PDSLin: Parallel Domain decomposition Schur complement based Linear solver

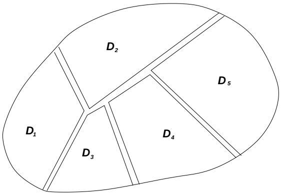
Ichitaro Yamazaki and Xiaoye Sherry Li
Lawrence Berkeley National Laboratory

October 31, 2010

Schur complement method for solving $Ax = b$, where A is a sparse matrix.

divide-and-conquer approach:

- devide entire problem into smaller non-overlapping subdomain problems.
- solve the subdomain problems to form interface problem (Schur complement).
- solve the interface problem.



"**Hybrid**" solver with various options (e.g., direct or iterative solver) for solving subdomain and interface problems.

Schur complement method:

1. extract subdomains using a parallel graph partitioning algorithm.
2. reorder A so that all the subdomains come before the interfaces:

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A22 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix},$$

   where A_{11} represents subdomains, A_{22} consists of separators, and A_{12} and A_{21} are the interfaces between A_{11} and A_{22}.

3. solve the subdomain problems to form

$$\begin{pmatrix} A_{11} & A_{12} \\ & S \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ \widehat{b}_2 \end{pmatrix},$$

   where $\widehat{b}_2 = b_2 - A_{21}A_{11}^{-1}b_1$ and S is the Schur complement given by

$$S = A_{22} - A_{21}A_{11}^{-1}A_{12}.$$

Schur complement method (continued):

4. solve the Schur complement system:

$$Sx_2 = \widehat{b}_2.$$

5. solve the subdomains system:

$$A_{11}x_1 = b_1 - A_{12}x_2.$$

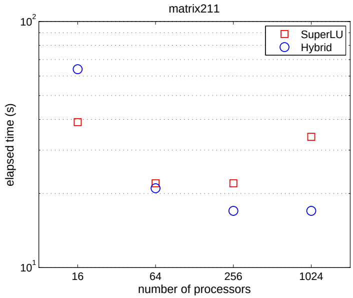
Remarks

- Most fill-ins occur in S, and a preconditioned iterative method is effective for solving the Schur complement system.
- Each subdomain system can be solved independently (i.e., A_{11} is block diagonal).
- There are several options to solve each subdomain system (e.g., SuperLU, SuperLU_DIST, preconditioned iterative or hybrid solver).

PDSLin software

- solves real or complex general problems with multiple RHSs.
- implemented in C and MPI, with Fortran90 interface.
- detailed Users' Guide.
- requires following external packages:
  - PT-Scotch (`www.labri.fr/perso/pelegrin/scotch/`) or ParMetis (`glaros.dtc.umn.edu/gkhome/metis/parmetis/overview`)
  - SuperLU_DIST (`crd.lbl.gov/~xiaoye/SuperLU/`, version 2.4 or above)
  - PETSc (`www.mcs.anl.gov/petsc/petsc-as/`, optional, version 2.3.3)
- contact: Ichitaro Yamazaki (ic.yamazaki@gmail.com), Xiaoye Li (xsli@lbl.gov), or Esmond Ng (egng@lbl.gov).

Parallel performance with Fusion MHD equations

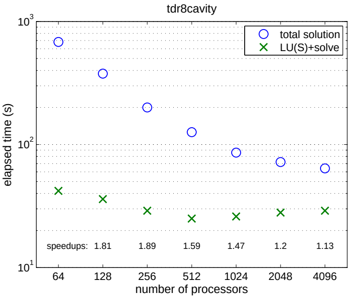


experimental setups:

- PT-SCOTCH to extract 8 subdomains of size $\approx 99k$ (parallel nested dissection)
- SuperLU_DIST to factor each subdomain.
- SuperLU_DIST to compute ILU precond., i.e., LU($\widetilde{S}$) with $\widetilde{S} \approx S$ of size $\approx 13k$, using 64 processors
- BiCGStab of PETSc to solve $Sy = c$ with $\frac{\|A\hat{x}-b\|_2}{\|b\|_2} < 10^{-12}$ (converged in ~ 10 iterations)

- M3D-C1 to model a fusion device (provided by Steve Jardin at PPPL, CEMM SciDAC).
- dimension: $801,378$ (real unsymmetric/indefinite).
- use two cores per node of Cray-XT4 (Franklin) at NERSC.

Parallel performance with Accelerator Maxwell equations



experimental setups:

- PT-SCOTCH to extract 64 subdomains of size $\approx 277k$ (nested dissection)
- SuperLU_DIST to factor each subdomain
- SuperLU_DIST to compute ILU precond., i.e., LU($\widetilde{S}$) with $\widetilde{S} \approx S$ of size $\approx 57k$, using 64 processors
- BiCGStab of PETSc to solve $Sy = c$ with $\frac{\|A\hat{x}-b\|_2}{\|b\|_2} < 10^{-12}$ (converged in ~ 10 iterations)

- Omega3P to design an ILC cavity (provided by Lie-Quan Lee at SLAC, ComPASS SciDAC).
- dimension: $17,799,228$ (real symmetric/highly indefinite).
- use one core per node of Cray-XT4 (Franklin) at NERSC.