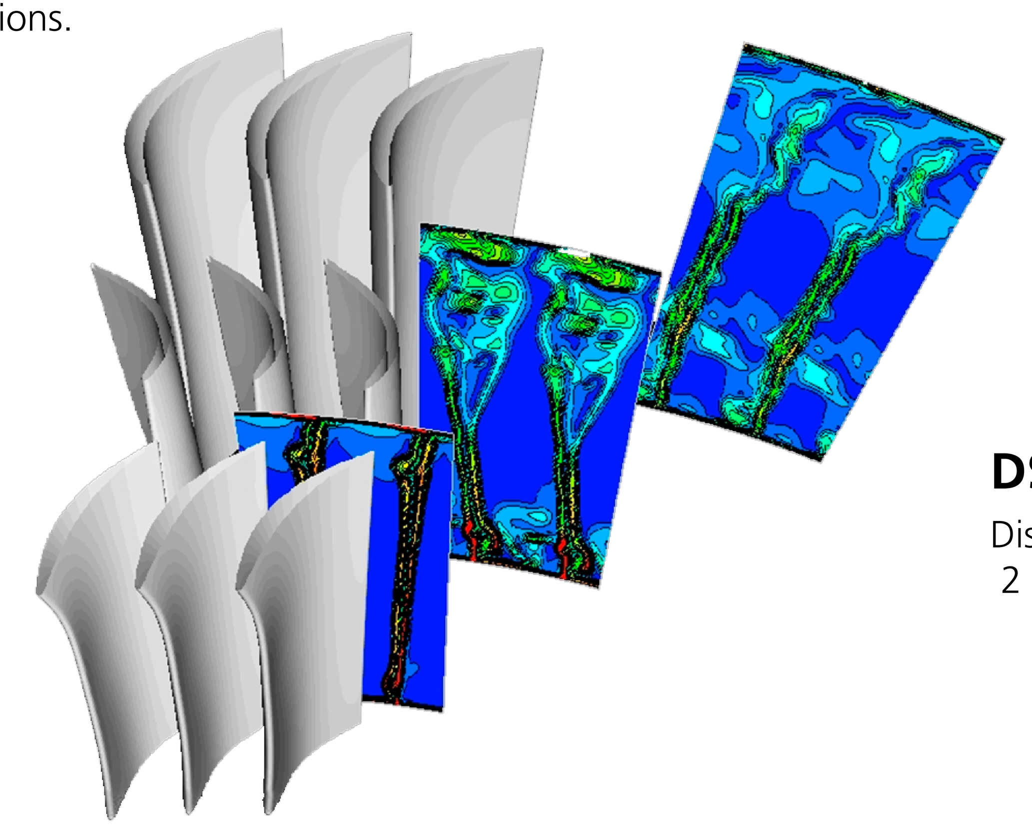
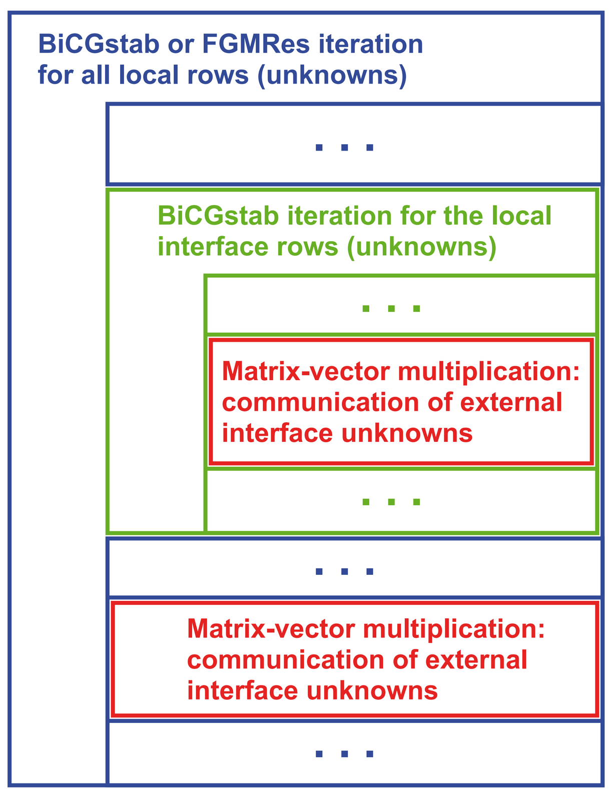
Parallel Iterative Solvers for Block-Structured CFD Problems

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.



DSC Method: Algorithm



Schematic view on each processor

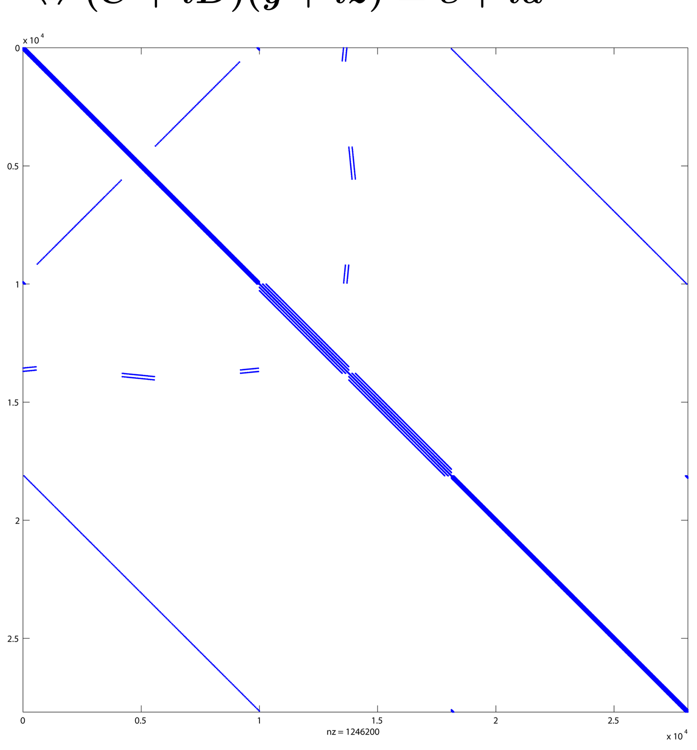
Typical *linearTRACE* Matrix Problem

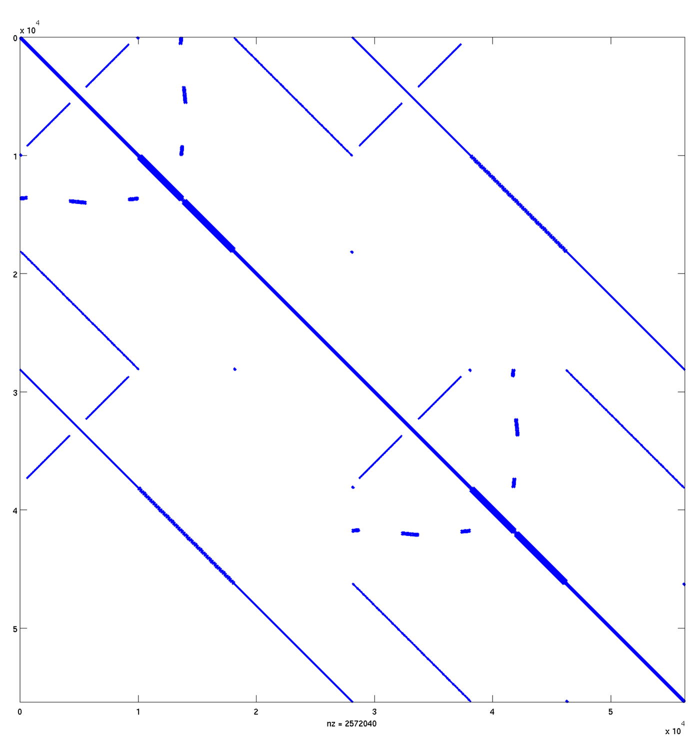
Complex TRACE matrix
n=28,120; nz=1,246,200; condition: 6.7·10⁶

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

Real TRACE matrix
n=56,240; nz=2,572,040; condition: 8.4·10⁶

$$\begin{pmatrix} C & -D \\ D & C \end{pmatrix} \begin{pmatrix} y \\ z \end{pmatrix} = \begin{pmatrix} c \\ d \end{pmatrix}$$
$$\Leftrightarrow Gw = e$$





Preconditioner for TRACE: Background

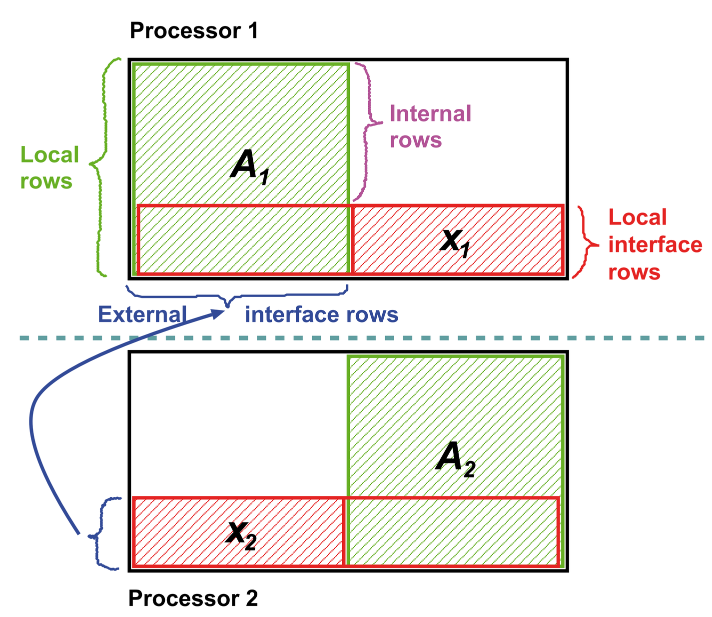
- Modules *linearTRACE* or *adjointTRACE* $Ax = b$
- A non-symmetric, complex or real, sparse
- Parallel iterative solver: (F)GMRes with preconditioning

$$P^{-1}Ax = P^{-1}b$$

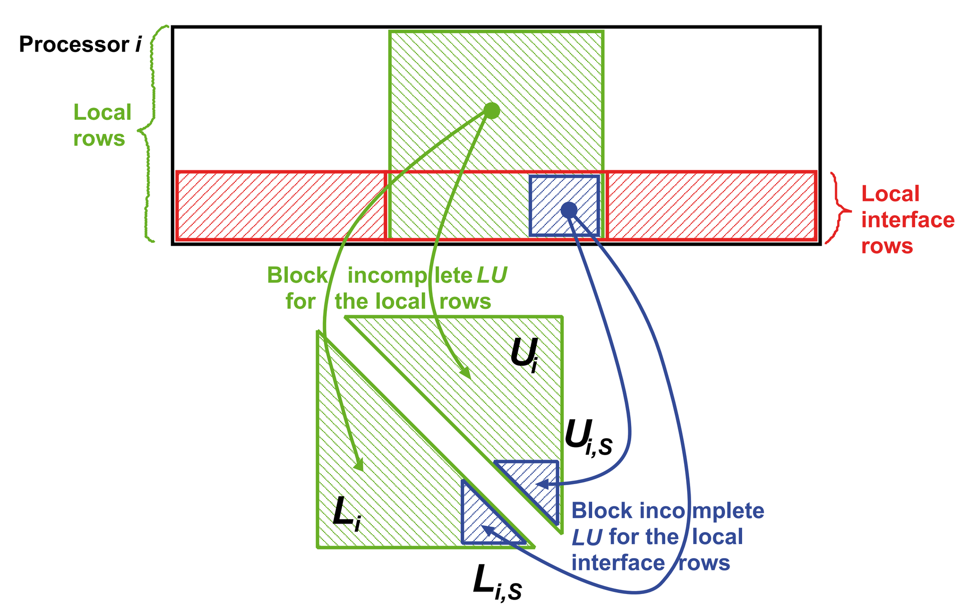
- Distinctly dominates the time behaviour
- Matrix-vector and vector-vector operations
- Preconditioning usually is the most time-consuming operation
  - Crucial for scalability
  - Status: block-local preconditioning
    - ILU, SSOR
    - Scalability limited
  - Goal: global, scalable preconditioner
    - Experiments with Distributed Schur Complement (DSC) methods

DSC Method: Definitions

Distributed Matrix, 2 processors

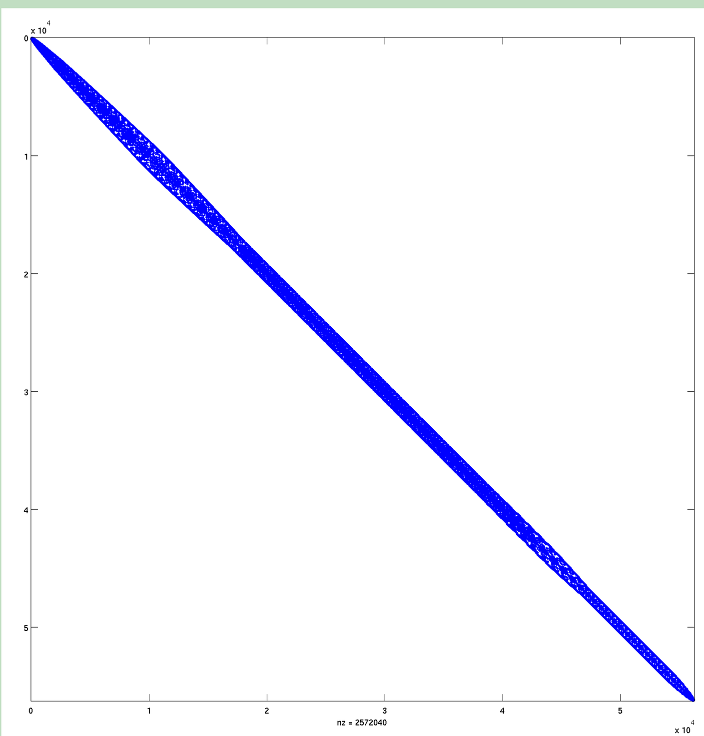


DSC Method: Incomplete LU Factorizations



Matrix Permutation for Bandwidth Reduction

After Reverse Cuthill-McKee (RCM)



Results of the Performance Analysis for TRACE Matrix Problems

- Block Jacobi preconditioning performs well for small processor counts.
- The DSC method pays off for higher processor counts.

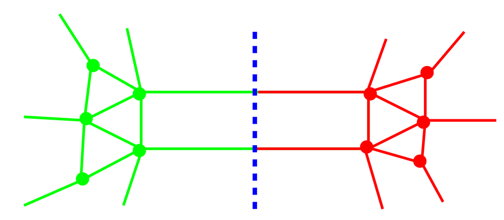
- Potential method of choice: intelligent solver with
  - problem and convergence dependent parameter control;
  - problem and convergence dependent preconditioning;
  - preconditioning dependent on the processor count.

DSC Method and Partitioning

Graph partitioning: *ParMETIS* (University of Minnesota)

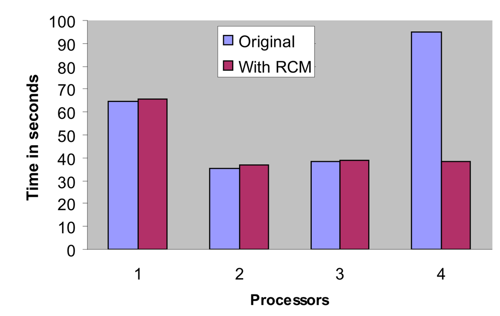
Goal:

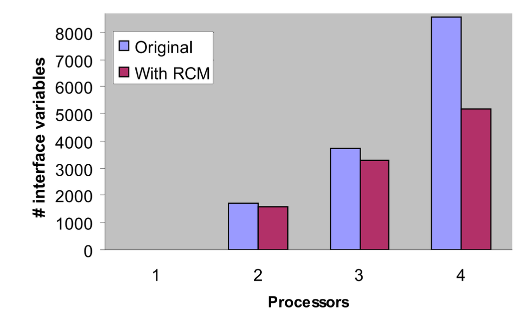
Minimize the number of edges cut ⟷ number of interface unknowns



Performance Tests on an Quad-Core Intel Xeon CPU L5420

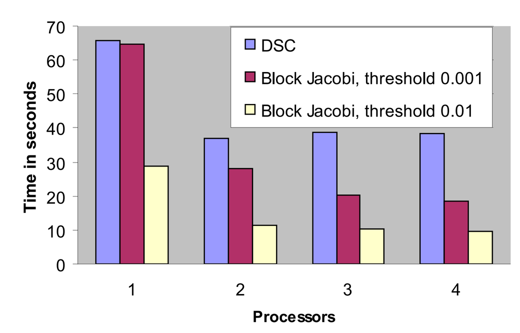
Comparison: DSC method for original and RCM permuted matrix

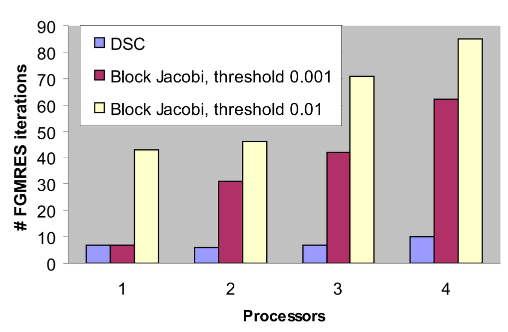




Number of interface variables is significantly lower with RCM.

Comparison: DSC Method versus Block Jacobi Preconditioning (with RCM)



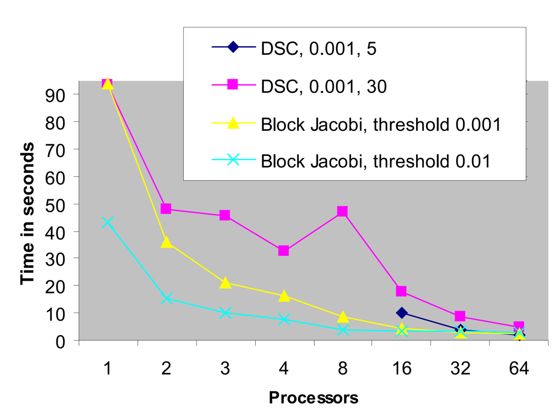


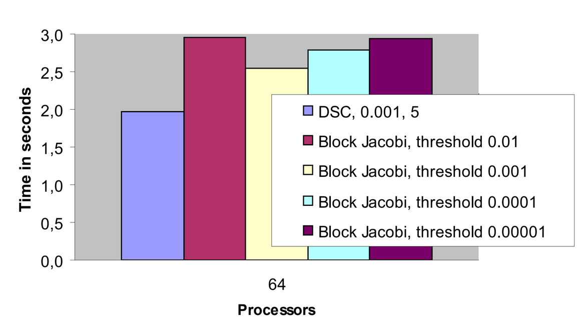
Number of iterations is stable for DSC, but Block Jacobi is faster.

Performance on a Cluster at DLR-AT

(AMD Opteron Processor 250; Dual-Processor Nodes; 2.4 GHz)

Comparison: DSC method versus Block Jacobi preconditioning (with RCM)





For a high processor count, the DSC method appears to pay off.

German Aerospace Center (DLR)

Simulation and Software Technology
Distributed Systems and Component Software

Linder Höhe, D-51147 Köln, Germany

Dr.-Ing. Achim Basermann
Phone: +49 2203 601-3326
Telefax: +49 2203 601-3070
Achim.Basermann@dlr.de

www.DLR.de/sc

Dr. Hans-Peter Kersken
Phone: +49 2203 601-2803
Telefax: +49 2203 601-3070
Hans-Peter.Kersken@dlr.de

www.DLR.de/sc

HICFD - Hocheffiziente Implementierung von CFD-Codes für HPC-Many-Core-Architekturen (BMBF-Projekt im Rahmen des „IKT 2020 - Forschung und Innovation")

www.hicfd.de

GEFÖRDERT VOM

Bundesministerium für Bildung und Forschung

DLR
Deutsches Zentrum für Luft- und Raumfahrt e.V.
in der Helmholtz-Gemeinschaft

Simulation and Software Technology