An Optimal Domain Decomposition Method for the C-Grid Navier-Stokes Jacobian

Jonas Thies[⋆] Sven Baars Mark v.d. Klok Fred Wubs[†]

[⋆] German Aerospace Center
Simulation and Software Technology

[†] University of Groningen
Department of Mathematics and Computer Science

project ESSEX

Problem Setting: Nonlinear PDE System

2^{nd} order PDE after space discretization

- $\mathrm{M}\frac{\partial \Phi}{\partial t} = \mathsf{F}(\Phi, t)$
- with suitable boundary and initial conditions

Steady state; Φ as $t \to \infty$.

Standard technique: time stepping

- may take very long
- no information about stability of solution
- low frequency modes affect solution on very long time scales

Problem Setting: Nonlinear PDE System

Example: 3D Boussinesq equations

$$
\begin{aligned}
\frac{\partial u}{\partial t} &= -\left((uu)_x + (vu)_y + (wu)_z\right) - p_x + \nu\nabla^2 u \\
\frac{\partial v}{\partial t} &= -\left((uv)_x + (vv)_y + (wv)_z\right) - p_y + \nu\nabla^2 v \\
\frac{\partial w}{\partial t} &= -\left((uw)_x + (vw)_y + (ww)_z\right) - p_z + \nu\nabla^2 w + g\alpha T \\
\frac{\partial T}{\partial t} &= -\left((uT)_x + (vT)_y + (wT)_z\right) + \kappa\nabla^2 T
\end{aligned}
$$

$$u_x + v_y + w_z = 0$$

Jacobian has saddle point structure:

$$\mathrm{J} = \begin{pmatrix} A & -\mathrm{Grad} \\ \mathrm{Div} & 0 \end{pmatrix}.$$

Example: Rayleigh-Bénard Convection

- Cube-shaped domain
- heated from below
- Rayleigh-Number

$$\mathrm{Ra} = \frac{\alpha g \Delta T d^3}{\nu \kappa}$$

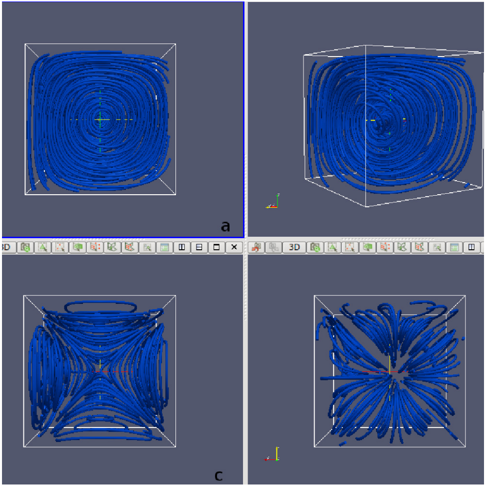
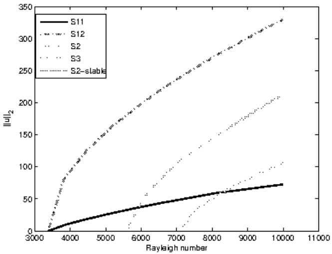


Figure. **Flow patterns near the first three primary bifurcations**
(a) x/y roll,
(b) diagonal roll,
(c) four rolls,
(d) toroidal roll

Continuation Methods

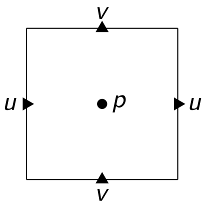
- step through parameter space rather than time
- predictor/corrector method
- Newton-Krylov
- linear stability analysis (Eigenproblems)

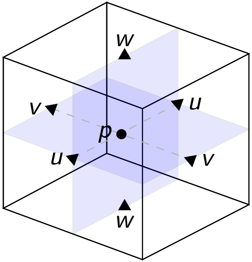
Bifurcation diagram



Depends crucially on strong preconditioning

C-grid and $\mathcal{F}$-matrices





- Discretization yields saddle point problem

$$\begin{bmatrix} A & B \\ B^T & O \end{bmatrix} \begin{bmatrix} \vec{u} \\ p \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}$$

$\mathcal{F}$-matrix: A is spd and B has at most two nonzeroes per row and row sum zero.

Computing an LU Decomposition of an $\mathcal{F}$-matrix

$$\begin{bmatrix} A & B \\ B^T & 0 \end{bmatrix} \begin{bmatrix} x_v \\ x_p \end{bmatrix} = \begin{bmatrix} f_v \\ f_p \end{bmatrix} \begin{matrix} \mathrm{V-nodes} \\ \mathrm{P-nodes} \end{matrix}$$

Algorithm: LU decomposition of an $\mathcal{F}$-matrix.

- Compute a fill-reducing ordering for the graph $F(A) \cup F(BB^T)$,
- during Gaussian elimination, insert the P-nodes to form 2×2 pivots whenever a coupling between a V-node and a P-node is encountered.

Theorem (De Niet/Wubs 2009)

In every step of this algorithm, the resulting Schur complement is an $\mathcal{F}$-matrix.

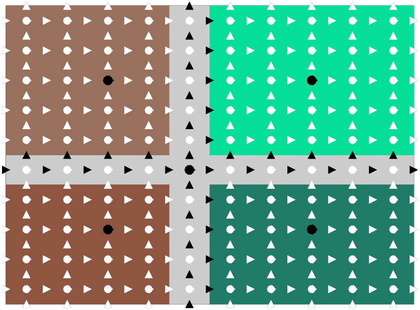
How is Fill Generated in the Direct Approach?

$$\left[\begin{array}{cc|cc} \alpha & \beta & a^T & b^T \\ \beta & 0 & \hat{b}^T & 0 \\ \hline a & \hat{b} & \hat{A} & \hat{B} \\ b & 0 & \hat{B}^T & O \end{array}\right]. \tag{1}$$

Elimination step:

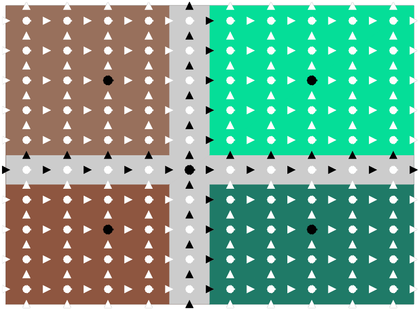
- Multiple of $\hat{b}\hat{b}^T$ is added to $\hat{A}$;
- $\hat{b}$ becomes denser as P-nodes are eliminated;
- So dropping in $\hat{A}$ doesn't make sense.

Domain Decomposition



- Subdomains and separators;
- Retain one pressure per subdomain.

Domain Decomposition

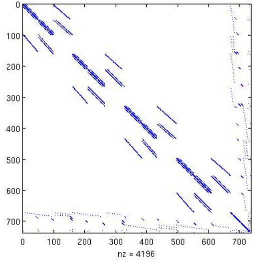


- Subdomains and separators;
- Retain one pressure per subdomain.

- This ordering exposes parallelism in the matrix:

$$K \Longrightarrow \begin{pmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{pmatrix},$$

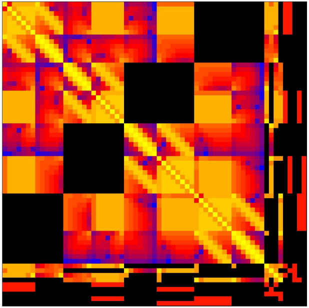
where K_{11} is block-diagonal.



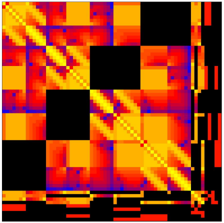
The Schur Complement

- LU-decomposition of the matrices on the subdomains, $K_{11} = L_{11} U_{11}$;
- Schur-complement: $S = K_{22} - K_{21} K_{11}^{-1} K_{12}$;
- retains $\mathcal{F}$-matrix-property of K;
- only a few rather dense 'B' columns (with at most two entries per row);

Schur complement
(four subdomains)

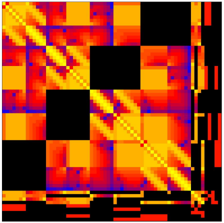


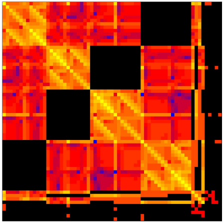
Sparse Approximation of the Schur Complement



- Still an $\mathcal{F}$-matrix;
- All V-nodes on a separator are now connected to the same 2 P-nodes;
- Use orthogonal transformation to disconnect them.

Sparse Approximation of the Schur Complement

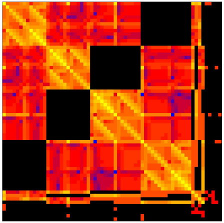




- Still an $\mathcal{F}$-matrix;
- All V-nodes on a separator are now connected to the same 2 P-nodes;
- Use orthogonal transformation to disconnect them.

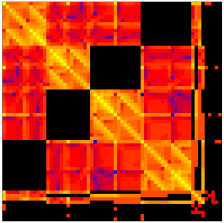
$\Longrightarrow$ Only one V-node per separator remains connected to P-nodes (V_Σ-nodes)

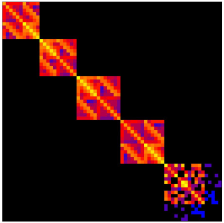
Dropping



- Use simple drop-by-position:
  - Drop all couplings between separator groups
  - ... and all couplings between V_Σ and regular V-nodes.

Dropping





- Use simple drop-by-position:
  - Drop all couplings between separator groups
  - ... and all couplings between V_Σ and regular V-nodes.

$\Longrightarrow$ Block diagonal preconditioner with a 'reduced matrix' S_2 in the lower right.

Robustness and $\mathcal{O}(N \log N)$ with an ILU???

Robustness and $\mathcal{O}(N \log N)$ with an ILU???

1. Orthogonal transformations:
    - Eliminate most V-P couplings to avoid fill;
    - 'Transfer operators' defining coarse problem S_2.

Robustness and $\mathcal{O}(N \log N)$ with an ILU???

1. Orthogonal transformations:
   - Eliminate most V-P couplings to avoid fill;
   - 'Transfer operators' defining coarse problem S_2.
2. Coarse problem S_2: solve for flux V_Σ through each separator;
   - Still an $\mathcal{F}$-matrix in case of the Stokes equations;

Robustness and $\mathcal{O}(N \log N)$ with an ILU???

1. Orthogonal transformations:
   - Eliminate most V-P couplings to avoid fill;
   - 'Transfer operators' defining coarse problem S_2.
2. Coarse problem S_2: solve for flux V_Σ through each separator;
   - Still an $\mathcal{F}$-matrix in case of the Stokes equations;
3. Constraint preconditioning:
   - no approximations in 'Grad' or 'Div' part;
   - mass is conserved exactly throughout.

Robustness and $\mathcal{O}(N \log N)$ with an ILU???

1. Orthogonal transformations:
   - Eliminate most V-P couplings to avoid fill;
   - 'Transfer operators' defining coarse problem S_2.
2. Coarse problem S_2: solve for flux V_Σ through each separator;
   - Still an $\mathcal{F}$-matrix in case of the Stokes equations;
3. Constraint preconditioning:
   - no approximations in 'Grad' or 'Div' part;
   - mass is conserved exactly throughout.
4. Drop-by-position
   - original properties preserved (symmetry, positiveness);
   - singular subsystems cannot occur.

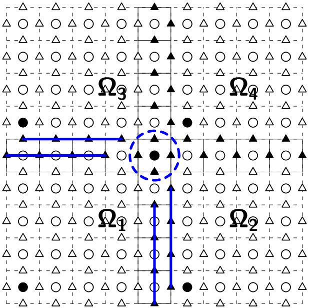
Robustness and $\mathcal{O}(N \log N)$ with an ILU???

1. Orthogonal transformations:
   - Eliminate most V-P couplings to avoid fill;
   - 'Transfer operators' defining coarse problem S_2.
2. Coarse problem S_2: solve for flux V_Σ through each separator;
   - Still an $\mathcal{F}$-matrix in case of the Stokes equations;
3. Constraint preconditioning:
   - no approximations in 'Grad' or 'Div' part;
   - mass is conserved exactly throughout.
4. Drop-by-position
   - original properties preserved (symmetry, positiveness);
   - singular subsystems cannot occur.
5. No segregation of variables:
   - velocity and pressure kept together;
   - no nested iterations.

Issue with Standard Cartesian Partitioning

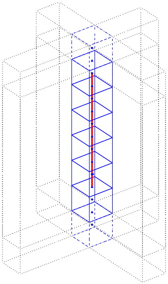
isolated pressure unknowns must be retained in Schur-Complement

2D



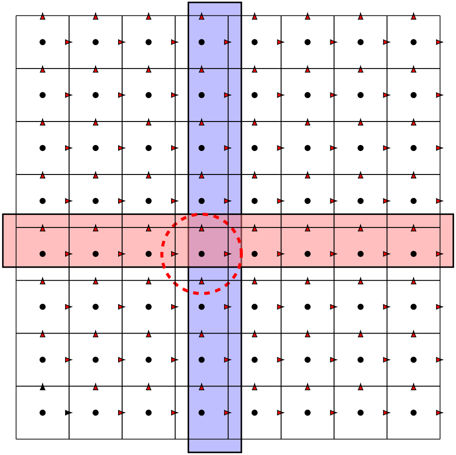
1 retained P-node per subdomain

3D

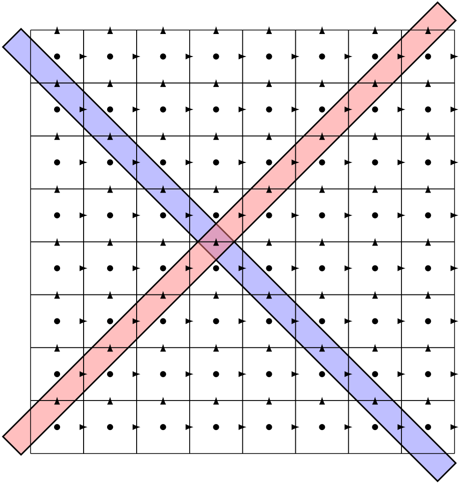


$\mathcal{O}(s)$ retained P-nodes for subdomain size s^3

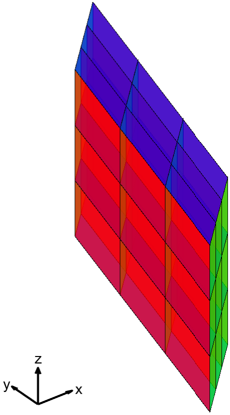
Solution in 2D: Skew Partitioning



Solution in 2D: Skew Partitioning



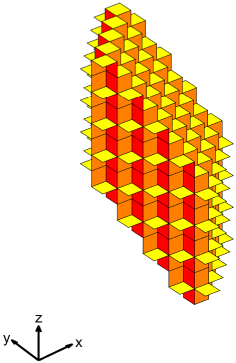
Skew Partitioning in 3D – the Parallelepipedal Subdomain



German: Spat

BSc. thesis of Mark v.d. Klok, implemented in HYMLS by Sven Baars

Skew Partitioning in 3D – the Parallelepipedal Subdomain

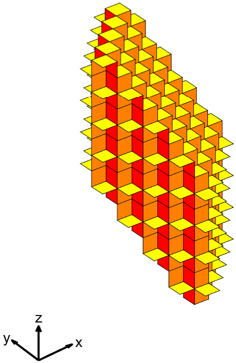


- space-filling single template

German: Spat

BSc. thesis of Mark v.d. Klok, implemented in HYMLS by Sven Baars

Skew Partitioning in 3D – the Parallelepipedal Subdomain

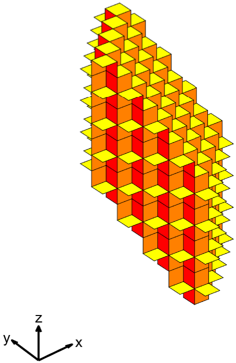


- space-filling single template
- stackable: suitable for multilevel

German: Spat

BSc. thesis of Mark v.d. Klok, implemented in HYMLS by Sven Baars

Skew Partitioning in 3D – the Parallelepipedal Subdomain

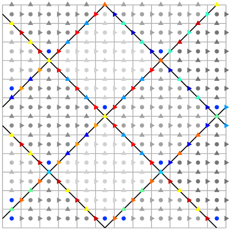


- space-filling single template
- stackable: suitable for multilevel
- structure-preserving: no isolated P-nodes

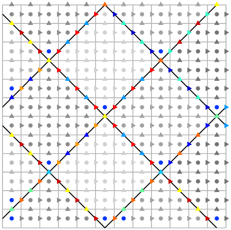
German: Spat

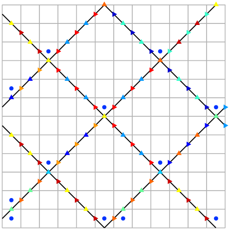
BSc. thesis of Mark v.d. Klok, implemented in HYMLS by Sven Baars

Summary: Staggered-grid Multi-Level ILU (SMILU)

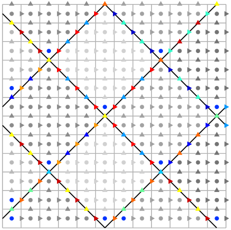


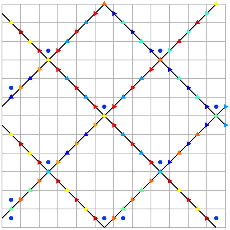
Summary: Staggered-grid Multi-Level ILU (SMILU)

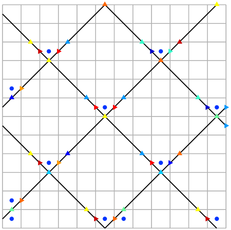




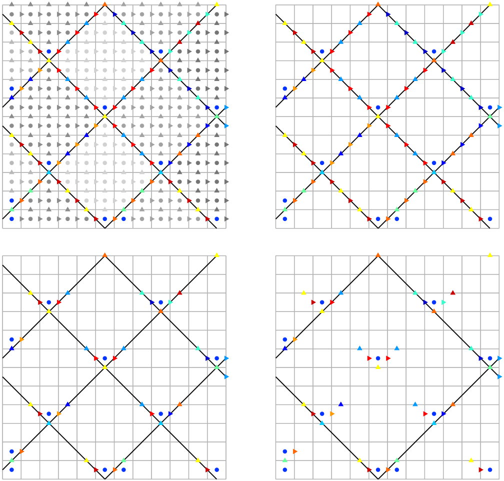
Summary: Staggered-grid Multi-Level ILU (SMILU)







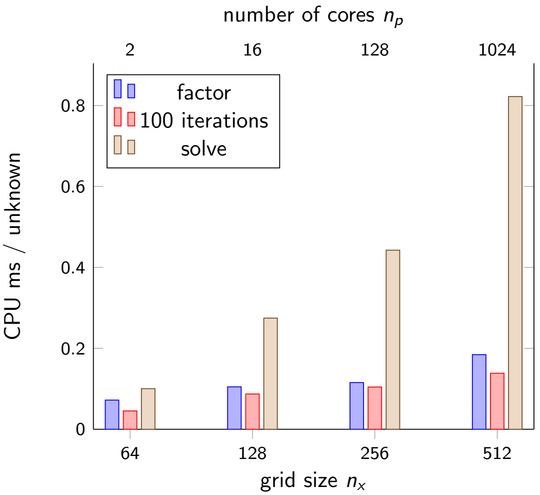
Summary: Staggered-grid Multi-Level ILU (SMILU)



A First Benchmark: 3D Stokes on Emmy @ RRZE

- Stokes-flow, cube-shaped domain
- 2×10 core Intel(R) IvyBridge cluster (approx. 500 nodes)
- GMRES(50)+SMILU
- 3–7 levels
- separator length $s = 8$ (subdomain size s^3 grid cells)
- coarsening factor $c = 2$ (8 subdomains combined)
- implementation: HYMLS (Trilinos/Epetra, flat MPI)

A First Benchmark: 3D Stokes on Emmy @ RRZE



Questions?

Contact

Jonas Thies

DLR Simulation and Software Technology
High Performance Computing

Jonas.Thies@DLR.de
Phone 02203 / 601 41 45
http://www.DLR.de/sc

Funding: DLR, NWO, DFG (SPPEXA/ESSEX)

Summary

- fully coupled preconditioner
- 'geometric' approach
- multi-level, $\mathcal{O}(N \log N)$

Next

- evaluate performance on bigger machines (e.g. SuperMUC)
- more complex flow problems (e.g. Rayleigh-Bénard)
- fully implicit ocean model