Strategies for Self-Correcting Algebraic Multigrid Methods

Scott MacLachlan

Scott.MacLachlan@colorado.edu

Department of Applied Mathematics

University of Colorado at Boulder

Collaborators

- Steve McCormick
- Tom Manteuffel
- Rob Falgout
- Marian Brezina
- John Ruge

Basic Multigrid V-Cycle

- Relax on $Ax = b$
- Restrict residual equation to coarser grid
- Solve coarse grid equation recursively
- Interpolate coarse grid correction and add to current approximation
- Relax on $Ax = b$

Overall Goal

- Interested in applying algebraic multigrid methods to "more difficult" problems, such as systems (e.g. elasticity)

Overall Goal

- Interested in applying algebraic multigrid methods to “more difficult” problems, such as systems (e.g. elasticity)
- Develop an algebraic multigrid solver with increased robustness properties while not sacrificing optimality

Overall Goal

- Interested in applying algebraic multigrid methods to "more difficult" problems, such as systems (e.g. elasticity)
- Develop an algebraic multigrid solver with increased robustness properties while not sacrificing optimality
- Develop a solver which defaults to simplicity if given a simple problem

Basic Multigrid Properties

- Simple (Gauss-Seidel) Relaxation is inefficient for $Ax = b$ on error components e that give relatively small residuals: Ae is “small” relative to e (e is said to be algebraically smooth)

Basic Multigrid Properties

- Simple (Gauss-Seidel) Relaxation is inefficient for $Ax = b$ on error components e that give relatively small residuals: Ae is “small” relative to e (e is said to be algebraically smooth)
  - Example: For s.p.d. matrices, Richardson iteration stalls iff the error e satisfies

$$\langle Ae, e \rangle \ll ||A|| \langle e, e \rangle$$

Basic Multigrid Properties

- Simple (Gauss-Seidel) Relaxation is inefficient for $Ax = b$ on error components e that give relatively small residuals: Ae is "small" relative to e (e is said to be algebraically smooth)
  - Example: For s.p.d. matrices, Richardson iteration stalls iff the error e satisfies

$$\langle Ae, e\rangle \ll \|A\|\langle e, e\rangle$$

- For efficient multigrid performance, relaxation and coarse grid correction must be complementary

Current Assumptions

- Coarse grids are predetermined and sufficient for full multigrid efficiency
  - Currently choosing coarse grids based on geometric or “classical” AMG criteria
  - Eventually hope to determine coarse grids adaptively as well (Compatible Relaxation)

Main Ideas

- Build a multigrid heirarchy such that slowly converging components of the method are incorporated as they are uncovered

Main Ideas

- Build a multigrid heirarchy such that slowly converging components of the method are incorporated as they are uncovered
- Complementarity of relaxation and coarse grid correction means that if relaxation is inefficient on a component then that component must be treated by coarse grid correction

Main Ideas

- Build a multigrid heirarchy such that slowly converging components of the method are incorporated as they are uncovered
- Complementarity of relaxation and coarse grid correction means that if relaxation is inefficient on a component then that component must be treated by coarse grid correction
- Components that are slow to converge for $Ax = b$ will also be slow for $Ax = 0$

Details of the Method

- Relax on $Ax = 0$ with a random initial guess to quickly resolve a representative of the slow-to-converge components

Details of the Method

- Relax on $Ax = 0$ with a random initial guess to quickly resolve a representative of the slow-to-converge components or show that relaxation is sufficient for the problem

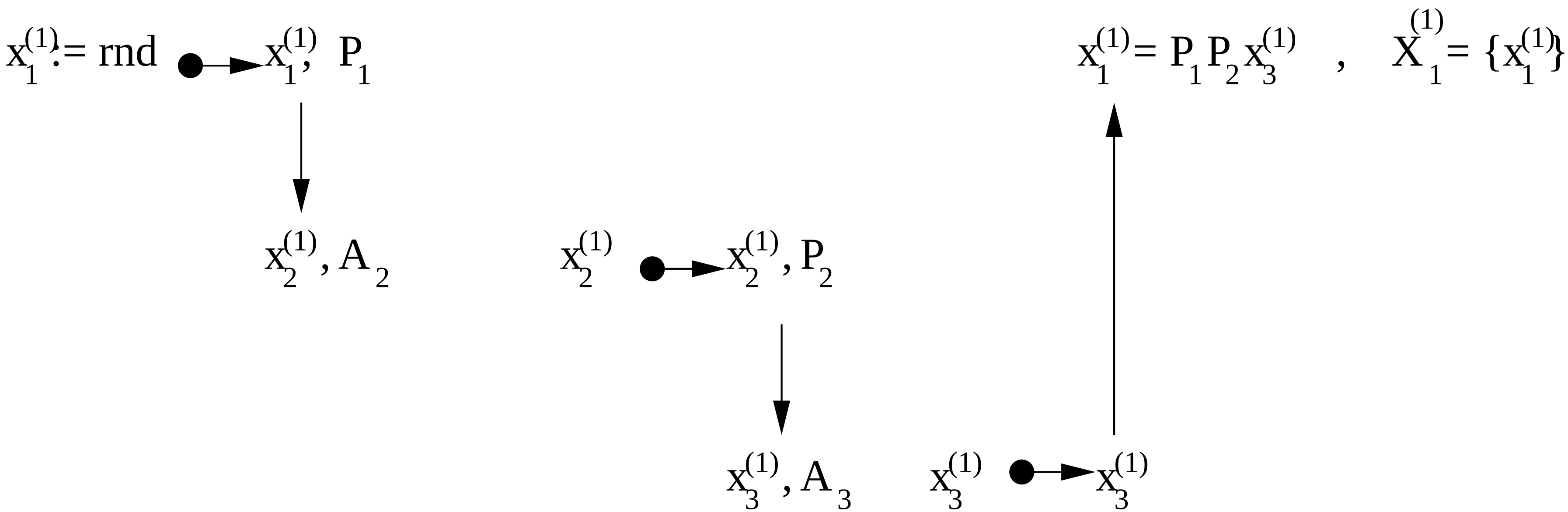
Details of the Method

- Relax on $Ax = 0$ with a random initial guess to quickly resolve a representative of the slow-to-converge components or show that relaxation is sufficient for the problem
- Define a 2-grid method by choosing a coarse grid and interpolation so that this component is in the range of interpolation (and using variational properties)

Details of the Method

- Relax on $Ax = 0$ with a random initial guess to quickly resolve a representative of the slow-to-converge components or show that relaxation is sufficient for the problem
- Define a 2-grid method by choosing a coarse grid and interpolation so that this component is in the range of interpolation (and using variational properties)
- Go to a multigrid method by recursion

Details . . .


$x_1^{(1)} := \text{rnd}$
$x_1^{(1)}, \ P_1$
$x_1^{(1)} = P_1 P_2 x_3^{(1)} \quad , \quad X_1^{(1)} = \{x_1^{(1)}\}$
$x_2^{(1)}, A_2$
$x_2^{(1)}$
$x_2^{(1)}, P_2$
$x_3^{(1)}, A_3$
$x_3^{(1)}$
$x_3^{(1)}$

Details . . .

- Have developed a multigrid heirarchy, want to know if it is good enough

Details . . .

- Have developed a multigrid heirarchy, want to know if it is good enough
- Test it on $Ax = 0$

Details . . .

- Have developed a multigrid heirarchy, want to know if it is good enough
- Test it on $Ax = 0$
- If good enough, stop

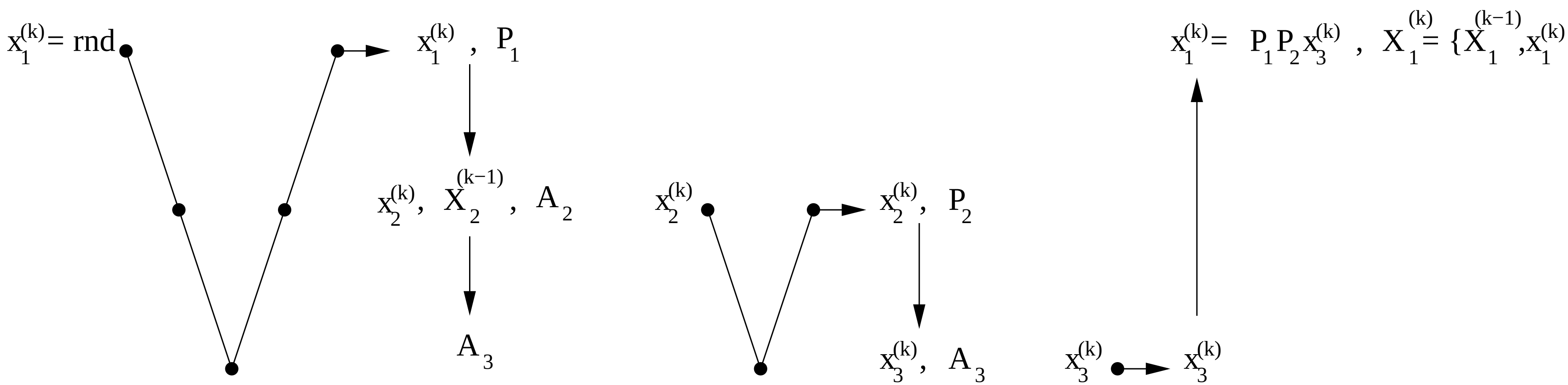
Details . . .

- Have developed a multigrid heirarchy, want to know if it is good enough
- Test it on $Ax = 0$
- If good enough, stop
- If not, define a new interpolation based on the error from this test AND the component from the first step

Details . . .

- Have developed a multigrid heirarchy, want to know if it is good enough
- Test it on $Ax = 0$
- If good enough, stop
- If not, define a new interpolation based on the error from this test AND the component from the first step
- Iterate . . .

Details . . .



$x_1^{(k)} = \text{rnd}$
$x_1^{(k)}$, P_1
$x_2^{(k)}$, $X_2^{(k-1)}$, A_2
A_3
$x_2^{(k)}$
$x_2^{(k)}$, P_2
$x_3^{(k)}$, A_3
$x_3^{(k)}$
$x_3^{(k)}$
$x_1^{(k)} = P_1 P_2 x_3^{(k)}$, $X_1^{(k)} = \{X_1^{(k-1)}, x_1^{(k)}\}$

Choosing Interpolation

- Seek to define interpolation to fit collection of algebraically smooth vectors

Choosing Interpolation

- Seek to define interpolation to fit collection of algebraically smooth vectors
- Algebraic smoothness means

$$(Ax)_i \approx 0$$

$$\text{or } a_{ii}x_i \approx -\sum_{j\in N_i} a_{ij}x_j$$

$$= -\sum_{j\in C_i} a_{ij}x_j - \sum_{k\in F_i} a_{ik}x_k$$

Choosing Interpolation

- Seek to define interpolation to fit collection of algebraically smooth vectors
- Algebraic smoothness means

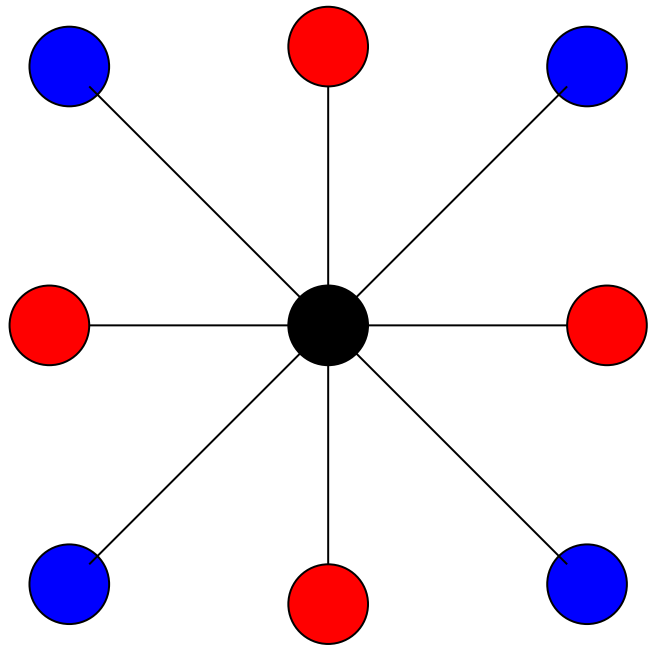
$$(Ax)_i \approx 0$$

$$\text{or } a_{ii}x_i \approx -\sum_{j \in N_i} a_{ij}x_j$$

$$= -\sum_{j \in C_i} a_{ij}x_j - \sum_{k \in F_i} a_{ik}x_k$$

- To define interpolation, need to collapse connections from F_i to C_i

N_i, the neighbourhood of i



Fine Grid Points

Coarse Grid Points

Choosing Interpolation . . .

- Seek to define interpolation to fit collection of algebraically smooth vectors

Choosing Interpolation . . .

- Seek to define interpolation to fit collection of algebraically smooth vectors
- If $k \in F_i$ is connected to a set of $j \in C_i$, we want to write

$$x_k = \sum_{j \in C_i} w_{kj} x_j$$

Choosing Interpolation . . .

- Seek to define interpolation to fit collection of algebraically smooth vectors
- If $k \in F_i$ is connected to a set of $j \in C_i$, we want to write

$$x_k = \sum_{j \in C_i} w_{kj} x_j$$

- Considering all representatively smooth vectors, this becomes the system

$$X_k^T = X_{C_i}^T w_k$$

Choosing Interpolation ...

- Seek to define interpolation to fit collection of algebraically smooth vectors
- If $k \in F_i$ is connected to a set of $j \in C_i$, we want to write

$$x_k = \sum_{j \in C_i} w_{kj} x_j$$

- Considering all representatively smooth vectors, this becomes the system

$$X_k^T = X_{C_i}^T w_k$$

- When we have accumulated enough vectors (more vectors to fit than points in C_i), we must solve this in a least-norm sense

Choosing Interpolation . . .

- Once we have w_{kj}, we write

$$x_k = \sum_{j \in C_i} w_{kj} x_j$$

Choosing Interpolation . . .

- Once we have w_{kj}, we write

$$x_k = \sum_{j \in C_i} w_{kj} x_j$$

- Then, using the definition of algebraic smoothness, we have

$$a_{ii} x_i \approx - \sum_{j \in C_i} a_{ij} x_j - \sum_{k \in F_i} a_{ik} x_k$$

$$a_{ii} x_i \approx - \sum_{j \in C_i} a_{ij} x_j - \sum_{k \in F_i} \sum_{j \in C_i} a_{ik} w_{kj} x_j$$

$$x_i \approx - \sum_{j \in C_i} \frac{a_{ij} + \sum_{k \in F_i} a_{ik} w_{kj}}{a_{ii}} x_j$$

Choosing Interpolation . . .

So, we define interpolation to a fine grid point i as

$$x_i = -\sum_{j \in C_i} \frac{a_{ij} + \sum_{k \in F_i} a_{ik} w_{kj}}{a_{ii}} x_j$$

Summary

- Premise: AMG good, but need assumptions on smoothness

Summary

- Premise: AMG good, but need assumptions on smoothness
- Relieve method from these presumptions

Summary

- Premise: AMG good, but need assumptions on smoothness
- Relieve method from these presumptions
- Coarse grid correction must address errors which look (locally) like representatives

Summary

- Premise: AMG good, but need assumptions on smoothness
- Relieve method from these presumptions
- Coarse grid correction must address errors which look (locally) like representatives
- Get new representatives through method, so they must be distinct

Summary

- Premise: AMG good, but need assumptions on smoothness
- Relieve method from these presumptions
- Coarse grid correction must address errors which look (locally) like representatives
- Get new representatives through method, so they must be distinct
- Get new interpolation by maintaining performance on previous representatives while accounting for new representative

Preliminary Numerics

- $-\nabla \cdot D(x,y)\nabla u(x,y) = 0$ on $[0,1]^2$
- Dirichlet boundary conditions
- Geometric choice of coarse grids
- Interpolation chosen as above, for 1 smooth vector

Preliminary Numerics . . .

- $D(x, y) = 1$ (Laplace)

size	convergence factor
32×32	0.096
64×64	0.14

- $D(x, y) = r_1 * 10^{2r_2}$; r_1, r_2 random, uniform on $[0, 1)$

size	convergence factor
32×32	0.30
64×64	0.50

Future Work

- Further investigation into choosing interpolation

Future Work

- Further investigation into choosing interpolation
- Adaptive choice of coarse grids

Future Work

- Further investigation into choosing interpolation
- Adaptive choice of coarse grids
- Can we do this effectively looking at $Ax = b$, or is a "setup" phase necessary?

Future Work

- Further investigation into choosing interpolation
- Adaptive choice of coarse grids
- Can we do this effectively looking at $Ax = b$, or is a "setup" phase necessary?
- Ensure optimal multigrid efficiency

Conclusions

- Have framework for self correcting multigrid solvers
- Preliminary numerics suggest this approach is feasible
- Method defaults to simple relaxation for problems where this is sufficient
- Much work still to be done