Distributed Schur Complement Solvers for Real and Complex Block-Structured CFD Problems

Dr.-Ing. Achim Basermann, Dr. Hans-Peter Kersken

German Aerospace Center (DLR)
Simulation- and Software Technology
Distributed Systems and Component Software
Porz-Wahnheide, Linder Höhe, D-51147 Cologne, Germany

achim.basermann@dlr.de
Locations and employees

**Germany:** 6000 employees across 29 research institutes and facilities at
- 13 sites.

Offices in **Brussels**, **Paris** and **Washington**.
Survey

- Motivation
- The *Distributed Schur Complement* method (DSC)
- Complex and real problem formulation
- Experiments with TRACE matrices
- Conclusions
Parallel Simulation System TRACE

- TRACE: Turbo-machinery Research Aerodynamic Computational Environment
- Developed by the Institute for Propulsion Technology of the German Aerospace Center (DLR-AT)
- Calculates internal turbo-machinery flows
- Finite volume method with block-structured grids
- The linearized TRACE modules require the parallel, iterative solution of large, sparse non-symmetric systems of linear equations.
Preconditioners for TRACE: Background

- Modules linearTRACE or adjointTRACE
  \[ Ax = b \]

- A non-symmetric, complex or real, sparse

- Parallel iterative solver: (F)GMRes with preconditioning
  \[ P^{-1} Ax = P^{-1} b \]

- Distinctly dominates the time behavior

- Matrix-vector and vector-vector operations

- Preconditioning usually is the most time-consuming operation

- Crucial for scalability

- **Status**: Block-local preconditioning
  - ILU, SSOR
  - Scalability limited

- **Goal**: global, scalable preconditioner
  - Experiments with Distributed Schur Complement (DSC) methods
DSC Method (1)

Distributed matrix, 2 processors

Processor 1

\[ A_1 \]

\[ X_1 \]

Local rows

Internal rows

Processor 2

\[ A_2 \]

\[ X_2 \]

External interface rows

Local interface rows
DSC Method (2)

DSC Algorithm

Schematic view on each processor
DSC Method (3)

Preconditioning within the DSC algorithm

Processor $i$

Local rows

Block incomplete LU for the local interface rows

Block incomplete LU for the local rows

$U_i$

$L_i$

$U_{i,s}$

$L_{i,s}$
DSC Method (4): Effect of Partitioning

Graph partitioning: ParMETIS (University of Minnesota)

Goal:
- Minimize the number of edges cut
- Minimize the number of interface unknowns

Undirected graph  →  Symmetrize the matrix structure
Matrix Experiments: Real or Complex Arithmetics?

Complex TRACE matrix
(n=28,120; nz=1,246,200; Cond.: $6.7 \cdot 10^6$)

\[ Ax = b \]
\[ \Leftrightarrow (C + iD)(y + iz) = c + id \]

Real TRACE matrix
(n=56,240; nz=2,572,040; Cond.: $8.4 \cdot 10^6$)

\[
\begin{pmatrix}
C & -D \\
D & C
\end{pmatrix}
\begin{pmatrix}
y \\
z
\end{pmatrix}
=
\begin{pmatrix}
c \\
d
\end{pmatrix}
\]
\[ \Leftrightarrow Gw = e \]
DSC Preconditioner: Matrix Permutation (complex)

**Background:** Fill-in reduction for ILUT preconditioning

- Original
- Minimum Degree (MD)
- Reverse Cuthill-McKee (RCM)
ILU Preconditioner: Fill-in in L and U (complex)

**MATLAB**: ILUT preconditioner; threshold = $10^{-3}$
ILU Preconditioner: Matrix Permutation (real)

**Background:** Fill-in reduction for ILUT preconditioning

Original  Minimum Degree (MD)  Reverse Cuthill-McKee (RCM)
ILU Preconditioner: Fill-in in $L$ and $U$ (real)

**MATLAB:** ILUT preconditioner; threshold $= 10^{-3}$
Performance: Complex or real Arithmetics?

**MATLAB:** ILUT preconditioning;
threshold $= 10^{-3}$; $|\text{rel. residual}| < 10^{-10}$
Performance on the AeroGrid Cluster of DLR
(Dual-processor nodes; Quad-Core Intel Harpertown; 2.83 GHz)

DSC method, real versus complex problem formulation
DSC Method: Performance (real)
(Dual-processor nodes; AMD Opteron 250; 2.4 GHz)

DSC method versus Block-Jacobi preconditioning (with RCM)

For a high processor count, the DSC method appears to pay off.
DSC Method: Strong Scaling (complex)
(Dual-processor nodes; Quad-Core Intel Harpertown; 2.83 GHz)

TRACE matrix THD
(n=378,400; nz=45,456,500; threshold = 10^{-3}; |rel. residual| < 10^{-5})

TRACE matrix UHBR
(n=4,497,520; nz=552,324,700; threshold = 10^{-3}; |rel. residual| < 10^{-10})
Conclusions

- Permutation (MD, RCM) crucial for ILUT performance; slight advantages for RCM (higher locality)

- Complex computations significantly faster than real ones (higher locality, better ratio of calculation to memory access)

- DSC method lets expect higher scalability than block-local methods.

Future work

- Development of an intelligent solver for TRACE with problem- and convergent-dependent parameter control and preconditioning

- Application of the DSC method as robust smoother in Multigrid methods
Questions?