Practical aspects of theoretical bounds on algebraic multigrid

Scott MacLachlan
scott.maclachlan@gmail.com

Delft Institute of Applied Mathematics, TU-Delft
Centrum voor Wiskunde en Informatica, Amsterdam

Joint work with
Yousef Saad, University of Minnesota
Luke Olson, University of Illinois at Urbana-Champaign

July 20, 2007

Algebraic Multigrid for Real-World Applications

Fast solvers for real problems

Applications include

- Markov-chain processes (Google)
- Maxillo-facial surgery
- Quantum Chromodynamics

Challenges include

- Higher-order and discontinuous finite elements
- Multiple scales
- Extreme heterogeneity

What is Algebraic Multigrid?

Algebraic Multigrid (AMG) is a family of techniques

Properties:

- Multigrid/Multilevel structure
  - Hierarchy of models on increasingly coarser scales
- Inexpensive processing on each scale
  - Jacobi/Gauss-Seidel/ILU iteration
- Additive/multiplicative coarse-grid correction

What is Algebraic Multigrid?

Algebraic Multigrid (AMG) is a family of techniques

Properties:

- Multigrid/Multilevel structure
  - Hierarchy of models on increasingly coarser scales
- Inexpensive processing on each scale
  - Jacobi/Gauss-Seidel/ILU iteration
- Additive/multiplicative coarse-grid correction

Coarse-grid models created algebraically

- System structure inferred from matrix entries
- Geometric/PDE information replaced by simple measures
- No/limited assumptions on problem origin

Gaussian Elimination

Goal is to factor $A = LDU$

First step: partition A:

$$A = \begin{bmatrix} a_{1,1} & a_{1,\star} \\ a_{\star,1} & A^{(2)} \end{bmatrix},$$

then factor the first row and column:

$$A = \begin{bmatrix} 1 & 0 \\ a_{1,1}^{-1} a_{\star,1} & I \end{bmatrix} \begin{bmatrix} a_{1,1} & 0 \\ 0 & A^{(2)} - a_{\star,1} a_{1,1}^{-1} a_{1,\star} \end{bmatrix} \begin{bmatrix} 1 & a_{1,1}^{-1} a_{1,\star} \\ 0 & I \end{bmatrix}$$

Apply this step recursively to $\hat{A}^{(2)} = A^{(2)} - a_{\star,1} a_{1,1}^{-1} a_{1,\star}$

Block Factorization

Can also do this elimination in blocks

Partition

$$A\mathbf{x} = \begin{bmatrix} A_{ff} & -A_{fc} \\ -A_{cf} & A_{cc} \end{bmatrix} \begin{pmatrix} \mathbf{x}_f \\ \mathbf{x}_c \end{pmatrix} = \begin{pmatrix} \mathbf{b}_f \\ \mathbf{b}_c \end{pmatrix} = \mathbf{b},$$

then A can be block factored as

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

where $\hat{A}_{cc} = A_{cc} - A_{cf}A_{ff}^{-1}A_{fc}$.

Block Factorization Solve

Easy to write inverse of block-factored form, so that

$$\begin{pmatrix} \mathbf{x}_f \\ \mathbf{x}_c \end{pmatrix} = \begin{bmatrix} I & A_{ff}^{-1}A_{fc} \\ 0 & I \end{bmatrix} \begin{bmatrix} A_{ff}^{-1} & 0 \\ 0 & \hat{A}_{cc}^{-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ A_{cf}A_{ff}^{-1} & I \end{bmatrix} \begin{pmatrix} \mathbf{b}_f \\ \mathbf{b}_c \end{pmatrix}.$$

Algorithm: solve $A\mathbf{x} = b$ by

1. $\mathbf{y}_f = A_{ff}^{-1}\mathbf{b}_f$
2. $\mathbf{y}_c = \mathbf{b}_c + A_{cf}\mathbf{y}_f$
3. Solve $\hat{A}_{cc}\mathbf{x}_c = \mathbf{y}_c$
4. $\mathbf{x}_f = \mathbf{y}_f + A_{ff}^{-1}A_{fc}\mathbf{x}_c$

Block Factorization Preconditioners

Idea: precondition A using

$$M = \begin{bmatrix} I & 0 \\ -A_{cf} D_{ff}^{-1} & I \end{bmatrix} \begin{bmatrix} D_{ff} & 0 \\ 0 & S \end{bmatrix} \begin{bmatrix} I & -D_{ff}^{-1} A_{fc} \\ 0 & I \end{bmatrix}.$$

Approximation to block factorization of A with

- $D_{ff} \approx A_{ff}$
- $S \approx A_{cc} - A_{cf} A_{ff}^{-1} A_{fc}$

If D_{ff} and S are easy to invert, then computing $M^{-1}\mathbf{r}$ is cheap

$$M^{-1}\mathbf{r} = \begin{bmatrix} I & D_{ff}^{-1} A_{fc} \\ 0 & I \end{bmatrix} \begin{bmatrix} D_{ff}^{-1} & 0 \\ 0 & S^{-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ A_{cf} D_{ff}^{-1} & I \end{bmatrix} \begin{pmatrix} \mathbf{r}_f \\ \mathbf{r}_c \end{pmatrix}$$

O. Axelsson, *Iterative Solution Methods*, 1994

Algebraic Recursive Multilevel Solver

Approximate A_{ff} by its ILU factors, $A_{ff} \approx D_{ff} = LU$.

Preconditioner is

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

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

Approximate Schur complement, S,

- computed using fill/truncation techniques as in ILU
- solved recursively

Y. Saad and B. Suchomel, Numer. Linear Algebra Appl. 2002, **9**:359-378
M. Bollhöfer and Y. Saad, SISC 2006, **27**:1627-1650

Additive Coarse-Grid Correction

Defining $P = \left[\begin{array}{c} D_{ff}^{-1} A_{fc} \\ I \end{array} \right]$, we can write

$$I - M^{-1} A = I - P S^{-1} P^T A - \begin{pmatrix} D_{ff}^{-1} & 0 \\ 0 & 0 \end{pmatrix} A.$$

Two ways of reducing errors:

- $\left(I - \begin{pmatrix} D_{ff}^{-1} & 0 \\ 0 & 0 \end{pmatrix} A \right)$ only reduces $\mathbf{e}_f$
- $\left(I - P S^{-1} P^T A \right)$ reduces errors only in Range(P)

Block factorization naturally defines an additive correction

Reduction-based AMG

Multiplicative variant of block factorization

$$I - M_{AMG}^{-1}A = \left(I - PS^{-1}P^TA\right)\left(I - \omega \begin{pmatrix} D_{ff}^{-1} & 0 \\ 0 & 0 \end{pmatrix} A\right)$$

Error partitioned into two subspaces:

M. Ries, U. Trottenberg, G. Winter, J. Lin. Alg. Applic., 1983
S. MacLachlan, T. Manteuffel, S. McCormick, Numer. Linear Algebra Appl. 2006.

Reduction-based AMG

Multiplicative variant of block factorization

$$I - M_{AMG}^{-1}A = \left(I - PS^{-1}P^TA\right)\left(I - \omega \begin{pmatrix} D_{ff}^{-1} & 0 \\ 0 & 0 \end{pmatrix} A\right)$$

Error partitioned into two subspaces:

- Errors in $R = \text{Range}\left(\begin{bmatrix} A_{ff}^{-1}A_{fc} \\ I \end{bmatrix}\right)$, must be reduced by coarse-grid correction

M. Ries, U. Trottenberg, G. Winter, J. Lin. Alg. Applic., 1983

S. MacLachlan, T. Manteuffel, S. McCormick, Numer. Linear Algebra Appl. 2006.

Reduction-based AMG

Multiplicative variant of block factorization

$$I - M_{AMG}^{-1} A = \left(I - PS^{-1}P^T A\right) \left(I - \omega \begin{pmatrix} D_{ff}^{-1} & 0 \\ 0 & 0 \end{pmatrix} A\right)$$

Error partitioned into two subspaces:

- Errors in $R = \text{Range}\left(\begin{bmatrix} A_{ff}^{-1} A_{fc} \\ I \end{bmatrix}\right)$, must be reduced by coarse-grid correction
- Errors in $(R)^{\perp}$, should be reduced by (fine-grid) relaxation

M. Ries, U. Trottenberg, G. Winter, J. Lin. Alg. Applic., 1983

S. MacLachlan, T. Manteuffel, S. McCormick, Numer. Linear Algebra Appl. 2006.

Classical Algebraic Multigrid

"Generalization" of AMGr:

- Full-grid relaxation using Jacobi or Gauss-Seidel
- Interpolation chosen to fit algebraically smooth errors
    - Assumption on errors that relaxation is slow to reduce
- Coarse grid chosen based on M-matrix properties
- Coarse-grid equations solved by recursion

Idea: Error-propagation, $(I - PS^{-1}P^TA)(I - D^{-1}A)$

- requires complementarity
- choose P to complement known properties of D

A. Brandt, S. McCormick, J. Ruge, in *Sparsity and Its Applications*, 1984
J. Ruge and K. Stüben, in *Multigrid Methods*, 1987

What About Convergence?

There is no single AMG convergence theory!

Two questions:

- Given a method, how does it perform?
- Given a class of problems, how do we design a method?

What About Convergence?

There is no single AMG convergence theory!

Two questions:

- Given a method, how does it perform?
- Given a class of problems, how do we design a method?

Ideal theory is predictive:

Computable: A priori information on expected performance

Sharp: Prediction is accurate

Goals

Aim for an algebraic theory for algebraic multigrid

Stay away from the PDEs

- Lots of theory for multigrid based on elliptic PDEs
- We apply AMG for a much larger class of problems
- Conditions on convergence should be from linear algebra

Convergence theory typically takes the form of a bound:

$$\mathcal{K}\left(M^{-\frac{1}{2}}AM^{-\frac{1}{2}}\right) \leq K \quad \text{or} \quad \|I - M^{-1}A\|_A \leq 1 - \frac{1}{K}$$

Upper and Lower Bounds

Upper convergence bounds give worst-case performance

Consider $\mathcal{K}\left(M^{-\frac{1}{2}}AM^{-\frac{1}{2}}\right) \leq K$ and $\|I - M^{-1}A\|_A \leq 1 - \frac{1}{K}$, K determines iterations needed to guarantee accuracy

Upper and Lower Bounds

Upper convergence bounds give worst-case performance

Consider $\mathcal{K}\left(M^{-\frac{1}{2}}AM^{-\frac{1}{2}}\right) \leq K$ and $\|I - M^{-1}A\|_A \leq 1 - \frac{1}{K}$, K determines iterations needed to guarantee accuracy

What about lower bounds?

- Indicate sharpness, potential problems
- Useful in algorithm development

Large lower bounds on $\mathcal{K}\left(M^{-\frac{1}{2}}AM^{-\frac{1}{2}}\right)$ do not guarantee bad CG convergence!

Additive Convergence Theory

Let A be symmetric and positive definite

- $\begin{bmatrix} D_{ff} & -A_{fc} \\ -A_{cf} & A_{cc} \end{bmatrix}$ be positive semi-definite
- $\mathbf{x}_f^T D_{ff} \mathbf{x}_f \leq \lambda_{\min} \mathbf{x}_f^T D_{ff} \mathbf{x}_f \leq \mathbf{x}_f^T A_{ff} \mathbf{x}_f \leq \lambda_{\max} \mathbf{x}_f^T D_{ff} \mathbf{x}_f$
- $\nu_{\min} \mathbf{x}_c^T S \mathbf{x}_c \leq \mathbf{x}_c^T \hat{A}_{cc} \mathbf{x}_c \leq \nu_{\max} \mathbf{x}_c^T S \mathbf{x}_c$

Then,

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

Y. Notay, Numer. Linear Algebra Appl. 2005, **12**:419-451

Additive Convergence Theory

Let A be symmetric and positive definite

- $\begin{bmatrix} D_{ff} & -A_{fc} \\ -A_{cf} & A_{cc} \end{bmatrix}$ be positive semi-definite
- $\mathbf{x}_f^T D_{ff} \mathbf{x}_f \leq \lambda_{\min} \mathbf{x}_f^T D_{ff} \mathbf{x}_f \leq \mathbf{x}_f^T A_{ff} \mathbf{x}_f \leq \lambda_{\max} \mathbf{x}_f^T D_{ff} \mathbf{x}_f$
- $\nu_{\min} \mathbf{x}_c^T S \mathbf{x}_c \leq \mathbf{x}_c^T \hat{A}_{cc} \mathbf{x}_c \leq \nu_{\max} \mathbf{x}_c^T S \mathbf{x}_c$

Then,

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

and

$$\frac{\lambda_{\max}}{\lambda_{\min}} \leq \kappa(M^{-\frac{1}{2}} A M^{-\frac{1}{2}})$$

Y. Notay, Numer. Linear Algebra Appl. 2005, **12**:419-451

AMGr Convergence

Let A be symmetric and positive-definite

- $\begin{bmatrix} D_{ff} & -A_{fc} \\ -A_{cf} & A_{cc} \end{bmatrix}$ be positive semi-definite
- $\mathbf{x}_f^T D_{ff} \mathbf{x}_f \leq \mathbf{x}_f^T A_{ff} \mathbf{x}_f \leq \lambda_{\max} \mathbf{x}_f^T D_{ff} \mathbf{x}_f$
- Choose relaxation as $I - \frac{2}{\lambda_{\max}+1} D_{ff}^{-1} A_{ff}$
- Take $P = \begin{bmatrix} D_{ff}^{-1} A_{fc} \\ I \end{bmatrix}$, $S = P^T A P$

Then

$$\|I - M_{AMG}^{-1} A\|_A \leq \left(1 - \left(\frac{2}{\lambda_{\max} + 1}\right)^2\right)^{\frac{1}{2}}$$

S. MacLachlan, T. Manteuffel, S. McCormick, Numer. Linear Algebra Appl. 2006.

Lower Bounds for AMG

Is a small $\lambda_{\text{max}}(D_{ff}^{-1}A_{ff})$ necessary for good AMG performance?

Consider $A = (n+1)I - \mathbf{1}\mathbf{1}^T = \begin{pmatrix} n & -1 & \cdots & -1 \\ -1 & n & \cdots & -1 \\ \vdots & \ddots & \ddots & \vdots \\ -1 & -1 & \cdots & n \end{pmatrix}$

Choose:

- A_{ff} to be upper $(n-1) \times (n-1)$ block
- $D_{ff} = I$
- Full-grid Richardson relaxation
- $P = \begin{bmatrix} D_{ff}^{-1}A_{fc} \\ I \end{bmatrix}$, $S = P^TAP$

$$\lambda_{\text{max}} = n+1, \text{ but } \|I - M_{AMG}^{-1}A\|_A \leq \tfrac{1}{2}$$

Theory and Algorithms

AMGr theory not predictive for classical AMG

Can we design a complete algorithm for which it is predictive?

- Given A
- Choose partition, $A = \left[\begin{smallmatrix} A_{ff} & -A_{fc} \\ -A_{cf} & A_{cc} \end{smallmatrix}\right]$
- Choose splitting, $A_{ff} = D_{ff} - R_{ff}$
- Estimate λ_{max}
- Use AMGr to guarantee convergence

Coarse-grid Selection

Key to success in AMGr is in the partitioning of A

$$A = \left[\begin{array}{cc} A_{ff} & -A_{fc} \\ -A_{cf} & A_{cc} \end{array} \right]$$

Need: Good approximation, D_{ff}, to A_{ff}
Need: Cheap computation of $D_{ff}^{-1}\mathbf{r}_f$, $D_{ff}^{-1}A_{fc}$
Need: Dimension of A_{cc} much smaller than A

Two Observations

1. Cost of $D_{ff}^{-1}\mathbf{r}_f$ depends on sparsity structure of D_{ff}
    - Cheapest when D_{ff} is diagonal

S. MacLachlan, Y. Saad, SISC, to appear

Two Observations

1. Cost of $D_{ff}^{-1}\mathbf{r}_f$ depends on sparsity structure of D_{ff}
    - Cheapest when D_{ff} is diagonal
2. Diagonally dominant A_{ff} can be approximated by its diagonal
    - More diagonally dominant → better approximation

S. MacLachlan, Y. Saad, SISC, to appear

Two Observations

1. Cost of $D_{ff}^{-1}\mathbf{r}_f$ depends on sparsity structure of D_{ff}
    - Cheapest when D_{ff} is diagonal
2. Diagonally dominant A_{ff} can be approximated by its diagonal
    - More diagonally dominant → better approximation

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

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

S. MacLachlan, Y. Saad, SISC, to appear

Two Observations

1. Cost of $D_{ff}^{-1}\mathbf{r}_f$ depends on sparsity structure of D_{ff}
   - Cheapest when D_{ff} is diagonal
2. Diagonally dominant A_{ff} can be approximated by its diagonal
   - More diagonally dominant → better approximation

A_{ff} is called θ-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 θ-dominant.

S. MacLachlan, Y. Saad, SISC, to appear

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 $\hat{\theta}_i = \dfrac{a_{ii}}{\displaystyle\sum_{j \in F \cup U} |a_{ij}|}$
- Whenever $\hat{\theta}_i \geq \theta$, $i \longrightarrow F$
- If $U \neq \emptyset$, then pick $j = \text{argmin}_{i \in U}\{\hat{\theta}_i\}$
  - $j \longrightarrow C$
  - Update $\hat{\theta}_i$ for all $i \in U$ with $a_{ji} \neq 0$

S. MacLachlan, Y. Saad, SISC, to appear

Choosing D_{ff}

Several ways to choose D_{ff} to bound $\lambda_{\max}$

Greedy coarsening algorithm guarantees $a_{ii} \geq \theta \sum_{j \in F} |a_{ij}|$

- $D_{ff} = (2 - \frac{1}{\theta}) diag(A_{ff})$
- $(D_{ff})_{ii} = (2 - \frac{1}{\theta_i}) a_{ii}$

In general, guarantee $\lambda_{\max} \leq \frac{1}{2\theta - 1}$
Bound on error-reduction per cycle by

$$\|I - M_{AMG}^{-1} A\|_A \leq \left(\frac{2\theta - 1}{\theta^2} \right)^{\frac{1}{2}}$$

S. MacLachlan, Y. Saad, SISC, to appear

Similar Approach

Coarsen based on compatible relaxation

- If $\|I - D_{ff}^{-1}A_{ff}\|_{A_{ff}}$ is small, then there is a P that gives good AMG performance
- Choose coarse grid by testing convergence of relaxation, $I - D_{ff}^{-1}A_{ff}$

Fix stencil of interpolation, P

Interpolate based on minimizing trace of P^TAP

- Unconstrained minimization leads to $P^TAP = \hat{A}_{cc}$
- Ensure stability of coarse-scale problem, but control iteration costs

R. Falgout and P. Vassilevski, SIAM J. Numer. Anal. 2004, **42**:1669-1693

J. Brannick and L. Zikatanov, in *Proc. DD16, 2007*

Sharp and Two-Sided Bounds

Many different bounds on AMG performance are possible

Sharp two-level bound, $\|I - M_{\text{AMG}}^{-1}A\|_A = 1 - \frac{1}{K}$, for

$$K = \max_{\mathbf{v}} \frac{\mathbf{v}^T \tilde{M} P (P^T \tilde{M} P)^{-1} P^T \tilde{M} \mathbf{v}}{\mathbf{v}^T A \mathbf{v}}$$

- Bound is sharp, but depends on eigenvalue problem

More recently, Zikatanov has shown lower bounds on K that can be used to gain lower bounds on AMG convergence

R. Falgout, P. Vassilevski, L. Zikatanov, Num. Linear Algebra Appl. 2005, **12**:471-494

Summary

- AMG is a family of algebraic multilevel solvers
- Coarse-grid corrections may be additive or multiplicative
- Want sharp, predictive theory for AMG performance
- Want AMG algorithms designed to satisfy theory
- Theory links performance to fine-grid spectral equivalence
- Couple coarse-grid selection and interpolation to bound convergence

Sharpness, Computability, Algorithms

Sharpness:

- Sharp convergence theory is a spectral theory
- Good convergence bounds require (sharp) eigenvalue bounds

Computability:

- Predictive theory is a useful tool
- Convergence bound must depend on easily calculated quantities

Algorithms:

- Classical algorithms motivated by heuristics
- More recently, use theory to motivate algorithms
- Limited success, but both algorithms and bounds improving