Multilevel Solvers

Performance shouldn’t degrade with increased problem size

Stationary iterative methods:
  • Norm of $I - B^{-1}A$ must be bounded uniformly below one

Preconditioned Krylov methods,
  • Condition number, $\kappa(B^{-\frac{1}{2}}AB^{-\frac{1}{2}})$, must be uniformly bounded

Multilevel techniques achieve this uniformity by exploiting multiscale structure
Block Factorization

Partition

\[
Ax = \begin{bmatrix}
A_{ff} & -A_{fc} \\
-A_{cf} & A_{cc}
\end{bmatrix}
\begin{bmatrix}
x_f \\
x_c
\end{bmatrix} = \begin{bmatrix}
b_f \\
b_c
\end{bmatrix} = b,
\]

then block factor,

\[
A = \begin{bmatrix}
I & 0 \\
-A_{cf}A_{ff}^{-1} & I
\end{bmatrix}
\begin{bmatrix}
A_{ff} & 0 \\
0 & \hat{A}_{cc}
\end{bmatrix}
\begin{bmatrix}
I & -A_{ff}^{-1}A_{fc} \\
0 & I
\end{bmatrix},
\]

where \( \hat{A}_{cc} = A_{cc} - A_{cf}A_{ff}^{-1}A_{fc} \).
Approximate $A_{ff}$ by its ILUT factors, $A_{ff} \approx LU$. Preconditioner is

$$B = \begin{bmatrix}
I & 0 \\
-A_{cf}U^{-1}L^{-1} & I
\end{bmatrix}
\begin{bmatrix}
LU & 0 \\
0 & S
\end{bmatrix}
\begin{bmatrix}
I & -U^{-1}L^{-1}A_{fc} \\
0 & I
\end{bmatrix},$$

where $S \approx A_{cc} - A_{cf}U^{-1}L^{-1}A_{fc}$. Coarse-grid problems

- computed using techniques akin to ILUT
- solved recursively

Multigrid Components

- Relaxation

Relax: \( x^{(1)} = x^{(0)} + D^{-1}r^{(0)} \)

- Use a smoothing process (such as Jacobi or Gauss-Seidel) to eliminate oscillatory errors
- Remaining error satisfies \( Ae^{(1)} = r^{(1)} = b - Ax^{(1)} \)
Multigrid

Multigrid Components

- Relaxation
- Restriction
- Transfer residual to coarse grid
- Compute $P^T r^{(1)}$

Relax: $x^{(1)} = x^{(0)} + D^{-1} r^{(0)}$

Restriction
Multigrid

Multigrid Components

- Relaxation
- Restriction
- Coarse-Grid Correction

Relax: \( x^{(1)} = x^{(0)} + D^{-1}r^{(0)} \)

Restriction

Solve: \( P^T A P x_c = P^T r^{(1)} \)

- Use coarse-grid correction to eliminate smooth errors
- Best correction, \( x_c \), in terms of \( A \)-norm satisfies

\[ P^T A P x_c = P^T r^{(1)} \]
Multigrid

Multigrid Components

- Relaxation
- Restriction
- Coarse-Grid Correction
- Interpolation

- Transfer correction to fine grid
- Compute $x^{(2)} = x^{(1)} + Px_c$

Relax: $x^{(1)} = x^{(0)} + D^{-1}r^{(0)}$

Solve: $P^TAPx_c = P^Tr^{(1)}$

Restriction Interpolation
Multigrid

Multigrid Components

- Relaxation
- Restriction
- Coarse-Grid Correction
- Interpolation
- Relaxation
- Relax once again to remove oscillatory error introduced in coarse-grid correction

Relax: \( x^{(1)} = x^{(0)} + D^{-1}r^{(0)} \)

Restriction

Interpolation

Solve: \( P^T A P x_c = P^T r^{(1)} \)

A Greedy Strategy for Coarse-Grid Selection - p.5
Multigrid Components

- Relaxation
- Restriction
- Coarse-Grid Correction
- Interpolation
- Relaxation

Relax: $x^{(1)} = x^{(0)} + D^{-1}r^{(0)}$

Restriction Interpolation

Solve: $P^T A P x_c = P^T r^{(1)}$

Direct solution of coarse-grid problem isn’t practical

Recursion!

Apply same methodology to solve coarse-grid problem
Algebraic Multigrid (AMG)

- Goal of coarsening is to complement fixed relaxation
- Variational formulation
  - Coarse-grid correction is optimal in $A$-norm
  - Algebraically smooth error must be in range of interpolation
- Choose coarse-grid, $C$, and interpolation, $P$,
  - using only algebraic information
  - with knowledge of (assumed) algebraically smooth errors

Partitioning

Choice of partition in

**ARMS:** affects sparsity in ILU
controls size of Schur complement

**AMG:** influences sparsity in $P$
determines size of CGO

Good partitioning

- adequately reduces dimension of coarse-scale problem
- allows sparse choices of $P$ or LU without sacrificing accuracy
- enables recursive solve for coarse-scale problem

Goal of partitioning is to enable efficient resolution of coarse-scale errors
Two-level Theory

- **Goal** is to use theory to inform algorithmic choices
- Solution on a given level depends only on quality of solution on next coarser level
- Multilevel theory can be intricate

Partition

\[
A x = \begin{bmatrix}
A_{ff} & -A_{fc} \\
-A_{cf} & A_{cc}
\end{bmatrix}
\begin{pmatrix}
x_f \\
x_c
\end{pmatrix} = \begin{pmatrix}
b_f \\
b_c
\end{pmatrix} = b
\]

Use two-level analysis to make choices within a multilevel scheme.
ARMS Analysis

Let

- \( B = \begin{bmatrix} I & 0 \\ -A_{cf} D_{ff}^{-1} I & I \\ D_{ff} & 0 \\ 0 & S \end{bmatrix} \begin{bmatrix} I \ 0 \\ -D_{ff}^{-1} A_{fc} \ 0 \\ 0 \ 1 \end{bmatrix} \)
- \( [D_{ff} \ -A_{fc} \ -A_{cf} \ Acc] \) be positive semi-definite
- \( x_f^T D_{ff} x_f \leq \lambda_{\min} x_f^T D_{ff} x_f \leq x_f^T A_{ff} x_f \leq \lambda_{\max} x_f^T D_{ff} x_f \)
- \( \nu_{\min} x_c^T S x_c \leq \hat{A}_{cc} x_c \leq \nu_{\max} x_c^T S x_c \)

Then,

\[
\kappa(B^{-\frac{1}{2}} A B^{-\frac{1}{2}}) \leq \left(1 + \sqrt{1 - \frac{1}{\lambda_{\max}}} \right) \frac{\lambda_{\max}^2 \nu_{\max}}{\min(\nu_{\min}, \lambda_{\min})}.
\]

Generalized AMG Measure

Let

- Relaxation be given by $I - D^{-1}A$
- $Q$ be a projection onto the range of $P$
- $\mu(Q, e) = \frac{\langle D(D+D^T-A)^{-1}D^T(I-Q)e,(I-Q)e \rangle}{\langle Ae,e \rangle} \leq K$ for $e \neq 0$
- $MG_2$ be a two-grid $V(0,1)$-cycle

Then,

$$\| MG_2 \|_A \leq \left( 1 - \frac{1}{K} \right)^{\frac{1}{2}}$$

Compatible Relaxation

“A general measure for the quality of the set of coarse variables is the convergence rate of the compatible relaxation”

One approach:
• Run relaxation on tentative $F$-set
• Identify points where compatible relaxation is slow
• Choose subset of these points to add to $C$

J. Brannick, Wednesday 11:00
Compatible Relaxation

Let

- $D$ be symmetric
- $2D - A$ be positive definite
- $x^T A x \leq \omega x^T D x$
- $\rho_f = \| I - D_{ff}^{-1} A_{ff} \|_{A_{ff}}$

Then,

$$\min_P \max_{e \neq 0} \mu(Q, e) \leq \frac{1}{(2 - \omega)(1 - \rho_f)}$$

For a given $F/C$ partition, the best possible measure depends on the equivalence between $D_{ff}$ and $A_{ff}$

Reduction-based AMG

Let

- Relaxation be fine-grid only, \( I - \frac{2}{\lambda_{\text{max}} + 1} D_{ff}^{-1} A_{ff} \)
- \( P = \begin{bmatrix} D_{ff}^{-1} A_{fc} \\ I \end{bmatrix} \)
- \( x_f^T D_{ff} x_f \leq x_f^T A_{ff} x_f \leq \lambda_{\text{max}} x_f^T D_{ff} x_f \)
- \( \begin{bmatrix} D_{ff} & -A_{fc} \\ -A_{cf} & A_{cc} \end{bmatrix} \) be positive semi-definite

Then

\[
\| MG_2 \|_A \leq \left( \frac{1}{\lambda_{\text{max}}} \left( \lambda_{\text{max}} - 1 + \left( \frac{\lambda_{\text{max}} - 1}{\lambda_{\text{max}} + 1} \right)^2 \right) \right)^{\frac{1}{2}}
\]

Coarsening

All three bounds depend on equivalence of $D_{ff}$ and $A_{ff}$

Good partition allows
- effective reduction, $|C| \ll |F|$
- efficient computation of $D_{ff}^{-1} y_f$ or $D_{ff}^{-1} A_{fc}$
- good equivalence, $\lambda_{\text{max}}$ small

A new approach to Compatible Relaxation

- Identify a property of $A_{ff}$ that guarantees good equivalence
- Choose $F$ so that this is always true
Diagonal Dominance

Jacobi on $A_{ff}$ converges if it is diagonally dominant

Stronger dominance $\rightarrow$ faster convergence

$A_{ff}$ is $\theta$-dominant if, for each $i \in F$,

$$a_{ii} \geq \theta \sum_{j \in F} |a_{ij}|$$

Coarsening Goal: Find largest set $F$ such that $A_{ff}$ is $\theta$-dominant.
Complexity

The problem, $\max\{|F| : A_{ff} \text{ is } \theta\text{-dominant}\}$, is NP-complete. Instead,

- Initialize $U = \{1, \ldots, n\}$, $F = C = \emptyset$
- For each point in $U$, compute $\hat{\theta}_i = \frac{a_{ii}}{\sum_{j \in F \cup U} |a_{ij}|}$
- Whenever $\hat{\theta}_i \geq \theta$, $i \rightarrow F$
- If $U \neq \emptyset$, then pick $j = \arg\min_{i \in U} \{\hat{\theta}_i\}$
  - $j \rightarrow C$
  - Update $\hat{\theta}_i$ for all $i \in U$ with $a_{ji} \neq 0$
Solvers

Two-level analysis gives uniform spectral equivalence of $A_{ff}$ with its diagonal, $D_{ff}$.

For multilevel solvers,

- **ARMS**: $D_{ff}$ is sparsest possible ILU of $A_{ff}$
- **AMG**: $D_{ff}^{-1} A_{fc}$ is very simple AMG interpolation operator
Solvers

Two-level analysis gives uniform spectral equivalence of $A_{ff}$ with its diagonal, $D_{ff}$.

For multilevel solvers,

**ARMS:** $D_{ff}$ is sparsest possible ILU of $A_{ff}$

**AMG:** $D_{ff}^{-1}A_{fc}$ is very simple AMG interpolation operator

Combination of dominance-based partitioning and classical algebraic coarsening leads to robust, efficient multilevel solvers

**AMG:** $V(1,1)$ cycles, Full Gauss-Seidel, greedy coarsening with second pass, classical AMG interpolation

**ARMS:** symmetrized ILU, fixed drop and fill thresholds, preconditioned GMRES
PDE Test Problems

Two-dimensional bilinear finite element discretizations of

\[-\nabla \cdot K(x, y) \nabla p(x, y) = 0.\]

**Problem 1:** \( K(x, y) = 1 \)

**Problem 2:** \( K(x, y) = 10^{-8} + 10(x^2 + y^2) \)

**Problem 3:** \( K(x, y) = 10^{-8} \) on 20% of the cells, chosen randomly; \( K(x, y) = 1 \) otherwise

**Problem 4:** \( K(x, y) = \begin{bmatrix} 1 & 0 \\ 0 & 0.01 \end{bmatrix} \)
## AMG Results

<table>
<thead>
<tr>
<th>Prob.</th>
<th>Grid</th>
<th>$c_A$</th>
<th>$t_{\text{setup}}$</th>
<th>$t_{\text{solve}}$</th>
<th># iters.</th>
<th>$\rho$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>512 × 512</td>
<td>1.33</td>
<td>1.3</td>
<td>0.7</td>
<td>5</td>
<td>0.13</td>
</tr>
<tr>
<td></td>
<td>1024 × 1024</td>
<td>1.33</td>
<td>5.1</td>
<td>2.5</td>
<td>5</td>
<td>0.14</td>
</tr>
<tr>
<td></td>
<td>2048 × 2048</td>
<td>1.33</td>
<td>21.9</td>
<td>10.5</td>
<td>5</td>
<td>0.14</td>
</tr>
<tr>
<td>2</td>
<td>512 × 512</td>
<td>1.33</td>
<td>1.3</td>
<td>0.6</td>
<td>5</td>
<td>0.13</td>
</tr>
<tr>
<td></td>
<td>1024 × 1024</td>
<td>1.33</td>
<td>5.1</td>
<td>2.5</td>
<td>5</td>
<td>0.14</td>
</tr>
<tr>
<td></td>
<td>2048 × 2048</td>
<td>1.33</td>
<td>21.7</td>
<td>10.4</td>
<td>5</td>
<td>0.14</td>
</tr>
<tr>
<td>3</td>
<td>512 × 512</td>
<td>2.06</td>
<td>2.3</td>
<td>1.2</td>
<td>6</td>
<td>0.35</td>
</tr>
<tr>
<td></td>
<td>1024 × 1024</td>
<td>2.08</td>
<td>9.6</td>
<td>4.8</td>
<td>6</td>
<td>0.40</td>
</tr>
<tr>
<td></td>
<td>2048 × 2048</td>
<td>2.10</td>
<td>41.0</td>
<td>19.8</td>
<td>6</td>
<td>0.46</td>
</tr>
<tr>
<td>4</td>
<td>512 × 512</td>
<td>2.39</td>
<td>1.5</td>
<td>1.0</td>
<td>5</td>
<td>0.13</td>
</tr>
<tr>
<td></td>
<td>1024 × 1024</td>
<td>2.41</td>
<td>6.2</td>
<td>4.1</td>
<td>5</td>
<td>0.20</td>
</tr>
<tr>
<td></td>
<td>2048 × 2048</td>
<td>2.43</td>
<td>25.8</td>
<td>17.7</td>
<td>5</td>
<td>0.20</td>
</tr>
</tbody>
</table>
## ARMS Results

<table>
<thead>
<tr>
<th>Prob.</th>
<th>Grid</th>
<th>$c_B$</th>
<th>$t_{\text{setup}}$</th>
<th>$t_{\text{solve}}$</th>
<th># iters.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>128 × 128</td>
<td>2.65</td>
<td>0.2</td>
<td>0.3</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td>256 × 256</td>
<td>2.67</td>
<td>1.3</td>
<td>2.2</td>
<td>44</td>
</tr>
<tr>
<td></td>
<td>512 × 512</td>
<td>2.68</td>
<td>11.0</td>
<td>22.5</td>
<td>82</td>
</tr>
<tr>
<td>2</td>
<td>128 × 128</td>
<td>2.39</td>
<td>0.2</td>
<td>0.3</td>
<td>31</td>
</tr>
<tr>
<td></td>
<td>256 × 256</td>
<td>2.35</td>
<td>0.8</td>
<td>2.9</td>
<td>56</td>
</tr>
<tr>
<td></td>
<td>512 × 512</td>
<td>2.32</td>
<td>3.0</td>
<td>28.2</td>
<td>97</td>
</tr>
<tr>
<td>3</td>
<td>128 × 128</td>
<td>1.40</td>
<td>0.2</td>
<td>0.3</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>256 × 256</td>
<td>1.42</td>
<td>0.7</td>
<td>2.2</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td>512 × 512</td>
<td>1.42</td>
<td>3.0</td>
<td>22.9</td>
<td>83</td>
</tr>
<tr>
<td>4</td>
<td>128 × 128</td>
<td>1.61</td>
<td>0.2</td>
<td>0.3</td>
<td>26</td>
</tr>
<tr>
<td></td>
<td>256 × 256</td>
<td>1.62</td>
<td>0.8</td>
<td>2.0</td>
<td>42</td>
</tr>
<tr>
<td></td>
<td>512 × 512</td>
<td>1.63</td>
<td>3.2</td>
<td>16.2</td>
<td>65</td>
</tr>
</tbody>
</table>
General ARMS Tests

- Test set from Rutherford-Appleton Labs
- 22 Selected problems, from 120K to 3.6M non-zeros
- Compared to ILUTP, fill factors adjusted to match ARMS preconditioner complexities

Results:
- ARMS converged in available memory (2GB + 1 GB swap) on 21 problems
- ILUTP converged for 13 problems, limited to memory or $2 \times$ ARMS iteration count
- ILUTP needed fewer iterations for 7 problems
- Equal performance for 4
- ARMS faster for 10

General ARMS Tests

- Test set from Rutherford-Appleton Labs
- 22 Selected problems, from 120K to 3.6M non-zeros
- Compared to ILUTP, fill factors adjusted to match ARMS preconditioner complexities

Results:
- ARMS converged in available memory (2GB + 1 GB swap) on 21 problems
- ILUTP converged for 13 problems, limited to memory or $2 \times$ ARMS iteration count
- ILUTP needed fewer iterations for 7 problems
- Equal performance for 4
- ARMS faster for 10

Nonsymmetric ARMS

Naïve Approach

• Choose row or column diagonal dominance
• Updates for row dominance require transpose

Nonsymmetric Permutations

• Choose offdiagonals as pivots to maximize dominance
• Simultaneously aim for row and column dominance

Results

• Test problems from earlier paper
• Naïve approach easily solves 31 of 45 problems
• Nonsymmetric permutation approach solves 43 of 45

Summary

- Theoretical motivation: fine-scale spectral equivalence
- Choose partition to guarantee good equivalence
- Diagonal dominance is simple, but effective
- Multilevel results show robustness and efficiency
Summary

- Theoretical motivation: fine-scale spectral equivalence
- Choose partition to guarantee good equivalence
- Diagonal dominance is simple, but effective
- Multilevel results show robustness and efficiency

Future Directions

- Symmetric ARMS with IC/MIC versus ILU
- Further explore non-symmetric ARMS
- More complicated measures