From classical to optimized Schwarz

by
Amik St-Cyr
National Center for Atmospheric Research
CISL/SCD/CSS

Plan of presentation

Part 1:
- Motivation of DDM
- Partitioning algorithms
- Classical Schwarz algorithm
- Matrix/discrete level
- Convergence
- Two level approach

Part 2:
The Robin method
Fourier analysis of Classical Schwarz
Fourier analysis for optimized Schwarz
Optimization over all Fourier modes
Examples FDM
High-Order methods (HOMs)
Optimized Schwarz in a massively parallel GCM
Conclusion
Part 1:
- Motivation of DDM
- Partitioning algorithms
- Classical Schwarz algorithm
- Matrix/discrete level
- Convergence
- Two level approach

DDM Motivation
- The global problem cannot fit into main memory, out of core computations: very slow swapping to disk
- (AND | OR) Concurrency can be exploited to solve the global problem: solving problem faster on parallel computers
- (AND | OR) The solution of the subproblems is “easier” than the global problem: direct methods on smaller subproblems cache friendly

Domain Decomposition
- Divide and conquer applied to PDEs:
- **Decompose** domain into many sub-domains
- **Solve** independently each smaller problem
- **Glue** the solutions together: convergence?

\[ \Omega \]
\[ Lu = f \]
Mesh partitioning: decompose the domain

- Geometric Based Algorithms
- Coordinate bisection
- Inertia bisection
- Graph Theory Based Algorithms
  - Graph bisection
  - Greedy algorithm
  - Spectral bisection
  - K-L algorithm
- Other Partitioning Algorithms
  - Global optimization algorithms
  - Reducing the bandwidth of the matrix
  - Index based algorithms
- The State of the Art
  - Hybrid approach
  - Multilevel approach
  - Parallel partitioning algorithms

Example: spectral bisection

\[
Lx = \begin{bmatrix}
-1 & -1 & 4 & -1 & -1 \\
-1 & -1 & 4 & -1 & -1 \\
-1 & -1 & 4 & -1 & -1 \\
-1 & -1 & 4 & -1 & -1 \\
-1 & -1 & 4 & -1 & -1 \\
\end{bmatrix} \quad x = \lambda x
\]

- Needs to be an eigenvector of Laplacian
- If composed of half +1 and half -1 it satisfies the two constraints
- Finding the Fielder vector: Lanczos algorithm
- Proceed recursively...

Practical DDM

Each part of problem solved on a compute node:

Partitioned meshes

Examples from Computational Fluid Dynamics: ParMetis

Courtesy of McGill’s CFD laboratory
Mesh partitioning

- Represents only the technical part of DDM
- Has deep ties with parallel computing: MIMD
- DDM denotes also the development of special algorithms to solve decomposed problems
- Algorithms: Schwarz, FETI, sub-structuring ...

Domain Decomposition

- Divide and conquer applied to PDEs
- Decompose domains into many sub-domains
- Solve independently each smaller problem
- Glue the solutions together: convergence?

Basic DD methods

(Overlapping) Schwarz (1870): existence of elliptic problems on non trivial domains

(Non-overlapping) Schur / sub-structuring methods

Kron (53) Przemieniecki (63)

2 classes of methods: overlapping and non-overlapping

Classical Schwarz

Suppose we need to solve:

\[ Lu = f \text{ in } \Omega, \quad Bu = g \text{ on } \partial \Omega \]

Partition the original domain into 2 domains:

\[
\begin{align*}
L u_1^{n+1} &= f \text{ in } \Omega_1, \\
L u_2^{n+1} &= f \text{ in } \Omega_2, \\
B(u_1^{n+1}) &= g \text{ on } \partial \Omega_1, \\
B(u_2^{n+1}) &= g \text{ on } \partial \Omega_2, \\
 u_1^{n+1} &= u_2^n \text{ on } \Gamma_{12}, \\
 u_2^{n+1} &= u_1^n \text{ on } \Gamma_{21}.
\end{align*}
\]
\[ \Delta u = 0, \text{ on } [-1, 1] \text{ with } u(-1) = u(1) = 0 \]
\[ \Delta u = 0, \text{ on } [-1, 1] \text{ with } u(-1) = u(1) = 0 \]
\[ \Delta u = 0, \text{ on } [-1, 1] \text{ with } u(-1) = u(1) = 0 \]
\( \Delta u = 0, \) on \([-1, 1]\) with \(u(-1) = u(1) = 0\)
\[ \Delta u = 0, \text{ on } [-1, 1] \text{ with } u(-1) = u(1) = 0 \]

Schwarz no overlap

Matrix formulation

Continuous problem:
\[
\begin{align*}
\mathcal{L} u_1^{n+1} &= f \text{ in } \Omega_1, \\
B(u_1^{n+1}) &= g \text{ on } \partial \Omega_1, \\
A_1 u_1^{n+1} &= f_1 + B_{21} u_2^n
\end{align*}
\]
\[
\begin{align*}
\mathcal{L} u_2^{n+1} &= f \text{ in } \Omega_2, \\
B(u_2^{n+1}) &= g \text{ on } \partial \Omega_2, \\
A_2 u_2^{n+1} &= f_2 + B_{12} u_1^n
\end{align*}
\]

Partition of unity: \( e = \bar{R}_1^T e_1 + \bar{R}_2^T e_2 \rightarrow u_1^{n+1} = \bar{R}_1^T u_1^{n+1} + \bar{R}_2^T u_2^{n+1} \)

Restriction operator:
\[ u_k^n = R_k u \]
Matrix formulation

If consistent then: \( A_1 R_1 - B_2 R_2 = R_1 A, \quad A_2 R_2 - B_1 R_1 = R_2 A. \)

Leading to:

\[
\tilde{u}^{n+1} = u^n + \sum_{i=1}^{2} \tilde{R}_i A_i^{-1} R_i (f - A u^n)
\]

RAS: Restricted Additive Schwarz
- Nonsymmetric
- Default option in PETSC
- Cai and Sarkis (1997)
- Equivalent to continuous

AS: Additive Schwarz
- Symmetric
- No continuous equivalent (EG02)
- Use with Krylov accelerator
- Nepomnyaschikh (86)

Preconditioning in Krylov methods:

\[
M_{RAS}^{-1} = \sum_{i=1}^{K} \tilde{R}_i A_i^{-1} R_i \quad M_{AS}^{-1} = \sum_{i=1}^{K} R_i^T A_i^{-1} R_i
\]

- In practice the restriction and extension are not created
- Matrix Problem can be reformulated: lower operation counts

Convergence theory

For symmetric positive definite matrices
- No results for Restricted Additive Schwarz
- Convergence rate not optimal
- Convergence rate not scalable

Developed by Lions, Dryja, Widlund, BRamble, Pasciak, Wang, Xu, Zhang etc ...

Subdomain diameter: \( H = \max_{1 \leq i \leq K} \text{diam}(\Omega_i) \)

Mesh size: \( h \)

Overlap size: \( \beta H, \beta \in (0, 1] \)

Convergence of PCG: \( ||u^{(k)} - u^*|| \leq 2 \gamma^k ||u^{(0)} - u^*|| \)

where \( \gamma = \frac{\sqrt{\kappa(M^{-1}A) - 1}}{\sqrt{\kappa(M^{-1}A) + 1}} \)
**Convergence additive Schwarz**

\[ \kappa(M^{-1}_{AS}A) \leq CH^{-2}(1 + \beta^{-1}) \]

- Diameter tends to zero as the number of subdomain increases
- The overlap size does not remove the diameter problem
- Estimate worsen when A has varying coefficients: \( \nabla \cdot (a(x) \nabla u) \)
- Diameter dependence prevents algorithmic scalability
- Schwarz didn’t care about the scalability!

**Scalable/Optimal DDM algorithm**

- A DDM is scalable if its rate of convergence does not deteriorate when the number of subdomains grows.
- A DDM for the solution of a linear system is optimal if its rate of convergence to the exact solution is independent of the size of the system.

The first definition involves \( H \)

The second involves \( h \)

**Practical scalability**

\[ \Omega \]

\[ Lu = f \]

If scalable the solution is reached 4 times faster!

If additive Schwarz is used it takes the half time to solve!!

**Two level methods**

- Add a very coarse problem solved on the entire domain
- Removes completely the subdomain diameter problem
- Not easy to parallelize! (Duplication of coarse solves)

\[ M^{-1}_{AS,2} = R_H^T A_H^{-1} R_H + \sum_{i=1}^{K} R_i^T A_i^{-1} R_i = \sum_{i=0}^{K} R_i^T A_i^{-1} R_i \]

Condition number: \[ \kappa(M^{-1}_{AS,2}A) \leq C(1 + \beta^{-1}) \]

Varying coefficients: \[ C(\beta)(1 + \log(H/h), C(\beta)(H/h) \]
Two level methods

- Still not perfect since overlap must be kept constant!
- The perfect method would have zero overlap and a condition number independent of $H$ and $h$
- Is it possible to construct such a Schwarz method ??
- If not how close can we get?

Part 2:

- The Robin method
- Fourier analysis of Classical Schwarz
- Fourier analysis for optimized Schwarz
- Optimization over all Fourier modes
- Examples FDM
- High-Order methods (HOMs)
- Optimized Schwarz in a massively parallel GCM
- Conclusion

DDM sources

Domain Decomposition
Barry Smith, Nathan Orevad, and William Gropp