\rightarrow$ unstructured meshes, FEM in space, implicit time-stepping
  $\rightarrow$ monolithic Multigrid + Newton solver techniques

- In particular: Solve on a space-time hierarchy



Space–time discretization

Discretization in time (θ–scheme) leads to a system

$$F(x) = A(x)x = b$$

in the form (here e.g. for 2 timesteps, IE in time):

$$A(x)x =$$

$$\left(\begin{array}{c|c|c}
\begin{array}{cc|cc}
\frac{I}{\Delta t}+N & & \nabla & \\
-I & \frac{I}{\Delta t}+N^* & & \nabla \\
\hline
-\nabla\cdot & & 0 & \\
 & -\nabla\cdot & & 0
\end{array} & -\frac{1}{\Delta t} & \\
\hline
-\frac{I}{\Delta t} & \begin{array}{cc|cc}
\frac{I}{\Delta t}+N & \frac{I}{\alpha} & \nabla & \\
-I & \frac{I}{\Delta t}+N^* & & \nabla \\
\hline
-\nabla\cdot & & 0 & \\
 & -\nabla\cdot & & 0
\end{array} & -\frac{I}{\Delta t} \\
\hline
 & -\frac{I}{\Delta t} & \begin{array}{cc|cc}
\frac{I}{\Delta t}+N & \frac{I}{\alpha} & \nabla & \\
-c(\gamma,\Delta t)I & \frac{I}{\Delta t}+N^* & & \nabla \\
\hline
-\nabla\cdot & & 0 & \\
 & -\nabla\cdot & & 0
\end{array}
\end{array}\right)
\left(\begin{array}{c}
y_0 \\ \lambda_0 \\ p_0 \\ \xi_0 \\ \hline y_1 \\ \lambda_1 \\ p_1 \\ \xi_1 \\ \hline y_2 \\ \lambda_2 \\ p_2 \\ \xi_2
\end{array}\right)$$

$\rightarrow$ **Sparse, (block) tridiagonal system**

Design of an efficient optimal control solver

- Nonlinearity: Newton method for quadratic convergence

$$x^{i+1} = x^i + F'^{-1}(x^i)(b - A(x^i)x^i)$$

- Linear subproblems: space-time MG solver (complexity $O(N)$)

  → using Block-Jacobi/Block-SOR smoothing techniques

- Linear subproblems in space: Monolithic Multigrid solver

  → using 'local Pressure-Schur complement' techniques in each timestep for the coupled Navier–Stokes subproblems

Space–time multigrid ingredients

Essential multigrid components:

- Mesh hierarchy for a space-time cylinder $Q = \Omega \times [0, T]$:
  Choose arbitrary space-time coarse mesh and refine!
- Prolongation/Restriction in space + time.
  Combination of FE in space & FD in time.
- An efficient smoother! E.g. with $\tilde{A} := A(x^i)$:

$$\tilde{A}v = d \qquad \Rightarrow \qquad v^{j+1} = v^j + \omega P^{-1}(d - \tilde{A}v^j)$$

What to use as preconditioner P?

Space–time multigrid ingredients

Essential multigrid components:

- Mesh hierarchy for a space-time cylinder $Q = \Omega \times [0, T]$:
  Choose arbitrary space-time coarse mesh and refine!
- Prolongation/Restriction in space + time.
  Combination of FE in space & FD in time.
- An efficient smoother! E.g. with $\tilde{A} := A(x^i)$:

$$\tilde{A}v = d \qquad \Rightarrow \qquad v^{j+1} = v^j + \omega P^{-1}(d - \tilde{A}v^j)$$

What to use as preconditioner P?

Space–time discretisation and preconditioner

$\tilde{A}$ in compressed form (omitting ∇ and $\nabla\cdot$ here):

$$
\left(\begin{array}{cc|cc|cc|c}
I+N & & & & & & \\
-I & \frac{I}{\Delta t}+N^* & & -\frac{I}{\Delta t} & & & \\
\hline
-\frac{I}{\Delta t} & & \frac{I}{\Delta t}+N & \frac{I}{\alpha} & & & \\
 & & -I & \frac{I}{\Delta t}+N^* & & -\frac{I}{\Delta t} & \\
\hline
 & & -\frac{I}{\Delta t} & & \frac{I}{\Delta t}+N & \frac{I}{\alpha} & \\
 & & & & -I & \frac{I}{\Delta t}+N^* & -\frac{I}{\Delta t} \\
\hline
 & & & & & \dots & \dots
\end{array}\right)
$$

$\Rightarrow$ Block-Jacobi preconditioner:

$$
P := P_{Jac} := \left(\begin{array}{cc|cc|cc|c}
I+N & & & & & & \\
-I & \frac{I}{\Delta t}+N^* & & & & & \\
\hline
 & & \frac{I}{\Delta t}+N & \frac{I}{\alpha} & & & \\
 & & -I & \frac{I}{\Delta t}+N^* & & & \\
\hline
 & & & & \frac{I}{\Delta t}+N & \frac{I}{\alpha} & \\
 & & & & -I & \frac{I}{\Delta t}+N^* & \\
\hline
 & & & & & & \dots
\end{array}\right)
$$

Space–time discretisation and preconditioner

$\tilde{A}$ in compressed form (omitting ∇ and $\nabla\cdot$ here):

$$
\left(\begin{array}{cc|cc|cc|c}
I+N & & & & & & \\
-I & \frac{I}{\Delta t}+N^* & & -\frac{I}{\Delta t} & & & \\
\hline
-\frac{I}{\Delta t} & & \frac{I}{\Delta t}+N & \frac{I}{\alpha} & & & \\
 & & -I & \frac{I}{\Delta t}+N^* & & -\frac{I}{\Delta t} & \\
\hline
 & & -\frac{I}{\Delta t} & & \frac{I}{\Delta t}+N & \frac{I}{\alpha} & \\
 & & & & -I & \frac{I}{\Delta t}+N^* & -\frac{I}{\Delta t} \\
\hline
 & & & & & \dots & \dots
\end{array}\right)
$$

$\Rightarrow$ Block-Jacobi preconditioner:

$$
P := P_{Jac} := \left(\begin{array}{cc|cc|cc|c}
I+N & & & & & & \\
-I & \frac{I}{\Delta t}+N^* & & & & & \\
\hline
 & & \frac{I}{\Delta t}+N & \frac{I}{\alpha} & & & \\
 & & -I & \frac{I}{\Delta t}+N^* & & & \\
\hline
 & & & & \frac{I}{\Delta t}+N & \frac{I}{\alpha} & \\
 & & & & -I & \frac{I}{\Delta t}+N^* & \\
\hline
 & & & & & & \dots
\end{array}\right)
$$

$\tilde{A}$ in compressed form (omitting ∇ and $\nabla\cdot$ here):

$$\left(\begin{array}{cc|cc|cc|c}
I+N & & & & & & \\
-I & \frac{I}{\Delta t}+N^* & & -\frac{I}{\Delta t} & & & \\
\hline
-\frac{I}{\Delta t} & & \frac{I}{\Delta t}+N & \frac{I}{\alpha} & & & \\
 & & -I & \frac{I}{\Delta t}+N^* & & -\frac{I}{\Delta t} & \\
\hline
 & & -\frac{I}{\Delta t} & & \frac{I}{\Delta t}+N & \frac{I}{\alpha} & \\
 & & & & -I & \frac{I}{\Delta t}+N^* & -\frac{I}{\Delta t} \\
\hline
 & & & & & \cdots & \cdots
\end{array}\right)$$

$\Rightarrow$ Block-Gauß-Seidel preconditioner:

$$P := P_{GS} := \left(\begin{array}{cc|cc|cc|c}
I+N & & & & & & \\
-I & \frac{I}{\Delta t}+N^* & & & & & \\
\hline
-\frac{I}{\Delta t} & & \frac{I}{\Delta t}+N & \frac{I}{\alpha} & & & \\
 & & -I & \frac{I}{\Delta t}+N^* & & & \\
\hline
 & & -\frac{I}{\Delta t} & & \frac{I}{\Delta t}+N & \frac{I}{\alpha} & \\
 & & & & -I & \frac{I}{\Delta t}+N^* & \\
\hline
 & & & & & \cdots & \cdots
\end{array}\right)$$

Space–time discretisation and preconditioner

$\tilde{A}$ in compressed form (omitting ∇ and $\nabla\cdot$ here):

$$
\left(\begin{array}{cc|cc|cc|c}
I+N & & & & & & \\
-I & \frac{I}{\Delta t}+N^* & & -\frac{I}{\Delta t} & & & \\
\hline
-\frac{I}{\Delta t} & & \frac{I}{\Delta t}+N & \frac{I}{\alpha} & & & \\
 & & -I & \frac{I}{\Delta t}+N^* & & -\frac{I}{\Delta t} & \\
\hline
 & & -\frac{I}{\Delta t} & & \frac{I}{\Delta t}+N & \frac{I}{\alpha} & \\
 & & & & -I & \frac{I}{\Delta t}+N^* & -\frac{I}{\Delta t} \\
\hline
 & & & & & \dots & \dots
\end{array}\right)
$$

$\Rightarrow$ Reversed Block-Gauß-Seidel preconditioner:

$$
P := P_{GS}^r := \left(\begin{array}{cc|cc|cc|c}
I+N & & & & & & \\
-I & \frac{I}{\Delta t}+N^* & & -\frac{I}{\Delta t} & & & \\
\hline
 & & \frac{I}{\Delta t}+N & \frac{I}{\alpha} & & & \\
 & & -I & \frac{I}{\Delta t}+N^* & & -\frac{I}{\Delta t} & \\
\hline
 & & & & \frac{I}{\Delta t}+N & \frac{I}{\alpha} & \\
 & & & & -I & \frac{I}{\Delta t}+N^* & \\
\hline
 & & & & & \dots & \dots
\end{array}\right)
$$

In each timestep:

$$A_{ii}(x)x_i = \left(\begin{array}{cc|cc} \frac{I}{\Delta t} + N_i & \frac{I}{\alpha} & \nabla & \\ -I & \frac{I}{\Delta t} + N_i^* & & \nabla \\ \hline -\nabla\cdot & & 0 & \\ & -\nabla\cdot & & 0 \end{array}\right) \begin{pmatrix} y_i \\ \lambda_i \\ p_i \\ \xi_i \end{pmatrix} = \begin{pmatrix} \mathrm{rhs}_i^y \\ \mathrm{rhs}_i^\lambda \\ \mathrm{rhs}_i^p \\ \mathrm{rhs}_i^\xi \end{pmatrix}$$

- Saddle-point character
- Application of A_{ii}^{-1}:
  - monolithic Multigrid solver in space
  - ‘local Pressure-Schur complement’ techniques

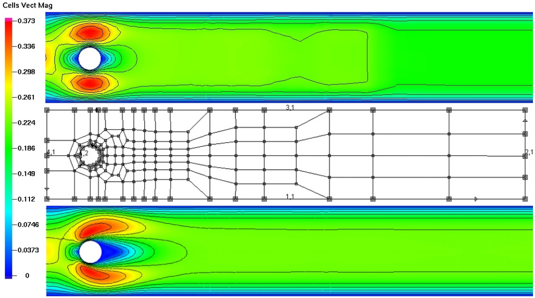
Discretisation:

- Finite Elements ($\rightarrow$ `FeatFlow`)

Numerical example

Flow-around-cylinder

- Target flow z: Stationary Stokes flow



Stokes, Target

Mesh

uncontrolled Nav.St.,
Initial condition

- Optimal control problem: Navier–Stokes, $Re = 20$, $t \in [0, 0.3]$.
- Coarse mesh: Standard DFG benchmark
  $\rightarrow$ 1404 DOF's in space, 16 timesteps, $\Delta t := 3/160$
  $\Rightarrow$ 23 868 DOF's, $\times 8$ per level

Convergence of the Newton solver

	Simulation			Optimisation			
lv.	#⊘NL	#⊘MG	T_{sim}	#NL	#MG	T_{opt}	$\frac{T_{opt}}{T_{sim}}$
2	3	11	2	5	16	112	56
3	3	11	15	5	27	1258	83
4	3	13	161	5	28	10226	63
5	3	15	1771	5	29	91529	52

- Nonlinear solver gained 8 digits
- Linear Solver: Space-Time-MG; gained 2 digits per step
- Space-preconditioner gained 2 digits per step
- $\alpha = 0.01$, $\gamma = 0.0$, FBSimSmoother

Pure single-grid solver

	Opt. with MG			Opt. without MG		
lv.	#NL	#MG	T_{opt}	#NL	#ite	T_{opt}
2	5	16	112	5	72	80
3	5	27	1258	5	160	1303
4	5	28	10226	5	360	23242

- Nonlinear solver gained 8 digits
- Space-time linear solver gained 2 digits per step
- Space-preconditioner gained 2 digits per step
- $\alpha = 0.01$, $\gamma = 0.0$, FBSimSmoother

$\Rightarrow$ Multigrid necessary

The role of the MG cycle

Choice of the MG cycle + hierarchy is crucial for the complexity!

Example (for simplicity): Heat equation

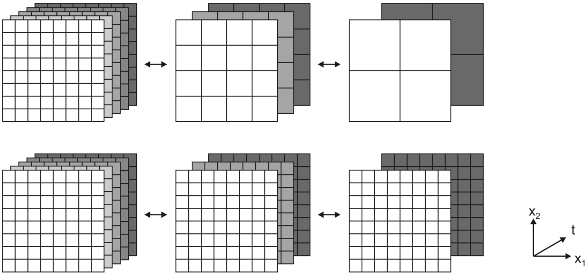
$$y_t - \Delta y = -\frac{1}{\alpha}\lambda \quad \text{in } Q$$

$$-\lambda_t - \Delta\lambda = (y - z) \quad \text{in } Q$$

Discr.: Space=Q_1, Time=IE or CN. Refinement/Coarsening:

1.) Space-time MG

2.) Pure time MG



The role of the MG cycle

- Time for NSM smoothing steps on level n: T_s^n.
- Total time for smoothing per MG step on level 2,...,n:

a) V-cycle: $T_s(2,\dots,n) = T_s^n + T_s^{n-1} + T_s^{n-2} + \dots + T_s^2$

b) W-cycle: $T_s(2,\dots,n) = T_s^n + 2T_s^{n-1} + 4T_s^{n-2} + \dots + 2^{n-2}T_s^2$

- Relationship of smoother time between levels (2D):

a) Full space-time multigrid: $T_s^{n-1} \approx 1/8\, T_s^n$

b) Time-multigrid: $T_s^{n-1} \approx 1/2\, T_s^n$

- Total numerical effort for smoothing $T_s(2,\dots,n)$:

	space-time MG	pure time-MG
V-cycle	$\leq 8/7\, T_s^n$	$\leq 2\, T_s^n$
W-cycle	$\leq 4/3\, T_s^n$	$\approx (n-1)\, T_s^n$

The role of the MG cycle

- Time for NSM smoothing steps on level n: T_s^n.
- Total time for smoothing per MG step on level 2,...,n:

a) V-cycle: $T_s(2,...,n) = T_s^n + T_s^{n-1} + T_s^{n-2} + ... + T_s^2$

b) W-cycle: $T_s(2,...,n) = T_s^n + 2\,T_s^{n-1} + 4\,T_s^{n-2} + ... + 2^{n-2}\,T_s^2$

- Relationship of smoother time between levels (2D):

a) Full space-time multigrid: $T_s^{n-1} \approx 1/8\;T_s^n$

b) Time-multigrid: $T_s^{n-1} \approx 1/2\;T_s^n$

- Total numerical effort for smoothing $T_s(2,...,n)$:

	space-time MG	pure time-MG
V-cycle	$\leq\; 8/7\;T_s^n$	$\leq 2\;T_s^n$
W-cycle	$\leq\; 4/3\;T_s^n$	$\approx (n-1)\;T_s^n$

The role of the MG cycle

In practise:

- $\Omega = [0,1]^2$, $[0, T] = [0,1]$, $\alpha = 0.001$, $\gamma = 1000$.
- analyt. solution: $y = t^2(1-t)^2 x_1$. λ, z appropriately.
- space=Q_1, time=IE
- MG-solver with BiCGStab(Block-GS)-smoother.

	Full space-time MG				Time-MG			
	V-cycle		W-cycle		V-cycle		W-cycle	
	#ite	time	#ite	time	#ite	time	#ite	time
$T_s(5,...,5)$	4	245	4	245	3	181	3	182
$T_s(4,...,5)$	4	268	4	291	3	272	3	366
$T_s(3,...,5)$	4	269	4	298	3	320	3	531
$T_s(2,...,5)$	4	270	4	300	3	345	3	663
	$8/7 \approx \frac{270}{245}$		$4/3 \approx \frac{300}{245}$		$2 \approx \frac{345}{181}$		$n-1 \approx \frac{663}{182}$	

Summary and Outlook

Shown:

- Concept and realisation of an optimal control flow solver with optimal complexity

Key points:

- linear complexity (due to MG techniques)
- flexible (due to FEM approach, higher order + 3D possible)
- robust (FEM-stab. possible, e.g. EO-FEM/interior penalty)
- generalisation to more complex problems possible

  (Non-Newtonian + Non-isothermal flow,
  boundary control, constraint control)

Choice of the MG cycle + hierarchy is crucial for the complexity!

Example (for simplicity): Heat equation

$$
\begin{aligned}
y_t - \Delta y &= -\frac{1}{\alpha}\lambda && \text{in } Q \\
-\lambda_t - \Delta \lambda &= (y - z) && \text{in } Q
\end{aligned}
$$

- Discr.: Space=Q_1, Time=IE or CN. Refinement/Coarsening:

Variant 1: Full space-time MG

lv.	#steps	#DOF
1	5	50
2	10	162
3	20	578
4	40	2178
5	80	8450

Variant 2: Pure time MG

lv.	#steps	#DOF
1	5	8450
2	10	8450
3	20	8450
4	40	8450
5	80	8450

Convergence of the Newton solver

	Simulation			Optimisation			
lv.	#⊘NL	#⊘MG	T_{sim}	#NL	#MG	T_{opt}	$\frac{T_{opt}}{T_{sim}}$
2	3	11	4	4	8	308	73
3	3	11	24	4	8	1727	72
4	3	13	208	5	11	15355	74

- Nonlinear solver gained 8 digits
- Linear Solver: Space-Time-MG; gained 2 digits per step
- Space-preconditioner gained 2 digits per step
- $\alpha = 0.01$, $\gamma = 0.0$, BiCGStab(FBGSSmoother)

Convergence of the Newton solver

	Simulation			Optimisation			
lv.	#⊘NL	#⊘MG	T_{sim}	#NL	#MG	T_{opt}	$\frac{T_{opt}}{T_{sim}}$
2	3	11	2	4	5	51	25
3	3	11	15	4	6	455	30
4	3	13	161	4	7	4483	28
5	3	15	1771	4	7	39925	23

- Nonlinear solver gained 8 digits
- Linear Solver: Time-MG; gained 2 digits per step
- Space-preconditioner gained 2 digits per step
- $\alpha = 0.01$, $\gamma = 0.0$, FBSimSmoother

Convergence for different α/γ

α	1.0			0.1			0.01		
γ	#NL	#MG	Time	#NL	#MG	Time	#NL	#MG	Time
0.0	4	11	516	4	15	705	5	27	1258
0.1	4	14	653	4	17	800	5	36	1675
0.5	4	15	701	4	23	1082	div	div	div
1.0	4	17	795	4	24	1123	div	div	div

- Nonlinear solver gained 8 digits
- Linear Solver: Space-Time-MG; gained 2 digits per step
- Space-preconditioner gained 2 digits per step
- Space/time level 3, FBSimSmoother

Convergence for different α/γ

α	1.0			0.1			0.01		
γ	#NL	#MG	Time	#NL	#MG	Time	#NL	#MG	Time
0.0	3	4	303	3	4	309	4	6	454
0.1	4	6	437	4	6	455	4	5	382
0.5	4	6	440	4	6	465	div	div	div
1.0	4	6	440	4	6	463	div	div	div

- Nonlinear solver gained 8 digits
- Linear Solver: Time-MG; gained 2 digits per step
- Space-preconditioner gained 2 digits per step
- Space/time level 3, FBSimSmoother

Convergence for different α/γ

α	1.0			0.1			0.01		
γ	#NL	#MG	Time	#NL	#MG	Time	#NL	#MG	Time
0.0	4	5	1260	4	7	1757	5	10	2519
0.1	4	6	1490	4	9	2215	4	9	2247
0.5	3	5	1295	4	10	2445	4	11	2711
1.0	3	6	1499	4	11	2719	4	11	2743

- Nonlinear solver gained 8 digits
- Linear Solver: Space-Time-MG; gained 2 digits per step
- Space-preconditioner gained 2 digits per step
- Space/time level 3, BiCGStab(FBGSSmoother)

$\Rightarrow$ Much more stable, but much slower!