INVENTEURS DU MONDE NUMÉRIQUE

C2S@Exa

September, Paris

Parallel Hierarchical Hybrid Algebraic Linear Solvers : current and future

HiePACS Inria Project
Joint Inria-CERFACS lab
INRIA Bordeaux Sud-Ouest

Motivations

$$\mathcal{A}x = b$$

The “spectrum” of linear algebra solvers

Direct

- ★ Robust/accurate for general problems
- ★ BLAS-3 based implementations

- ★ Memory/CPU prohibitive for large $3D$ problems
- ★ Limited parallel scalability

Iterative

- ★ Problem dependent efficiency/controlled accuracy
- ★ Only mat-vect required, fine grain computation

- ★ Less memory computation, possible trade-off with CPU
- ★ Attractive “build-in” parallel features

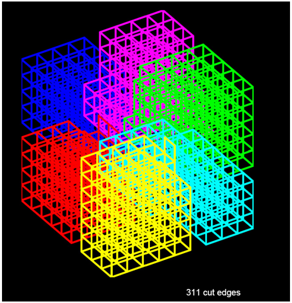
Goal: design Hybrid Linear Solvers

Develop robust scalable parallel hybrid direct/iterative linear solvers

- ★ Exploit the efficiency and robustness of the sparse direct solvers
- ★ Develop robust parallel preconditioners for iterative solvers
- ★ Take advantage of the natural scalable parallel implementation of iterative solvers

Domain Decomposition (DD)

- ★ Natural approach for PDE's
- ★ Extend to general sparse matrices
- ★ Partition the problem into subdomains, subgraphs
- ★ Use a direct solver on the subdomains
- ★ Robust preconditioned iterative solver



Two Hybrid Linear Solvers

Algebraic non-overlapping domain decomposition

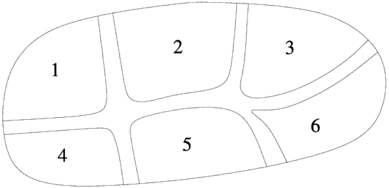
- ★ Partitioning of the adjacency graph of the sparse matrix
- ★ Perform a partial Gaussian elimination (sparse direct solution on the internal variables)
- ★ Solve the Schur complement system using a preconditioned Krylov subspace method
- ★ Backsolve for the internal variables

Two parallel implementations

- ★ HIPS (Hierarchical Iterative Parallel Solver)
  Incomplete LU factorization of the Schur complement based on a hierarchical interface decomposition ordering
- ★ MaPHyS (Massively Parallel Hybrid Solver)
  Algebraic additive Schwarz preconditioner for the Schur complement

Based on a non-overlapping domain decomposition

$$\begin{pmatrix} A_B & F \\ E & A_C \end{pmatrix}$$



B : Interior nodes of subdomains (direct factorization).

C : Interface nodes.

Special decomposition and ordering of the subset C :

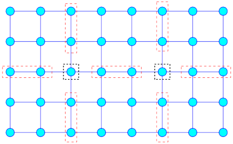
Goal : Building a global Schur complement preconditioner (ILU) from the local domain matrices only.

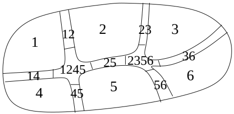
HIPS: domain interface based fill-in policy

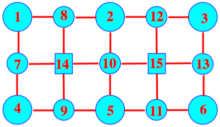
[P.Hénon, Y. Saad - SIAM SISC 06] [J.Gaidamour, P.Hénon -IEEE CSE 08]

Special decomposition and ordering of the subset C :

Hierachical interface decomposition into connectors :







Rules :

- ⋆ No creation of edge (fill-in) outside the local domain matrices.
- ⋆ Allow edges between connectors adjacent to the same subdomain.

⇒ keep the parallelism (communication only between adjacent subdomains).

HIPS: main features

Robust block incomplete factorization of the Schur complement

- Hierachy of separators (wirebasket like - faces , edges, vertices)
- Block incomplete factorization with “geometrical” fill-in policy to express parallelism
  (Global factorization using only local sub-domain matrices)
- MIS ordering to express parallelism within incomplete factorisation steps

HIPS: numerical and parallel features

- Memory control via treshold parameters (interior, Schur approximation and incomplete factorization)
- Parallel load balancing: mapping of several subdomains on each MPI process

MaPHyS: Algebraic Additive Schwarz preconditioner [L.Carvalho, L.G., G.Meurant- NLAA - 01]

$$\mathcal{S} = \sum_{i=1}^{N} \mathcal{R}_{\Gamma_i}^T \mathcal{S}^{(i)} \mathcal{R}_{\Gamma_i}$$

$$\mathcal{S} = \begin{pmatrix} \ddots & & & & \\ & \mathcal{S}_{kk} & \mathcal{S}_{k\ell} & & \\ & \mathcal{S}_{\ell k} & \mathcal{S}_{\ell\ell} & \mathcal{S}_{\ell m} & \\ & & \mathcal{S}_{m\ell} & \mathcal{S}_{mm} & \mathcal{S}_{mn} \\ & & & \mathcal{S}_{nm} & \mathcal{S}_{nn} \end{pmatrix} \Longrightarrow \mathcal{M} = \begin{pmatrix} \ddots & & & & \\ & \begin{bmatrix} \mathcal{S}_{kk} & \mathcal{S}_{k\ell} \\ \mathcal{S}_{\ell k} & \mathcal{S}_{\ell\ell} \end{bmatrix}^{-1} & & & \\ & & \begin{bmatrix} \mathcal{S}_{\ell\ell} & \mathcal{S}_{\ell m} \\ \mathcal{S}_{m\ell} & \mathcal{S}_{mm} \end{bmatrix}^{-1} & & \\ & & & \begin{bmatrix} \mathcal{S}_{mm} & \mathcal{S}_{mn} \\ \mathcal{S}_{nm} & \mathcal{S}_{nn} \end{bmatrix} & \end{pmatrix}$$

$$\mathcal{M} = \sum_{i=1}^{N} \mathcal{R}_{\Gamma_i}^T (\bar{\mathcal{S}}^{(i)})^{-1} \mathcal{R}_{\Gamma_i}$$

Similarity with Neumann-Neumann preconditioner [J.F Bourgat, R. Glowinski, P. Le Tallec and M. Vidrascu - 89] [Y.H. de Roek, P. Le Tallec and M. Vidrascu - 91]

where $\bar{\mathcal{S}}^{(i)}$ is obtained from $\mathcal{S}^{(i)}$

$$\underbrace{\mathcal{S}^{(i)} = \begin{pmatrix} \mathcal{S}_{kk}^{(\iota)} & \mathcal{S}_{k\ell} \\ \mathcal{S}_{\ell k} & \mathcal{S}_{\ell\ell}^{(\iota)} \end{pmatrix}}_{\text{local Schur}} \Longrightarrow \underbrace{\bar{\mathcal{S}}^{(i)} = \begin{pmatrix} \mathcal{S}_{kk} & \mathcal{S}_{k\ell} \\ \mathcal{S}_{\ell k} & \mathcal{S}_{\ell\ell} \end{pmatrix}}_{\text{local assembled Schur}}$$

$$\searrow \quad \sum_{\iota \in adj} \mathcal{S}_{\ell\ell}^{(\iota)} \quad \nearrow$$

MaPHyS: main future features

Ongoing/future software development

- ★ Improved software flexibility (rely on MUMPS, PastiX, Scotch, ...)
- ★ Hybrid MPI-Thread implementation (PhD thesis - DIP Inria/Total) on top of PaStiX
- ★ Implementation on top of runtime systems - cf Emmanuel's talk

Ongoing/future numerical development

- ★ Improved numerical robustness - hierarchical toward global preconditioning (ADT Maphys@Exa via C2S@Exa)
- ★ Coarse space mechanisms through augmentation and/or deflation
- ★ New Krylov subspace methods (block variants, hiden/avoiding communicaitons, ...) - shared with sparse directs (cf Emmanuel's talk)