A monolithic space-time multigrid solver for distributed control of the time-dependent Navier-Stokes system

Michael Köster, Stefan Turek

(supported by M.Hinze, Uni Hamburg)

Michael Köster

Institute for Applied Mathematics
TU Dortmund

Uppsala, June 29th – July 3 2009

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

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

No constraints (for simplicity).

Design goals for optimal control tools

Aim: Moderate performance measure; for C not too large (≈ 10):

$$\frac{\text{costs for optimization}}{\text{costs for simulation}} \leq C$$

By modern numerical CFD techniques
($\rightarrow$ special FEM on solution adapted grids, fast MG+Newton solvers)

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

Aim:
$$\text{costs for optimization} \stackrel{!}{=} O(N)$$

Distributed Control of nonstationary flow

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

Distributed Control of nonstationary flow

Corresponding KKT-System (unconstrained case):

$$
\begin{aligned}
y_t + N(y)y + \nabla p + \tfrac{1}{\alpha}\lambda &= 0 && \text{in } Q \\
-\nabla \cdot y &= 0 && \text{in } Q \\[1em]
-\lambda_t + N^*(y)\lambda + \nabla \xi - y &= -z && \text{in } Q \\
-\nabla \cdot \lambda &= 0 && \text{in } Q
\end{aligned}
$$

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

Properties of the KKT system

Observation:

- KKT-system $\longrightarrow$ *elliptic* BVP in space/time

Idea:

- Apply highly efficient ingredients from CFD (Multigrid + Newton) to this BVP!

Feasible?

Space–time discretisation

KKT-System:

$$\begin{pmatrix} y_t \\ -\lambda_t \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} N(y) & \frac{1}{\alpha} & \nabla & \\ -I & N^*(y) & & \nabla \\ -\nabla\cdot & & 0 & \\ & -\nabla\cdot & & 0 \end{pmatrix} \begin{pmatrix} y \\ \lambda \\ p \\ \xi \end{pmatrix} = \begin{pmatrix} 0 \\ -z \\ 0 \\ 0 \end{pmatrix}$$

$$\Rightarrow \quad \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} + \begin{pmatrix} N(w) & \nabla \\ -\nabla\cdot & 0 \end{pmatrix} \begin{pmatrix} w \\ q \end{pmatrix} = \begin{pmatrix} f \\ 0 \end{pmatrix}$$

$\Rightarrow$ similar to a generalised Navier–Stokes equation

Space–time discretization

Discretization in space+time leads to a system

$$A(x)x = b$$

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

$$A(x)x = \left(\begin{array}{c|c|c}
\begin{array}{cc|cc} NST & & -B & \\ -M & NST^* & & -B \\ \hline -B^T & & 0 & \\ & -B^T & & 0 \end{array} & -\frac{M}{\Delta t} & \\ \hline
-\frac{M}{\Delta t} & \begin{array}{cc|cc} NST & \frac{M}{\alpha} & -B & \\ -M & NST^* & & -B \\ \hline -B^T & & 0 & \\ & -B^T & & 0 \end{array} & -\frac{M}{\Delta t} \\ \hline
 & -\frac{M}{\Delta t} & \begin{array}{cc|cc} NST & \frac{M}{\alpha} & -B & \\ -c(\gamma, \Delta t)M & NST^* & & -B \\ \hline -B^T & & 0 & \\ & -B^T & & 0 \end{array}
\end{array}\right)
\left(\begin{array}{c} y_0 \\ \lambda_0 \\ p_0 \\ \xi_0 \\ \hline y_1 \\ \lambda_1 \\ p_1 \\ \xi_1 \\ \hline y_2 \\ \lambda_2 \\ p_2 \\ \xi_2 \end{array}\right)$$

$\rightarrow$ Sparse, (block) tridiagonal system

Design of an efficient optimal control solver

- Nonlinearity: Newton method for quadratic convergence

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

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

  $\longrightarrow$ using Block-Jacobi/Block-SOR smoothing techniques

- Linear subproblems in space: Monolithic Multigrid solver

  $\longrightarrow$ 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 $\tilde{A} := A'(x^i)$:

$$v^{j+1} = v^j + \omega P^{-1}(b - \tilde{A}v^j), \qquad j = 1, ..., \mathrm{NSM}$$

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 $\tilde{A} := A'(x^i)$:

$$v^{j+1} = v^j + \omega P^{-1}(b - \tilde{A}v^j), \qquad j = 1, ..., \mathrm{NSM}$$

What to use as preconditioner P?

Space–time discretisation and preconditioner

$\tilde{A}$ in compressed form (omitting B and B^T here):

$$
\left(\begin{array}{cc|cc|cc|cc}
M & & & & & & & \\
-M & NST^* & & -\frac{M}{\Delta t} & & & & \\
\hline
-\frac{M}{\Delta t} & & NST & \frac{1}{\alpha}M & & & & \\
& & -M & NST^* & & -\frac{M}{\Delta t} & & \\
\hline
& & -\frac{M}{\Delta t} & & NST & \frac{1}{\alpha}M & & \\
& & & & -M & NST^* & -\frac{M}{\Delta t} & \\
\hline
& & & & & \ldots & \ldots &
\end{array}\right)
$$

$\Rightarrow$ Block-Jacobi preconditioner:

$$
P := P_{Jac} := \left(\begin{array}{cc|cc|cc|c}
M & & & & & & \\
-M & NST^* & & & & & \\
\hline
& & NST & \frac{1}{\alpha}M & & & \\
& & -M & NST^* & & & \\
\hline
& & & & NST & \frac{1}{\alpha}M & \\
& & & & -M & NST^* & \\
\hline
& & & & & & \ldots
\end{array}\right)
$$

$\tilde{A}$ in compressed form (omitting B and B^T here):

$$
\left(\begin{array}{cc|cc|cc|cc}
M & & & & & & & \\
-M & NST^* & & -\frac{M}{\Delta t} & & & & \\
\hline
-\frac{M}{\Delta t} & & NST & \frac{1}{\alpha}M & & & & \\
 & & -M & NST^* & & -\frac{M}{\Delta t} & & \\
\hline
 & & -\frac{M}{\Delta t} & & NST & \frac{1}{\alpha}M & & \\
 & & & & -M & NST^* & & -\frac{M}{\Delta t} \\
\hline
 & & & & & \dots & & \dots
\end{array}\right)
$$

$\Rightarrow$ Block-Jacobi preconditioner:

$$
P := P_{Jac} := \left(\begin{array}{cc|cc|cc|c}
M & & & & & & \\
-M & NST^* & & & & & \\
\hline
 & & NST & \frac{1}{\alpha}M & & & \\
 & & -M & NST^* & & & \\
\hline
 & & & & NST & \frac{1}{\alpha}M & \\
 & & & & -M & NST^* & \\
\hline
 & & & & & & \dots
\end{array}\right)
$$

Space–time discretisation and preconditioner

$\tilde{A}$ in compressed form (omitting B and B^T here):

$$
\left(\begin{array}{cc|cc|cc|cc}
M & & & & & & & \\
-M & NST^* & & -\frac{M}{\Delta t} & & & & \\
\hline
-\frac{M}{\Delta t} & & NST & \frac{1}{\alpha}M & & & & \\
 & & -M & NST^* & & -\frac{M}{\Delta t} & & \\
\hline
 & & -\frac{M}{\Delta t} & & NST & \frac{1}{\alpha}M & & \\
 & & & & -M & NST^* & & -\frac{M}{\Delta t} \\
\hline
 & & & & & \dots & & \dots
\end{array}\right)
$$

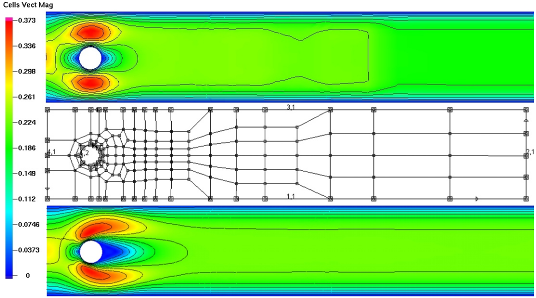
$\Rightarrow$ Block-Gauß-Seidel preconditioner:

$$
P := P_{GS} := \left(\begin{array}{cc|cc|cc|cc}
M & & & & & & & \\
-M & NST^* & & & & & & \\
\hline
-\frac{M}{\Delta t} & & NST & \frac{1}{\alpha}M & & & & \\
 & & -M & NST^* & & & & \\
\hline
 & & -\frac{M}{\Delta t} & & NST & \frac{1}{\alpha}M & & \\
 & & & & -M & NST^* & & \\
\hline
 & & & & & \dots & & \dots
\end{array}\right)
$$

Numerical example

Flow-around-cylinder

- Target flow z: Stokes flow, $t \in [0, 1]$, starting from rest



Stokes at $t = 1$

Mesh

uncontrolled Nav.St.

- Optimal control problem: Navier–Stokes, $Re = 20$
- Coarse mesh: Standard DFG benchmark
  $\rightarrow$ 1404 DOF's in space, 5 timesteps, $\Delta t := 0.2$
  $\Rightarrow$ 8 424 DOF's, $\times 8$ per level

Numerical example

Convergence of the Newton solver

		Simulation				Optimisation	
Δt	Space-Lv.	#NL	#MG	⌀#NL	⌀#MG	#NL	#MG
1/20	3	63	312	3	16	4	47
1/40	4	123	709	3	18	4	14
1/80	5	246	1589	3	20	4	9

- Nonlinear solver gained 5 digits
- Space-time MG gained 2 digits per step
- Space-preconditioner gained 2 digits per step

Numerical example

Convergence of the Newton solver

Δt	Space-Lv.	T_{sim}	T_{opt}	$\frac{T_{\text{opt}}}{T_{\text{sim}}}$
1/20	3	27.0	1384.42	51.3
1/40	4	209.6	3895.59	18.6
1/80	5	2227.1	22882.87	10.3

$\Rightarrow C \approx 10 - 20$ on reasonable refinement levels.

Numerical example

technische universität dortmund

Convergence of the solver for different α and γ

α	0.05		0.01		0.005	
γ	#NL	#MG	#NL	#MG	#NL	#MG
0	4	14	4	14	4	16
0.1	4	14	4	14	5	17
0.3	4	15	5	20	5	23
0.5	5	16	6	29	5	79

$\Delta t = 1/40$, Space-level 4

$\Rightarrow$ Smaller α + larger γ $\longrightarrow$ worse convergence rates

$\Rightarrow$ stronger smoother with 'black box' character required

Numerical example

Convergence of the solver for different ν

		$\nu = 1/1000$		$\nu = 1/250$	
Δt	Space-Lv.	#NL	#MG	#NL	#MG
1/10	2	4	36	4	13
1/20	3	4	24	3	8
1/40	4	4	14	3	7
1/80	5	4	13	3	8

$\Delta t = 1/40$, Space-level 4

$\Rightarrow$ Stabilisation necessary for higher RE numbers

e.g. EO-stabilisation (consistent, only in space)

$$j(u,v) = \sum_{\text{edge}E} \gamma |E|^2 \int_E [\nabla u][\nabla v] d\sigma$$

Numerical example

Convergence of the solver for different ν

		$\nu = 1/1000$		$\nu = 1/250$	
Δt	Space-Lv.	#NL	#MG	#NL	#MG
1/10	2	4	36	4	13
1/20	3	4	24	3	8
1/40	4	4	14	3	7
1/80	5	4	13	3	8

$\Delta t = 1/40$, Space-level 4

$\Rightarrow$ Stabilisation necessary for higher RE numbers

e.g. EO-stabilisation (consistent, only in space)

$$j(u, v) = \sum_{\text{edge} E} \gamma |E|^2 \int_E [\nabla u][\nabla v] d\sigma$$

Summary and Outlook

Shown:

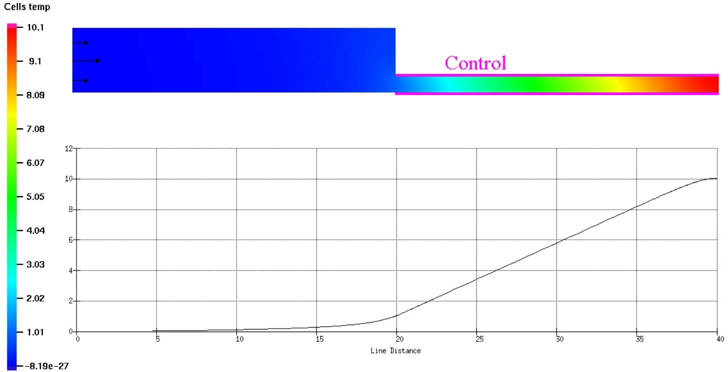
- Concept and realisation of an optimal control flow solver with linear complexity ($C \approx 10 - 20$)

Key points:

- linear complexity (due to MG techniques)
- flexible (due to FEM approach, 3D possible)
- robust (FEM-stabilisation possible, e.g. EO-FEM/interior penalty)
- generalisation to more complex problems possible (Non-Newtonian + Non-isothermal flow, boundary control, constraint control)

Example: Temperature by friction

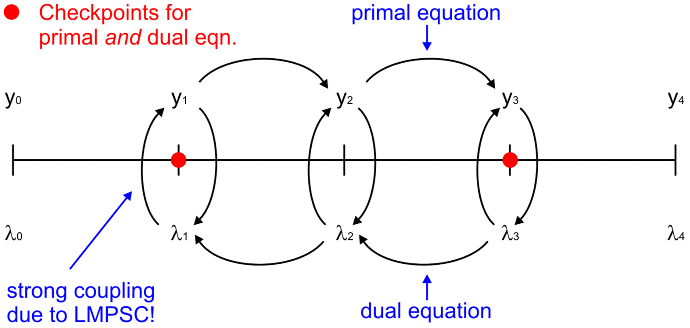
Friction leads to a temperature increase in the rear area of the channel:



$$
\begin{aligned}
y_t - \nabla \cdot (\nu(y,\Theta)\mathbf{D}(y)) + y\nabla y + \nabla p - Gr\Theta g &= f_1 \\
\Theta_t - (1/Pr)\Delta\Theta + y\nabla\Theta + D(y) : D(y) &= f_2
\end{aligned}
$$

$\longrightarrow$ How to control to prevent the temperature increase?

Checkpointing in the One-shot approach



- Checkpoints $\rightarrow$ nonlinear subproblems of the same kind.
- High computational costs necessary for recomputation
  $\rightarrow$ due to strong coupling by LPSC!