

APPM 6640

Multigrid Methods

Steve McCormick & John Ruge

303-492-0662

stevem@colorado.edu & john.ruge@colorado.edu
<http://amath.colorado.edu/faculty/stevem>

Office Hours

MW 2-4⁴⁵ pm (except for class!) in ECOT 322

CU-Boulder

1 of 322

CU-Boulder

2 of 322

Sources

MGNet Newsletter & software repository

<http://www.mgnet.org/>

MathSciNet Many papers electronically available

<http://www.ams.org/mathscinet>

Copper Mountain Conference March 22-27, 2009

<http://amath.colorado.edu/faculty/copper>

We hope to support your attendance there.

Course web site:

<http://grandmaster.colorado.edu/appm6640/>

These slides are there.

CU-Boulder

3 of 322

CU-Boulder

4 of 322

APPM 6640 Syllabus

- Coursework
 - No exams
 - Homework exercises & computing assignments
 - Project & presentation--joint is OK
 - Can tailor some or all work & later lectures to your interests
- Philosophy
 - Big on fundamentals, principles, & simple ideas
 - Think about the scientific method in general
 - Understand by concrete examples & experience
 - Know the whole picture
 - Ask questions & interject comments
 - Make sure I explain what you need
- Text
 - A Multigrid Tutorial, 2nd edition, 2nd printing
 - Supplemental material as needed

Homework exercises

Due 1 week after the relevant chapter is covered.

- | | |
|-----------------------|----------------------------|
| 1: 3, 4 | 6: 1, 2, 6 |
| 2: 1-3, 8, 10, 13, 16 | 7: 1, 2, 7, 10, 12, 15, 21 |
| 3: 2-6 | 8: 2, 6, 8 |
| 4: 6, 7, 11 | 9: 1, 3 |
| 5: 1-3, 12, 13 | 10: 1, 3, 5, 8, 9 |

Computing assignments

Due 2 weeks after the relevant chapter is covered.

Start your assignments early:

multigrid algorithms can be tricky to code!

Use whatever computing language/platform you like.

2: 20 4: 13, 15 6: 6 by computer!!!

7: Convert your 2-D code developed for Chapter 4 (standard coarsening & point relaxation) to solve (7.14).

Verify the code using $\epsilon = 1$. Test its performance using $\epsilon = 1/2, 1/5, 1/10, 2, 5, 10$. Interpret your results.

Project & presentation

Due at the end of the semester.

- Project definition is intentionally vague.
- Think about what has captured your attention.
- See us, but only after you've thought about it.
- Computing &/or theory &/or application &/or exploration.
- I'd like to see a presentation (10 minutes) plus a short (<5 pages) writeup about what you did.
- Presentation: Prepare! Speak to your peers, not me.
- Check course web site for suggestions.

Successful Scientific Inquiry

- Attitude
 - You can do it! Be positive.
 - But is it really right? Be critical.
 - Don't hope or guess. THINK!
 - Control your emotions! Expect ups & downs.
- Method
 - Start simply. Reduce issue to the simplest possible case.
 - Take tiny steps, but keep the big picture in mind.
 - Study concrete examples.
 - Look for analogies. Can A be done in any way like how B was done?
- Creativity
 - What do you really want? What end are you really aiming for?
 - What do you really need? What you're trying may be sufficient to do what you want, but would an easier-to-prove weaker result do instead?
- Intelligence
 - It doesn't hurt to try to be "smart" too.

A Multigrid Tutorial

2nd Edition, 2nd Printing

By

William L. Briggs

← THANKS!

CU-Denver

Van Emden Henson

← THANKS!

LLNL

Steve McCormick

CU-Boulder

Outline

by chapter

- 1. Model Problems
- 2. Basic Iterative Methods
 - Convergence tests
 - Analysis
- 3. Elements of Multigrid
 - Relaxation
 - Coarsening
- 4. Implementation
 - Complexity
 - Diagnostics
- 5. Some Theory
 - Spectral vs. algebraic
- 6. Nonlinear Problems
 - Full approximation scheme
- 7. Selected Applications
 - Neumann boundaries
 - Anisotropic problems
 - Variable meshes
 - Variable coefficients
- 8. Algebraic Multigrid (AMG)
 - Matrix coarsening
- 9. Multilevel Adaptive Methods
 - FAC
- 10. Finite Elements
 - Variational methodology

Suggested reading

CHECK THE MG LIBRARY & MGNET REPOSITORY

- A. Brandt, "Multi-level Adaptive Solutions to Boundary Value Problems," *Math Comp.*, 31, 1977, pp 333-390.
- A. Brandt, "Multigrid techniques: 1984 guide with applications to computational fluid dynamics," *GMD*, 1984.
- W. Hackbusch, "Multi-Grid Methods & Applications," Springer, 1985.
- W. Hackbusch & U. Trottenberg, "Multigrid Methods", Springer-Verlag, 1982.
- S. McCormick, ed., "Multigrid Methods," SIAM Frontiers in Applied Math. III, 1987.
- U. Trottenberg, C. Oosterlee, & A. Schüller, "Multigrid," Academic Press, 2000.
- P. Wesseling, "An Introduction to Multigrid Methods," Wiley, 1992.

Multilevel methods have been developed for...

- PDEs, CFD, porous media, elasticity, electromagnetics.
- Purely algebraic problems, with no physical grid; for example, network & geodetic survey problems.
- Image reconstruction & tomography.
- Optimization (e.g., the traveling salesman & long transportation problems).
- Statistical mechanics, Ising spin models.
- Quantum chromo dynamics.
- Quadrature & generalized FFTs.
- Integral equations.
-
-
-

Everyone uses multilevel methods

- Multigrid, multilevel, multiscale, multiphysics, ...
Use local "governing rules" on the finest level to resolve the state of the system at these detailed scales, but--recognizing that these "rules" have broader implications that are hard to determine there--use coarser levels to resolve larger scales. Continual feedback is essential because improving one scale impacts other scales.
- Common uses
Sight, art, politics, thinking (scientific research), cooking, team sports, ...

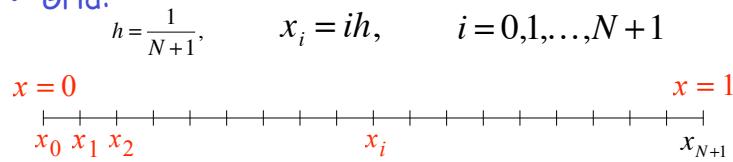
1. Model problems

- 1-D boundary value problem:

$$-u''(x) + \sigma u(x) = f(x) \quad 0 < x < 1, \quad \sigma \geq 0$$

$$u(0) = u(1) = 0$$

- Grid:



- Let $v_i \approx u(x_i)$ & $f_i \approx f(x_i)$ for $i = 0, 1, \dots, N+1$.

This discretizes the variables, but what about the equations?

Approximate $u''(x)$ via Taylor series

- Approximate 2nd derivative using Taylor series:

$$u(x_{i+1}) = u(x_i) + h u'(x_i) + \frac{h^2}{2!} u''(x_i) + \frac{h^3}{3!} u'''(x_i) + O(h^4)$$

$$+ u(x_{i-1}) = u(x_i) - h u'(x_i) + \frac{h^2}{2!} u''(x_i) - \frac{h^3}{3!} u'''(x_i) + O(h^4)$$

- Summing & solving:

$$u''(x_i) = \frac{u(x_{i+1}) - 2u(x_i) + u(x_{i-1})}{h^2} + O(h^2)$$

Approximate equation via finite differences

- Approximate the BVP

$$-u''(x) + \sigma u(x) = f(x) \quad 0 < x < 1, \quad \sigma \geq 0$$

$$u(0) = u(1) = 0$$

by a finite difference scheme:

$$\frac{-v_{i-1} + 2v_i - v_{i+1}}{h^2} + \sigma v_i = f_i \quad i = 1, 2, \dots, N$$

$$v_0 = v_{N+1} = 0$$

Discrete model problem

Letting $\mathbf{v} = (v_1, v_2, \dots, v_N)^T$ &
 $\mathbf{f} = (f_1, f_2, \dots, f_N)^T$

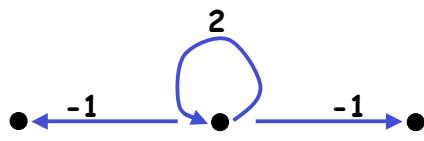
we obtain the matrix equation $\mathbf{Av} = \mathbf{f}$, where \mathbf{A} is $N \times N$, symmetric, positive definite, &

$$\mathbf{A} = \frac{1}{h^2} \begin{pmatrix} 2 + \sigma h^2 & -1 & & & \\ -1 & 2 + \sigma h^2 & -1 & & \\ & -1 & 2 + \sigma h^2 & \ddots & \\ & & \ddots & \ddots & -1 \\ & & & -1 & 2 + \sigma h^2 & -1 \\ & & & & -1 & 2 + \sigma h^2 \end{pmatrix} \quad \mathbf{v} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \\ v_{N-1} \\ v_N \end{pmatrix} \quad \mathbf{f} = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_{N-1} \\ f_N \end{pmatrix}$$

Stencil notation

$$A = [-1 \ 2 \ -1]$$

dropping h^{-2} & σ for convenience



Basic solution methods

- Direct
 - Gaussian elimination
 - Factorization
 - Fast Poisson solvers (FFT-based, reduction-based, ...)
- Iterative
 - Richardson, Jacobi, Gauss-Seidel, ...
 - Steepest Descent, Conjugate Gradients, ...
 - Incomplete Factorization, ...
- Notes:
 - This simple 1-D problem can be solved efficiently in many ways. Pretend it can't & that it's very hard, because it shares many characteristics with some very hard problems. If we keep things as simple as possible by studying this model, we've got a chance to really understand what's going on.
 - But, to keep our feet on the ground, let's go to 2-D anyway...

2-D model problem

- Consider the problem

$$-u_{xx} - u_{yy} + \sigma u = f(x, y), \quad 0 < x < 1, \quad 0 < y < 1$$

$$u = 0 \text{ when } x = 0, x = 1, y = 0, \text{ or } y = 1 \quad \sigma \geq 0$$

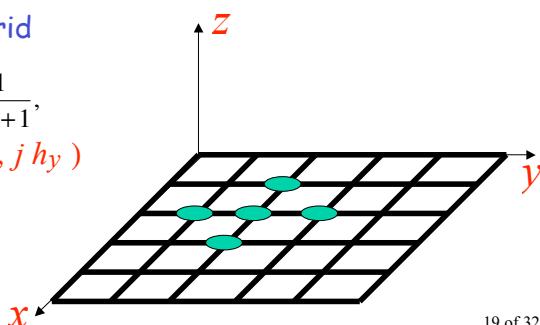
- Consider the grid

$$h_x = \frac{1}{L+1}, \quad h_y = \frac{1}{M+1},$$

$$(x_i, y_j) = (i h_x, j h_y)$$

$$0 \leq i \leq L+1$$

$$0 \leq j \leq M+1$$



Discretizing the 2-D problem

- Let $v_{ij} \approx u(x_i, y_j)$ & $f_{ij} \approx f(x_i, y_j)$. Again, using 2nd-order finite differences to approximate u_{xx} & u_{yy} we arrive at the approximate equation for the unknown $u(x_i, y_j)$, for $i=1,2, \dots, L$ & $j=1,2, \dots, M$:

$$\frac{-v_{i-1,j} + 2v_{ij} - v_{i+1,j}}{h_x^2} + \frac{-v_{i,j-1} + 2v_{ij} - v_{i,j+1}}{h_y^2} + \sigma v_{ij} = f_{ij}$$

$$v_{i,j} = 0 : \quad i = 0, \quad i = L+1, \quad j = 0, \quad j = M+1$$

- Order the unknowns (& also the vector f) lexicographically by y-lines:

$$v = (v_{1,1}, v_{1,2}, \dots, v_{1,M}, v_{2,1}, v_{2,2}, \dots, v_{2,M}, \dots, v_{L,1}, v_{L,2}, \dots, v_{L,M})^T$$

Resulting linear system

- We obtain a block-tridiagonal system $\mathbf{A}\mathbf{v} = \mathbf{f}$:

$$\begin{pmatrix} A_1 & -I_x & & \\ -I_x & A_2 & -I_x & \\ & -I_x & A_3 & \ddots & \\ & & \ddots & \ddots & -I_x \\ & & & -I_x & A_{L-1} & -I_x \\ & & & & -I_x & A_L \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \\ v_{L-1} \\ v_L \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_{L-1} \\ f_L \end{pmatrix}$$

where I_x is the h_x^{-2} times the identity matrix &

$$A_i = \begin{pmatrix} \frac{2}{h_x^2} + \frac{2}{h_y^2} + \sigma & -\frac{1}{h_y^2} & & \\ -\frac{1}{h_y^2} & \frac{2}{h_x^2} + \frac{2}{h_y^2} + \sigma & -\frac{1}{h_y^2} & \\ & -\frac{1}{h_y^2} & \frac{2}{h_x^2} + \frac{2}{h_y^2} + \sigma & \ddots \\ & & \ddots & -\frac{1}{h_y^2} \\ & & & -\frac{1}{h_y^2} & \frac{2}{h_x^2} + \frac{2}{h_y^2} + \sigma & -\frac{1}{h_y^2} \\ & & & & -\frac{1}{h_y^2} & \frac{2}{h_x^2} + \frac{2}{h_y^2} + \sigma \end{pmatrix}$$

CU-Boulder

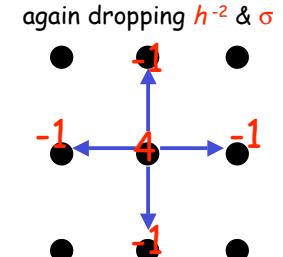
21 of 322

Stencils

preferred for grid issues

Stencils are much better for showing the grid picture:

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & \frac{2}{h_x^2} + \frac{2}{h_y^2} + \sigma & -1 \\ 0 & -1 & 0 \end{bmatrix}$$



again dropping h^{-2} & σ

Stencils show local relationships--grid point interactions.

CU-Boulder

22 of 322

Outline

Chapters 1-5:

- ✓ Model Problems
- Basic Iterative Methods
 - Convergence tests
 - Analysis
- Elements of Multigrid
 - Relaxation
 - Coarsening
- Implementation
 - Complexity
 - Diagnostics
- Some Theory
 - Spectral vs. algebraic

Homework Due

Chapters 6-10:

- Nonlinear Problems
 - Full approximation scheme
- Selected Applications
 - Neumann boundaries
 - Anisotropic problems
 - Variable meshes
 - Variable coefficients
- Algebraic Multigrid (AMG)
 - Matrix coarsening
- Multilevel Adaptive Methods
 - FAC
- Finite Elements
 - Variational methodology

CU-Boulder

23 of 322

2. Basic iterative methods

- Consider the $N \times N$ matrix equation $\mathbf{A}\mathbf{u} = \mathbf{f}$ & let \mathbf{v} be an approximation to \mathbf{u} .

- Two important measures:

- The Error: $\mathbf{e} = \mathbf{u} - \mathbf{v}$ with norms

$$\|\mathbf{e}\|_\infty = \max |e_i| \quad \& \quad \|\mathbf{e}\|_2 = \sqrt{\sum_{i=1}^N e_i^2}$$

- The Residual: $\mathbf{r} = \mathbf{f} - \mathbf{A}\mathbf{v}$ with

$$\|\mathbf{r}\|_\infty \quad \& \quad \|\mathbf{r}\|_2$$

What does $\|\mathbf{r}\|$ measure???

Why have both \mathbf{r} & \mathbf{e} ???

CU-Boulder

24 of 322

Residual correction

- Since $e = u - v$, we can write $Au = f$ as $A(v + e) = f$

which means that $Ae = f - Av \equiv r$.

- Residual Equation:

$$Ae = r$$

- Residual Correction:

$$u = v + e$$

What does this do for us???

Relaxation

- Consider the 1-D model problem

$$-u_{i-1} + 2u_i - u_{i+1} = h^2 f_i \quad 1 \leq i \leq N \quad u_0 = u_{N+1} = 0$$

- Jacobi (simultaneous displacement): Solve the i^{th} equation for v_i holding all other variables fixed:

$$v_i^{(new)} = \frac{1}{2}(v_{i-1}^{(old)} + v_{i+1}^{(old)} + h^2 f_i) \quad 1 \leq i \leq N$$

Jacobi in matrix form

- Let $A = D - L - U$, where D is diagonal & $-L$ & $-U$ are the strictly lower & upper triangular parts of A .
- Then $Au = f$ becomes

$$\begin{aligned} (D - L - U)u &= f \\ Du &= (L + U)u + f \\ u &= D^{-1}(L + U)u + D^{-1}f \end{aligned}$$

- Let $R_J = D^{-1}(L + U)$. $R_J = D^{-1}(D - A) = I - D^{-1}A$
- R_J is called the error propagation or iteration matrix.
- Then the iteration is

$$v^{(new)} = R_J v^{(old)} + D^{-1}f$$

Error propagation matrix & the error

- From the derivation,

$$\begin{aligned} u &= D^{-1}(L + U)u + D^{-1}f \\ u &= R_J u + D^{-1}f \end{aligned}$$

- the iteration is

$$v^{(new)} = R_J v^{(old)} + D^{-1}f$$

- subtracting,

$$u - v^{(new)} = R_J u + D^{-1}f - (R_J v^{(old)} + D^{-1}f)$$

- or

$$u - v^{(new)} = R_J u - R_J v^{(old)}$$

- hence,

Error propagation!

$$e^{(new)} = R_J e^{(old)}$$

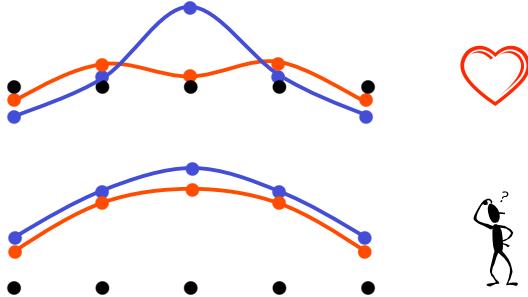
$$R_J = I - D^{-1}A$$

A picture

$$R_J = D^{-1} (L + U) = \begin{bmatrix} \frac{1}{2} & 0 & \frac{1}{2} \\ & \ddots & & \ddots \end{bmatrix}$$

so Jacobi is an error averaging process:

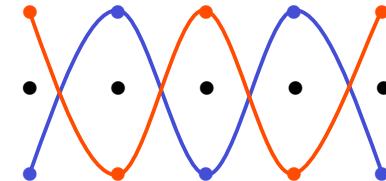
$$e_i^{(\text{new})} \leftarrow (e_{i-1}^{(\text{old})} + e_{i+1}^{(\text{old})})/2$$



CU-Boulder

29 of 322

But...



CU-Boulder

30 of 322

Another matrix look at Jacobi

$$\begin{aligned} v^{(\text{new})} &\leftarrow D^{-1}(L + U)v^{(\text{old})} + D^{-1}f \quad (L + U = D - A) \\ &= (I - D^{-1}A)v^{(\text{old})} + D^{-1}f \end{aligned}$$

$$v^{(\text{new})} = v^{(\text{old})} - D^{-1}(Av^{(\text{old})} - f) = v^{(\text{old})} + D^{-1}r$$

- Exact: $u = u - D^{-1}(Au - f)$
- Subtracting: $e^{(\text{new})} = e^{(\text{old})} - D^{-1}Ae^{(\text{old})}$
- Exact: $u = u - A^{-1}(Au - f) = A^{-1}f$
- General form: $u = u - B(Au - f)$ with $B \sim A^{-1}$
- Damped Jacobi: $u = u - \omega D^{-1}(Au - f)$ with $0 < \omega < 1/\rho(D^{-1}A)$
- Gauss-Seidel: $u = u - (D - L)^{-1}(Au - f)$

Note that $R_\omega = I - \omega D^{-1}A$ is a polynomial in A when $D = I$. We exploit this simplicity (symmetry, etc.) in what follows. !!!This special property doesn't usually hold in practice!!!

Weighted Jacobi

safer ($0 < \omega < 1$) changes

- Consider the iteration

$$v_i^{(\text{new})} \leftarrow (1 - \omega)v_i^{(\text{old})} + \frac{\omega}{2}(v_{i-1}^{(\text{old})} + v_{i+1}^{(\text{old})} + h^2 f_i)$$

- Letting $A = D - L - U$, the matrix form is

$$\begin{aligned} v^{(\text{new})} &= [(1 - \omega)I + \omega D^{-1}(L + U)]v^{(\text{old})} + \omega h^2 D^{-1}f \\ &= R_\omega v^{(\text{old})} + \omega h^2 D^{-1}f \end{aligned}$$

- Note that

$$R_\omega = [(1 - \omega)I + \omega R_J]$$

- It is easy to see that if $e^{(\text{approx})} = u - v^{(\text{approx})}$, then

$$e^{(\text{new})} = R_\omega e^{(\text{old})}$$

CU-Boulder

31 of 322

32 of 322

Gauss-Seidel (1-D)

- Solve equation i for u_i & update immediately.
- Equivalently: set each component of r to zero in turn.
- Component form: for $i = 1, 2, \dots, N$, set

$$v_i \leftarrow \frac{1}{2}(v_{i-1} + v_{i+1} + h^2 f_i)$$

- Matrix form: $A = (D - L - U)$
 $(D - L)u = Uu + f$
 $u = (D - L)^{-1}Uu + (D - L)^{-1}f$
- Let $R_G = (D - L)^{-1}U$ $R_G = (D - L)^{-1}(D - L - A) = I - (D - L)^{-1}A$
- Then iterate: $v^{(new)} \leftarrow R_G v^{(old)} + (D - L)^{-1}f$
- Error propagation: $e^{(new)} \leftarrow R_G e^{(old)}$

CU-Boulder

33 of 322

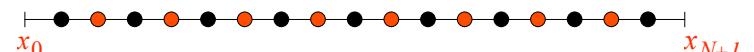
Red-black Gauss-Seidel

- Update the EVEN points:

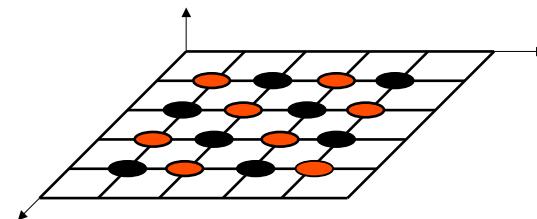
$$v_{2i} \leftarrow \frac{1}{2}(v_{2i-1} + v_{2i+1} + h^2 f_{2i})$$

- Update the ODD points:

$$v_{2i+1} \leftarrow \frac{1}{2}(v_{2i} + v_{2i+2} + h^2 f_{2i+1})$$



- 2-D:



34 of 322

Test?

$$Au = f$$

Need to know how we're doing!!!

- What f ?

$$Au = 0$$

- What v ?

$$v = \text{rand}$$

CU-Boulder

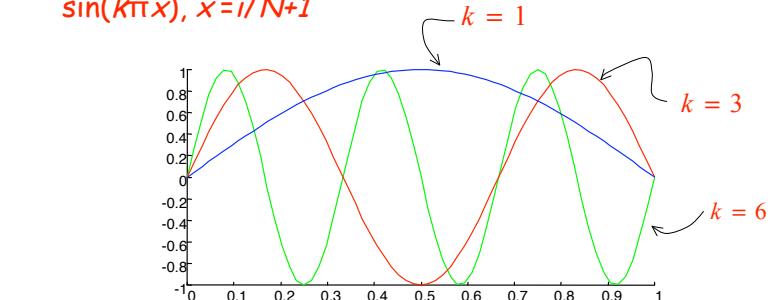
35 of 322

Numerical experiments

- Solve $Au = 0$, $-u_{i-1} + 2u_i - u_{i+1} = 0$
- Use Fourier modes as initial iterates, with $N = 63$:

$$\vec{v}_k = (\vec{v}_i)_k = \left(\sin\left(\frac{ik\pi}{N+1}\right) \right) \quad 1 \leq i \leq N, \quad 1 \leq k \leq N$$

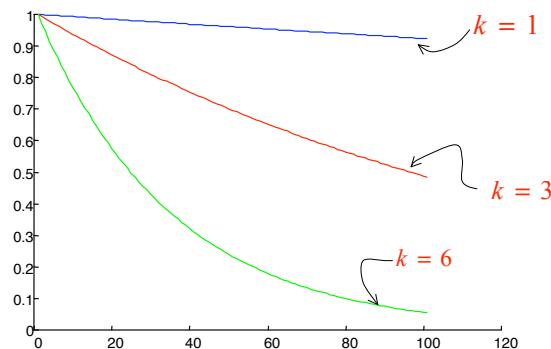
$$\sin(k\pi x), x = i/N+1$$



36 of 322

Convergence factors differ for different error components

Error, $\|e\|_\infty$, in weighted ($\omega=2/3$) Jacobi on $Au=0$ for 100 iterations using initial guesses v_1 , v_3 , & v_6

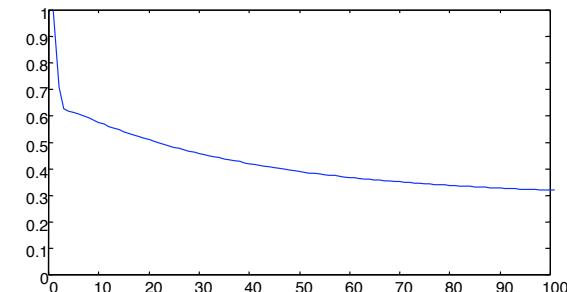


CU-Boulder

37 of 322

Stalling convergence relaxation shoots itself in the foot

- Weighted ($\omega=2/3$) Jacobi on 1-D problem.
- Initial guess: $v = \frac{1}{3} \left[\left(\sin\left(\frac{i\pi}{N+1}\right) \right) + \left(\sin\left(\frac{6i\pi}{N+1}\right) \right) + \left(\sin\left(\frac{32i\pi}{N+1}\right) \right) \right]$
- Error $\|e\|_\infty$ plotted against iteration number:



CU-Boulder

38 of 322

Analysis of stationary linear iteration

- Let $v^{(\text{new})} = Rv^{(\text{old})} + g$. The exact solution is unchanged by the iteration: $u = Ru + g$.
- Subtracting: $e^{(\text{new})} = Re^{(\text{old})}$.
- Let $e^{(0)}$ be the initial error & $e^{(i)}$ be the error after the i^{th} iteration. After n iterations, we have $e^{(n)} = R^n e^{(0)}$.

We can deal with 2^4 , but $\begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix}^4$???

What if $\begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix}^4 e = 2^4 e$???

CU-Boulder

39 of 322

Quick review of eigenvectors & eigenvalues

- The number λ is an eigenvalue of a matrix B & $w \neq 0$ its associated eigenvector if $Bw = \lambda w$.
- The eigenvalues & eigenvectors are characteristics of a given matrix.
- Eigenvectors are linearly independent, & if there is a complete set of N distinct eigenvectors for an $N \times N$ matrix, then they form a basis: for any v , there exist unique scalars v_k such that

$$v = \sum_{k=1}^N v_k w_k$$

- Propagation:

$$B^n v = \sum_{k=1}^N \lambda^k v_k w_k$$

Why is an eigenvector useful???

CU-Boulder

40 of 322

"Fundamental Theorem of Iteration"

R is convergent ($R^n \rightarrow 0$ as $n \rightarrow \infty$) iff

$$\rho(R) = \max |\lambda| < 1.$$

Thus, $e^{(n)} = R^n e^{(0)} \rightarrow 0$ for any initial vector $v^{(0)}$
iff $\rho(R) < 1$.

$\rho(R) < 1$ assures convergence of R iteration.

$\rho(R)$ is the spectral convergence factor.

But ρ doesn't tell you much by itself--it's generally valid only asymptotically. It's useful for the symmetric case in particular because it's equal to $\|R\|_2$, so we'll use it here.

Rayleigh quotient vs. spectral radius

assume A is symmetric (w_k orthonormal) & nonnegative definite ($\lambda \geq 0$)

- $RQ(v) \leq \rho(A)$: $v = \sum v_k w_k$

$$\begin{aligned} RQ(v) &= \frac{\langle Av, v \rangle}{\langle v, v \rangle} = \frac{\langle A \sum v_k w_k, \sum v_k w_k \rangle}{\langle \sum v_k w_k, \sum v_k w_k \rangle} \\ &= \frac{\langle \sum \lambda_k v_k w_k, \sum v_k w_k \rangle}{\langle \sum v_k w_k, \sum v_k w_k \rangle} = \frac{\sum \lambda_k v_k^2}{\sum v_k^2} \leq \lambda_N = \rho(A) \end{aligned}$$

- $\sup_{v \neq 0} RQ(v) = \rho(A)$:

$$RQ(w_N) = \frac{\langle Aw_N, w_N \rangle}{\langle w_N, w_N \rangle} = \frac{\langle \lambda_N w_N, w_N \rangle}{\langle w_N, w_N \rangle} = \lambda_N = \rho(A)$$

Euclidean norm vs. spectral radius

use RQ

- $\|R\|_2 = \rho^{1/2}(R^T R)$:

$$\begin{aligned} \|R\|_2^2 &= \sup_{e \neq 0} \|Re\|_2^2 / \|e\|_2^2 \\ &= \sup_{e \neq 0} \langle Re, Re \rangle / \langle e, e \rangle \\ &= \sup_{e \neq 0} \langle R^T Re, e \rangle / \langle e, e \rangle = \rho(R^T R) \end{aligned}$$

note: $\|Re\|_2 \leq \|R\|_2 \cdot \|e\|_2$

- $\|A\|_2 = \rho^{1/2}(A^2) = \rho(A)$ for symmetric A !!!

$\rho(R)$ vs. $\|R\|_2$

$$\rho(R) = \sup |\lambda(R)|.$$

$\rho(R)$ is norm independent.

$\rho(R)$ is asymptotic:
could get large $\|R\|_2$ but $\rho(R) < 1$
 \Rightarrow "convergence". Next slide.

$$\|R\|_2 = \sup_{e \neq 0} \|Re\|_2 / \|e\|_2 = \rho^{1/2}(R^T R).$$

$\|R\|_2$ depends on $\|\cdot\|_2$.

$\|R\|_2$ bounds $\|e^{(n+1)}\|_2 / \|e^{(n)}\|_2$:
worst case! Probably pessimistic initially, but sharp sooner or later.

$$\rho(R) = \inf \|R\|$$

taken over all matrix norms.

$$\rho(R) = \|R\|_2$$

for symmetric R .

Example: $R = \begin{pmatrix} 0 & K \\ 0 & 0 \end{pmatrix}$, K large

$\rho(R) = 0$ but $\|R\|_2 = K!$

$$e^{(0)} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \Rightarrow \frac{\|e^{(1)}\|_2}{\|e^{(0)}\|_2} = \frac{\|Re^{(0)}\|_2}{\|e^{(0)}\|_2} = K$$

Thus, 1 iteration with $e^{(0)}$ shows dramatic L^2 divergence!

But $R^2 = 0$, so $e^{(2)} = Re^{(1)} = R^2 e^{(0)} = 0!$

Thus, 2 iterations with $e^{(0)}$ show complete convergence!

On one hand, this is special ($\lambda=0$, large K , 2x2), so this behavior would be more subtle & persistent in general.

On the other, this behavior would vanish for symmetric R .

Convergence factor & rate

- How many iterations are enough to guarantee reduction of the initial error by 10^{-d} ?

$$\frac{\|e^{(n)}\|}{\|e^{(0)}\|} \leq \|R^n\| \leq \|R\|^n \sim 10^{-d}$$

- So, we have

$$n \sim \frac{d}{-\log_{10} \rho(R)}$$

- Convergence factor = $\|R\|$ or $\rho(R)$.

- Convergence rate = $-\log_{10} \|R\|$ $\frac{\text{digits}}{\text{iteration}}$ or $-\log_{10}(\rho(R))$.

Convergence analysis: Weighted Jacobi

$$\begin{aligned} R_\omega &= (1-\omega)I + \omega D^{-1}(L+U) \\ &= I - \omega D^{-1}A \\ R_\omega &= I - \frac{\omega}{2} \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & 2 & -1 & \\ & & \ddots & \ddots & \ddots \\ & & & -1 & 2 \end{pmatrix} \quad \text{1-D} \end{aligned}$$

For our 1-D model, the eigenvectors of weighted Jacobi R_ω & the eigenvectors of A are the same! Why???

$$\lambda(R_\omega) = 1 - \frac{\omega}{2} \lambda(A) \quad \text{This is very special!!!}$$

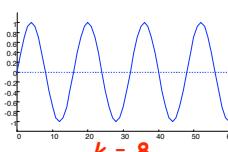
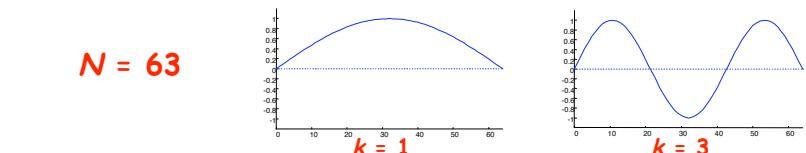
Eigenpairs of (scaled) A

The eigenvectors of A are Fourier modes!

$$\lambda_k(A) = 4 \sin^2 \left(\frac{k\pi}{2(N+1)} \right), \quad w_{k,i} = \sin \left(\frac{ik\pi}{N+1} \right)$$

$$\begin{matrix} [-1 & 2 & -1] \\ \lambda_N \approx 4 \\ \lambda_1 \approx \pi^2 h^2 \end{matrix}$$

$N = 63$



Eigenvectors of R_ω = eigenvectors of A

$$\lambda_k(R_\omega) = 1 - 2\omega \sin^2\left(\frac{k\pi}{2(N+1)}\right)$$

- Expand the initial error in terms of the eigenvectors:

$$e^{(0)} = \sum_{k=1}^N c_k w_k$$

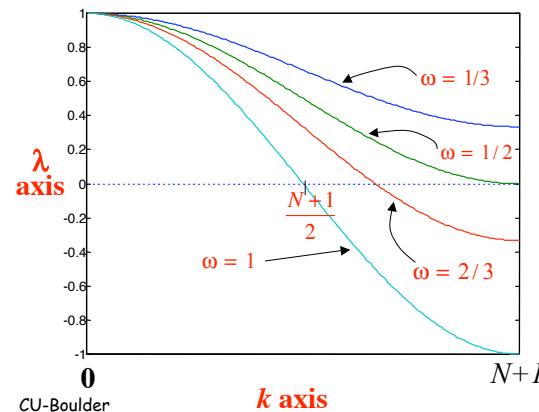
- After n iterations:

$$R^n e^{(0)} = \sum_{k=1}^N c_k \lambda_k^n w_k$$

- The k^{th} error mode is reduced by $\lambda_k(R_\omega)$ each iteration.

Relaxation suppresses eigenmodes unevenly

- Look carefully at $\lambda_k(R_\omega) = 1 - 2\omega \sin^2\left(\frac{k\pi}{2(N+1)}\right)$.



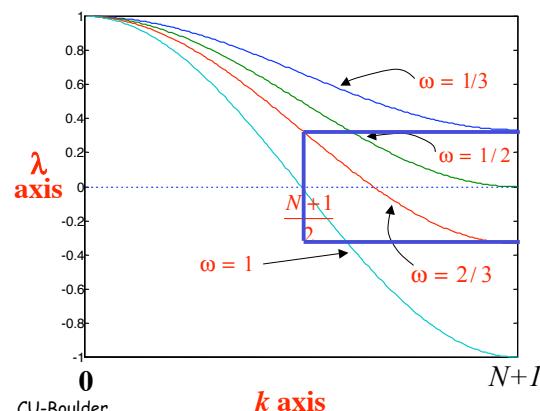
Note that if $0 < \omega \leq 1$, then $|\lambda_k(R_\omega)| < 1$ for $k = 1, 2, \dots, N$.

For $0 < \omega \leq 1$,

$$\begin{aligned}\lambda_k &= 1 - 2\omega \sin^2\left(\frac{\pi k}{2(N+1)}\right) \\ &= 1 - 2\omega \sin^2\left(\frac{\pi h}{2}\right) \\ &= 1 - O(h^2) \approx 1\end{aligned}$$

Low frequencies are “undamped”

Notice that no value of ω will efficiently damp out long waves or low frequencies.



What value of ω gives the best damping of short waves or high frequencies $N/2 \leq k \leq N$

Choose ω such that

$$\lambda_{N/2}(R_\omega) = -\lambda_N(R_\omega)$$

$$\frac{\lambda_N}{2} = -\lambda_{N/2}$$

$$\Rightarrow \omega \approx \frac{2}{3}$$

Smoothing factor

- The smoothing factor is the largest magnitude of the iteration matrix eigenvalues corresponding to the oscillatory Fourier modes:

$$\text{smoothing factor} = \max |\lambda_k(R)| \quad \text{for } N/2 \leq k \leq N.$$

- Why only the upper spectrum? “MG” spectral radius?

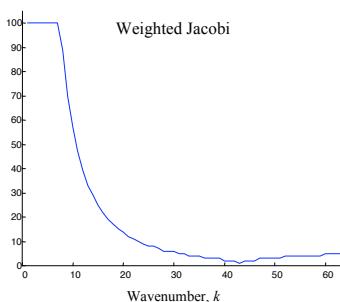
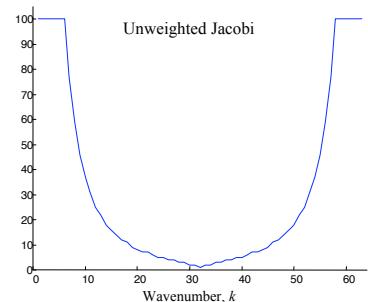
- For R_ω with $\omega=2/3$, the smoothing factor is $1/3$:

$$|\lambda_{N/2}| = |\lambda_N| = 1/3 \quad \& \quad |\lambda_k| < 1/3 \quad \text{for } N/2 < k < N.$$

- But $|\lambda_k| \approx 1 - \omega k^2 \pi^2 h^2$ for long waves ($k \ll N/2$).



Convergence of Jacobi on $Au=0$



- Jacobi on $Au=0$ with $N=63$. Number of iterations needed to reduce initial error $\|e\|_\infty$ by 0.01.

- Initial guess :

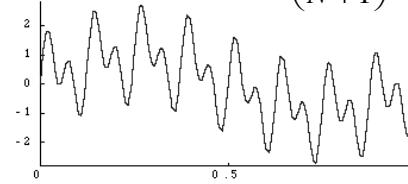
$$v_{k,i} = \sin\left(\frac{ik\pi}{N+1}\right)$$

CU-Boulder

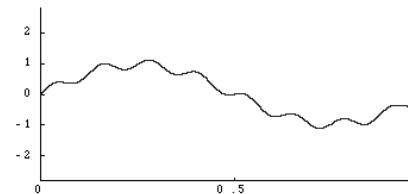
53 of 322

Weighted Jacobi = smoother (error)

- Initial error: $v_i = \sin\left(\frac{2i\pi}{N+1}\right) + \frac{1}{2}\sin\left(\frac{16i\pi}{N+1}\right) + \frac{1}{2}\sin\left(\frac{32i\pi}{N+1}\right)$



- Error after 35 iteration sweeps:



Many relaxation schemes are smoothers: oscillatory error modes are quickly eliminated, but smooth modes are slowly damped.

CU-Boulder

CU-Boulder

54 of 322

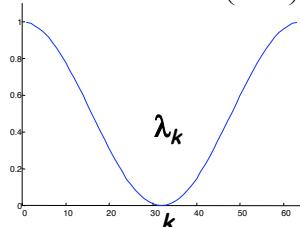
Similar analysis for other smoothers

- Gauss-Seidel relaxation applied to the 3-point difference matrix A (1-D model problem):

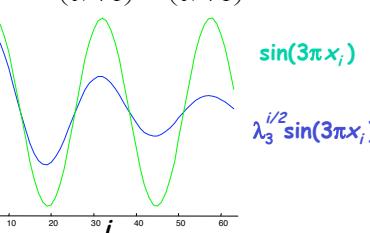
$$R_G = (D-L)^{-1}U$$

- A little algebra & trigonometry shows that

$$\lambda_k(R_G) = \cos^2\left(\frac{k\pi}{N+1}\right)$$



$$(w_k)_i = \cos^j\left(\frac{k\pi}{N+1}\right) \sin\left(\frac{ik\pi}{N+1}\right)$$

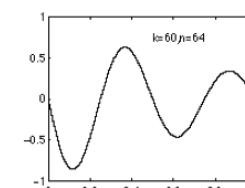
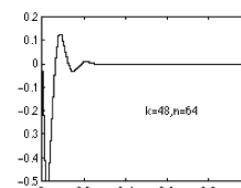
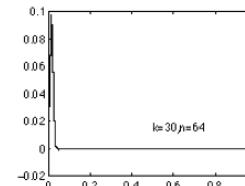
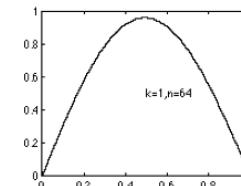


What's w_k look like for large k ?

CU-Boulder

55 of 322

Gauss-Seidel eigenvectors



These are VERY different from Jacobi's eigenvectors.
It's not clear how smoothness depends on k .

You cannot expect G-S to quickly reduce Fourier modes.

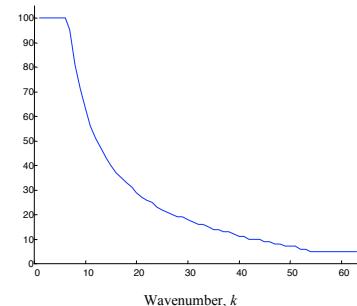
You can only expect G-S to produce smooth results!

56 of 322

Gauss-Seidel convergence

$$Au = 0$$

Eigenvectors of R_G are not the same as those of A !!!
Gauss-Seidel mixes the modes of A .



Gauss-Seidel on $Au = 0$, with $N = 63$. Number of iterations needed to reduce initial error $\|u\|_\infty$ by 0.01.

Initial guess (modes of A):

$$v_{ki} = \sin\left(\frac{ik\pi}{N+1}\right)$$

OK, so G-S does reduce oscillatory Fourier modes.
But you will see that this depends on ordering.

In fact, red-black G-S does not reduce high Fourier modes.

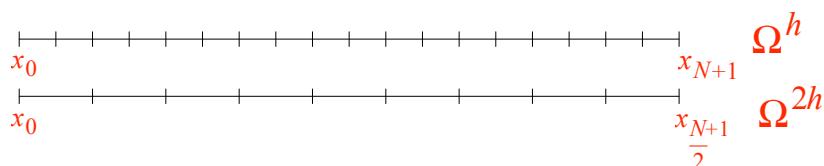
CU-Boulder

57 of 322

3. Elements of multigrid

1st observation toward multigrid

- Many relaxation schemes have the smoothing property: oscillatory error modes are quickly eliminated, while smooth modes are often very slow to disappear.
- We'll turn this adversity around: the idea is to use coarse grids to take advantage of smoothing.



How?

CU-Boulder

59 of 322

Outline

Chapters 1-5:

- ✓ Model Problems
- ✓ Basic Iterative Methods
 - Convergence tests
 - Analysis
- Elements of Multigrid
 - Relaxation
 - Coarsening
- Implementation
 - Complexity
 - Diagnostics
- Some Theory
 - Spectral vs. algebraic

Homework Due
Computing Too

CU-Boulder

Chapters 6-10:

- Nonlinear Problems
 - Full approximation scheme
- Selected Applications
 - Neumann boundaries
 - Anisotropic problems
 - Variable meshes
 - Variable coefficients
- Algebraic Multigrid (AMG)
 - Matrix coarsening
- Multilevel Adaptive Methods
 - FAC
- Finite Elements
 - Variational methodology

58 of 322

Reason #1 for coarse grids: Nested iteration

- Coarse grids can be used to compute an improved initial guess for the fine-grid relaxation. This is advantageous because:

- Relaxation on the coarse-grid is much cheaper: half as many points in 1-D, one-fourth in 2-D, one-eighth in 3-D,...
- Relaxation on the coarse grid has a marginally faster convergence factor ($|\lambda_1(R)| \approx 1 - \omega\pi^2 h^2$):

$$1 - O(4h^2) \quad \text{instead of} \quad 1 - O(h^2)$$

CU-Boulder

60 of 322

Idea!

Nested iteration

⋮

- Relax on $Au = f$ on Ω^{4h} to obtain initial guess v^{2h} .
- Relax on $Au = f$ on Ω^{2h} to obtain initial guess v^h .
- Relax on $Au = f$ on Ω^h to obtain ... final solution???
- What is $A^{2h}u^{2h} = f^{2h}$?
Analogous to $Auh = fh$ for now.
- How do we migrate between grids?
Hang on...
- What if the error still has large smooth components when we get to the fine grid Ω^h ?
Hang on...

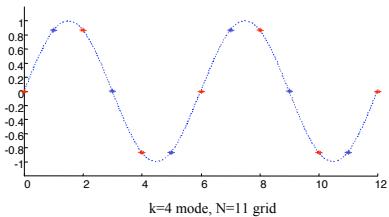
CU-Boulder

61 of 322

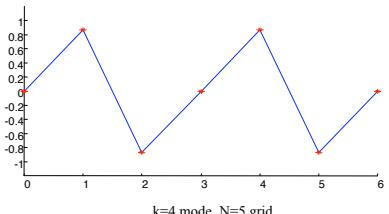
For $k=1,2,\dots,(N-1)/2$, the k^{th} mode is preserved on the coarse grid.

$$w_{k,2i}^h = \sin\left(\frac{2ik\pi}{N+1}\right) = \sin\left(\frac{ik\pi}{(N+1)/2}\right) = w_{k,i}^{2h}$$

Also, note that
 $w_{(N+1)/2}^h \rightarrow 0$
on the coarse grid.



What happens to the modes between $(N+1)/2$ & N ?

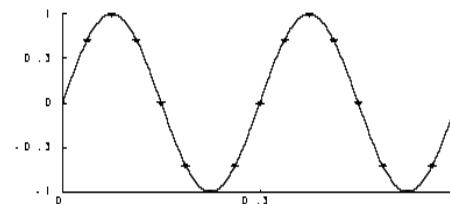


CU-Boulder

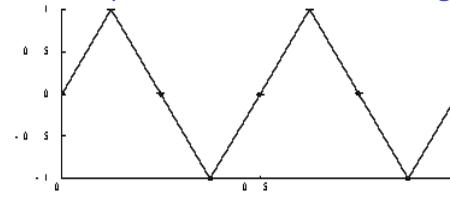
63 of 322

Reason #2 for coarse grids:
Smooth error becomes more oscillatory

- A smooth function:



can be represented by linear interpolation from a coarser grid:



CU-Boulder

On the coarse grid, the smooth error appears to be relatively higher in frequency: in this example it is the 4-mode out of a possible 15 on the fine grid, ~1/4 the way up the spectrum. On the coarse grid, it is the 4-mode out of a possible 7, ~1/2 the way up the spectrum.

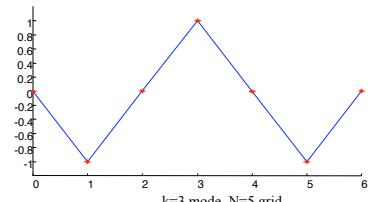
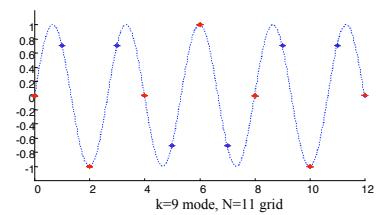
Relaxation on $2h$ is (cheaper &) faster on this mode!!!

62 of 322

For $k>(N+1)/2$, w_k^h is disguised on the coarse grid: aliasing!!!

For $k>(N+1)/2$, the k^{th} mode on the fine grid is aliased & appears as the $(N+1-k)^{\text{th}}$ mode on the coarse grid:

$$\begin{aligned} (w_k^h)_{2i} &= \sin\left(\frac{(2i)\pi k}{N+1}\right) \\ &= -\sin\left(\frac{2i\pi(N+1-k)}{N+1}\right) \\ &= -\sin\left(\frac{i\pi(N+1-k)}{(N+1)/2}\right) \\ &= -(w_{N+1-k}^{2h})_i \end{aligned}$$

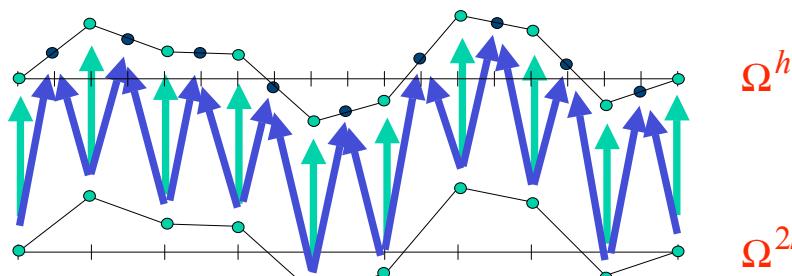


CU-Boulder

64 of 322

1-D interpolation (prolongation)

- Values at points on the coarse grid map unchanged to the fine grid.
- Values at fine-grid points NOT on the coarse grid are the averages of their coarse-grid neighbors.



CU-Boulder

65 of 322

1-D interpolation (prolongation) to migrate from coarse to fine grids

- Mapping from the coarse grid to the fine grid:

$$I_{2h}^h : \Omega^{2h} \rightarrow \Omega^h$$

- Let v^h , v^{2h} be defined on Ω^h , Ω^{2h} . Then

$$I_{2h}^h v^{2h} = v^h$$

where

$$v_{2i}^h = v_i^{2h} \quad \text{for } 0 \leq i \leq \frac{N+1}{2} \text{ (including boundaries).}$$

$$v_{2i+1}^h = \frac{1}{2}(v_i^{2h} + v_{i+1}^{2h}) \quad \text{for } 0 \leq i \leq \frac{N-1}{2}.$$

CU-Boulder

66 of 322

1-D prolongation operator P

- $P = I_{2h}^h$ is a linear operator: $\mathcal{R}^{(N-1)/2} \rightarrow \mathcal{R}^N$.

- $N=7$:

$$\begin{pmatrix} 1/2 & & & \\ 1 & & & \\ 1/2 & 1/2 & & \\ 1 & & & \\ 1/2 & 1/2 & & \\ 1 & & & \\ 1/2 & & & \end{pmatrix}_{7 \times 3} \begin{pmatrix} v_1^{2h} \\ v_2^{2h} \\ v_3^{2h} \end{pmatrix}_{3 \times 1} = \begin{pmatrix} v_1^h \\ v_2^h \\ v_3^h \\ v_4^h \\ v_5^h \\ v_6^h \\ v_7^h \end{pmatrix}_{7 \times 1}$$

- I_{2h}^h has full rank, so $\eta(P) = \{0\}$. rank = max # linearly independent cols or rows

When is $I_{2h}^h v^{2h} = 0$?

CU-Boulder

67 of 322

"Give To" stencil for P

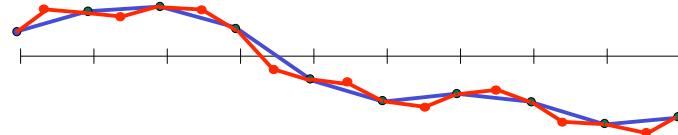
$$[\begin{matrix} 1/2 & 1 & 1/2 \end{matrix}] \\ \begin{matrix} 0 & \times & 1/2 & 1 & 1/2 & \times & 0 \end{matrix}$$

CU-Boulder

68 of 322

How well could v^{2h} approximate u ?

- Imagine that a coarse-grid approximation v^{2h} has been found. How well **could** it approximate the exact solution u ?



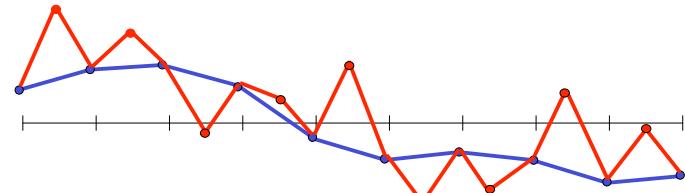
- If u is smooth, a coarse-grid interpolant v^{2h} **might** do very well.

Moral

- If what we want to compute is **smooth**, a coarse-grid interpolant **could** do very well.
- If what we want to compute is **oscillatory**, a coarse-grid interpolant **cannot** do very well.
- What if u is not smooth? Can we make it so?
- Can we make something smooth?
- How about e ? Can we smooth it? How would we get e & use it to get u ? $Ae = r$ & $u \leftarrow v + e$!
- Thus, use **nested iteration** on residual equation to approximate the error after smoothing!!!
- Just because the coarse grid **can** approximate e well doesn't mean we know **how** to do it! But we will soon!

How well could v^{2h} approximate u ?

- Imagine that a coarse-grid approximation v^{2h} has been found. How well **could** it approximate the exact solution u ?



- If u is oscillatory, a coarse-grid interpolant v^{2h} **cannot** work well.

2nd observation toward multigrid

- The residual equation:** Let v be an approximation to the solution of $Au = f$, where the residual $r = f - Av$. Then the error $e = u - v$ satisfies $Ae = r$.
- After relaxing on $Au = f$ on the fine grid, e will be smooth, so the coarse grid can approximate e well. This will be cheaper & e should be more oscillatory there, so relaxation will be more effective.
- Therefore, we go to a coarse grid & relax on the residual equation $Ae = r$.

What's a good initial guess on grid $2h$? $e = 0$!

How do we get to grid $2h$?

Stay tuned...



Coarse-grid correction 2-grid

- Relax on $Au = f$ on Ω^h to get an approximation v^h .
- Compute $r = f - Av^h$.
- Transfer $Ae = r$ to Ω^{2h} somehow & relax on it to obtain an approximation to the error, e^{2h} .
- Correct the approximation $v^h \leftarrow v^h + I_{2h}^h e^{2h}$.

This is the essence of multigrid.

We need a way to transfer $Ae = r$ to Ω^{2h} .

A way to coarsen $Ae = r$

- Assume we've relaxed so much that e is smooth.
- Ansatz: $e = Pv^{2h}$ for some coarse-grid v^{2h} .
- How do we characterize e so we can hope to compute it?

$$Ae = r \quad \Rightarrow \quad A P v^{2h} = r$$

$7 \times 7 \quad 7 \times 3 \quad 3 \times 1 = 7 \times 1$

- Too many equations now & too few unknowns!
- How about just eliminating every other equation?
- How about multiplying both sides by some 3×7 matrix?

$$(P^T \begin{smallmatrix} A^{2h} \\ 3 \times 3 \end{smallmatrix} P) v^{2h} = P^T r$$

$3 \times 7 \quad 7 \times 7 \quad 7 \times 3 \quad 3 \times 1 = 3 \times 1$

1-D restriction by injection

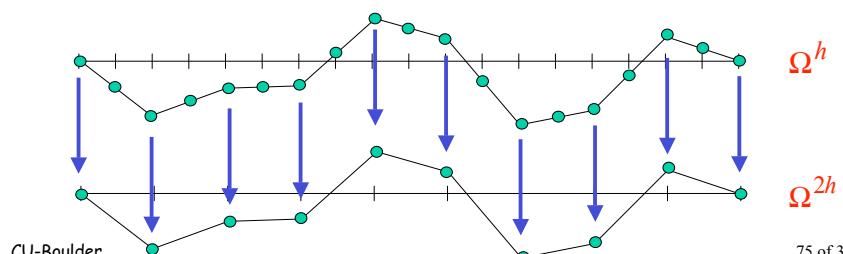
- Mapping from the fine grid to the coarse grid:

$$I_h^{2h} : \Omega^h \rightarrow \Omega^{2h}$$

- Let v^h, v^{2h} be defined on Ω^h, Ω^{2h} . Then

$$I_h^{2h} v^h = v^{2h}$$

where $v_i^{2h} = v_{2i}^h$.



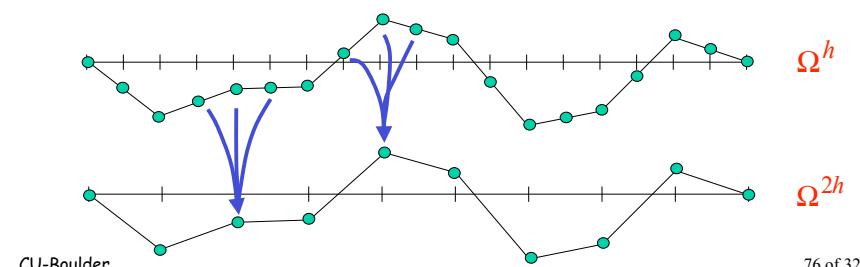
1-D restriction by full weighting

- Let v^h, v^{2h} be defined on Ω^h, Ω^{2h} . Then

$$I_h^{2h} v^h = v^{2h}$$

where

$$v_i^{2h} = \frac{1}{4}(v_{2i-1}^h + 2v_{2i}^h + v_{2i+1}^h).$$



1-D restriction (full-weighting)

- $R = I_h^{2h}$ is a linear operator: $\Re^N \rightarrow \Re^{(N-1)/2}$.

- $N = 7$:

$$\begin{pmatrix} 1/4 & 1/2 & 1/4 \\ & 1/4 & 1/2 & 1/4 \\ & & 1/4 & 1/2 & 1/4 \end{pmatrix} \begin{pmatrix} v_1^h \\ v_2^h \\ v_3^h \\ v_4^h \\ v_5^h \\ v_6^h \\ v_7^h \end{pmatrix} = \begin{pmatrix} v_1^{2h} \\ v_2^{2h} \\ v_3^{2h} \end{pmatrix}.$$

- I_h^{2h} has rank $\frac{N-1}{2}$, so $\dim(\eta(R)) = \frac{N+1}{2}$.

Look at the columns of R associated with grid $2h$.

Prolongation & restriction are often nicely related

- For the 1-D examples, linear interpolation & full weighting are

$$I_{2h}^h = \frac{1}{2} \begin{pmatrix} 1 \\ 2 \\ 1 & 1 \\ 2 \\ 1 & 1 \\ 2 \\ 1 \end{pmatrix} \quad I_h^{2h} = \frac{1}{4} \begin{pmatrix} 1 & 2 & 1 \\ 1 & 2 & 1 \\ 1 & 2 & 1 \end{pmatrix}$$

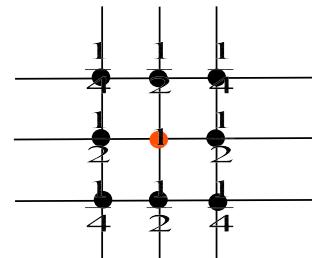
- So they're related by the variational condition

$$I_{2h}^h = c (I_h^{2h})^T \quad \text{for } c \text{ in } \Re.$$

2-D prolongation

$$\begin{aligned} v_{2i,2j}^h &= v_{ij}^{2h} \\ v_{2i+1,2j}^h &= \frac{1}{2}(v_{ij}^{2h} + v_{i+1,j}^h) \\ v_{2i,2j+1}^h &= \frac{1}{2}(v_{ij}^{2h} + v_{i,j+1}^h) \\ v_{2i+1,2j+1}^h &= \frac{1}{4}(v_{ij}^{2h} + v_{i+1,j}^h + v_{i,j+1}^h + v_{i+1,j+1}^h) \end{aligned}$$

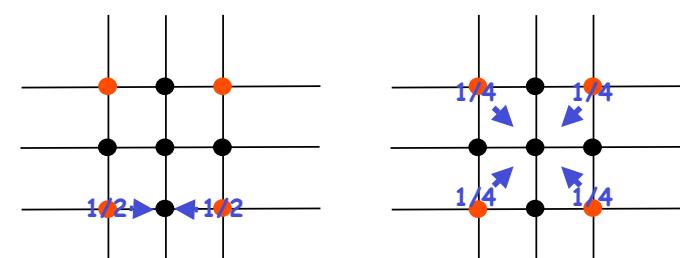
We denote the operator by using a "give to" stencil []. Centered over a C-point ●, it shows what fraction of the C-point's value contributes to a neighboring F-point ●.



"Get From" interpolation

$$\begin{bmatrix} 1/4 & 1/4 \\ 1/2 & 1/2 \\ 1/4 & 1/4 \end{bmatrix}$$

Centered over a fine-grid point ●.

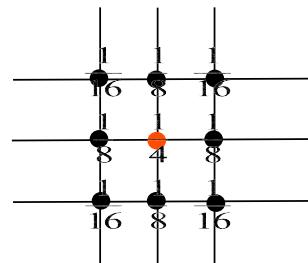


2-D restriction (full weighting)

$$\begin{bmatrix} \frac{1}{16} & \frac{1}{8} & \frac{1}{16} \\ \frac{1}{8} & \frac{1}{4} & \frac{1}{8} \\ \frac{1}{16} & \frac{1}{8} & \frac{1}{16} \end{bmatrix}$$

We denote the operator by using a "get from" stencil []. Centered over a C-point ●, it shows what fraction of the value of the neighboring F-point ● contributes to the value at the C-point.

CU-Boulder



81 of 322

Now we put all these ideas together

- Nested Iteration
 - effective on smooth solution (components).
- Relaxation
 - effective on oscillatory error (components).
- Residual Equation
 - characterizes the error.
 - enables nested iteration for smooth error (components)!!!
- Prolongation (variables) & Restriction (equations)
 - provides pathways between coarse & fine grids.

CU-Boulder

82 of 322

2-grid coarse-grid correction

$$\underline{v^h \leftarrow CG(v^h, f^h), \alpha_1, \alpha_2}$$

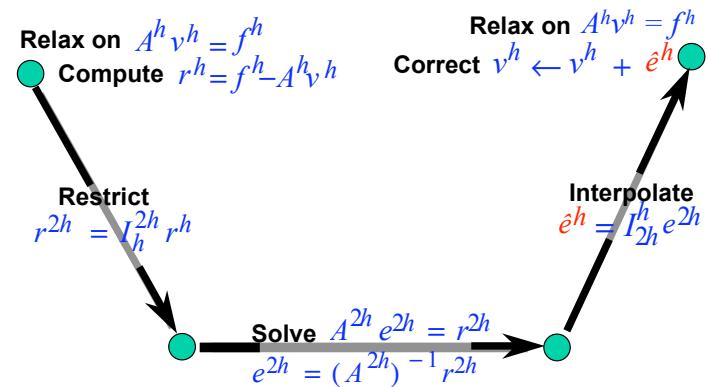
- 1) Relax α_1 times on $A^h v^h = f^h$ on Ω^h with arbitrary initial guess v^h . If $h = h_{\text{coarsest}}$, then go to 6.
- 2) Compute $r^h = f^h - A^h v^h$.
- 3) Compute $r^{2h} = I_h^{2h} r^h$.
- 4) "Solve" $A^{2h} e^{2h} = r^{2h}$ on Ω^{2h} .
- 5) Correct fine-grid solution $v^h \leftarrow v^h + I_{2h}^h e^{2h}$.
- 6) Relax α_2 times on $A^h v^h = f^h$ on Ω^h .

What is e^{2h} here?

CU-Boulder

83 of 322

2-grid coarse-grid correction



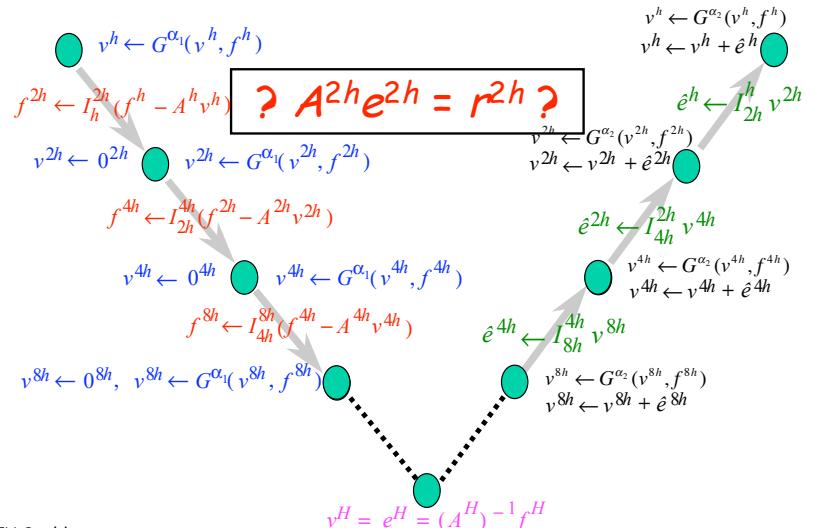
CU-Boulder

84 of 322

What is A^{2h} ?

- For this scheme to work, we must have A^{2h} , a coarse-grid operator. For the moment, we will simply assume that A^{2h} is "the coarse-grid version" of the fine-grid operator A^h .
- Later we'll return to the question of constructing A^{2h} .

How do we "solve" the coarse-grid residual equation? Recursion!



V-cycle (recursive form)

$$\underline{v^h \leftarrow M V^h (v^h, f^h), \alpha_1, \alpha_2}$$

- Relax α_1 times on $A^h u^h = f^h$ with initial v^h given.
- If Ω^h is the coarsest grid, go to 4;
else:

$$f^{2h} \leftarrow I^{2h}_h(f^h - A^h v^h)$$

$$v^{2h} \leftarrow 0$$

$$v^{2h} \leftarrow M V^{2h} (v^{2h}, f^{2h})$$
- Correct: $v^h \leftarrow v^h + I^{2h}_{2h} v^{2h}$.
- Relax α_2 times on $A^h u^h = f^h$ with initial guess v^h .

Outline

Chapters 1-5:

- ✓ Model Problems
- ✓ Basic Iterative Methods
 - Convergence tests
 - Analysis
- ✓ Elements of Multigrid
 - Relaxation
 - Coarsening
- Implementation
 - Complexity
 - Diagnostics
- Some Theory
 - Spectral vs. algebraic

Chapters 6-10:

- Nonlinear Problems
 - Full approximation scheme
- Selected Applications
 - Neumann boundaries
 - Transonic problems
 - Variable meshes
 - Variable coefficients
- Algebraic Multigrid (AMG)
 - Matrix coarsening
- Multilevel Adaptive Methods
 - FAC
- Finite Elements
 - Variational methodology

Homework Due

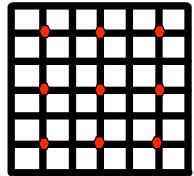
4. Implementation

Storage cost: v^h & f^h on each level.

- In 1-D, a coarse grid has about half as many points as the fine grid.



- In 2-D, a coarse grid has about one-fourth as many points as the fine grid.



- In d-dimensions, a coarse grid has about 2^{-d} as many points as the fine grid.

- Total storage cost: $N^d (1 + 2^{-d} + 2^{-2d} + 2^{-3d} + \dots + 2^{-Md}) < \frac{N^d}{1 - 2^{-d}}$ less than 2, 4/3, & 8/7 the cost of storage on the fine grid for 1-D, 2-D, & 3-D problems, respectively.

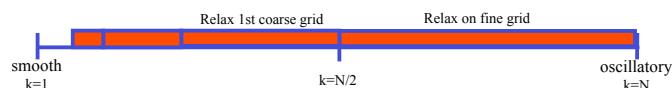
Computational cost

- Let one Work Unit (WU) be the cost of one relaxation sweep on the fine grid.
- Ignore the cost of restriction & interpolation (typically about 20% of the total cost).
- Consider a V-cycle with 2 pre-coarse-grid correction sweep ($\alpha_1 = 2$) & 1 post-coarse-grid correction sweep ($\alpha_2 = 1$).
- Cost of a V-cycle (in WUs):

$$3(1 + 2^{-d} + 2^{-2d} + 2^{-3d} + \dots + 2^{-Md}) < \frac{3}{1 - 2^{-d}}$$
- Cost is about 2, 4/3, & 8/7 X 3 WUs per V-cycle in 1, 2, & 3 dimensions, respectively.

Convergence analysis

- First, a heuristic argument:
 - The convergence factor for the oscillatory error modes (smoothing factor) is small & bounded uniformly in h .
smoothing factor = $\max |\lambda_k(R)|$ for $N/2 \leq k \leq N$.
 - Multigrid focuses the relaxation process on attenuating the oscillatory components on each level.



⇒ The overall multigrid convergence factor is small & bounded uniformly in h !

Bounded uniformly in $h \neq$ independent of h .

Revisiting the model problem

- The BVP:

$$-u''(x) = f(x) \quad 0 < x < 1, \quad \sigma \geq 0$$

$$u(0) = u(1) = 0$$
- The finite difference scheme:

$$\frac{-v_{i-1} + 2v_i - v_{i+1}}{h^2} = f_i \quad i = 1, 2, \dots, N$$

$$v_0 = v_{N+1} = 0$$
- Truncation error:

$$\frac{-u_{i-1}^{(h)} + 2u_i^{(h)} - u_{i+1}^{(h)}}{h^2} + O(h^2) = f_i \quad i = 1, 2, \dots, N$$

$$u^{(h)} = \text{exact PDE solution vector}$$

Convergence analysis

a little more precisely...

- Continuous problem: $Au = f, u_i = u(x_i)$
- Discrete problem: $A^h u^h = f^h, v^h \approx u^h$
- Global/discretization error: $E_i = u(x_i) - u_i^h$
 $\|E\| \leq K h^p$
 $(p=2 \text{ for model problem \& proper norm})$
- Algebraic error: $e^h = u^h - v^h$
- For tolerance ε , assume the aim is to find v^h so that the total error, $\|e\| = \|u^h - v^h\| \leq \varepsilon$, where $u^h = (u(x_i))$.
- Then this objective is assured as follows:
 $\|e\| \leq \|u^h - u^h\| + \|u^h - v^h\| = \|E\| + \|e^h\| \leq \varepsilon$.

$$\begin{aligned}
 Au^{(h)} + O(h^2) &= f \\
 \Rightarrow Au^{(h)} &= f + O(h^2) \\
 \Rightarrow Au &= f \quad (u = \text{discrete sol'n}) \\
 \Rightarrow A(u^{(h)} - u) &= O(h^2) \\
 \Rightarrow AE &= O(h^2).
 \end{aligned}$$

$E = \text{discretization error}$

So $\|E\| = \|A^{-1}O(h^2)\|$

$$\begin{aligned}
 &\leq \|A^{-1}\| \cdot \|O(h^2)\| \\
 &= \lambda_{\max}(A^{-1}) \cdot O(h^2) = O(h^2)/\lambda_{\min}(A)
 \end{aligned}$$

or

$$\|E\| = O(h^2).$$

We can satisfy the convergence objective by imposing two conditions

- 1) $\|E\| \leq \varepsilon/2$. Achieve this condition by choosing an appropriately small grid spacing h :

$$Kh^p = \varepsilon/2.$$

- 2) $\|e^h\| \leq \varepsilon/2$. Achieve this condition by iterating until $\|e^h\| \leq \varepsilon/2 = Kh^p$ on grid h ; then we've

converged to the level of discretization error.

- Once discretization error & algebraic error are in balance, then it would be better to go to grid $h/2$ than to iterate more!

Convergence to the level of discretization error

- Use an MV scheme with convergence factor $\gamma < 1$ bounded uniformly in h (fixed α_1 & α_2).
- Assume a d -dimensional problem on an $N \times N \times \dots \times N$ grid with $h = N^{-1}$.
- Initial error is $\|e^h\| = \|u^h - 0\| = \|u^h\| = O(1)$.
- Must reduce this to $\|e^h\| = O(h^p) = O(N^{-p})$.
- We can determine the number of V-cycles needed for this, if we can bound the convergence factor, γ .

Work to converge to the level of discretization error

- Using θ V-cycles with convergence factor γ gives an overall convergence factor of γ^θ .
- We therefore have $\gamma^\theta = O(N^{-p})$, or $\theta = O(\log N)$.
- Since 1 V-cycle costs $O(1)$ WUs & 1 WUs is $O(N^d)$, then converging to the level of discretization error using the MV method cost $O(N^d \log N)$.
- This compares to fast direct methods (fast Poisson solvers). But multigrid can do even better...

CU-Boulder

97 of 322

Numerical example

- Consider the 2-D model problem (with $\sigma = 0$):

$$u_{xx} - u_{yy} = 2[(1 - 6x^2)y^2(1 - y^2) + (1 - 6y^2)x^2(1 - x^2)]$$
in the unit square, with $u = 0$ Dirichlet boundary.
- The solution to this problem is

$$u(x, y) = -(x^4 - x^2)(y^4 - y^2).$$
- We examine effectiveness of MV cycling to solve this problem on $(N+1) \times (N+1)$ grids [$(N-1) \times (N-1)$ interior points] for $N = 16, 32, 64, 128$.

CU-Boulder

98 of 322

| V-cycle | $n = 16$ | | | | $n = 32$ | | | | |
|----------|-------------|-------|-----------|-----------|-------------|-------|-----------|-------|--|
| | $\ r^k\ _h$ | ratio | $\ e\ _h$ | ratio | $\ r^k\ _h$ | ratio | $\ e\ _h$ | ratio | |
| 0 | 6.75e+02 | | 5.45e-01 | | 2.60e+03 | | 5.61e-01 | | |
| 1 | 4.01e+00 | 0.01 | 1.05e-02 | 0.02 | 1.97e+01 | 0.01 | 1.38e-02 | 0.02 | |
| 2 | 1.11e-01 | 0.03 | 4.10e-04 | 0.04 | 5.32e-01 | 0.03 | 6.32e-04 | 0.05 | |
| 3 | 3.96e-03 | 0.04 | 1.05e-04 | 0.26 | 2.06e-02 | 0.04 | 4.41e-05 | 0.07 | |
| 4 | 1.63e-04 | 0.04 | 1.05e-04 | 0.98* | 9.79e-04 | 0.05 | 2.58e-05 | 0.59 | |
| 5 | 7.45e-06 | 0.05 | 1.05e-04 | 1.00* | 5.20e-05 | 0.05 | 2.58e-05 | 1.00* | |
| 6 | 3.75e-07 | 0.05 | 1.05e-04 | 1.00 | 2.96e-06 | 0.06 | 2.58e-05 | 1.00* | |
| 7 | 2.08e-08 | 0.06 | 1.05e-04 | 1.00 | 1.77e-07 | 0.06 | 2.58e-05 | 1.00* | |
| 8 | 1.24e-09 | 0.06 | 1.05e-04 | 1.00 | 1.10e-08 | 0.06 | 2.58e-05 | 1.00* | |
| 9 | 7.74e-11 | 0.06 | 1.05e-04 | 1.00 | 7.16e-10 | 0.06 | 2.58e-05 | 1.00* | |
| 10 | 4.99e-12 | 0.06 | 1.05e-04 | 1.00 | 4.79e-11 | 0.07 | 2.58e-05 | 1.00* | |
| 11 | 3.27e-13 | 0.07 | 1.05e-04 | 1.00 | 3.29e-12 | 0.07 | 2.58e-05 | 1.00* | |
| 12 | 2.18e-14 | 0.07 | 1.05e-04 | 1.00* | 2.31e-13 | 0.07 | 2.58e-05 | 1.00* | |
| 13 | 1.33e-15 | 0.11 | 1.05e-04 | 1.00* | 1.80e-14 | 0.08 | 2.58e-05 | 1.00* | |
| 14 | 1.04e-15 | 0.45 | 1.05e-04 | 1.00* | 6.47e-15 | 0.36 | 2.58e-05 | 1.00* | |
| 15 | 6.61e-16 | 0.63 | 1.05e-04 | 1.00* | 5.11e-15 | 0.79 | 2.58e-05 | 1.00* | |
| <hr/> | | | | | | | | | |
| $n = 64$ | | | | $n = 128$ | | | | | |
| V-cycle | $\ r^k\ _h$ | ratio | $\ e\ _h$ | ratio | $\ r^k\ _h$ | ratio | $\ e\ _h$ | ratio | |
| 0 | 1.06e+04 | | 5.72e-01 | | 4.16e+04 | | 5.74e-01 | | |
| 1 | 7.56e+01 | 0.01 | 1.39e-02 | 0.02 | 2.97e+02 | 0.01 | 1.39e-02 | 0.02 | |
| 2 | 2.07e+00 | 0.03 | 6.87e-04 | 0.05 | 8.25e+00 | 0.03 | 6.92e-04 | 0.05 | |
| 3 | 8.30e-02 | 0.04 | 4.21e-05 | 0.06 | 3.37e-01 | 0.04 | 4.22e-05 | 0.06 | |
| 4 | 4.10e-03 | 0.05 | 7.05e-06 | 0.17 | 1.65e-02 | 0.05 | 3.28e-06 | 0.08 | |
| 5 | 2.29e-04 | 0.06 | 3.44e-06 | 0.91* | 8.99e-04 | 0.05 | 1.63e-06 | 0.50 | |
| 6 | 1.39e-05 | 0.06 | 3.44e-06 | 1.00 | 5.29e-05 | 0.06 | 8.61e-06 | 0.99* | |
| 7 | 8.92e-07 | 0.06 | 6.44e-06 | 1.00 | 3.29e-06 | 0.06 | 1.61e-06 | 1.00* | |
| 8 | 5.97e-08 | 0.07 | 6.44e-06 | 1.00* | 2.14e-07 | 0.06 | 1.61e-06 | 1.00* | |
| 9 | 4.10e-09 | 0.07 | 6.44e-06 | 1.00* | 1.43e-08 | 0.07 | 1.61e-06 | 1.00* | |
| 10 | 2.87e-10 | 0.07 | 6.44e-06 | 1.00* | 9.82e-10 | 0.07 | 1.61e-06 | 1.00* | |
| 11 | 2.04e-11 | 0.07 | 6.44e-06 | 1.00* | 6.84e-11 | 0.07 | 1.61e-06 | 1.00* | |
| 12 | 1.46e-12 | 0.07 | 6.44e-06 | 1.00* | 4.83e-12 | 0.07 | 1.61e-06 | 1.00* | |
| 13 | 1.09e-13 | 0.07 | 6.44e-06 | 1.00* | 3.64e-13 | 0.07 | 1.61e-06 | 1.00* | |
| 14 | 2.60e-14 | 0.24 | 6.44e-06 | 1.00* | 1.03e-13 | 0.28 | 1.61e-06 | 1.00* | |
| 15 | 2.30e-14 | 0.88 | 6.44e-06 | 1.00* | 9.19e-14 | 0.89 | 1.61e-06 | 1.00* | |

A warning about bounds

- Bounds like $\|e_{n+1}\| \leq \gamma \|e_n\|$ & $\|u^{(h)} - u^h\| = O(h)$ are only just that--bounds!
- If you see behavior that suggests that these bounds are sharp (e.g., halving h halves the discretization error), then great. If you don't see this behavior, don't assume things are wrong.
- Think about this:

$$O(h^2) = O(h) \text{ but generally } O(h) \neq O(h^2) !!!$$

(Any process that is $O(h^2)$ is also $O(h)$, but the converse isn't necessarily true.)

CU-Boulder

99 of 322

CU-Boulder

100 of 322

Numerical results MV cycling

Shown are the results of 15 V(2,1)-cycles. We display, after each cycle, residual norms, total error norms, & ratios of these norms to their values after the previous cycle.

$N = 16, 32, 64, 128$.

$$\begin{aligned} \|r^h\|_h &= h \|r^h\|_2 \\ &\text{scaled residual error} \\ \|e\|_h &= h \|u^{(h)} - v^h\|_2 \\ &\text{scaled discrete total error} \end{aligned}$$

Reconsideration

You want to approximate u^h .

A good iteration is the V-cycle.

What's a good way to start it?

Can you do better than $v^h \leftarrow 0$?

Start on the coarse grid!

Use nested iteration for the V-cycle!

Look again at nested iteration

- Idea:* It's cheaper to solve a problem (fewer iterations) if the initial guess is good.
- How to get a good initial guess:
 - "Solve" the problem on the coarse grid first.
 - Interpolate the coarse solution to the fine grid.
- Now, let's use the V-cycle as the solver on each grid level! This defines the **Full Multigrid (FMG) cycle**.

Full multigrid (FMG)

$$\underline{v^h \leftarrow FMG(f^h)}$$

- Initialize $f^h, f^{2h}, f^{4h}, \dots, f^H$
- Solve on coarsest grid $v^H = (A^H)^{-1} f^H$
- Interpolate initial guess $v^{2h} \leftarrow I_{4h}^{2h} v^{4h}$
- Perform V-cycle $v^{2h} \leftarrow MV^{2h}(v^{2h}, f^{2h})$
- Interpolate initial guess $v^h \leftarrow I_{2h}^h v^{2h}$
- Perform V-cycle $v^h \leftarrow MV^h(v^h, f^h)$

$$v^H = (A^H)^{-1} f^H$$

$$v^{2h} \leftarrow I_{4h}^{2h} v^{4h}$$

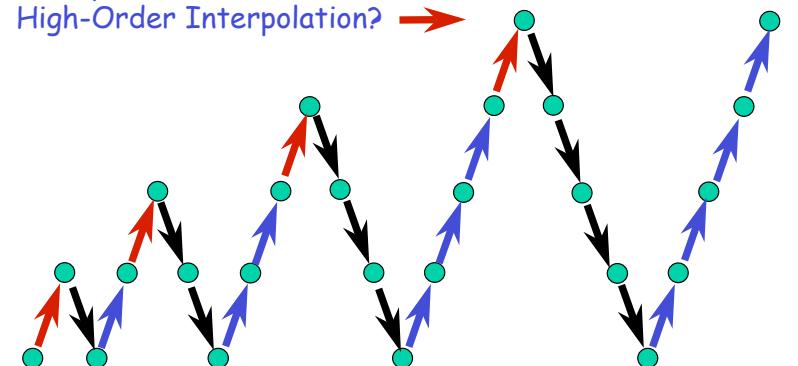
$$v^{2h} \leftarrow MV^{2h}(v^{2h}, f^{2h})$$

$$v^h \leftarrow I_{2h}^h v^{2h}$$

$$v^h \leftarrow MV^h(v^h, f^h)$$

FMG-cycle

- Restriction
- Interpolation
- High-Order Interpolation?



FMG-cycle (recursive form)

$$v^h \leftarrow \text{FMG}(f^h), \eta$$

- 1) Initialize f^h, f^{2h}, \dots, f^H .
- 2) If $h = H$, then go to 4 (where MV is a direct solve);
else: $v^{2h} \leftarrow \text{FMG}(f^{2h})$.
- 3) Set initial guess: $v^h \leftarrow I_{2h}^h v^{2h}$. We use $\eta = 1$.
- 4) Perform $v^h \leftarrow \text{MV}(v^h, f^h)$ η times.

FMG cycle cost

One V(2,1)-cycle is performed per level, at a cost of $3/(1 - 2^{-d})$ WUs per grid (where the WU is for the size of the finest grid involved).

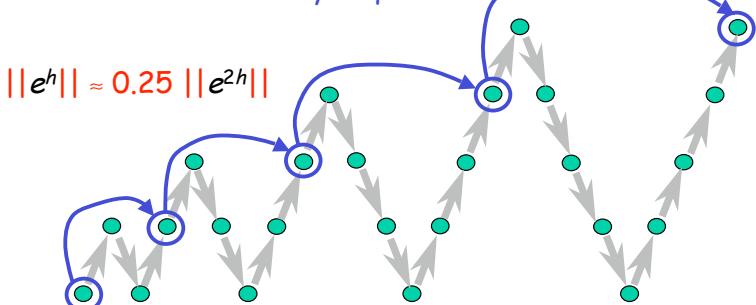
The size for the WU for coarse-grid j is 2^{-jd} times the size for the WU for the fine grid (grid 0). Hence, the cost of the FMG(2,1) cycle in WUs is less than

$$[3/(1 - 2^{-d})](1 + 2^{-d} + 2^{-2d} + \dots) = 3/(1 - 2^{-d})^2.$$

$$d=1: 12 \text{ WUs}; \quad d=2: 16/3 \text{ WUs}; \quad d=3: 192/49 \text{ WUs}.$$

Has discretization error been reached by FMG?

If discretization error is achieved, then $\|e^h\| = O(h^2)$ & the error norms at the "solution" points in the cycle should form a Cauchy sequence:



We need to be more careful...

Comparing the right things

- Problem: We are thinking that u^{2h} approximates u^h to order $O(h^2)$, when all we really know is that u^h approximates $u^{(h)}$ to order $O(h^2)$ (any h).
- We know that $u^{(2h)}$ and $u^{(h)}$ are the "same", right? So, if u^{2h} approximates $u^{(2h)}$ to order $O(4h^2)$ and u^h approximates $u^{(h)}$ to order $O(h^2)$, shouldn't u^{2h} approximate u^h to order $O(4h^2)$? How, exactly?
- When we interpolate u^{2h} to grid h , what errors does interpolation introduce?
- Sorting out these comparison is a bit technical.
- In other words, here comes the algebra...

Interpolation stability

how interpolation affects error

- Property: $\|Pe^{2h}\| \leq \beta \|e^{2h}\|$
- Reasoning:

$$\begin{aligned}\|Pe^{2h}\| &\leq \|P\| \|e^{2h}\| \\ &= \|P^T P\|^{1/2} \|e^{2h}\|\end{aligned}$$

$$P^T P = \begin{pmatrix} \frac{1}{2} & 1 & \frac{1}{2} & & \\ & \frac{1}{2} & 1 & \frac{1}{2} & \\ & & \frac{1}{2} & 1 & \frac{1}{2} \\ & & & \frac{1}{2} & 1 \end{pmatrix} \begin{pmatrix} \frac{1}{2} & & & \\ & 1 & & \\ & & \frac{1}{2} & \frac{1}{2} \\ & & & 1 \\ & & & & \frac{1}{2} \end{pmatrix} = \begin{pmatrix} \frac{3}{2} & \frac{1}{4} & & \\ \frac{1}{4} & \frac{3}{2} & \frac{1}{4} & \\ & \frac{1}{4} & \frac{3}{2} & \\ & & & \frac{1}{2} \end{pmatrix}$$

$$\Rightarrow \|P^T P\|^{1/2} \leq \sqrt{2}$$

In practice, $\beta \approx 1$.

Approximation property

how the discrete solutions approximate each other

Property: $\|u^h - Pu^{2h}\| \leq \alpha Kh^2$

Reasoning:

$$\begin{aligned}|(u^{(h)} - Pu^{(2h)})_i| &= |u(x_i) - (u(x_{i-1}) + u(x_{i+1}))/2| \leq |u''(\xi)| h^2/2 \\ \Rightarrow \|u^{(h)} - Pu^{(2h)}\| &\leq cKh^2 \\ \Rightarrow \|u^h - Pu^{2h}\| &\leq \|u^h - u^{(h)}\| + \|u^{(h)} - Pu^{(2h)}\| + \|Pu^{(2h)} - Pu^{2h}\| \\ &\leq Kh^2 + cKh^2 + \|P\| \cdot \|u^{(2h)} - u^{2h}\| \\ &\leq Kh^2 + cKh^2 + \beta K(2h)^2 \\ &\leq (1+c+4\beta)Kh^2\end{aligned}$$

In practice, $\alpha \approx 5$.

FMG accuracy

$$\|e^h\| \leq Kh^2$$

Assume:

$$\begin{aligned}\|e^{2h}\| &\leq K(2h)^2 && \text{induction hypothesis} \\ \|u^h - Pu^{2h}\| &\leq \alpha Kh^2 && \text{approximation property } (\alpha \approx 5) \\ \|Pu^{2h}\| &\leq \beta \|u^{2h}\| && \text{interpolation stability } (\beta \approx 1)\end{aligned}$$

Triangle inequality:

$$\begin{aligned}\|e^h\| &= \|u^h - Pv^{2h}\| \\ &\leq \|u^h - Pu^{2h}\| + \|Pu^{2h} - Pv^{2h}\| \\ &\leq \alpha Kh^2 + \beta K(2h)^2 \\ &= (\alpha + 4\beta)Kh^2 \\ \Rightarrow \|e^h\| &\leq "9"Kh^2\end{aligned}$$

So we need only reduce $\|e^h\|$ by "0.1"!!!

Numerical example

- Consider again the 2-D model problem (with $\sigma = 0$):
 $-u_{xx} - u_{yy} = 2[(1 - 6x^2)y^2(1 - y^2) + (1 - 6y^2)x^2(1 - x^2)]$
 inside the unit square, with $u = 0$ on the boundary.
- We examine the effectiveness of FMG cycling to solve the problem on $(N+1) \times (N+1)$ grids [$(N-1) \times (N-1)$ interior points] for $N = 16, 32, 64, 128$.

FMG results

3 FMG cycles & comparison with MV cycle results

| N | FMG(1,0) $\ e\ _h$ | ratio | FMG(1,1) $\ e\ _h$ | ratio | FMG(2,1) $\ e\ _h$ | ratio | FMG(1,1) WU | V(2,1) cycles | V(2,1) WU |
|------|-----------------------|-------|-----------------------|----------|-----------------------|-------|----------------|------------------|--------------|
| 2 | 5.86e-03 | | 5.86e-03 | | 5.86e-03 | | | | |
| 4 | 5.37e-03 | 0.917 | 2.49e-03 | 0.424 | 2.03e-03 | 0.347 | 7/2 | 3 | 12 |
| 8 | 2.78e-03 | 0.518 | 9.12e-04 | 0.367 | 6.68e-04 | 0.328 | 7/2 | 4 | 16 |
| 16 | 1.19e-03 | 0.427 | 2.52e-04 | 1.03e-04 | 1.72e-04 | 0.257 | 7/2 | 4 | 16 |
| 32 | 4.70e-04 | 0.395 | 6.00e-05 | 2.58e-05 | 4.00e-05 | 0.233 | 7/2 | 5 | 20 |
| 64 | 1.77e-04 | 0.377 | 1.36e-05 | 6.44e-06 | 9.36e-06 | 0.234 | 7/2 | 5 | 20 |
| 128 | 6.49e-05 | 0.366 | 3.12e-05 | 1.61e-06 | 2.26e-06 | 0.241 | 7/2 | 6 | 24 |
| 256 | 2.33e-05 | 0.359 | 7.35e-07 | 0.235 | 5.56e-07 | 0.246 | 7/2 | 7 | 28 |
| 512 | 8.26e-06 | 0.354 | 1.77e-07 | 0.241 | 1.38e-07 | 0.248 | 7/2 | 7 | 28 |
| 1024 | 2.90e-06 | 0.352 | 4.35e-08 | 0.245 | 3.41e-08 | 0.249 | 7/2 | 8 | 32 |
| 2048 | 1.02e-06 | 0.351 | 1.08e-08 | 0.247 | 8.59e-09 | 0.250 | 7/2 | 9 | 36 |

$$\|e\|_h = h \|u^{(h)} - v^h\|_2$$

scaled discrete total error

Diagnostic tools

for debugging the code, the method, the problem

- Finding mistakes in codes, algorithms, concepts, & the problem itself challenges our scientific abilities.
- This challenge is especially tough for multigrid:
 - Interactions between multilevel processes can be very subtle.
 - It's often not easy to know how well multigrid should perform.
- Achi Brandt:
 - "The amount of computational work should be proportional to the amount of real physical changes in the computed solution."
 - "Stalling numerical processes must be wrong."
- The "computational culture" is best learned by lots of experience & interaction, but some discussion helps.

Tool # 1: Be methodical

- Modularize your code.
- Test the algebraic solver first.
- Test the discretization next.
- Test the FMG solver last.
- Beware of boundaries, scales, & concepts.
- Ask whether the problem itself is well posed.

Tool # 2: Start simply

- Start from something that already works if you can.
- Introduce complexities slowly & methodically, testing thoroughly along the way.
- Start with a very coarse fine grid (no oxymoron intended).
- Start with two levels if you can, using a direct solver or lots of cycles on coarse grids if nothing else.

If you find trouble, your first job is to find the simplest case where that trouble is still evident!!!

Tool # 3: Expose trouble

Start simply, but don't let niceties mask trouble:

- Set reaction/Helmholtz terms to zero.
- Take infinite or very big time steps.
- Don't take 1-D too seriously, not even 2-D.

Tool # 4: Test fixed point property

Relaxation shouldn't alter the exact solution of the linear system (up to machine precision).

- Create a right side: $f = Au^*$ with u^* given.
- Make sure u^* satisfies the right boundary conditions.
- Test relaxation starting with u^* : Is $r = 0$, is it zero after relaxation, does u^* change?
- Test coarse-grid correction starting with u^* : Is the correction zero?

Tool # 5: Test on $Au=0$

- The exact solution $u^* = 0$ is known!
- Residual norm $\|Au\|$ & error norm $\|u\|$ are computable.
- Norms $\|Au\|$ & $\|u\|$ should eventually decrease steadily with a rate that might be predicted by mode analysis.
- Multigrid can converge so fast that early stalling suggests trouble when it's just that all machine-representable numbers in a nonzero u^* have already been computed! Computing $r = f - Au$ & updating u shouldn't have trouble with machine precision if you have $u^* = 0$ & thus $f = 0$.

Tool # 6: Zero out residual

- Using a normal test, try multiplying the residual by 0 before you go to the coarse grid.
- Check to see that the coarse-grid corrections are 0.
- Compare this test with a relaxation-only test--the results should be identical.

Tool # 7: Print out residual norms

- Use the discrete L^2 norm:

$$\|r\|_h = (h^d \sum r_i^2)^{1/2} = h^{d/2} \|r\|_2$$

- Output $\|r\|_h$ after each pre- & post-relaxation sweep.
- These norms should decline to zero steadily for each h .
- The norm after post-relaxation should be consistently smaller than after pre-relaxation--by the predictable convergence factor at least.

A warning about residuals

- Residuals could mask large smooth errors:
 - If $\|e\|=O(h^2)$, then $\|r\|=O(h^2)$ for smooth e , but $\|r\|=O(1)$ for oscillatory e . ($\lambda_N/\lambda_1 = O(h^2)!!!$)
 - This means that if we are controlling the error by monitoring $\|r\|$, if we don't know the nature of e , & if we don't want to risk large e by requiring only $\|r\|=O(1)$, then we may have to work very hard to make $\|r\|=O(h^2)$. (We could in effect be trying to make $\|e\|=O(h^4) !!!$)
- Multigrid tends to balance the errors, so $\|r\|=O(h^2)$ tends to mean $\|e\|=O(h^2)$.

Tool # 8: Graph the error

- Run a test on a problem with known solution ($Au=0$).
- Plot algebraic error before & after fine-grid relaxation.
- Is the error oscillatory after coarse-grid correction?
- Is the error much smoother after fine-grid relaxation?
- Are there any strange characteristics near boundaries, interfaces, or other special phenomena?

Tool # 9: Test two-level cycling

- Replace the coarse-grid V-cycle recursive call with a direct solver if possible, or iterate many times with some method known to "work" (test $\|r\|$ to be sure it's very small), or use many recursive V-cycle calls.
- This can be used to test performance between two coarser levels, especially if residual norm behavior identifies trouble on a particular level.

Tool # 10: Beware of boundaries

- Boundaries usually require special treatment of the stencils, intergrid transfers, & sometimes relaxation.
- Special treatment often means special trouble, typically exposed in later cycles as it begins to infect the interior.
- Replace the boundary by periodic or Dirichlet conditions.
- Relax more at the boundary, perhaps using direct solvers.
- Make sure your coarse-grid approximation at the boundary is guided by good discretization at the fine-grid boundary.

Tool # 11: Test for symmetry

- If your problem is symmetric or includes a symmetric case, test for it.
- Check symmetry of the fine-grid & coarse-grid matrices: are a_{ij} & a_{ji} relatively equal (to machine precision).
- Be especially watchful for asymmetries near boundaries.

Tool # 12: Check for compatibility

- This is a bit ahead of schedule, but consider the problem
 $-u'' = f$ with $u'(0) = u'(1) = 0$.
- It's singular: If $u = 1$, then $-u'' = 0$ & $u'(0) = u'(1) = 0$.
- It's solvable iff $f \in \text{Range}(\partial_{xx}) = \eta^\perp(\partial_{xx}) = \{1\}^\perp$ or $f \perp 1$.
- First fix the grid h right side: $f^h \leftarrow f^h - (\langle f^h, 1 \rangle / \langle 1, 1 \rangle)1$.
- Do this on coarse grids too: $f^{2h} \leftarrow f^{2h} - (\langle f^{2h}, 1 \rangle / \langle 1, 1 \rangle)1$.
- Uniqueness is also a worry: $u^h \leftarrow u^h - (\langle u^h, 1 \rangle / \langle 1, 1 \rangle)1$.

Tool # 13: Test for linearity

- Again, this is a bit ahead of schedule, but if you're writing a nonlinear FAS code, it should agree with the linear code when you test it on a linear problem. Try it.
- Move gradually to the target nonlinear test problem by putting a parameter in front of the nonlinear term, then running tests as the parameter changes slowly from 0 to 1.

Tool # 14: Use a known PDE solution

- Set up the source term ($f = Au^*$ in Ω) & data ($g = u^*$ on Γ).
- Do multigrid results compare qualitatively with sampled u^* ?
- Monitor $\|u^* - u^h\|_h$.
- Test a case with no discretization error (let $u^* = 2^{\text{nd}}$ degree polynomial). The algebraic error should tend steadily to 0.
- Test a case with discretization error (let u^* have a nonzero 3rd derivative). The algebraic error should decrease rapidly at first, then stall at discretization error level. Check error behavior as you decrease h . Does it behave like $O(h^2)$ (h halved \Rightarrow error halved) or whatever it should behave like?

Computing assignments

- Document:
norms/weights, $V(v_1, v_2)$, errors, labels (table, graph)
- Use various scenarios:
 $Ax = 0$, $Ax = f$, varying N & v_i & ω , Jacobi/Gauss-Seidel
- Thoroughly test:
don't stop until you get what you expect.
compare with known solution, text, others.
study discretization & algebraic errors.
report on "asymptotic" factors.
- Be kind to the reader:
code = zzz... tables = + tables & graphs = ++
tables & graphs & discussion (clear, concise) = +++
- Discuss, discuss, discuss:
what do you see & think? what did you learn?

Tool # 15: Test FMG accuracy

- Make sure first that the algebraic solver converges as predicted, with uniformly bounded convergence factors.
- Test the discretization using Tool # 14.
- Compare FMG total error to discretization error for various h . You might need to tune the FMG process here (play with the number of cycles & relaxation sweeps).

Outline

Chapters 1-5:

- ✓ Model Problems
- ✓ Basic Iterative Methods
 - Convergence tests
 - Analysis
- ✓ Elements of Multigrid
 - Relaxation
 - Coarsening
- ✓ Implementation
 - Complexity
 - Diagnostics
- Some Theory
 - Spectral vs. algebraic

Chapters 6-10:

- Nonlinear Problems
 - Full approximation scheme
- Selected Applications
 - Neumann boundaries
 - Anisotropic problems
 - Variable meshes
 - Variable coefficients
- Algebraic Multigrid (AMG)
 - Matrix coarsening
- Multilevel Adaptive Methods
 - FAC
- Finite Elements
 - Variational methodology

Homework Due
Computing Tag

5. Some theory

What is A^{2h} ?

- Recall the 2-grid coarse-grid correction scheme:
 - 1) Relax on $A^h u^h = f^h$ on Ω^h to get v^h .
 - 2) Compute $f^{2h} = I_h^{2h}(f^h - A^h v^h)$.
 - 3) Solve $A^{2h} u^{2h} = f^{2h}$ on Ω^{2h} .
 - 4) Correct fine-grid solution $v^h \leftarrow v^h + I_{2h}^h u^{2h}$.
- Assume that $e^h \in \text{Range}(I_{2h}^h)$. Then the residual equation can be written

$$r^h = A^h e^h = A^h I_{2h}^h u^{2h} \text{ for some } u^{2h} \in \Omega^{2h}.$$

This characterizes u^{2h} , but with too many equations.

- How does A^h act on $\text{Range}(I_{2h}^h)$?

Building A^{2h} : The Galerkin condition

- The residual equation on the coarse grid is

$$I_h^{2h} A^h I_{2h}^h u^{2h} = I_h^{2h} r^h$$

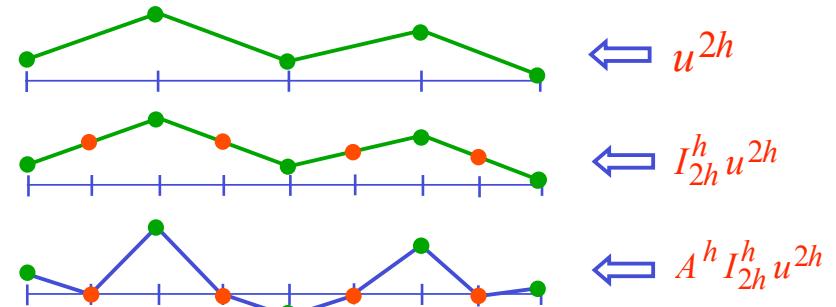
- We thus identify the coarse-grid operator as

$$A^{2h} = I_h^{2h} A^h I_{2h}^h \quad A^{2h} = RAP$$

- How do we know RAP is symmetric?

If $P^T = \alpha R$ (so that $R^T = (1/\alpha)P$), then $(RAP)^T = P^T A^T R^T = \alpha(1/\alpha) RAP = RAP$.

How does A^h act on $\text{Range}(I_{2h}^h)$?



Thus, the odd rows of $A^h I_{2h}^h$ are zero (1-D only) & $r_{2i+1} = 0$. So we keep the even rows of $A^h I_{2h}^h$ for the residual equations on Ω^{2h} . We do this by applying restriction, either injection or full weighting: $I_h^{2h} A^h I_{2h}^h u^{2h} = I_h^{2h} r^h$.

We use full weighting from now on unless otherwise stated.

Computing the i^{th} row of A^{2h}

- Compute $A^{2h} \hat{e}_i^{2h}$, where $\hat{e}_i^{2h} = (0, 0, \dots, 0, 1, 0, \dots, 0)^T$.

↑ Why is this the i^{th} row of A^{2h} ?

| | |
|---------------------|--|
| | \hat{e}_i^{2h} |
| 0 | \hat{e}_i^{2h} |
| 0 | $I_{2h}^h \hat{e}_i^{2h}$ |
| $-\frac{1}{2h^2}$ | $A^h I_{2h}^h \hat{e}_i^{2h}$ |
| $-\frac{1}{(2h)^2}$ | $I_h^{2h} A^h I_{2h}^h \hat{e}_i^{2h}$ |
| $\frac{1}{(2h)^2}$ | |
| $\frac{2}{(2h)^2}$ | |
| $-\frac{1}{(2h)^2}$ | |

The i^{th} row of A^{2h} looks a lot like the i^{th} row of A^h !

- The i^{th} row of A^{2h} is $\frac{1}{(2h)^2} [-1 \quad 2 \quad -1]$, which is the Ω^{2h} version of A^h .
- Note that **IF** relaxation on Ω^h leaves only error in the range of interpolation, then solving $A^{2h} u^{2h} = f^{2h}$ determines the error exactly!
- This is generally not feasible, but this logic leads to a very plausible representation for A^{2h} .

Variational properties of coarsening

- The definition for A^{2h} that resulted from the foregoing line of reasoning is useful for both theoretical & practical reasons. Together with the commonly used relationship between restriction & prolongation, we have the variational properties:

$$A^{2h} = I_h^{2h} A^h I_{2h}^h$$

$$I_{2h}^h = c (I_h^{2h})^T$$

Galerkin Condition

for c in \mathfrak{R} .

Properties of restriction in a little more detail...

- Full Weighting: $I_h^{2h} : \Omega^h \rightarrow \Omega^{2h}$ or $I_h^{2h} : \mathfrak{R}^N \rightarrow \mathfrak{R}^{(N-1)/2}$
- $N=7$:

$$I_h^{2h} = \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 & & & & \\ & 1 & 2 & 1 & & & \\ & & & 1 & 2 & 1 & \end{bmatrix}_{3 \times 7}$$
- I_h^{2h} has rank $\frac{N-1}{2}$ & null space $\eta(I_h^{2h})$ with dim $\frac{N+1}{2}$.

Spectral properties of restriction

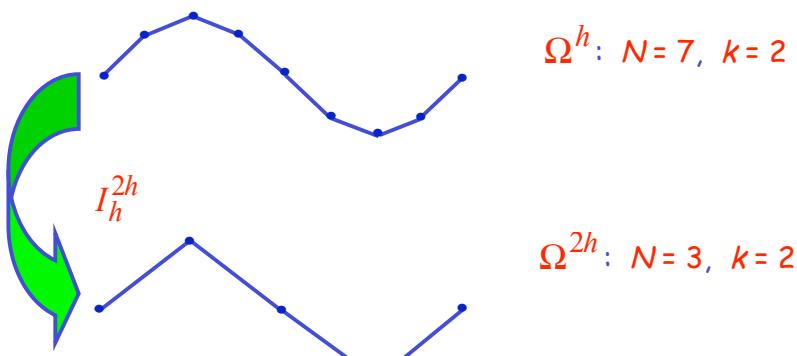
- How does I_h^{2h} act on the eigenvectors of A^h ?
- Consider $w_{k,j}^h = \sin\left(\frac{jk\pi}{N+1}\right)$, $1 \leq k \leq N$, $0 \leq j \leq N$.
- A little algebra & trigonometry shows that

$$\begin{aligned} (I_h^{2h} w_k^h)_j &= \cos^2\left(\frac{k\pi}{2(N+1)}\right) w_{k,j}^{2h} \\ &\equiv c_k w_{k,j}^{2h} \end{aligned}$$

for $1 \leq k \leq (N-1)/2$.

Spectral properties (cont'd)

- i.e., $I_h^{2h} [k^{\text{th}} \text{ mode on } \Omega^h] = c_k [k^{\text{th}} \text{ mode on } \Omega^{2h}]$



CU-Boulder

141 of 322

Spectral properties (cont'd)

- Let $k' = N+1 - k$ for $1 \leq k \leq (N-1)/2$, so that $(N-3)/2 \leq k' \leq N$.
- A little algebra & trigonometry shows that

$$(I_h^{2h} w_k^h)_j = -\sin^2\left(\frac{k\pi}{2(N+1)}\right) w_{k,j}^{2h}$$

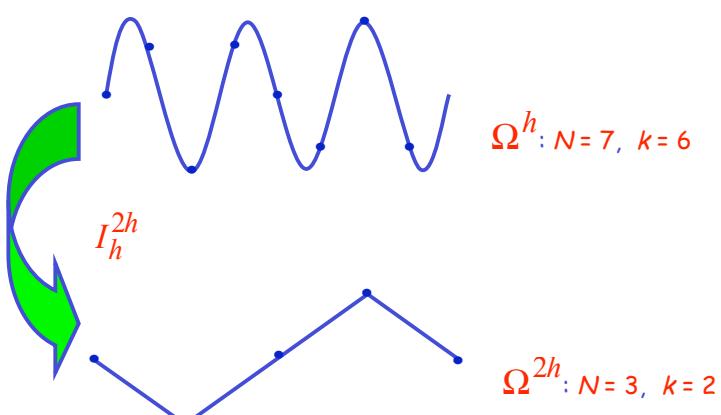
$$\equiv s_k w_{k,j}^{2h}$$

CU-Boulder

142 of 322

Spectral properties (cont'd)

i.e., $I_h^{2h} [(N+1-k)^{\text{th}} \text{ mode on } \Omega^h] = -s_k [k^{\text{th}} \text{ mode on } \Omega^{2h}]$



CU-Boulder

143 of 322

Spectral properties (cont'd)

- Summarizing:

$$\left. \begin{aligned} I_h^{2h} w_k^h &= c_k w_k^{2h} \\ I_h^{2h} w_{k'}^h &= -s_k w_{k'}^{2h} \\ I_h^{2h} w_{(N+1)/2}^h &= 0 \end{aligned} \right\} \left\{ \begin{array}{l} 1 \leq k \leq \frac{N-1}{2} \\ k' = N+1-k \end{array} \right.$$

- Complementary modes:

$$W_k = \text{span} \{w_k^h, w_{k'}^h\}$$

$$I_h^{2h} W_k \rightarrow \{w_k^{2h}\}^\infty$$

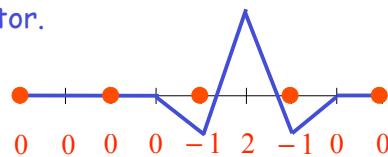
CU-Boulder

144 of 322

Null space of restriction

- Observe that $\eta(I_h^{2h}) = \text{span}(A^h \hat{e}_i^h)$, where i is odd & \hat{e}_i^h is the i^{th} unit vector.

- Let $\eta_i = A^h \hat{e}_i^h$.



- While the η_i looks oscillatory, it generally contains all Fourier modes of A^h :

$$\eta_i = \sum_{k=1}^N a_k w_k \quad a_k \neq 0$$

- All the Fourier modes of A^h are needed to represent the null space of restriction!

Properties of interpolation

- Interpolation: $I_{2h}^h : \Omega^{2h} \rightarrow \Omega^h$ or $I_{2h}^h : \mathcal{R}^{(N-1)/2} \rightarrow \mathcal{R}^N$

- $N = 7$:

$$I_{2h}^h = \frac{1}{2} \begin{bmatrix} 1 \\ 2 \\ 1 & 1 \\ 2 \\ 1 & 1 \\ 2 \\ 1 \end{bmatrix}$$

- I_{2h}^h has full rank & null space $\{0\}$.

Spectral properties of interpolation

- How does I_{2h}^h act on the eigenvectors of A^{2h} ?
- Consider $(w_k^{2h})_j = \sin\left(\frac{jk\pi}{(N+1)/2}\right)$, $1 \leq k \leq (N-1)/2$, $0 \leq j \leq (N+1)/2$.
- A bit of work shows that the modes of A^{2h} are NOT "preserved" by I_{2h}^h , but that the space W_k is "preserved":

$$\begin{aligned} I_{2h}^h w_k^{2h} &= \cos^2\left(\frac{k\pi}{2(N+1)}\right) w_k^h - \sin^2\left(\frac{k\pi}{2(N+1)}\right) w_k^h \\ &= c_k w_k^h - s_k w_k^h \end{aligned}$$

Spectral properties of interpolation

$$I_{2h}^h w_k^{2h} = c_k w_k^h - s_k w_k^h$$

- Interpolation of smooth Ω^{2h} modes excites oscillatory modes on Ω^h .

- Note that if $k \ll \frac{N}{2}$, then

$$\begin{aligned} I_{2h}^h w_k^{2h} &= \left(1 - O\left(\frac{k^2}{N^2}\right)\right) w_k^h + O\left(\frac{k^2}{N^2}\right) w_k^h \\ &\approx w_k^h \end{aligned}$$

- I_{2h}^h is 2nd-order interpolation.

Range of interpolation

- The range of I_{2h}^h is the span of the columns of I_{2h}^h .
- Let ξ_i be the i^{th} column of I_{2h}^h .



$$\xi_i^h = \sum_{k=1}^N b_k w_k^h, \quad b_k \neq 0$$

- All the Fourier modes of A^h are needed to represent $\text{Range}(I_{2h}^h)$.

CU-Boulder

149 of 322

Use all the facts to analyze the coarse-grid correction scheme
relax only before correction

- Relax once on Ω^h : $v^h \leftarrow Rv^h + Bf^h$.
 - Compute & restrict residual $f^{2h} \leftarrow I_h^{2h}(f^h - A^h v^h)$.
 - Solve residual equation $v^{2h} = (A^{2h})^{-1} f^{2h}$.
 - Correct fine-grid solution $v^h \leftarrow v^h + I_{2h}^h v^{2h}$.
- The entire process appears as
- $$v^h \leftarrow Rv^h + Bf^h + I_{2h}^h (A^{2h})^{-1} I_h^{2h} (f^h - A^h (Rv^h + Bf^h))$$
- The exact solution satisfies
- $$u^h = Ru^h + Bf^h + I_{2h}^h (A^{2h})^{-1} I_h^{2h} (f^h - A^h (Ru^h + Bf^h))$$

CU-Boulder

150 of 322

CG error propagation

- Subtracting the previous two expressions, we get

$$e^h \leftarrow \left[I - I_{2h}^h (A^{2h})^{-1} I_h^{2h} A^h \right] R e^h$$

$$e^h \leftarrow CG e^h$$

- How does CG act on the modes of A^h ? Assume e^h consists of the modes w_k^h & $w_{k'}^h$ for $1 \leq k \leq \frac{N-1}{2}$ & $k' = N+1-k$.

- We know how R^α , A^h , I_h^{2h} , $(A^{2h})^{-1}$, I_{2h}^h act on w_k^h & $w_{k'}^h$.

CG error propagation

- For now, assume no relaxation. Then

$$W_k = \text{span}\{w_k^h, w_{k'}^h\}$$

is invariant under CG:

$$CG w_k^h = s_k w_k^h + s_{k'} w_{k'}^h$$

$$CG w_{k'}^h = c_k w_k^h + c_{k'} w_{k'}^h$$

where

$$c_k = \cos^2\left(\frac{k\pi}{2(N+1)}\right) \quad s_k = \sin^2\left(\frac{k\pi}{2(N+1)}\right)$$

CU-Boulder

151 of 322

CU-Boulder

152 of 322

CG error propagation for $k \ll N$

- Consider the case $k \ll N$ (extremely smooth & oscillatory modes):

$$w_k \rightarrow O\left(\frac{k^2}{N^2}\right) w_k + O\left(\frac{k^2}{N^2}\right) w_{k'}$$

$$w_{k'} \rightarrow \left(1 - O\left(\frac{k^2}{N^2}\right)\right) w_k + \left(1 - O\left(\frac{k^2}{N^2}\right)\right) w_{k'}$$

- Hence, CG eliminates the smooth modes but does not damp the oscillatory modes of the error!

CG with relaxation

- Next, include one relaxation sweep. Assume that the relaxation R preserves the modes of A^h (although this is often unnecessary). Let λ_k denote the eigenvalue of R associated with w_k . For $k \ll N/2$:

$$w_k \rightarrow \lambda_k s_k w_k + \lambda_k s_k w_{k'} \quad \text{Small!}$$

$$w_{k'} \rightarrow \lambda_{k'} c_k w_k + \lambda_{k'} c_k w_{k'} \quad \text{Small!}$$

Crucial observation

- Between relaxation & coarse-grid correction, both smooth & oscillatory components of the error are effectively damped.
- This is the "spectral" picture of how multigrid works. We examine now another viewpoint, the "algebraic" picture of multigrid.

Recall the variational properties

- All the analysis that follows assumes that the variational properties hold:

$$A^{2h} = I_h^{2h} A^h I_{2h}^h$$

$$I_{2h}^h = c (I_h^{2h})^T$$

Galerkin Condition

for c in \mathfrak{R} .

Fundamental Theorem of Linear Algebra

$$N(B) = R(B^T)^\perp$$

- If $x \in N(B)$, then, for any $y \in R(B^T)$, we have $y = B^T z$ and, hence,

$$\langle x, y \rangle = \langle x, B^T z \rangle = \langle Bx, z \rangle = 0.$$

So, $x \in R(B^T)^\perp$ and, thus, $N(B) \subset R(B^T)^\perp$.

- But if $x \in R(B^T)^\perp$, then $x \perp B^T z$ for any z , so $0 = \langle x, B^T z \rangle = \langle Bx, z \rangle$.

Picking $z = Bx$ shows $x \in N(B)$, so $N(B) \supset R(B^T)^\perp$.

Subspace decomposition of Ω^h

- If $u^h \in N(I_h^{2h} A^h)$, then, for any u^{2h} , we have

$$0 = \langle I_h^{2h} A^h u^h, u^{2h} \rangle = \langle A^h u^h, I_{2h} u^{2h} \rangle,$$

so

$$R(I_{2h}^h) \perp_{A^h} N(I_h^{2h} A^h),$$

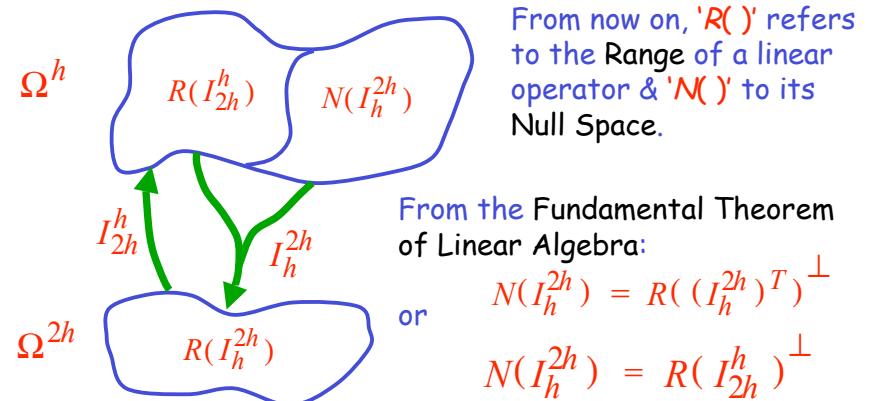
where $x \perp_{A^h} y$ means $\langle A^h x, y \rangle = 0$. "energy" inner product

- Moreover, any e^h can be written as $e^h = s^h + t^h$, where $s^h \in R(I_{2h}^h)$ & $t^h \in N(I_h^{2h} A^h)$.
- Hence, we get the "energy-orthogonal" decomposition

$$\Omega^h = R(I_{2h}^h) \oplus N(I_h^{2h} A^h).$$

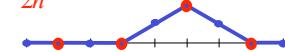
Algebraic interpretation of CG

Consider the subspaces that make up Ω^h & Ω^{2h} .



Characteristics of the subspaces

- Since $s^h = I_{2h}^h q^{2h}$ for some $q^{2h} \in \Omega^{2h}$, we associate s^h with the smooth components of e^h . But, s^h generally has all Fourier modes in it. Recall the basis vectors for I_{2h}^h :



- Similarly, we associate t^h with oscillatory components of e^h , although t^h generally has all Fourier modes in it as well. Recall that $N(I_h^{2h})$ is spanned by $\eta_i = A^h \hat{e}_i^h$, so $N(I_h^{2h} A^h)$ is spanned by the unit vectors $\hat{e}_i^h = (0, 0, \dots, 0, 1, 0, \dots, 0)^T$ for odd i , which "look" oscillatory.



Algebraic analysis CG

- Recall that (without relaxation)

$$CG = I - I_{2h}^h (A^{2h})^{-1} I_h^{2h} A^h.$$

- First note that if $s^h \in R(I_{2h}^h)$, then $CG s^h = 0$. This follows since $s^h = I_{2h}^h q^{2h}$ for some $q^{2h} \in \Omega^{2h}$ & therefore

$$CG s^h = \left[I - I_{2h}^h (A^{2h})^{-1} I_h^{2h} A^h \right] I_{2h}^h q^{2h} = 0.$$

A^{2h} by Galerkin property

- It follows that $N(CG) = R(I_{2h}^h)$, that is, the null space of CG is the range of interpolation.

What does this imply?

CU-Boulder

161 of 322

How does the algebraic picture fit with the spectral view?

- We may view Ω^h in two ways:

$$\Omega^h = \left\{ \begin{array}{l} \text{Low frequency modes} \\ 1 \leq k \leq N/2 \end{array} \right\} \oplus \left\{ \begin{array}{l} \text{High frequency modes} \\ N/2 < k < N \end{array} \right\}$$

that is,

$$\Omega^h = L \oplus H$$

or

$$\Omega^h = R(I_{2h}^h) \oplus N(I_h^{2h} A^h).$$

Are these "orthogonal" decompositions?

CU-Boulder

163 of 322

More algebraic analysis of CG

- Next, note that if $t^h \in N(I_h^{2h} A^h)$, then

$$CG t^h = \left[I - I_{2h}^h (A^{2h})^{-1} I_h^{2h} A^h \right] t^h$$

0

$$\Rightarrow CG t^h = t^h$$

- Thus, CG is the identity on $N(I_h^{2h} A^h)$.

What does this imply?

CU-Boulder

162 of 322

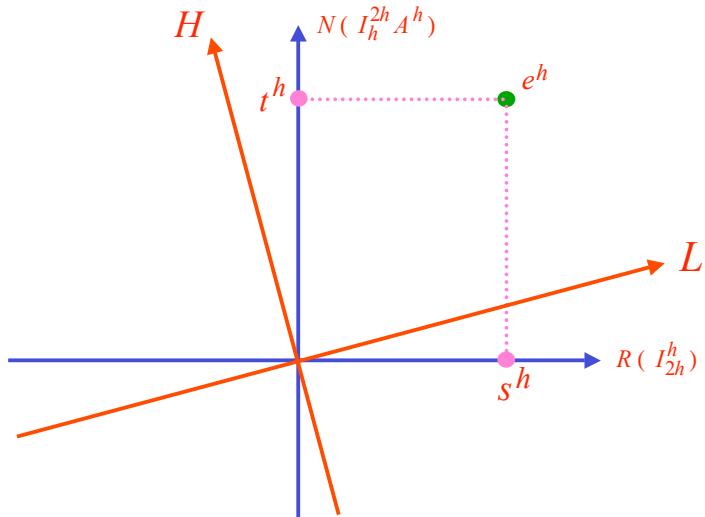
Actually, each view is just part of the picture

- The operations we have examined work on different spaces!
- While $N(I_h^{2h} A^h)$ is mostly oscillatory, it isn't H , & while $R(I_{2h}^h)$ is mostly smooth, it isn't L .
- Relaxation eliminates error from H .
- Coarse-grid correction eliminates error from $R(I_{2h}^h)$.

CU-Boulder

164 of 322

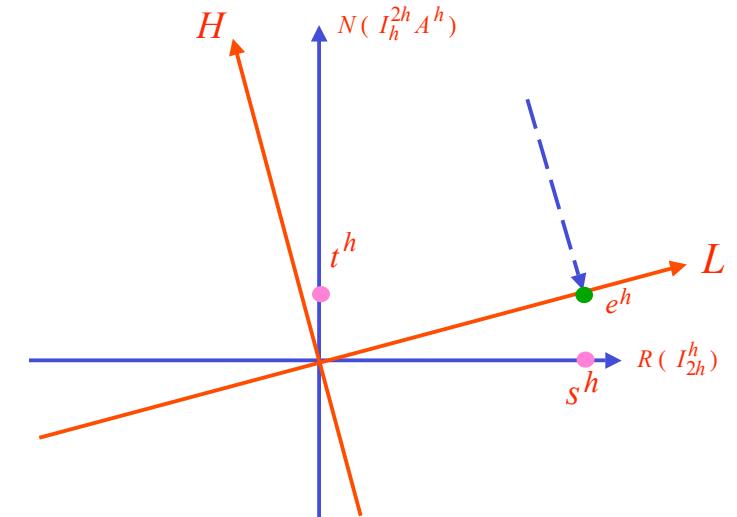
How it actually works (cartoon)



CU-Boulder

165 of 322

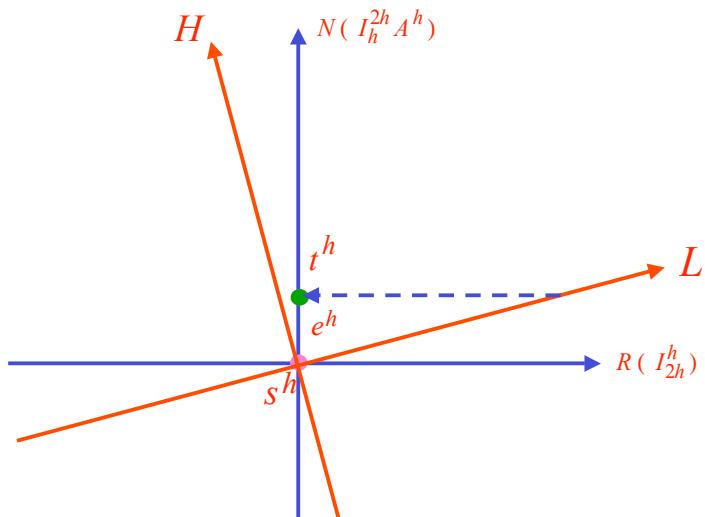
Relaxation eliminates H , but increases the error in $R(I_{2h}^h)$



CU-Boulder

166 of 322

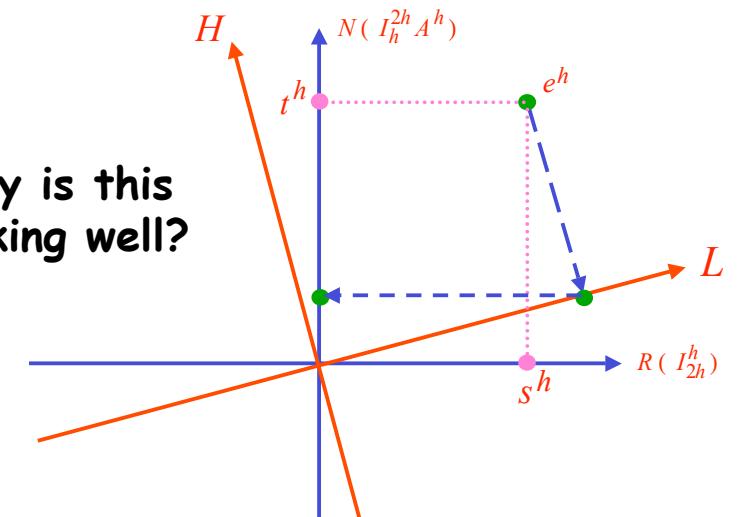
CG eliminates error in $R(I_{2h}^h)$, but increases error in H



CU-Boulder

167 of 322

CG eliminates error in $R(I_{2h}^h)$, but increases error in H

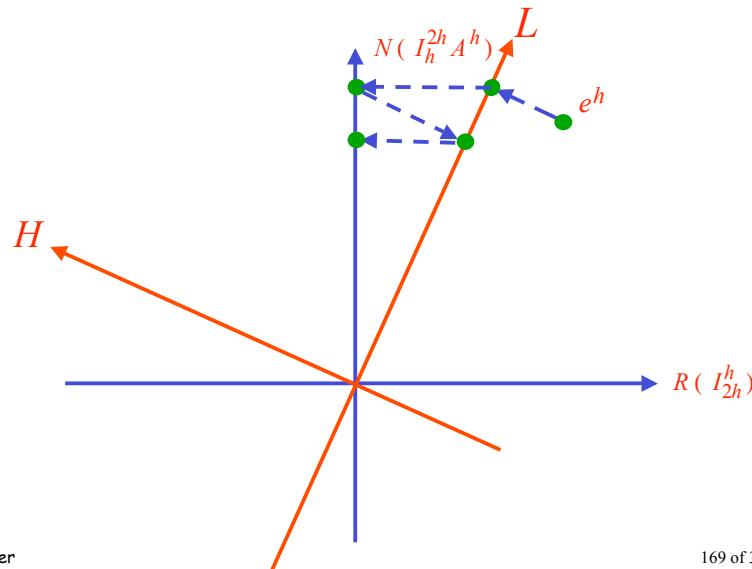


CU-Boulder

168 of 322

Why is this working well?

What if L points away from $R(I_{2h}^h)$?



CU-Boulder

169 of 322

Outline

Chapters 1-5:

- ✓ Model Problems
- ✓ Basic Iterative Methods
 - Convergence tests
 - Analysis
- ✓ Elements of Multigrid
 - Relaxation
 - Coarsening
- ✓ Implementation
 - Complexity
 - Diagnostics
- ✓ Some Theory
 - Spectral vs. algebraic

Homework Due

CU-Boulder

170 of 322

Chapters 6-10:

- Nonlinear Problems
 - Full approximation scheme
- Selected Applications
 - Neumann boundaries
 - Anisotropic problems
 - Variable meshes
 - Variable coefficients
- Algebraic Multigrid (AMG)
 - Matrix coarsening
- Multilevel Adaptive Methods
 - FAC
- Finite Elements
 - Variational methodology

Some reflection

- What do you want out of this course?
- Do you seem to be getting there?
- Can we get there more directly?
- Start thinking about your project!

6. Nonlinear problems

HANG ON !!!

- How should we approach the nonlinear system

$$A(u) = f$$
& can we use MG to solve it?
- A fundamental relation we've relied on is the linear residual equation:

$$Au - Av = f - Av \Rightarrow Ae = r.$$
- We can't rely on this now since a nonlinear $A(u)$ generally means

$$A(u) - A(v) \neq A(e).$$

CU-Boulder

171 of 322

CU-Boulder

172 of 322

The nonlinear residual equation

We still base our development around the residual equation, now the nonlinear residual equation:

$$\begin{aligned} A(u) &= f \\ \Rightarrow A(u) - A(v) &= f - A(v) \\ \Rightarrow A(u) - A(v) &= r \end{aligned}$$

How can we use this equation as the basis for a solution method?

Newton's method for scalar $F: \mathbb{R} \rightarrow \mathbb{R}$

- Best known & most important nonlinear problem solver!
- We wish to solve $F(x) = 0$.
- Ex: $F(x) = x e^x - 1, F'(x) = (1 + x) e^x$.
exponent, not error
- Expand F in a Taylor series about x :
 $F(x+s) = F(x) + sF'(x) + s^2 F''(\xi)$.
- Ex: $(x+s)e^{(x+s)} - 1 = x e^x - 1 + s(1+x) e^x + \text{h.o.t.}$
- Dropping higher-order terms (h.o.t.), if $x+s$ is a solution,
 $0 = F(x) + sF'(x) \Rightarrow s = -F(x)/F'(x)$.
- We thus arrive at Newton's method:

$$x \leftarrow x - F(x)/F'(x)$$

Newton's method for systems

- Ex: $-u''(x) + u(x) e^{u(x)} = f$ may be discretized as

$$A(u) \equiv \begin{pmatrix} \frac{2u_1 - u_2}{h^2} + u_1 e^{u_1} \\ \vdots \\ \frac{-u_{i-1} + 2u_i - u_{i+1}}{h^2} + u_i e^{u_i} \\ \vdots \\ \frac{-u_{N-1} + 2u_N}{h^2} + u_N e^{u_N} \end{pmatrix} = \begin{pmatrix} f(x_1) \\ \vdots \\ f(x_i) \\ \vdots \\ f(x_N) \end{pmatrix} = \underline{f}.$$

- Taylor series about v : $A(v+e) = A(v) + J(v)e + \text{h.o.t.}$

$$J(v) \equiv \left(\frac{\partial a(u_j)}{\partial a(u_i)} \right)_{u=v} = \begin{pmatrix} \frac{2}{h^2} + (1+v_1)e^{v_1} & -\frac{1}{h^2} & & \\ -\frac{1}{h^2} & \frac{2}{h^2} + (1+v_2)e^{v_2} & -\frac{1}{h^2} & \\ & \ddots & \ddots & \\ & -\frac{1}{h^2} & \frac{2}{h^2} + (1+v_N)e^{v_N} & \end{pmatrix}.$$

error, not exponent

Newton's method for general systems

- The system $A(u) = f$ in vector form is

$$\begin{pmatrix} a_1(u_1, u_2, \dots, u_N) \\ a_2(u_1, u_2, \dots, u_N) \\ \vdots \\ a_N(u_1, u_2, \dots, u_N) \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{pmatrix}$$

- Expanding $A(v+e)$ in a Taylor series about v :

$$A(v+e) = A(v) + J(v)e + \text{h.o.t.}$$

$$J(v) = \left(\frac{\partial a_i}{\partial u_j}(v) \right)$$

Newton for systems (cont'd)

- $J(v)$ is the Jacobian

$$J(v) = \begin{pmatrix} \frac{\partial a_1}{\partial u_1} & \frac{\partial a_1}{\partial u_2} & \dots & \frac{\partial a_1}{\partial u_N} \\ \frac{\partial a_2}{\partial u_1} & \frac{\partial a_2}{\partial u_2} & \dots & \frac{\partial a_2}{\partial u_N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial a_N}{\partial u_1} & \frac{\partial a_N}{\partial u_2} & \dots & \frac{\partial a_N}{\partial u_N} \end{pmatrix}_{u=v}$$

- If $u = v + e$ is a solution, $f = A(v) + J(v)e + \text{h.o.t.}$, so
 $e \approx [J(v)]^{-1}(f - A(v))$.

- This leads to the iteration

$$v \leftarrow v + [J(v)]^{-1}(f - A(v))$$

CU-Boulder

177 of 322

Newton's via the residual equation

- The nonlinear residual equation is

$$A(v + e) - A(v) = r.$$

- Expanding $A(v + e)$ in a two-term Taylor series about v & ignoring h.o.t.:

$$A(v) + J(v)\hat{e} - A(v) = r \quad \hat{e} \approx e$$

or

$$J(v)\hat{e} = r.$$

- Newton's method is thus:

$$v \leftarrow v + [J(v)]^{-1}r, \quad r = f - A(v)$$

CU-Boulder

178 of 322

How does multigrid fit in?

- One obvious method is to use multigrid to solve $J(v)\hat{e} = r$ at each iteration step. This method is called Newton-MG & can be very effective.
- However, we might want to use multigrid ideas to treat the nonlinearity directly.
- To do that, we need to specialize multigrid components (relaxation & coarsening) for the nonlinear case.

CU-Boulder

179 of 322

What is nonlinear relaxation?

- Several of the common relaxation schemes have nonlinear counterparts. For $A(u) = f$, we describe nonlinear Gauss-Seidel:

For each $i = 1, 2, \dots, N$:

Change the value of v_i so that the i^{th} equation is satisfied, that is, so that $(A(v))_i = f_i$.

- Equivalently:

For each $i = 1, 2, \dots, N$:

Find $s \in \mathbb{R}$ such that $(A(v + s\varepsilon_i))_i = f_i$, where ε_i is the i^{th} canonical unit basis vector.

CU-Boulder

180 of 322

How is nonlinear Gauss-Seidel done?

- Each $(A(v))_i = f_i$ is a nonlinear scalar equation for v_i . We can use the scalar Newton's method to solve!
 - Example: $-u''(x) + u(x) e^{u(x)} = f$ may be discretized so that $(A(v))_i = f_i$ is given by

$$\frac{-v_{i-1} + 2v_i - v_{i+1}}{h^2} + v_i e^{y_i} = f_i \quad 1 \leq i \leq N$$

- Newton iteration for v_i is given by

$$v_i \leftarrow v_i - \frac{\frac{-v_{i-1} + 2v_i - v_{i+1}}{h^2} + v_i e^{v_i} - f_i}{\frac{2}{h^2} + (1 + v_i) e^{v_i}}$$

CU-Boulder

181 of 322

Look at the coarse residual equation

- We must evaluate the quantities on Ω^{2h} in
$$A^{2h}(v^{2h} + e^{2h}) - A^{2h}(v^{2h}) = r^{2h}.$$
 - Given v^h , a fine-grid approximation, we restrict the residual to the coarse grid:
$$r^{2h} = I_h^{2h}(f^h - A^h(v^h)).$$
 - For v^{2h} , we restrict v^h by $v^{2h} = I_h^{2h}v^h$.

Thus,

$$A^{2h}(I_h^{2h}v^h + e^{2h}) = A^{2h}(I_h^{2h}v^h) + I_h^{2h}(f^h - A^h(v^h))$$

CU-Boulder

183 of 322

How do we do coarsening for nonlinear multigrid?

- Recall the nonlinear residual equation
$$A(v + e) - A(v) = r.$$
 - In multigrid, we obtain an approximate solution v^h on the fine grid, then solve the residual equation on the coarse grid.
 - The residual equation on Ω^{2h} appears as

$$A^{2h}(v^{2h} + e^{2h}) - A^{2h}(v^{2h}) = r^{2h}.$$

CU-Boulder

182 of 322

We've obtained a coarse-grid equation of the form $A^{2h}(\mathbf{u}^{2h}) = \mathbf{f}^{2h}$

- Consider the coarse-grid equation:

- We solve $A^{2h}(u^{2h}) = f^{2h}$ for $u^{2h} = I_h^{2h}v^h + e^{2h}$ & obtain

$$e^{2h} = u^{2h} - I_h^{2h} v^h$$

- We then apply the correction:

$$v^h \leftarrow v^h + I_{2h}^h e^{2h}$$

CU-Boulder

184 of 322

Full approximation scheme (FAS) 2-grid form

- Perform nonlinear relaxation on $A^h(u^h) = f^h$ to obtain an approximation v^h .
- Restrict the approximation & its residual:

$$v^{2h} = I_h^{2h} v^h \quad r^{2h} = I_h^{2h}(f^h - A(v^h)).$$
- Solve the coarse-grid equation:

$$A^{2h}(u^{2h}) = A^{2h}(v^{2h}) + r^{2h}.$$
- Extract $2h$ approximation to h error:

$$e^{2h} = u^{2h} - v^{2h}.$$
- Interpolate & correct:

$$v^h \leftarrow v^h + I_h^h e^{2h}.$$

CU-Boulder

185 of 322

A few observations about FAS

$$A^{2h}(I_h^{2h}v^h + e^{2h}) = A^{2h}(I_h^{2h}v^h) + I_h^{2h}(f^h - A^h(v^h))$$

- If A is a linear operator, then FAS reduces directly to the linear two-grid correction scheme:

$$A^{2h}(I_h^{2h}v^h + e^{2h}) = A^{2h}(I_h^{2h}v^h) + I_h^{2h}(f^h - A^h(v^h))$$

- An exact solution to the fine-grid problem is a fixed point of the FAS iteration:

$$A^{2h}(I_h^{2h}v^h + e^{2h}) = A^{2h}(I_h^{2h}v^h) + I_h^{2h}(f^h - A^h(v^h))$$

CU-Boulder

186 of 322

A few more observations about FAS

$$A^{2h}(I_h^{2h}v^h + e^{2h}) = A^{2h}(I_h^{2h}v^h) + I_h^{2h}(f^h - A^h(v^h))$$

- The FAS coarse-grid equation can be written as

$$A^{2h}(u^{2h}) = f^{2h} + \tau_h^{2h}$$

where $\tau_h^{2h} = A^{2h}(I_h^{2h}v^h) - I_h^{2h}A^h(v^h)$ is the so-called tau correction term & f^{2h} is the original $2h$ source term, provided you choose it that way: $f^{2h} = I_h^{2h}f^h$.
- In general, since $\tau_h^{2h} \neq 0$, the solution u^{2h} to the FAS coarse-grid equation is not the same as the solution to the original coarse-grid problem

$$A^{2h}(u^{2h}) = f^{2h}.$$
- The tau correction is as a way to alter the coarse-grid equation to enhance its approximation properties.

CU-Boulder

187 of 322

Still more observations about FAS

- A true multilevel FAS process is recursive, using FAS to solve the nonlinear Ω^{2h} problem using Ω^{4h} .
- Hence, FAS is generally employed in a V- or W-cycling scheme.

CU-Boulder

188 of 322

Even more observations about FAS

- For linear problems, we use FMG to obtain a good initial guess on the fine grid. Convergence of nonlinear iterations depends critically on having a good initial guess.
- When FMG is used for nonlinear problems, the interpolant $I_{2h}^h u^{2h}$ is generally accurate enough to be in the basin of attraction of the fine-grid solver.
- Thus, whether FAS, Newton, or Newton-multigrid is used on each level, one FMG cycle should provide a solution accurate to the level of discretization, unless the nonlinearity is extremely strong.

Intergrid transfers for FAS

- Generally speaking, the standard operators (linear interpolation, full weighting) work effectively in FAS schemes.
- For strongly nonlinear problems or for the coarse-grid approximation that is to become a fine-grid initial guess, higher-order interpolation (e.g., cubic interpolation) may be beneficial.

What is $A^{2h}(u^{2h})$ in FAS?

As in the linear case, there are two basic possibilities:

1. $A^{2h}(u^{2h})$ is determined by discretizing the nonlinear operator, $A(u)$, in the same fashion as was employed to obtain $A^h(u^h)$, except that the coarser mesh spacing is used.
2. $A^{2h}(u^{2h})$ is determined from the Galerkin condition

$$A^{2h}(u^{2h}) = I_h^{2h} A^h(I_{2h}^h u^{2h})$$

where the action of the Galerkin product can be captured in an implementable formula.

The first method is usually easier & more common.

Nonlinear problems: An example

- Consider

$$-\Delta u(x,y) + \gamma u(x,y) e^{u(x,y)} = f(x,y)$$

on the unit square, $[0,1] \times [0,1]$, with homogeneous Dirichlet boundary conditions & a regular $h = 1/128$ Cartesian grid.

- Suppose the exact solution is

$$u(x,y) = (x^2 - x^3) \sin(3\pi y).$$

Discretization of the nonlinear example

- The operator can be written (sloppily) as

$$\frac{1}{h^2} \begin{pmatrix} -1 & -1 \\ -1 & 4 & -1 \\ -1 & & -1 \end{pmatrix} u_{i,j}^h + \gamma u_{i,j}^h e^{u_{i,j}^h} = f_{i,j}$$

$A^h(u^h)$

- Relaxation (nonlinear Gauss-Seidel) is given by

$$v_{i,j}^h \leftarrow v_{i,j}^h - \frac{(A^h(v^h))_{i,j} - f_{i,j}}{\frac{4}{h^2} + \gamma(1 + v_{i,j}^h)e^{v_{i,j}^h}}$$

FAS & Newton's method on

$$-\Delta u(x,y) + \gamma u(x,y) e^{u(x,y)} = f(x,y)$$

$$N = 127$$

- FAS V(2,1)-cycles until $\|r\| < 10^{-10}$.

| | γ | | | |
|----------------------|----------|-------|-------|-------|
| | 1 | 10 | 100 | 1000 |
| convergence factor | 0.135 | 0.124 | 0.098 | 0.072 |
| number of FAS cycles | 12 | 11 | 11 | 10 |

- Newton's Method with exact inner solves until $\|r\| < 10^{-10}$.

| | γ | | | |
|-----------------------------|----------|----------|----------|----------|
| | 1 | 10 | 100 | 1000 |
| convergence factor | 4.00E-05 | 7.00E-05 | 3.00E-04 | 2.00E-04 |
| number of Newton iterations | 3 | 3 | 3 | 4 |

Newton, Newton-MG, & FAS on

$$-\Delta u(x,y) + \gamma u(x,y) e^{u(x,y)} = f(x,y)$$

$$N = 127, \gamma = 10$$

- Newton uses exact solves, Newton-MG is inexact Newton with a fixed number of inner V(2,1)-cycles for the Jacobian problem, overall stopping criterion $\|r\| < 10^{-10}$.

| Method | Outer iterations | Inner iterations | Megaflops |
|-----------|------------------|------------------|-----------|
| Newton | 3 | | 1660.6 |
| Newton-MG | 3 | 20 | 56.4 |
| Newton-MG | 4 | 10 | 38.5 |
| Newton-MG | 5 | 5 | 25.1 |
| Newton-MG | 10 | 2 | 22.3 |
| Newton-MG | 19 | 1 | 24.6 |
| FAS | 11 | | 27.1 |

Compare FMG-FAS & FMG-Newton-MG

$$-\Delta u(x,y) + \gamma u(x,y) e^{u(x,y)} = f(x,y)$$

- We will do one FMG cycle using a single FAS V(2,1)-cycle as the "solver" at each new level. We then follow that with as many FAS V(2,1)-cycles as is needed to obtain $\|r\| < 10^{-10}$.

- Next, we will do one FMG cycle using a Newton-MG step at each new level (with one linear V(2,1)-cycle as the Jacobian "solver.") We then follow that with as many Newton-multigrid steps as is needed to obtain $\|r\| < 10^{-10}$.

Compare FMG-FAS & FMG-Newton-MG

$$-\Delta u(x,y) + \gamma u(x,y) e^{u(x,y)} = f(x,y)$$

$$N=127, \gamma=10$$

| Cycle | $\ r^h\ $ | $\ e^h\ $ | Mflops | Cycle | $\ r^h\ $ | $\ e^h\ $ | Mflops | Cycle |
|---------|-----------|-----------|--------|-------|-----------|-----------|--------|------------|
| FMG-FAS | 1.10E-02 | 2.00E-05 | 3.1 | | 1.06E-02 | 2.50E-05 | 2.4 | FMG-Newton |
| FAS V | 6.80E-04 | 2.40E-05 | 5.4 | | 6.70E-04 | 2.49E-05 | 4.1 | Newton-MG |
| FAS V | 5.00E-05 | 2.49E-05 | 7.6 | | 5.10E-05 | 2.49E-05 | 5.8 | Newton-MG |
| FAS V | 3.90E-06 | 2.49E-05 | 9.9 | | 6.30E-06 | 2.49E-05 | 7.5 | Newton-MG |
| FAS V | 3.20E-07 | 2.49E-05 | 12.2 | | 1.70E-06 | 2.49E-05 | 9.2 | Newton-MG |
| FAS V | 3.00E-08 | 2.49E-05 | 14.4 | | 5.30E-07 | 2.49E-05 | 10.9 | Newton-MG |
| FAS V | 2.90E-09 | 2.49E-05 | 16.7 | | 1.70E-07 | 2.49E-05 | 12.6 | Newton-MG |
| FAS V | 3.00E-10 | 2.49E-05 | 18.9 | | 5.40E-08 | 2.49E-05 | 14.3 | Newton-MG |
| FAS V | 3.20E-11 | 2.49E-05 | 21.2 | | 1.70E-08 | 2.49E-05 | 16.0 | Newton-MG |
| | | | | | 5.50E-09 | 2.49E-05 | 17.7 | Newton-MG |
| | | | | | 1.80E-09 | 2.49E-05 | 19.4 | Newton-MG |
| | | | | | 5.60E-10 | 2.49E-05 | 21.1 | Newton-MG |
| | | | | | 1.80E-10 | 2.49E-05 | 22.8 | Newton-MG |
| | | | | | 5.70E-11 | 2.49E-05 | 24.5 | Newton-MG |

Remembering coarse-grid correction

$$Au = f$$

- Relax (damped Jacobi) to smooth $e = u - v$:

$$v \leftarrow v - \omega D^{-1}(Av - f).$$

- Form the residual equation $Ae = r$:

$$Ae = A(u - v) = f - Av = r.$$

- Use premise that smooth error $\Rightarrow e = I_{2h}^h e^{2h}$:

$$AI_{2h}^h e^{2h} = r.$$

fewer unknowns

- Use transpose $I_{2h}^{2h} = (I_{2h}^h)^T$ to reduce equations:

$$I_h^{2h} AI_{2h}^h e^{2h} = I_h^{2h} r.$$

fewer equations

$$A^{2h}$$

Motivating FAS for nonlinear A

$$A(v + e) = f \rightarrow A^{2h}(v^{2h} + e^{2h}) = "f^{2h}"$$

$$A^{2h}(I_h^{2h} v^{2h} + e^{2h}) = A^{2h}(I_h^{2h} v^{2h}) + I_h^{2h}(f^{2h} - A^h(v^{2h}))$$

? ?

- What overriding principle can we find to get from h to $2h$?
- With known v , how do we discretize a PDE of the form

$$A(v + e) = f?$$

- Example:

$$\gamma uu' - u'' = f \rightarrow \gamma(v + e)(v + e)' - (v + e)'' = f.$$

- Expand to get an equation in e of form $g + ae + be' + \dots$:

$$\underbrace{\gamma vv' - v''}_{A(v)} + \gamma(v'e + ve' + ee') - e'' = f.$$

Differential residual equation

$$\gamma(v'e + ve' + ee') - e'' = f - A(v) \equiv r(v)$$

- Right side (analogous to injection):

$$r(v) \rightarrow r_j^h = r(x_j).$$

- Coefficients:

$$v \rightarrow v_j^h = v(x_j), \quad v' \rightarrow (v')_j^h = (v(x_{j+1}) - v(x_{j-1}))/2h.$$

- Unknowns:

$$e \rightarrow e_j^h, \quad e' \rightarrow (e')_j^h = (e_{j+1}^h - e_{j-1}^h)/2h,$$

$$e'' \rightarrow (e'')_j^h = (e_{j+1}^h - 2e_j^h + e_{j-1}^h)/h^2.$$

Leading to FAS...

$$\gamma u u' - u'' = f \rightarrow A(v+e) = f \rightarrow \gamma(v'e + ve' + ee') - e'' = f - A(v)$$

- Fine-grid residual equation (at h point $2j$):

$$\gamma \frac{v_{2j+1}^h - v_{2j-1}^h}{2h} e_{2j}^h + \gamma v_{2j}^h \frac{e_{2j+1}^h - e_{2j-1}^h}{2h} + \gamma e_{2j}^h \frac{e_{2j+1}^h - e_{2j-1}^h}{2h} + \frac{-e_{2j+1}^h + 2e_{2j}^h - e_{2j-1}^h}{h^2} = r_{2j}^h$$

- Coarse-grid residual equation (at $2h$ point j):

$$\gamma \frac{v_{2j+2}^h - v_{2j-2}^h}{4h} e_{2h}^h + \gamma v_{2j}^h \frac{e_{2h+1}^h - e_{2h-1}^h}{4h} + \gamma e_{2h}^h \frac{e_{2h+1}^h - e_{2h-1}^h}{4h} + \frac{-e_{2h+1}^h + 2e_{2h}^h - e_{2h-1}^h}{(2h)^2} = (I_h^{2h} r^h)_j$$

- FAS

$$A^{2h} (I_h^{2h} v^h + e^{2h}) - A^{2h} (I_h^{2h} v^h) = I_h^{2h} (f^h - A^h(v^h))$$

One Newton-MG step

- Step 0. Given v , form the grid h linear correction equation:

$$\frac{-e_{j-1} + 2e_j - e_{j+1}}{h^2} + (1+v_j)e^{v_j} e_j = r_j \equiv f_j - \left(\frac{-v_{j-1} + 2v_j - v_{j+1}}{h^2} + v_j e^{v_j} \right)$$

Initialize the Newton correction approximation: $e = 0$.

- Step 1: Relax on the grid h linear equation.

- Step 2: Solve the grid $2h$ error correction equation:

$$\frac{-e_{j-1}^{2h} + 2e_j^{2h} - e_{j+1}^{2h}}{(2h)^2} + (1+v_{2j})e^{v_{2j}} e_j^{2h} = r_{2j} - \left(\frac{-e_{2j-1} + 2e_{2j} - e_{2j+1}}{h^2} + (1+v_{2j})e^{v_{2j}} e_j \right)$$

- Step 3: Correct the grid h Newton correction:

$$e \leftarrow e + I_h^h e^{2h}$$

- Step 4: Stop if you've "solved" the linear equation well enough for Newton correction e & set $v \leftarrow v + e$. Else, leave v alone & return to Step 1.

Example: Newton-MG vs. FAS

- PDE (er, ODE):

$$-u''(x) + u(x) e^{u(x)} = f(x).$$

- Discretization:

$$\frac{-v_{j-1} + 2v_j - v_{j+1}}{h^2} + v_j e^{v_j} = f_j.$$

One FAS step

- Step 0. Given v , form the grid h nonlinear equation:

$$\frac{-v_{j-1} + 2v_j - v_{j+1}}{h^2} + v_j e^{v_j} = f_j$$

- Step 1: Relax on the grid h nonlinear equation to improve v .

- Step 2: Solve the grid $2h$ FAS correction equation:

$$\frac{-e_{j-1}^{2h} + 2e_j^{2h} - e_{j+1}^{2h}}{(2h)^2} + (v_{2j}^h + e_j^{2h})e^{v_{2j}^h + e_j^{2h}} - v_{2j}^h e^{v_{2j}^h} = (I_h^{2h} r^h)_j$$

- Step 3: Correct the grid h approximation v :

$$v \leftarrow v + I_{2h}^h e^{2h}$$

Outline

Chapters 1-5:

- ✓ Model Problems
- ✓ Basic Iterative Methods
 - Convergence tests
 - Analysis
- ✓ Elements of Multigrid
 - Relaxation
 - Coarsening
- ✓ Implementation
 - Complexity
 - Diagnostics
- ✓ Some Theory
 - Spectral vs. algebraic

Homework Due Computing Too!

CU-Boulder

205 of 322

Chapters 6-10:

- ✓ Nonlinear Problems
 - Full approximation scheme
- ✓ Selected Applications
 - Neumann boundaries
 - Anisotropic problems
 - Variable meshes
 - Variable coefficients
- Algebraic Multigrid (AMG)
 - Matrix coarsening
- Multilevel Adaptive Methods
 - FAC
- Finite Elements
 - Variational methodology

7. Selected applications

7a. Neumann boundary conditions

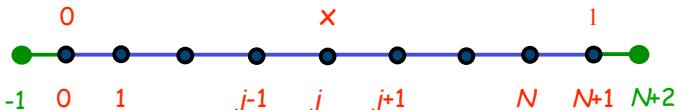
- Consider the 1-D problem

$$-u''(x) = f(x), \quad 0 < x < 1,$$

$$u'(0) = u'(1) = 0.$$

- We discretize on the interval $[0,1]$ with $h = 1/(N+1)$ grid spacing & nodes $x_j = jh$, $j = 0, 1, 2, \dots, N+1$.

- We extend the interval with two ghost points:



CU-Boulder

206 of 322

Central differences at boundary

- We use differences as before, but now also for the derivative in the Neumann condition:

$$u'(0) \approx \frac{u_1 - u_{-1}}{2h}$$

$$u''(x_j) \approx \frac{-u_{j-1} + 2u_j - u_{j+1}}{h^2}$$

$$u'(1) \approx \frac{u_{N+2} - u_N}{2h}$$

- This yields the system

$$\frac{-u_{j-1} + 2u_j - u_{j+1}}{h^2} = f_j \quad 0 \leq j \leq N+1$$

$$\frac{u_1 - u_{-1}}{2h} = 0$$

$$\frac{u_{N+2} - u_N}{2h} = 0$$

Eliminating the ghost points

- Use the boundary conditions to eliminate u_{-1}, u_{N+2}

$$\frac{u_1 - u_{-1}}{2h} = 0 \implies u_{-1} = u_1 \quad \frac{u_{N+2} - u_N}{2h} = 0 \implies u_{N+2} = u_N$$

- Eliminating the ghost points in the $j=0$ & $j=N+1$ equations gives the $(N+2) \times (N+2)$ system of equations:

$$\frac{-u_{j-1} + 2u_j - u_{j+1}}{h^2} = f_j \quad 1 \leq j \leq N$$

$$\frac{2u_0 - 2u_1}{h^2} = f_0$$

$$\frac{-2u_N + 2u_{N+1}}{h^2} = f_{N+1}$$

CU-Boulder

207 of 322

CU-Boulder

208 of 322

Write the system in matrix form

- We can write $A^h u^h = f^h$, where

$$A^h = \frac{1}{h^2} \begin{pmatrix} 2 & -2 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -2 & 2 \end{pmatrix}.$$

- Note that A^h is $(N+2) \times (N+2)$ & nonsymmetric, & the system involves unknowns u_0^h & u_{N+1}^h at the boundaries.

We must consider compatibility

- The problem $-u''(x) = f(x)$, for $0 < x < 1$, with $u'(0) = u'(1) = 0$, is not well-posed!
- If $u(x)$ is a solution, then so is $u(x) + \text{constant}$.
- We cannot be certain a solution exists. If one does, it must satisfy

$$-\int_0^1 u''(x) dx = \int_0^1 f(x) dx \quad \Rightarrow \quad -[u'(1) - u'(0)] = \int_0^1 f(x) dx$$

$0 = \int_0^1 f(x) dx$

- This integral compatibility condition is necessary! If $f(x)$ doesn't satisfy it, there is no solution!

The well-posed system

- The compatibility condition is necessary for a solution to exist. In general, it is also sufficient:
 $-\partial^2/\partial x^2$ is a well-behaved operator on the space of functions $u(x)$ that have zero mean.
- Thus, we may conclude that if $f(x)$ satisfies the compatibility condition, then the problem is well-posed:

$$\begin{aligned} -u''(x) &= f(x), \quad 0 < x < 1, \\ u'(0) &= u'(1) = 0, \\ \int_0^1 u(x) dx &= 0. \end{aligned}$$

- The last says: of all possible solutions $u(x) + \text{constant}$, we choose the one with zero mean.

The discrete problem is not well posed

- Since all row sums of A^h are zero, then $1^h \in N(A^h)$.
- It's easy to see that $\dim(N(A^h)) = 1$, so $N(A^h) = \text{span}\{1^h\}$.
- By the Fundamental Theorem of Linear Algebra, $A^h u^h = f^h$ has a solution if & only if $f^h \in N((A^h)^T)^\perp$.
- It is easy to show that $N((A^h)^T) = c(1/2, 1, 1, \dots, 1, 1/2)^T$.
- Thus, $A^h u^h = f^h$ has a solution if & only if $f^h \perp c(1/2, 1, 1, \dots, 1, 1/2)^T$.
- That is,

$\frac{1}{2}f_0^h + \sum_{j=1}^N f_j^h + \frac{1}{2}f_{N+1}^h = 0$

We have two issues to consider

$$A^h u^h = f^h$$

- Solvability: A solution exists iff $f^h \in N((A^h)^T)^\perp$.
- Uniqueness: If u^h is a solution, then so is $u^h + v^h$ for any $v^h \in N(A^h)$.
- Note that if $A^h = (A^h)^T$, then $N((A^h)^T) = N(A^h)$ & solvability & uniqueness can be handled together.
- This is easily done. Multiply the first & last equations by 1/2, giving

$$\hat{A}^h = \frac{1}{h^2} \begin{pmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & 2 & -1 & \\ & & \ddots & \ddots & \ddots \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 & -1 \\ & & & & & 1 & \end{pmatrix}$$

CU-Boulder

213 of 322

The new system is symmetric

- We have the symmetric system $\hat{A}^h u^h = \hat{f}^h$:

$$\frac{1}{h^2} \begin{pmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & 2 & -1 & \\ & & \ddots & \ddots & \ddots \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 & -1 \\ & & & & & 1 & \end{pmatrix} \begin{pmatrix} u_0^h \\ u_1^h \\ u_2^h \\ \vdots \\ u_N^h \\ u_{N+1}^h \end{pmatrix} = \begin{pmatrix} f^{h/2} \\ f_1^h \\ f_2^h \\ \vdots \\ f_N^h \\ f_{N+1/2}^h \end{pmatrix}$$

- Solvability is guaranteed by ensuring that \hat{f}^h is orthogonal to the constant vector 1^h :

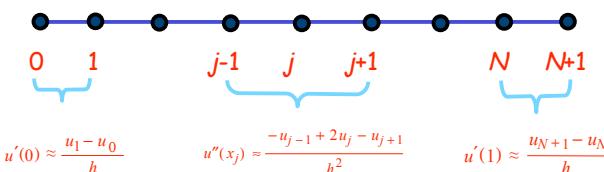
$$\left\langle \hat{f}^h, 1^h \right\rangle = \sum_{j=0}^{N+1} \hat{f}_j^h = 0$$

CU-Boulder

214 of 322

One-sided differences at boundary a similar result

- No ghost points:



- This yields the system

$$\frac{-u_{j-1} + 2u_j - u_{j+1}}{h^2} = f_j \quad 1 \leq j \leq N$$

$$\frac{u_1 - u_0}{h^2} = 0 \quad \frac{u_{N+1} - u_N}{h^2} = 0$$

CU-Boulder

215 of 322

The well-posed discrete system

- The $(N+3) \times (N+2)$ system is:

$$\frac{-u_{j-1} + 2u_j - u_{j+1}}{h^2} = f_j \quad 1 \leq j \leq N$$

$\frac{u_0 - u_1}{h^2} = f_0$ for one-sided scheme

$$\frac{-u_N + u_{N+1}}{h^2} = f_{N+1}$$

$\sum_{i=0}^{N+1} u_i^h = 0$ (choose the zero mean solution)

or, more simply

$$\hat{A}^h u^h = \hat{f}^h$$

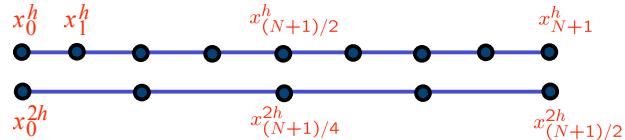
$$\left\langle u^h, 1^h \right\rangle = 0$$

CU-Boulder

216 of 322

Multigrid for the Neumann problem

- We must have the interval endpoints on all grids



- Relaxation is performed at all points, including endpoints:

$$v_0^h \leftarrow v_0^h + h^2 \hat{f}_0^h \quad v_j^h \leftarrow \frac{v_{j-1}^h + v_{j+1}^h + h^2 \hat{f}_j^h}{2} \quad v_{N+1}^h \leftarrow v_N^h + h^2 \hat{f}_{N+1}^h$$

- We add a global Gram-Schmidt step after relaxation on each level to enforce the zero-mean condition:

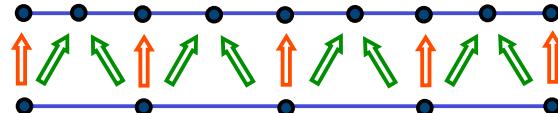
$$v^h \leftarrow v^h - \frac{\langle v^h, 1^h \rangle}{\langle 1^h, 1^h \rangle} 1^h$$

CU-Boulder

217 of 322

Interpolation must include the endpoints

- We use linear interpolation:



$$I_{2h}^h = \begin{pmatrix} 1 & & & & \\ 1/2 & 1/2 & & & \\ & 1 & & & \\ & 1/2 & 1/2 & & \\ & & 1 & & \\ & & 1/2 & 1/2 & \\ & & & 1 & \\ & & & 1/2 & 1/2 \\ & & & & 1 \end{pmatrix}$$

CU-Boulder

218 of 322

Restriction also treats the endpoints

For restriction, we use $I_h^{2h} = \frac{1}{2}(I_{2h}^h)^T$, yielding the values

$$\hat{f}_0^{2h} = \frac{1}{2}\hat{f}_0^h + \frac{1}{4}\hat{f}_1^h$$

$$\hat{f}_j^{2h} = \frac{1}{4}\hat{f}_{2j-1}^h + \frac{1}{2}\hat{f}_{2j}^h + \frac{1}{4}\hat{f}_{2j+1}^h$$

$$\hat{f}_{N+1}^{2h} = \frac{1}{4}\hat{f}_N^h + \frac{1}{2}\hat{f}_{N+1}^h$$

CU-Boulder

219 of 322

The coarse-grid operator

- We compute the coarse-grid operator using the Galerkin condition

$$\hat{A}^{2h} = I_h^{2h} \hat{A}^h I_{2h}^h.$$

| $j^{2h} = 0$ | 1 |
|--|--------------------------------------|
| \hat{e}_0^{2h} | 1 0 |
| $I_{2h}^h \hat{e}_0^{2h}$ | 1 $\frac{1}{2}$ 0 |
| $\hat{A}^h I_{2h}^h \hat{e}_0^{2h}$ | $\frac{1}{2h^2}$ 0 $-\frac{1}{2h^2}$ |
| $I_h^{2h} \hat{A}^h I_{2h}^h \hat{e}_0^{2h}$ | $\frac{1}{4h^2}$ $-\frac{1}{4h^2}$ |

$$\hat{A}^{2h} = \frac{1}{4h^2} \begin{pmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & 2 & -1 & \\ & & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 1 \end{pmatrix}$$

CU-Boulder

220 of 322

Coarse-grid solvability

- Assuming \hat{f}^h satisfies $\langle \hat{f}^h, 1^h \rangle = 0$, the solvability condition, we can show that theoretically the coarse-grid problem $\hat{A}^{2h} u^{2h} = I_h^{2h} (\hat{f}^h - \hat{A}^h v^h)$ is also solvable.
- To be certain numerical round-off does not perturb solvability, we incorporate a Gram-Schmidt-like step each time a new right-hand side \hat{f}^{2h} is generated for the coarse grid:

$$\hat{f}^{2h} \leftarrow \hat{f}^{2h} - \frac{\langle \hat{f}^{2h}, 1^{2h} \rangle}{\langle 1^{2h}, 1^{2h} \rangle} 1^{2h}.$$

CU-Boulder

221 of 322

V(2,1)
cycles

Neumann problem: An example

Consider the problem

$$-u''(x) = 2x - 1, \quad 0 < x < 1, \quad u'(0) = u'(1) = 0,$$

which has $u(x) = \frac{x^2}{2} - \frac{x^3}{3} + c$ as a solution for any c ($c = -1/12$ gives the zero mean solution).

| grid size N | $\ r^h\ $ | average conv. factor | $\ e^h\ $ | number of cycles |
|------------------|-----------|-------------------------|-----------|---------------------|
| 31 | 6.30E-11 | 0.079 | 9.70E-05 | 9 |
| 63 | 1.90E-11 | 0.089 | 2.40E-05 | 10 |
| 127 | 2.60E-11 | 0.093 | 5.90E-06 | 10 |
| 255 | 3.70E-11 | 0.096 | 1.50E-06 | 10 |
| 511 | 5.70E-11 | 0.100 | 3.70E-07 | 10 |
| 1027 | 8.60E-11 | 0.104 | 9.20E-08 | 10 |
| 2047 | 2.10E-11 | 0.112 | 2.30E-08 | 10 |
| 4095 | 5.20E-11 | 0.122 | 5.70E-09 | 10 |

CU-Boulder

222 of 322

7. Selected applications

7b. Anisotropic problems

- All problems considered thus far have had $-h^2$ as the off-diagonal entries.
- We consider two situations when the matrix has two different constants on the off-diagonals. These situations arise when
 - the (2-d) differential equation has constant but different coefficients for the derivatives in the coordinate directions
 - the discretization has constant but different mesh spacing in the different coordinate directions

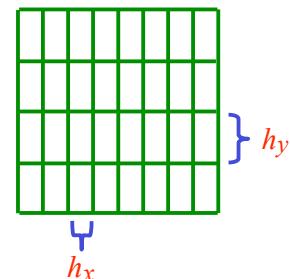
CU-Boulder

223 of 322

We consider two types of anisotropy

- Different coefficients on the derivatives

$$-\alpha_{xx} u_{xx} - \alpha_{yy} u_{yy} = f$$
 discretized on a uniform grid with spacing h .
- Different mesh spacings:



$$h_x = h = \frac{1}{N+1}$$

$$h_y = \frac{h_x}{\sqrt{\alpha}}$$

CU-Boulder

224 of 322

Both problems lead to the same stencil

$$\frac{-u_{j-1,k} + 2u_{j,k} - u_{j+1,k}}{h^2} + \alpha \frac{-u_{j,k-1} + 2u_{j,k} - u_{j,k+1}}{h^2}$$



$$A^h = \frac{1}{h^2} \begin{pmatrix} -\alpha & & \\ -1 & 2+2\alpha & -1 \\ & -\alpha & \end{pmatrix}$$

$$\frac{-u_{j-1,k} + 2u_{j,k} - u_{j+1,k}}{h^2} + \frac{-u_{j,k-1} + 2u_{j,k} - u_{j,k+1}}{\left(\frac{h}{\sqrt{\alpha}}\right)^2}$$



CU-Boulder

225 of 322

Why standard multigrid fails

$$A^h = \frac{1}{h^2} \begin{pmatrix} -\alpha & & \\ -1 & 2+2\alpha & -1 \\ & -\alpha & \end{pmatrix}$$

- Note that A^h has weak connections in the y -direction. MG convergence factors degrade as α gets small, with poor performance already at $\alpha = 0.1$.
- Consider the limiting case $\alpha \Rightarrow 0$: $A^h \Rightarrow \frac{1}{h^2} \begin{pmatrix} 0 & & \\ -1 & 2 & -1 \\ & 0 & \end{pmatrix}$
- Collection of disconnected 1-D problems!
- Point relaxation will smooth oscillatory errors in the x -direction (strong connections), but with no connections in the y -direction, the errors in that direction will generally be random; point relaxation provides no smoothing in the y -direction.

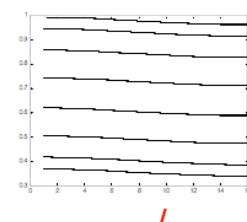
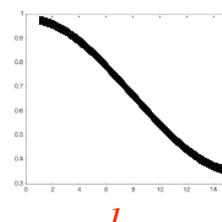
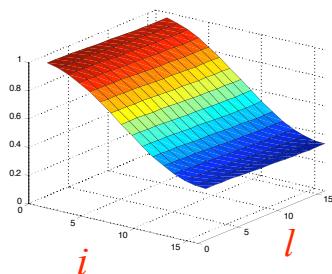
CU-Boulder

226 of 322

We analyze weighted Jacobi

- The eigenvalues of the weighted Jacobi iteration matrix for this problem are

$$\lambda_{i,l} = 1 - \frac{2\omega}{1+\alpha} \left(\sin^2 \left(\frac{i\pi}{2(N+1)} \right) + \alpha \sin^2 \left(\frac{l\pi}{2(N+1)} \right) \right)$$



CU-Boulder

227 of 322

Two strategies for anisotropy

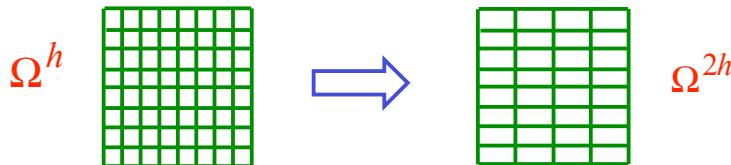
- Semicoarsening:** Because we expect MG-like convergence for the 1-D problems along lines of constant y , we could coarsen the grid in the x -direction, but not in the y -direction.
- Line relaxation:** Because the equations are strongly coupled in the x -direction, we could solve simultaneously for entire lines of unknowns in the x -direction (along lines of constant y).

CU-Boulder

228 of 322

Semicoarsening with point relaxation

- Point relaxation on $A^h = \frac{1}{h^2} \begin{pmatrix} -\alpha & & \\ -1 & 2+2\alpha & -1 \\ & -\alpha & \end{pmatrix}$ smoothes in the x -direction. Coarsen by removing every other y -line.



- We do not coarsen along the remaining y -lines.
- Semicoarsening is not as "fast" as full coarsening. The number of points on Ω^{2h} is about half the number of points on Ω^{2h} , instead of the usual one-fourth.

CU-Boulder

229 of 322

Interpolation with semicoarsening

- We interpolate in the 1-D way along each line of constant y .
- The formulas for interpolating the correction from the coarse to the fine grid for the 2D model problem are

$$v_{2j,k}^h = v_{2j,k}^h + v_{j,k}^{2h}$$

$$v_{2j+1,k}^h = v_{2j+1,k}^h + \frac{v_{j,k}^{2h} + v_{j+1,k}^{2h}}{2}$$

CU-Boulder

230 of 322

Line relaxation with full coarsening

- The other approach to this problem is to do full coarsening, but to relax entire x -lines (constant y) of variables simultaneously.
- Instead of lexicographic ordering, it's easier here to order by x -lines. Write A^h in block form as

$$A^h = \begin{pmatrix} D & -cI & & \\ -cI & D & -cI & \\ & -cI & D & -cI \\ & & \ddots & \ddots & -cI \\ & & & -cI & D \end{pmatrix}$$

where

$$c = \frac{\alpha}{h^2} \quad \&$$

$$D = \frac{1}{h^2} \begin{pmatrix} 2+2\alpha & -1 & & \\ -1 & 2+2\alpha & -1 & \\ & & \ddots & \\ & & -1 & 2+2\alpha \end{pmatrix}.$$

CU-Boulder

231 of 322

Line relaxation

- One sweep of line relaxation consists of solving a tridiagonal system for each line of constant y .
- The k^{th} such system has the form $D v_k^h = g_k^h$, where v_k^h is the k^{th} subvector of v^h with entries $(v_k^h)_j = v_{j,k}^h$ & the k^{th} right-hand side subvector is

$$(g_k^h)_j = f_{j,k}^h + \frac{\alpha}{h^2} (v_{j,k-1}^h + v_{j,k+1}^h).$$

- Because D is tridiagonal, the k^{th} system can be solved very efficiently.

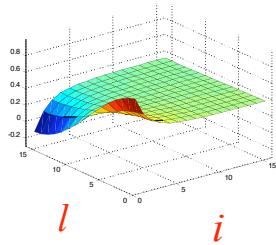
CU-Boulder

232 of 322

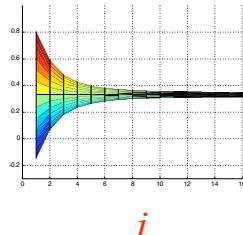
Why line relaxation works

- The eigenvalues of the weighted block Jacobi iteration matrix are

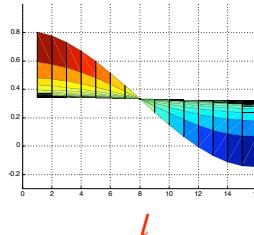
$$\lambda_{i,l} = 1 - \frac{2\omega}{2\sin^2\left(\frac{i\pi}{2(N+1)}\right) + \alpha} \left(\sin^2\left(\frac{i\pi}{2(N+1)}\right) + \alpha \sin^2\left(\frac{l\pi}{2(N+1)}\right) \right)$$



CU-Boulder



233 of 322



Semicoarsening & line relaxation

- We might not know the direction of weak coupling or it might vary.
- Suppose we want a method that can handle either

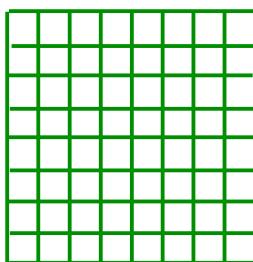
$$A_1^h = \frac{1}{h^2} \begin{pmatrix} -\alpha & & \\ -1 & 2+2\alpha & -1 \\ & -\alpha & \end{pmatrix} \quad \text{or} \quad A_2^h = \frac{1}{h^2} \begin{pmatrix} -1 & & \\ -\alpha & 2+2\alpha & -\alpha \\ & -1 & \end{pmatrix}$$

- We could use semicoarsening in the x -direction to handle A_1^h & line relaxation in the y -direction to take care of A_2^h .

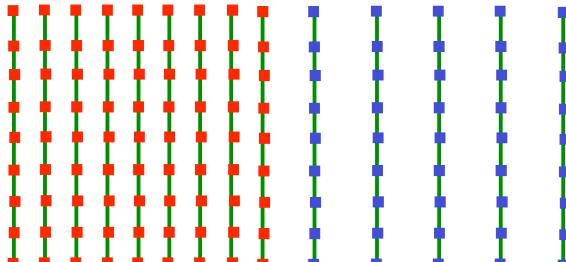
CU-Boulder

234 of 322

Semicoarsening & line relaxation



- The original grid.



- Original grid viewed as a stack of "pencils." Line relaxation is used to solve problem along each "pencil".
- Coarsening is done by deleting every other pencil.

An anisotropic example

- Consider $-u_{xx} - \alpha u_{yy} = f$ with $u = 0$ on the boundaries of the unit square, & stencil given by

$$A^h = \frac{1}{h^2} \begin{pmatrix} -\alpha & & \\ -1 & 2+2\alpha & -1 \\ & -\alpha & \end{pmatrix}.$$

- Suppose that $f(x,y) = 2(y - y^2) + 2\alpha(x - x^2)$ so that the exact solution is $u(x,y) = (y - y^2)(x - x^2)$.
- Note: If α is small, then the x -direction dominates, while if α is large, then the y -direction dominates.

CU-Boulder

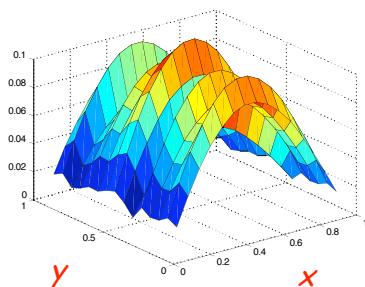
235 of 322

CU-Boulder

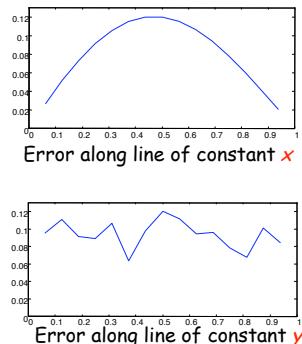
236 of 322

What is smooth error?

- Consider $\alpha = 0.001$ & suppose point Gauss-Seidel is applied to a random initial guess. The error after 50 sweeps appears as:



CU-Boulder



237 of 322

We experiment with 3 methods

- Standard V(2,1)-cycling, with point Gauss-Seidel relaxation, full coarsening, & linear interpolation
- Semicoarsening in the x -direction. Coarse & fine grids have the same number of points in the y -direction. 1-D full weighting & linear interpolation are used in the x -direction, with no y -coupling in the intergrid transfers
- Semicoarsening in the x -direction combined with line relaxation in the y -direction. 1-D full weighting & interpolation.

CU-Boulder

238 of 322

With semicoarsening, the operator must change

- To account for unequal mesh spacing, the residual & relaxation operators must use a modified stencil

$$A = \begin{pmatrix} -\frac{\alpha}{h_y^2} & & \\ & -\frac{1}{h_x^2} \left(\frac{2}{h_x^2} + \frac{2\alpha}{h_y^2} \right) - \frac{1}{h_x^2} & \\ & -\frac{1}{h_y^2} & \end{pmatrix}.$$

- Note that as grids become coarser, h_x grows while h_y remains constant.

CU-Boulder

239 of 322

How do the 3 methods work for various values of α ?

$N = 15$

- Asymptotic convergence factors:

| scheme | 1000 | 100 | 10 | 1 | 0.1 | 0.01 | 0.001 | 1E-04 | α |
|--------------------|------|------|------|------|------|------|-------|-------|----------|
| V(2,1)-cycles | 0.95 | 0.94 | 0.58 | 0.13 | 0.58 | 0.90 | 0.95 | 0.95 | |
| semiC in x | 0.94 | 0.99 | 0.98 | 0.93 | 0.71 | 0.28 | 0.07 | 0.07 | |
| semiC / line relax | 0.04 | 0.08 | 0.08 | 0.08 | 0.07 | 0.07 | 0.08 | 0.08 | |

y -direction strong x -direction strong

- Note: semicoarsening in x works well for $\alpha < .001$ but degrades noticeably even at $\alpha = .1$.

CU-Boulder

240 of 322

A semicoarsening subtlety

- Suppose α is small, so that semicoarsening in x is used. As we progress to coarser grids, h_x^{-2} gets small but h_y^{-2} remains constant.
- If, on some coarse grid, h_x^{-2} becomes comparable to αh_y^{-2} , then the problem effectively becomes recoupled in the y -direction. Continued semicoarsening can produce artificial anisotropy, strong in the y -direction.
- When this occurs, it is best to stop semicoarsening & use full coarsening on any further coarse grids.

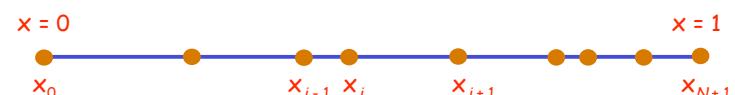
7. Selected applications

7c. Variable meshes

- Non-uniform grids are commonly used for domain or data irregularities or emerging solution features.
- Consider how we might approach the 1-D problem

$$\begin{aligned} u''(x) &= f(x), \quad 0 < x < 1, \\ u(0) &= u(1) = 0 \end{aligned}$$

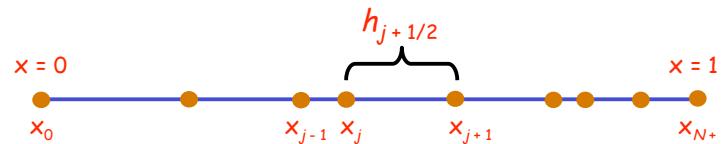
posed on the following nonuniform grid:



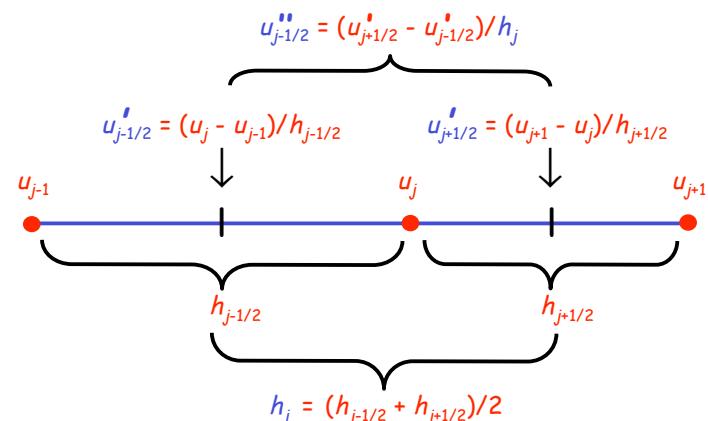
We need some notation for the mesh spacing

Let N be a positive integer. We define the spacing interval between x_j & x_{j+1} :

$$h_{j+1/2} \equiv x_{j+1} - x_j, \quad j = 0, 1, \dots, N.$$



Building second divided differences



We define the discrete differential operator

- Using second-order finite differences (& wading through a mess of algebra!), we obtain the discrete representation

$$-\alpha_j^h u_{j-1}^h + (\alpha_j^h + \beta_j^h) u_j^h - \beta_j^h u_{j+1}^h = f_j^h \quad 1 \leq j \leq N$$

$$u_0^h = u_{N+1}^h = 0$$

where

$$\alpha_j^h = \frac{2}{h_{j-1/2}(h_{j-1/2} + h_{j+1/2})} \quad \& \quad \beta_j^h = \frac{2}{h_{j+1/2}(h_{j-1/2} + h_{j+1/2})}.$$

- Multiply by $(h_{j-1/2} + h_{j+1/2})/2$ yields an SPD matrix.

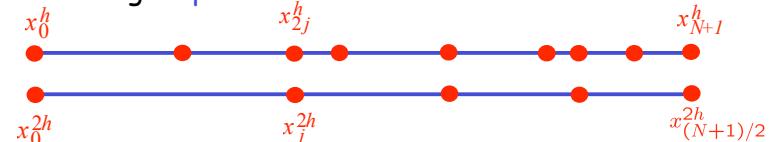
Interpolation formula

If $v^h = I_{2h}^h v^{2h}$, then, for $1 \leq j \leq (N-1)/2$, we have

$$v_{2j}^h = v_j^{2h}, \quad v_{2j+1}^h = \frac{h_{2j+3/2} v_j^{2h} + h_{2j+1/2} v_{j+1}^{2h}}{h_{2j+1/2} + h_{2j+3/2}}.$$

We modify standard multigrid to accommodate variable spacing

- We choose every other fine-grid point as a coarse-grid point:



- In $[x_0^h, x_2^h]$, linear means

$$v(x) = v_0^{2h} + (v_1^{2h} - v_0^{2h})(x - x_0^h)/(x_2^h - x_0^h).$$

- Now just plug in $x = x_1^h$ to get...

We use the variational properties to derive restriction & A^{2h}

$$A^{2h} = I_h^{2h} A^h I_{2h}^h \quad I_h^{2h} = \frac{1}{2} (I_{2h}^h)^T$$

- This produces a stencil on Ω^{2h} that is similar, but not identical, to the fine-grid stencil. If the resulting system is scaled by $(h_{j-1/2} + h_{j+1/2})$, then the Galerkin product is the same as the fine-grid stencil.
- For 2-D problems, this approach can be generalized readily to tensor-product grids. However, for general irregular grids, AMG is a better choice.

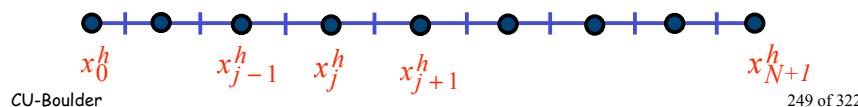
7. Selected applications

7d. Variable coefficients

- A common difficulty is variable coefficients, exemplified in 1-D by

$$(a(x) u'(x))' = f(x), \quad 0 < x < 1,$$

$$u(0) = u(1) = 0,$$
 where $a(x)$ is a positive function on $[0,1]$.
- We seek to develop a conservative, or self-adjoint, method for discretizing this problem.
- Assume we have available to us the values of $a(x)$ ($a_{j+1/2} \equiv a(x_{j+1/2})$) at midpoints of the uniform grid

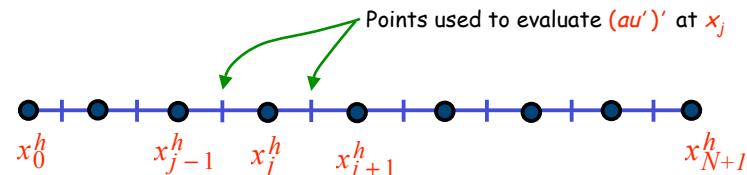


249 of 322

Discretize using central differences

We can use second-order differences to approximate the derivatives. To use a grid spacing of h , we evaluate $a(x)u'(x)$ at points midway between the gridpoints:

$$(a(x)u'(x))' \Big|_{x_j} = \frac{(au')|_{x_{j+1/2}} - (au')|_{x_{j-1/2}}}{h} + O(h^2)$$



250 of 322

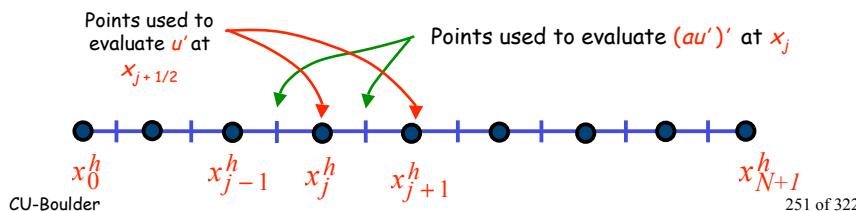
Discretize using central differences (cont'd)

To evaluate $(au')|_{x_{j+1/2}}$, we must sample $a(x)$ at the point $x_{j+1/2}$ & use second-order differences:

$$(au')|_{x_{j+1/2}} \approx a_{j+1/2} \frac{u_{j+1} - u_j}{h} \quad (au')|_{x_{j-1/2}} \approx a_{j-1/2} \frac{u_j - u_{j-1}}{h}$$

where

$$a_{j+1/2} \equiv a(x_{j+1/2}).$$



251 of 322

The basic stencil

We combine the differences for u' & for $(au')'$ to obtain the operator

$$-(a(x_j)u'(x_j))'(x_j) \approx -\frac{a_{j+\frac{1}{2}} \left(\frac{u_{j+1} - u_j}{h} \right) - a_{j-\frac{1}{2}} \left(\frac{u_j - u_{j-1}}{h} \right)}{h}$$

& the problem becomes

$$\frac{1}{h^2} \left(-a_{j-\frac{1}{2}} u_{j-1} + \left(a_{j-\frac{1}{2}} + a_{j+\frac{1}{2}} \right) u_j - a_{j+\frac{1}{2}} u_{j+1} \right) = f_j \quad 1 \leq j \leq N$$

$$u_0 = u_{N+1} = 0.$$

CU-Boulder

252 of 322

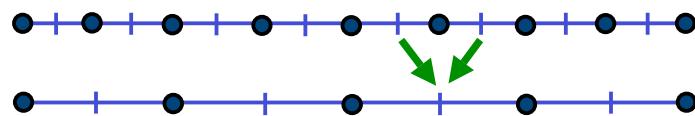
Coarsening the variable coefficient problem

- A reasonable approach is to use a standard multigrid algorithm with linear interpolation, full weighting, & the stencil

$$A^{2h} = \frac{1}{(2h)^2} \begin{bmatrix} -a_{j-\frac{1}{2}}^{2h} & a_{j-\frac{1}{2}}^{2h} + a_{j+\frac{1}{2}}^{2h} & -a_{j+\frac{1}{2}}^{2h} \end{bmatrix}$$

where

$$a_{j+1/2}^{2h} = \frac{a_{2j+1/2}^h + a_{2j+3/2}^h}{2}.$$



- The same stencil is obtained by the Galerkin relation.

Variable mesh vs. variable coefficients after scaling by $\eta_j = (h_{j-1/2} + h_{j+1/2})/2$ & h

- Variable mesh

$$-\frac{1}{h_{j-1/2}} u_{j-1}^h + \left(\frac{1}{h_{j-1/2}} + \frac{1}{h_{j+1/2}} \right) u_j^h - \frac{1}{h_{j+1/2}} u_{j+1}^h = \eta_j f_j^h$$

- Variable coefficients

$$\frac{1}{h} \left(-a_{j-\frac{1}{2}} u_{j-1} + \left(a_{j-\frac{1}{2}} + a_{j+\frac{1}{2}} \right) u_j - a_{j+\frac{1}{2}} u_{j+1} \right) = h f_j$$

- Correspondence

$$\frac{1}{h_{j-1/2}} \Leftrightarrow \frac{a_{j-1/2}}{h} \quad \frac{1}{h_{j+1/2}} \Leftrightarrow \frac{a_{j+1/2}}{h}$$

A variable coefficient example

- We use V(2,1) cycle, full weighting, linear interpolation.
- We use $a(x) = 1 + \rho \sin(k\pi x)$ & $a(x) = 1 + \rho \text{rand}(x)$.

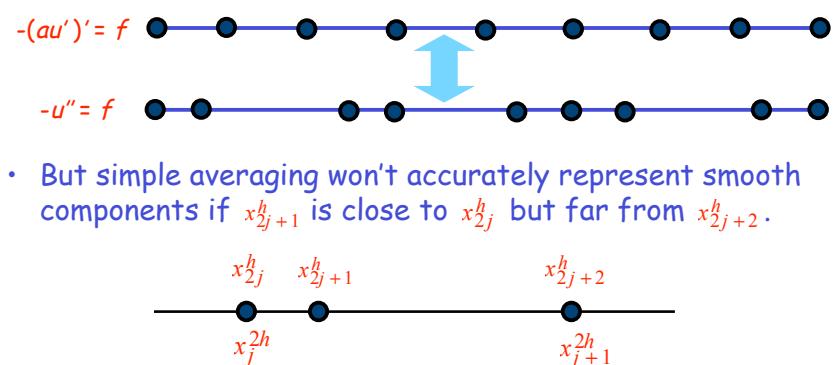
$$N=1024$$

$$a(x) = 1 + \rho \sin(k\pi x) \quad a(x) = 1 + \rho \text{rand}(x)$$

| ρ | k=3 | k=25 | k=50 | k=100 | |
|--------|-------|-------|-------|-------|-------|
| 0 | 0.085 | 0.085 | 0.085 | 0.085 | 0.085 |
| 0.25 | 0.084 | 0.098 | 0.098 | 0.094 | 0.083 |
| 0.5 | 0.093 | 0.185 | 0.194 | 0.196 | 0.173 |
| 0.75 | 0.119 | 0.374 | 0.387 | 0.391 | 0.394 |
| 0.85 | 0.142 | 0.497 | 0.511 | 0.514 | 0.472 |
| 0.95 | 0.191 | 0.681 | 0.69 | 0.694 | 0.672 |

Standard multigrid degrades if $a(x)$ is highly variable

- MG for variable coefficients is equivalent to MG (with simple averaging) for Poisson's equation on a variable mesh.



- But simple averaging won't accurately represent smooth components if x_{2j+1}^h is close to x_{2j}^h but far from x_{2j+2}^h .

One remedy is to apply operator-induced interpolation

- Assume that relaxation does not change smooth error, so the residual is approximately zero.

Applying at x_{2j+1}^h yields

$$\frac{-a_{2j+1/2}^h e_{2j}^h + (a_{2j+1/2}^h + a_{2j+3/2}^h) e_{2j+1}^h - a_{2j+3/2}^h e_{2j+2}^h}{h^2} = 0$$

- Solving for e_{2j+1}^h :

$$e_{2j+1}^h = \frac{a_{2j+1/2}^h e_{2j}^h + a_{2j+3/2}^h e_{2j+2}^h}{a_{2j+1/2}^h + a_{2j+3/2}^h}$$

Thus, operator-induced interpolation is

$$v_{2j}^{2h} = v_j^{2h},$$

$$v_{2j+1}^{2h} = \frac{a_{2j+1/2}^h v_j^{2h} + a_{2j+3/2}^h v_{j+1}^{2h}}{a_{2j+1/2}^h + a_{2j+3/2}^h}.$$

The case $a(x) \equiv 1$ reduces to standard interpolation:

$$v_{2j+1}^h = \frac{v_j^{2h} + v_{j+1}^{2h}}{2}.$$

As usual, the restriction & coarse-grid operators are defined by the Galerkin relations:

$$A^{2h} = I_h^{2h} A^h I_{2h}^h, \quad I_h^{2h} = c(I_{2h}^h)^T.$$

Outline

Chapters 1-5:

- Model Problems
- Basic Iterative Methods
 - Convergence tests
 - Analysis
- Elements of Multigrid
 - Relaxation
 - Coarsening
- Implementation
 - Complexity
 - Diagnostics
- Some Theory
 - Spectral vs. algebraic

Homework Due
Computing Too

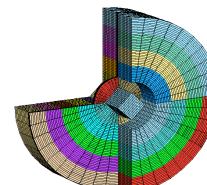
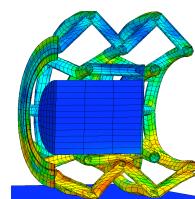
Chapters 6-10:

- Nonlinear Problems
 - Full approximation scheme
- Selected Applications
 - Neumann boundaries
 - Anisotropic problems
 - Variable meshes
 - Variable coefficients
- Algebraic Multigrid (AMG)
 - Matrix coarsening
- Multilevel Adaptive Methods
 - FAC
- Finite Elements
 - Variational methodology

8. Algebraic multigrid (AMG)

unstructured grids, variable coefficients,...

- Automatically defines coarse "grid".
- AMG has two distinct phases:
 - setup phase: define MG components.
 - solution phase: perform MG cycles.
- The AMG approach differs from geometric MG from the start:
 - fix relaxation (point Gauss-Seidel).
 - choose coarse "grids" & prolongation, P , so that error not reduced by relaxation is in $\text{range}(P)$. (Geometric MG fixes coarse grids & P , then chooses suitable operators & smoothers.)
- AMG uses the Galerkin principle so that coarse-grid correction eliminates error in $\text{range}(P)$.



Assume A is SPD unless otherwise stated.

AMG has two phases

- Setup Phase

- Select coarse "grids," Ω^{m+1} , $m = 1, 2, \dots$
- Define interpolation, I_{m+1}^m , $m = 1, 2, \dots$
- Define restriction & coarse-grid operators,
 $I_m^{m+1} = (I_{m+1}^m)^T$, $A^{m+1} = I_m^{m+1} A^m I_{m+1}^m$.

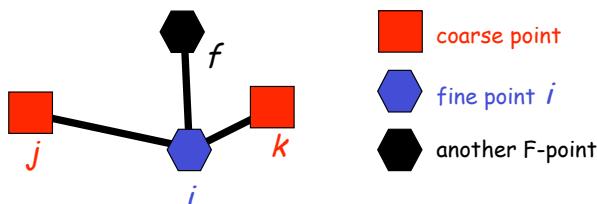
- Solve Phase

Standard MG processes: V-cycle, W-cycle, FMG, FAS, ...

- Only the selection of coarse grids does not parallelize well using existing techniques!

Operator-induced interpolation

pretend that we have a graph of A
& that we've identified coarse points



$$r_i = 0 \Rightarrow a_{ii} e_i + a_{ij} e_j + a_{ik} e_k + a_{if} e_f = 0.$$

To define e_i in terms of e_j and e_k , we must eliminate e_f .
Erase it?!

$$e_f = e_i \Rightarrow (a_{ii} + a_{if})e_i + a_{ij} e_j + a_{ik} e_k = 0$$

\Rightarrow F-points determined by C-points!

AMG fundamental concept: smooth error = "small" residuals/energy

- Error propagation via weighted Jacobi smoothing:

$$e^{k+1} = (I - \omega D^{-1}A) e^k.$$

- Error that is slow to converge satisfies

$$(I - \omega D^{-1}A) e \approx e \Rightarrow \omega D^{-1}A e \approx 0$$

$$\Rightarrow r \approx 0$$

- A little more precisely, assuming that $\omega D \approx I$, then slow-to-converge error has relatively small energy:

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

AMG uses strong connection to determine MG components

- Smoothing assumption:

$$r \approx 0 \text{ or } \langle Ae, e \rangle \approx 0.$$

- We say that i is strongly connected to j if

$$-a_{ij} \geq \theta \max_{k \neq i} \{-a_{ik}\}, \quad 0 < \theta \leq 1.$$

- For M-matrices, some algebra leads to

$$\langle Ae, e \rangle \approx \sum_{i \neq j} -\frac{a_{ij}}{2} (e_i - e_j)^2 \approx 0.$$

- So smooth error is more or less constant along strong connections.

Some useful set definitions

- The set of strong connections of a variable u_i , that is, the variables upon whose values the value of u_i depends, is defined as

$$S_i = \left\{ j : -a_{ij} \geq \theta \max_{k \neq i} \{-a_{ik}\} \right\}.$$

- The set of points strongly connected to a variable u_i is denoted $S_i^T = \{j : i \in S_j\}$.
- The set of coarse-grid variables is denoted C .
- The set of fine-grid variables is denoted F .
- The set of interpolatory coarse-grid variables used to interpolate the value of the fine-grid variable u_i is denoted C_i .

CU-Boulder

265 of 322

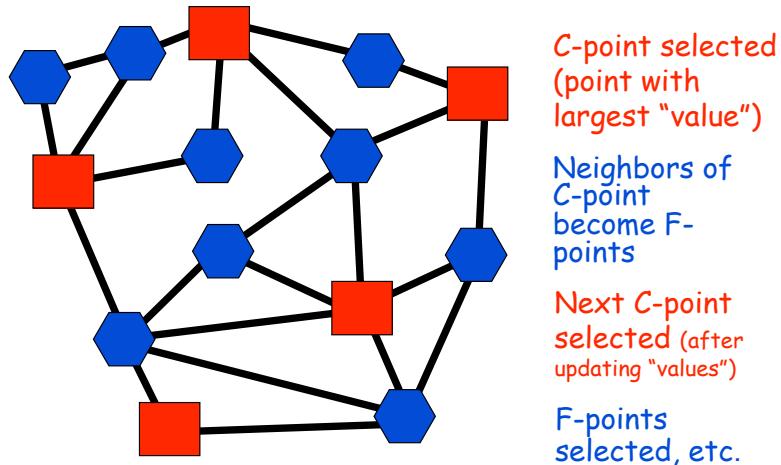
Choosing the coarse grid

- Two Criteria
 - (C1) For each $i \in F$, each point $j \in S_i$ should either be in C or should be strongly connected to at least one point in C_i .
 - (C2) C should be a maximal subset with the property that no two C -points are strongly connected to each other.
- Satisfying (C1) & (C2) is sometimes impossible. We use (C2) as a guide while enforcing (C1).

CU-Boulder

266 of 322

Selecting the coarse-grid points

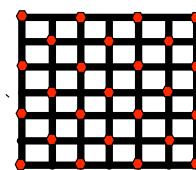


CU-Boulder

267 of 322

Examples: Laplacian operator

5-pt FD, 9-pt FE (quads), & 9-pt FE (stretched quads)

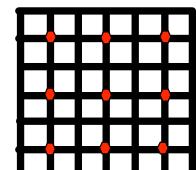


$$\begin{pmatrix} -1 & -1 \\ -1 & 4 & -1 \\ -1 & \end{pmatrix}$$

5-pt FD

9-pt FE (stretched quads)

$$\begin{pmatrix} -1 & -4 & -1 \\ 2 & 8 & 2 \\ -1 & -4 & -1 \end{pmatrix}$$



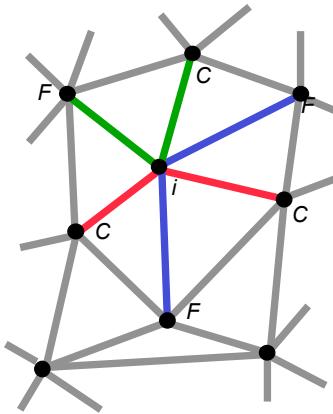
$$\begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix}$$

9-pt FE (quads)

CU-Boulder

268 of 322

Prolongation is based on smooth error, strong connections (from M-matrices)



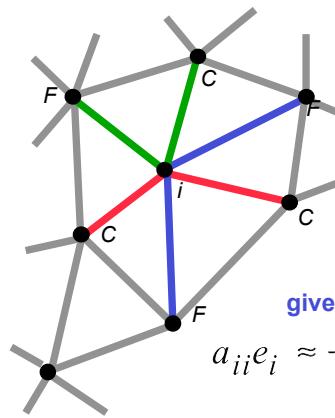
Smooth error is given by:

$$r_i = a_{ii} e_i + \sum_{j \in N_i} a_{ij} e_j \approx 0$$

Prolongation :

$$(Pe)_i = \begin{cases} e_i & , i \in C \\ \sum_{k \in C_i} \omega_{ik} e_k & , i \in F \end{cases}$$

Prolongation is based on smooth error, strong connections (from M-matrices)



Sets:

C_i Strongly connected C-pts.

D_i^S Strongly connected F-pts.

D_i^W Weakly connected points.

The definition of smooth error,

$$a_{ii} e_i \approx - \sum_{j \neq i} a_{ij} e_j$$

$$a_{ii} e_i \approx - \sum_{j \in C_i} a_{ij} e_j - \sum_{j \in D_i^S} a_{ij} e_j - \sum_{j \in D_i^W} a_{ij} e_j$$

Strong C

Strong F

Weak pts.

Finally, the prolongation weights are defined

In the smooth-error relation, use $e_j = e_i$ for weak connections. For strong F-points, use

$$e_j = \left(\sum_{k \in C_i} a_{jk} e_k \right) / \left(\sum_{k \in C_i} a_{jk} \right)$$

yielding the prolongation weights:

$$w_{ij} = - \frac{a_{ij} + \sum_{j \in D_i^S} \frac{a_{ik} a_{kj}}{\sum_{m \in C_i} a_{km}}}{a_{ii} + \sum_{n \in D_i^W} a_{in}}$$

AMG setup costs a bad rap

- Many geometric MG methods need to compute prolongation & coarse-grid operators.
- The only additional expense in the AMG setup phase is the coarse-grid selection algorithm.
- AMG setup phase is only 10-25% more expensive than in geometric MG & may be considerably less than that!

AMG performance: Sometimes a success story

- AMG performs extremely well on the model problem (Poisson's equation, regular grid)- optimal convergence factor (e.g., 0.14) & scalability w.r.t. problem size.
- AMG appears to be both scalable & efficient on diffusion problems on unstructured grids (e.g., 0.1-0.3).
- AMG handles anisotropic diffusion coefficients on irregular grids reasonably well.
- AMG handles anisotropic operators on structured & unstructured grids relatively well (e.g., 0.35).

CU-Boulder

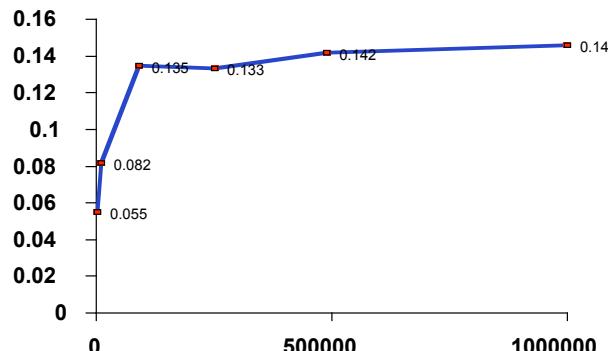
273 of 322

How does it perform (vol II)?

structured meshes, rectangular domains

5-point Laplacian on regular rectangular grids

Convergence factor (y-axis) plotted against number of nodes (x-axis)



CU-Boulder

275 of 322

How does it perform (vol I)?

regular grids, plain, old, vanilla problems,
unit square, $N = 64$, Dirichlet boundaries

- The Laplace operator:

| Stencil | Convergence | | Time per Cycle | Setup Times |
|-------------------|-------------|------------|-------------------|----------------|
| | per cycle | Complexity | | |
| 5-pt | 0.054 | 2.21 | 0.29 | 1.63 |
| 5-pt skew | 0.067 | 2.12 | 0.27 | 1.52 |
| 9-pt (-1, 8) | 0.078 | 1.30 | 0.26 | 1.83 |
| 9-pt (-1, -4, 20) | 0.109 | 1.30 | 0.26 | 1.83 |

- Anisotropic 5-Point Laplacian: $-\varepsilon U_{xx} - U_{yy}$

| Epsilon | 0.001 | 0.01 | 0.1 | 0.5 | 1 | 2 | 10 | 100 | 1000 |
|-------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Convergence/cycle | 0.084 | 0.093 | 0.058 | 0.069 | 0.054 | 0.079 | 0.087 | 0.093 | 0.083 |

CU-Boulder

274 of 322

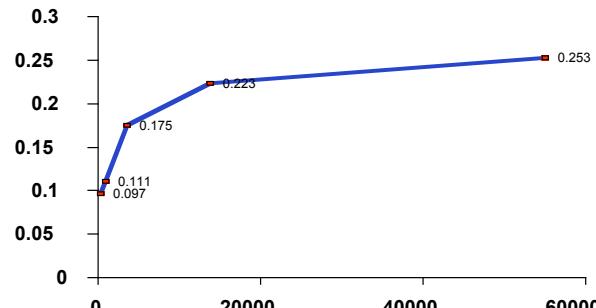
How does it perform (vol III)?

unstructured meshes, rectangular domains

Laplacian on random unstructured grids

(regular triangulations, 15-20% nodes randomly collapsed into neighboring nodes)

Convergence factor (y-axis) plotted against number of nodes (x-axis)



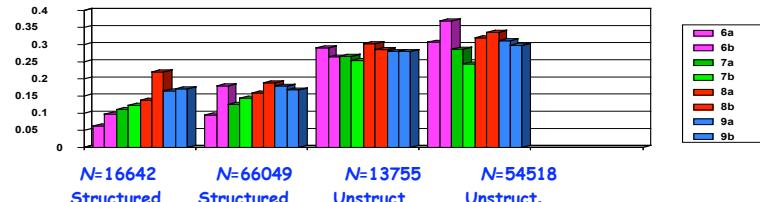
CU-Boulder

276 of 322

How does it perform (vol IV)?

isotropic diffusion, structured/unstructured grids

$\nabla \bullet (d(x,y) \nabla u)$ on structured, unstructured grids



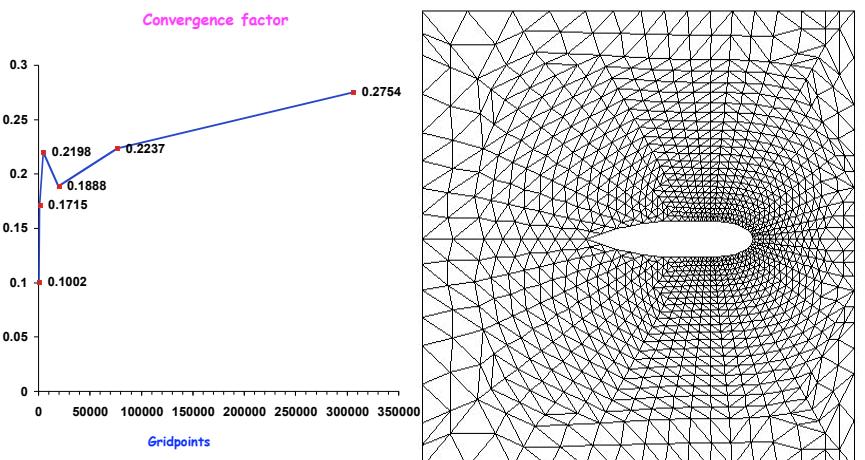
Problems used: "a" means parameter $c = 10$, "b" means $c = 1000$

$$6: d(x,y) = 1.0 + c|x-y| \quad 8: d(x,y) = \begin{cases} 1.0 & 0.125 \leq \max\{|x-0.5|, |y-0.5|\} \leq 0.25 \\ c & \text{otherwise} \end{cases}$$

$$7: d(x,y) = \begin{cases} 1.0 & x \leq 0.5 \\ c & x > 0.5 \end{cases} \quad 9: d(x,y) = \begin{cases} 1.0 & 0.125 \leq \sqrt{(x-0.5)^2 + (y-0.5)^2} \leq 0.25 \\ c & \text{otherwise} \end{cases}$$

How does it perform (vol V)?

Laplacian operator, unstructured grids



AMG for systems

- How can we do AMG on systems?

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix}$$

- Naïve approach: "Block" AMG (block Gauss-Seidel, using scalar AMG to "solve" at each cycle)

$$u \leftarrow (A_{11})^{-1} (f - A_{12}v)$$

$$v \leftarrow (A_{22})^{-1} (g - A_{21}u)$$

Great Idea! Except that it often doesn't work!

Block AMG doesn't account for strong inter-variable coupling.

AMG for systems: A solution

- To solve the system problem, allow interaction between the unknowns at all levels:

$$A^k = \begin{pmatrix} A_{11}^k & A_{12}^k \\ A_{21}^k & A_{22}^k \end{pmatrix} \quad \& \quad I_{k+1}^k = \begin{pmatrix} (I_{k+1}^k)_u & 0 \\ 0 & (I_{k+1}^k)_v \end{pmatrix}.$$

- This is called the "unknown-based" approach.
- 2-D biharmonic, Dirichlet boundaries, unit square, uniform quadrilateral mesh:

| Mesh spacing | 0.125 | 0.0625 | 0.03135 | 0.015625 |
|--------------------|-------|--------|---------|----------|
| Convergence factor | 0.22 | 0.35 | 0.42 | 0.44 |

Adaptive AMG (α AMG) to broaden applicability

adaptive interpolation
to determine sense of smoothness

+

adaptive C-point choice
to determine good coarse point

Standard interpolation

Standard AMG collapses stencils by assuming smooth error is locally constant (Poisson "sense of smoothness"):

$$\text{*** smooth } \mathbf{e} \approx \mathbf{c} \text{ ***} \quad (\text{constant}) \quad e_j = \left(\sum_{k \in C_i} a_{jk} e_k \right) / \left(\sum_{k \in C_i} a_{jk} \right)$$

$$a_{ii} e_i \approx - \sum_{j \in C_i} a_{ij} e_j - \sum_{j \in D_i^s} a_{ij} e_j - \sum_{j \in D_i^w} a_{ij} e_j$$

Strong C **Strong F** **Weak pts.**

$e_j = e_i$

Isn't standard interpolation OK?
Isn't smooth $\mathbf{e} \approx \mathbf{c}$ (constant) always?

- Suppose someone tried to make A "nice" for relaxation by scaling the diagonal so it's the identity:

$$\underline{A} \leftarrow D^{-1/2} A D^{-1/2}, \quad D = \text{diag}(a_{ii}).$$

- Relaxation still gives small residuals:

$$\underline{A}\underline{e} \approx 0.$$

- But:

$$\underline{A}\underline{e} = D^{-1/2} A D^{-1/2} \underline{e} \approx 0 \Rightarrow A D^{-1/2} \underline{e} \approx 0 \Rightarrow \underline{e} \approx D^{1/2} \mathbf{c}.$$

- So "smooth" here means $e_i \approx c \sqrt{a_{ii}}$. This could vary a lot!

Pre-relaxation

- What if we found a smooth error \mathbf{x} that's far from \mathbf{c} ?
- If we found that $x_j = 1.6x_i$ for $j \in D_i^w$, say, then we could just make the replacement $e_j \rightarrow 1.6e_i$.
- If, say, $x_j = 0.2x_k$ for $j \in D_i^s$ & all $k \in C_i$, then we could set

$$e_j \rightarrow \left(\sum_{j \in C_i} a_{jk} 0.2e_k \right) / \left(\sum_{j \in C_i} a_{jk} \right).$$

- How do we obtain a representative smooth error \mathbf{x} ?

Relax on $\underline{A}\underline{x} = 0$!!!

Adaptive C-point choice

- AMG uses the ansatz that residuals are 0 at F-points.
- What if we had $r_f = 0$ exactly?

$$\begin{pmatrix} A_{ff} & A_{fc} \\ A_{cf} & A_{cc} \end{pmatrix} \begin{pmatrix} e_f \\ e_c \end{pmatrix} = \begin{pmatrix} 0 \\ r_c \end{pmatrix} \Rightarrow A_{ff}e_f + A_{fc}e_c = 0 \Rightarrow \begin{pmatrix} e_f \\ e_c \end{pmatrix} = \begin{pmatrix} -A_{ff}^{-1}A_{fc} \\ I \end{pmatrix} e_c$$

- Choosing $P = \begin{pmatrix} -A_{ff}^{-1}A_{fc} \\ I \end{pmatrix}$ (ideal interpolation) makes MG exact:

$$\begin{aligned} MGe &= [CG]Pe_c = [I - P[P^T AP]^{-1}P^T A]Pe_c \\ &= Pe_c - P[P^T AP]^{-1}[P^T AP]e_c = 0. \end{aligned}$$

Compatible relaxation (CR)

- How do we make the residuals zero at F-points?

Relax on $A_{ff}x_f = -A_{fc}x_c$!

- To ensure that this works so that $r_f \rightarrow 0$ fast, we want A_{ff} to be well conditioned. Thus, we can assess whether we have good C-points by "CR":

$$A_{ff}x_f = 0.$$

- Quick CR convergence means that the F-point residuals will be small after relaxation & that there is a P (the ideal one) that gives good MG convergence.
- Is there a good local P ? Maybe, since CR is local & fast.
- Hopefully, we can use pre-relaxation to compute a good P .

Outline

Chapters 1-5:

- ✓ Model Problems
- ✓ Basic Iterative Methods
 - Convergence tests
 - Analysis
- ✓ Elements of Multigrid
 - Relaxation
 - Coarsening
- ✓ Implementation
 - Complexity
 - Diagnostics
- ✓ Some Theory
 - Spectral vs. algebraic

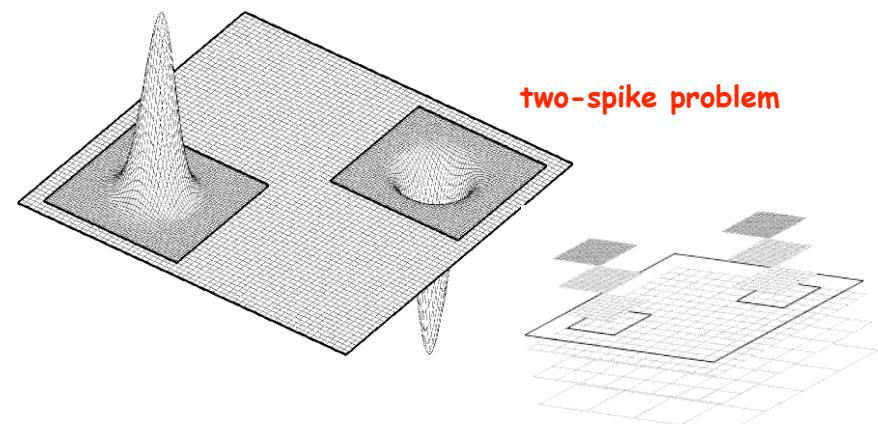
Chapters 6-10:

- ✓ Nonlinear Problems
 - Full approximation scheme
- ✓ Selected Applications
 - Neumann boundaries
 - Anisotropic problems
 - Variable meshes
 - Variable coefficients
- ✓ Algebraic Multigrid (AMG)
 - Matrix coarsening
 - Multilevel Adaptive Methods
 - FAC
 - Finite Elements
 - Variational methodology

Homework Due

9. Multilevel adaptive methods

fast adaptive composite grid method (FAC)

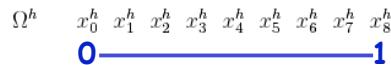


Local enhancement to resolve special regions of activity or interest.

Model 1-D problem

$$-u''(x) = f(x), \quad 0 < x < 1$$

$$u(0) = u(1) = 0$$



$$h^{-2}(-u_{i-1}^h + 2u_i^h - u_{i+1}^h) = f_i^h, \quad 1 \leq i \leq 7$$

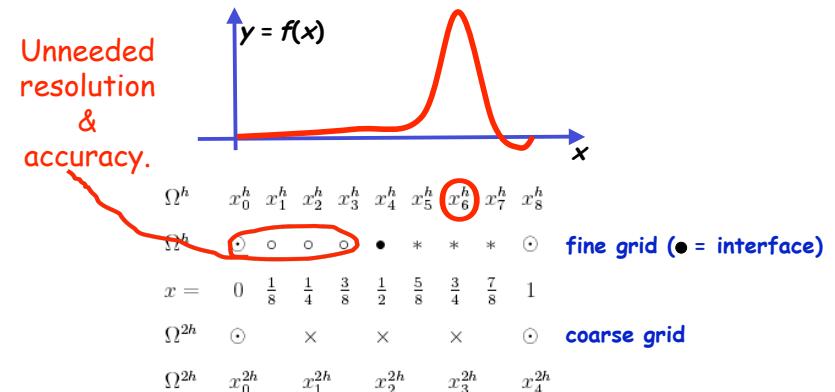
$$u_0^h = u_8^h = 0, \quad h = \frac{1}{8}$$

CU-Boulder

289 of 322

Local refinement

suppose $f(x)$ has a spike at $x = 3/4$
but is smooth elsewhere



CU-Boulder

290 of 322

Strategy

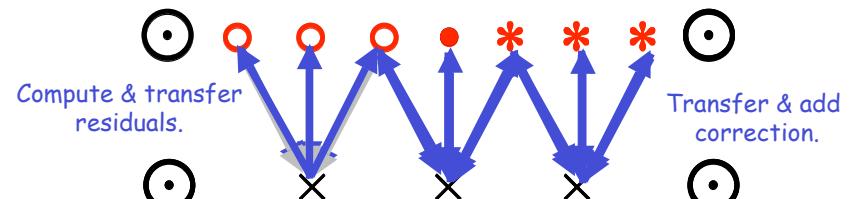
- Recognize that there's little value of having the fine grid in the smooth region, $[0, 1/2]$.
- Start with uniform grid & standard MG, then:
 - first eliminate relaxation in $[0, 1/2]$; and
 - then eliminate intergrid transfers & residual calculations.
- Then interpret this process via the composite grid (= $2h$ -points in $[0, 1/2]$ + h -points in $[1/2, 1]$).
- We'll try absurdly hard to eliminate work @ $x = 1/2$, but we have in mind multi-dimensions & smaller patches.

CU-Boulder

291 of 322

Local-Relaxation/Global-Correction MG

→ → WARNING: You won't get zero residuals on all of grid h . ← ←
Remember that the concept is right here, but we need to
make it efficient--without changing the results!!!
Consider the usual global grids. Relax only in the local region.
Now correct via coarse grid as usual. $x = 1/2$ becomes a boundary point.



We start by using local relaxation & eliminating unnecessary residual transfers.

To save all of the work of computing & transferring residuals & corrections in regions where they don't change, we need some messy ALGEBRA!

Trust me on the more messy stuff...

CU-Boulder

292 of 322

Eliminate relaxation

so r^h changes only on $2h$ v^h changes only here

notation

$$\Omega^h \quad x_0^h \quad x_1^h \quad x_2^h \quad x_3^h \quad x_4^h \quad x_5^h \quad x_6^h \quad x_7^h \quad x_8^h$$

$$\Omega^{2h} \quad x_0^{2h} \quad x_1^{2h} \quad x_2^{2h} \quad x_3^{2h} \quad x_4^{2h}$$

$$v^h = (v_i^h)$$

Once & forall:
it changes only
on $2h$.

- Initialize $v^h = 0$ & $f_1^{2h} \leftarrow (f_1^h + 2f_2^h + f_3^h)/4$.
- Relax on v^h on the local fine grid (x_5^h, x_6^h, x_7^h).
- Compute $r^h = f^h - A^h v^h$ (@ $x_3^h - x_7^h$) & transfer to $2h$:
 $f_2^{2h} \leftarrow (r_3^h + 2r_4^h + r_5^h)/4$ & $f_3^{2h} \leftarrow (r_5^h + 2r_6^h + r_7^h)/4$.
- Compute an approximation, v^{2h} , to the solution of the $2h$ residual equation, $A^{2h} u^{2h} = f^{2h}$.
- Update the residual at x_1^{2h} for later cycles:
 $f_1^{2h} \leftarrow f_1^{2h} - (-v_0^{2h} + 2v_1^{2h} - v_2^{2h})/(2h)^2$.
- Correct: $v^h \leftarrow v^h + I_{2h}^h v^{2h}$.

cycle

Why save v^h outside of local region?

293 of 322

Eliminate more

$$\Omega^h \quad x_0^h \quad x_1^h \quad x_2^h \quad x_3^h \quad x_4^h \quad x_5^h \quad x_6^h \quad x_7^h \quad x_8^h$$

$$\Omega^{2h} \quad x_0^{2h} \quad x_1^{2h} \quad x_2^{2h} \quad x_3^{2h} \quad x_4^{2h}$$

f_2^{2h} involves r_3^h & r_4^h , so we "need" $v_2^h - v_7^h$.

Store v^h only @ $x_3^h - x_7^h$ & save v_2^h on $2h$ (call it w_1^{2h}).

- Initialize $v^h = 0$, $w_1^{2h} = 0$, & $f_1^{2h} \leftarrow (f_1^h + 2f_2^h + f_3^h)/4$.
- Relax on v^h on the local fine grid (x_5^h, x_6^h, x_7^h).
- Compute $r^h = f^h - A^h v^h$ (@ $x_2^h - x_7^h$) & transfer to $2h$:
 $f_2^{2h} \leftarrow (r_3^h + 2r_4^h + r_5^h)/4$ & $f_3^{2h} \leftarrow (r_5^h + 2r_6^h + r_7^h)/4$.
- Compute an approximation, v^{2h} , to the solution of the $2h$ residual equation, $A^{2h} u^{2h} = f^{2h}$.
- Update the residual at x_1^{2h} for later cycles:
 $f_1^{2h} \leftarrow f_1^{2h} - (-v_0^{2h} + 2v_1^{2h} - v_2^{2h})/(2h)^2$.
- Accumulate the $2h$ approximation: $w_1^{2h} \leftarrow w_1^{2h} + v_1^{2h}$.
- Correct: $v^h \leftarrow v^h + I_{2h}^h v^{2h}$ (@ $x_3^h - x_7^h$).

cycle

CU-Boulder

294 of 322

Eliminate the rest

$$\Omega^h \quad x_0^h \quad x_1^h \quad x_2^h \quad x_3^h \quad x_4^h \quad x_5^h \quad x_6^h \quad x_7^h \quad x_8^h$$

$$\Omega^{2h} \quad x_0^{2h} \quad x_1^{2h} \quad x_2^{2h} \quad x_3^{2h} \quad x_4^{2h}$$

r_3^h doesn't change on h . Compute change in r_3^h on $2h$.

$2h$ residual @ $x_2^{2h} = g_2^{2h} - (-w_1^{2h} + 2v_4^h - v_6^h)/(2h)^2$, where $g_2^{2h} = (f_3^h + 2f_4^h + f_5^h)/4$. (Messy algebra!)

- Initialize $v^h = 0$, $w_1^{2h} = 0$, & $f_1^{2h} \leftarrow (f_1^h + 2f_2^h + f_3^h)/4$.
- Relax on v^h on the local fine grid (x_5^h, x_6^h, x_7^h).
- Compute the right sides for $2h$:
 $f_2^{2h} \leftarrow g_2^{2h} - (-w_1^{2h} + 2v_4^h - v_6^h)/(2h)^2$ &
 $f_3^{2h} \leftarrow (r_5^h + 2r_6^h + r_7^h)/4$.
- Compute an approximation, v^{2h} , to the solution of the $2h$ residual equation, $A^{2h} u^{2h} = f^{2h}$.
- Update the residual at x_1^{2h} for later cycles:
 $f_1^{2h} \leftarrow f_1^{2h} - (-v_0^{2h} + 2v_1^{2h} - v_2^{2h})/(2h)^2$.
- Accumulate the $2h$ approximation: $w_1^{2h} \leftarrow w_1^{2h} + v_1^{2h}$.
- Correct: $v^h \leftarrow v^h + I_{2h}^h v^{2h}$ (@ $x_4^h - x_7^h$).

cycle

295 of 322

Interpretation composite grid

$$\begin{array}{ccccccccc} \Omega^h & \odot & & \bullet & * & * & * & \odot \\ \Omega^{2h} & \odot & \times & \times & \times & & \odot \\ \Omega^c & \odot & \times & \bullet & * & * & * & \odot \\ v_0^c & v_1^c & v_2^c & v_3^c & v_4^c & v_5^c & v_6^c \end{array}$$

What equation in v^h are we solving?

$$\begin{aligned} (-v_0^c + 2v_1^c - v_2^c)/(2h)^2 &= f_1^c \equiv (f_1^h + 2f_2^h + f_3^h)/4 \\ (-v_{i-1}^c + 2v_i^c - v_{i+1}^c)/h &= f_i^c \equiv f_{i+2}^h \quad i = 3, 4, 5 \end{aligned} \quad \text{usual}$$

$$(-v_1^c + 3v_2^c - 2v_3^c)/(2h)^2 = f_2^c \equiv (f_3^h + 2f_4^h)/4 \quad @\text{interface}$$

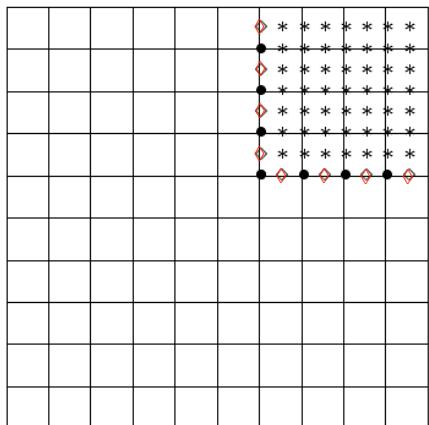
→ → The key is to treat the interface correctly. ← ←

CU-Boulder

296 of 322

Issues

- Adaptivity
- Error estimates
- Norms (proper scaling)
- Multiple dimensions
 - Slave points ◊
 - More complicated stencils
- Data structures
- Parallel algorithms (AFAC)
- Time-space



Two-spike example (Laplacian)

- Global grid h & one double-patch refined level $h/2$.
- V(1,0)-cycles, Gauss-Seidel.
- Asymptotic convergence of the solver.
- Scaled L^2 discretization error estimate.

| Global h | Convergence factor | Discrete L^2 norm of discretization error |
|------------|--------------------|---|
| 1/32 | 0.362 | 2.34e-2 |
| 1/64 | 0.367 | 5.742e-3 |
| 1/128 | 0.365 | 1.43e-3 |

Outline

Chapters 1-5:

- ✓ Model Problems
- ✓ Basic Iterative Methods
 - Convergence tests
 - Analysis
- ✓ Elements of Multigrid
 - Relaxation
 - Coarsening
- ✓ Implementation
 - Complexity
 - Diagnostics
- ✓ Some Theory
 - Spectral vs. algebraic

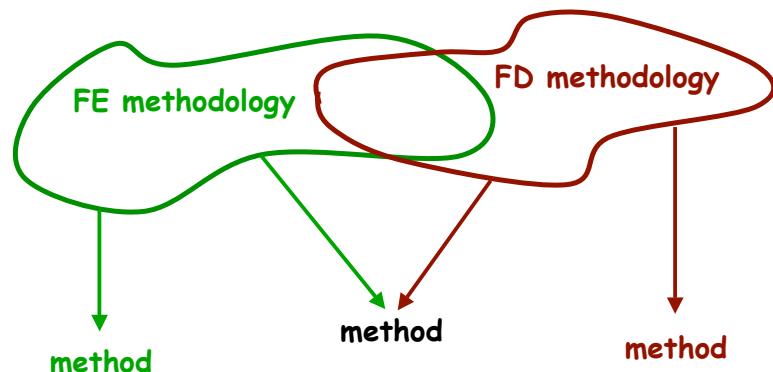
Homework Due

Chapters 6-10:

- ✓ Nonlinear Problems
 - Full approximation scheme
- ✓ Selected Applications
 - Neumann boundaries
 - Anisotropic problems
 - Variable meshes
 - Variable coefficients
- ✓ Algebraic Multigrid (AMG)
 - Matrix coarsening
- ✓ Multilevel Adaptive Methods
 - FAC
- Finite Elements
 - Variational methodology

10. Finite elements

FE, a variational discretization methodology



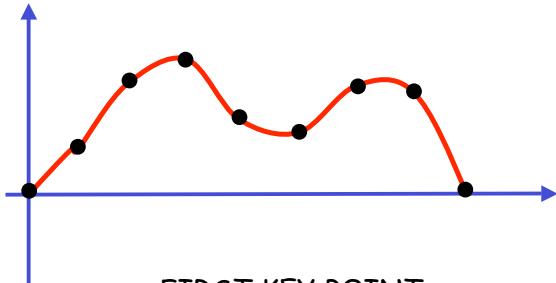
FD: differences at nodes & truncation error.

FE: weak form on discrete functions & approximation property

Other methods: finite volume, collocation, spectral, ...

Discretization

What do you see?



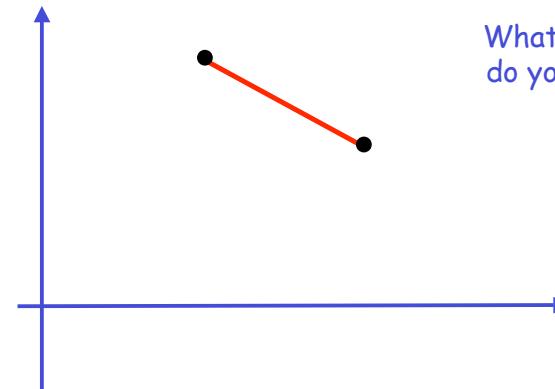
→ → FIRST KEY POINT ← ←
FE sees grid points (nodes or dofs)
only as characterizations of
continuum functions.

CU-Boulder

301 of 322

Localize

What choice
do you have?



→ → SECOND KEY POINT ← ←

FE functions are localized to ensure a sparse matrix.

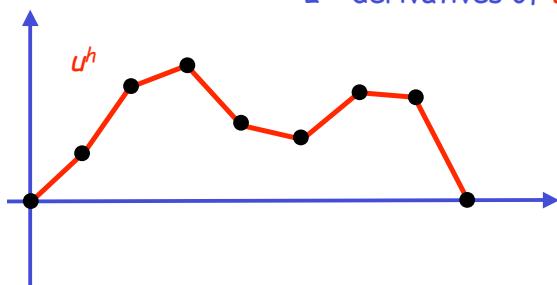
CU-Boulder

302 of 322

Continuous piecewise linear functions

??? - $u'' = f$???

How do you take
2nd derivatives of u^h ?



→ → THIRD KEY POINT ← ←
FE integrates away 2nd derivatives.

CU-Boulder

303 of 322

Weak form!

$$\begin{aligned}
 -u'' &= f & u &= \text{trial function} \\
 \Rightarrow -u''v &= fv & \forall v & v = \text{test function} \\
 \Rightarrow \int (-u'')v dx &= \int fv dx & \forall v \\
 \Rightarrow \int u'v' dx - u'|_0^1 &= \int fv dx & \forall v \\
 \Rightarrow \int u'v' dx &= \int fv dx & \forall v \exists v(0) = v(1) = 0 \\
 \Rightarrow (u', v') &= (f, v) & \forall v \exists v(0) = v(1) = 0
 \end{aligned}$$

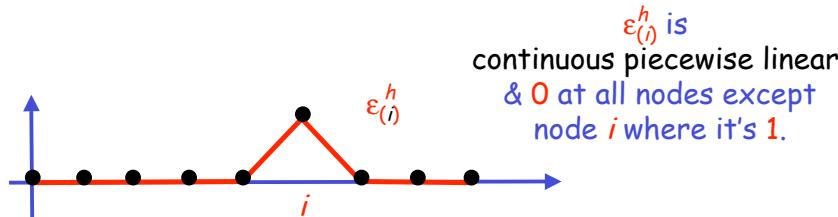
? Need basis for space of admissible u^h & v .

The \Rightarrow can actually be reversed
to \Leftarrow if u is smooth enough.

So the weak & strong forms are "equivalent".

304 of 322

Representation a basis of "hat" functions



Any continuous piecewise linear function can be represented by

$$u^h = \sum_i u_i^h \varepsilon_{(i)}^h.$$

↑ Now we're back to using node values.

CU-Boulder

305 of 322

FE basics

Self-adjoint positive definite \Rightarrow weak functional.

- Model problem $\Omega \subset \mathbb{R}^2$:

$$Lu \equiv -u_{xx} - u_{yy} = f \quad \text{in } \Omega$$

$$u = 0 \quad \text{on } \partial\Omega$$
- Sobolev spaces (wider than classical spaces!):
 $L^2(\Omega) = \{u : \int_{\Omega} u^2 d\Omega < \infty\}$
 $H_0^1(\Omega) = \{u : u, u_x, u_y \in L^2(\Omega), u|_{\partial\Omega} = 0\}$
 $(u, v) = \int_{\Omega} u v d\Omega$
- L is self-adjoint positive definite (more later):
 $(Lu, v) = (\nabla u, \nabla v) = \int_{\Omega} (u_x v_x + u_y v_y) d\Omega = (u, Lv)$
 $(Lu, u) > 0 \text{ if } u \neq 0$

We use (Lu, v) for simplicity,
but really we mean $(\nabla u, \nabla v)$.

$$\nabla u = \begin{pmatrix} u_x \\ u_y \end{pmatrix}$$

- Duality: Solving $Lu = f$ is equivalent to minimizing the weak functional

$$F(u) = "(Lu, u)" / 2 - (f, u)$$

$$= (\nabla u, \nabla u) / 2 - (f, u)$$
- Short story:
1st derivative test

$$\nabla F(u) = Lu - f = 0$$

2nd derivative test

$$F''(u) = L > 0$$

This is formal:
 Lu is not defined on all of $H_0^1(\Omega)$.

CU-Boulder

307 of 322

Weak form

$$(u^h, v) = (f, v) \quad \forall v \in \mathcal{V} \text{ s.t. } v(0) = v(1) = 0.$$

$$(\sum_j u_j^h \varepsilon_{(j)}^h, v) = (f, v) \quad \forall v \in \mathcal{V} \text{ s.t. } v(0) = v(1) = 0.$$

$$(\sum_j u_j^h \varepsilon_{(j)}^h, \varepsilon_{(i)}^h) = (f, \varepsilon_{(i)}^h) \quad \forall i.$$

$$\sum_j (\varepsilon_{(i)}^h, \varepsilon_{(j)}^h) u_j^h = (f, \varepsilon_{(i)}^h) \quad \forall i.$$

So the matrix is

$$A^h = \left(a_{ij}^h \right) = \left((\varepsilon_{(i)}^h, \varepsilon_{(j)}^h) \right).$$

CU-Boulder

306 of 322

Long story

- Using symmetry & linearity of L & bilinearity of the inner product:

$$F(u+v) = (Lu+v, u+v)/2 - (f, u+v)$$

$$= (Lu, u)/2 + (Lu, v) + (Lv, v)/2$$

$$- (f, u) - (f, v)$$

$$= F(u) + (Lu, v) - (f, v) + (Lv, v)/2$$

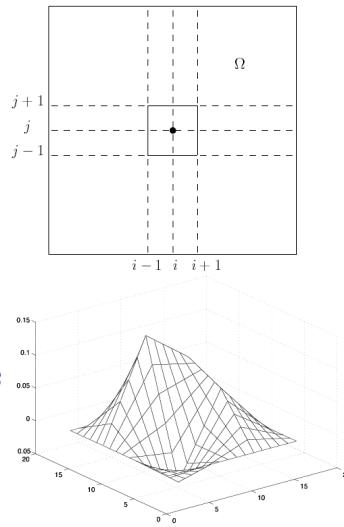
$$= F(u) + (Lu-f, v) + (Lv, v)/2.$$
- The last term, $(Lv, v)/2$, is positive for $v \neq 0$, but it can be neglected for very small v . Thus,
 $F(u+v) \geq F(u) \quad \forall v \in H_0^1(\Omega) \Leftrightarrow (Lu, v) = (f, v) \quad \forall v \in H_0^1(\Omega)$
 $\Leftrightarrow "Lu = f".$

CU-Boulder

308 of 322

FE constructs

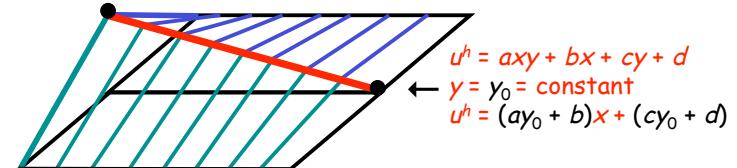
- Assume that Ω is the unit square.
- Consider an $n \times n$ grid of square "cells".
- Continuous piecewise bilinear elements: $H^h \subset H_0^1(\Omega)$.
- Each u^h in H^h is determined by its node values. This is how we'll represent them!
- Within each square:
 $u^h = axy + bx + cy + d$ &
 u^h is linear in a coordinate direction.
- If u^h on one side of an element matches u^h on the other at the nodes, then it matches on the common edge. So u^h is continuous.



CU-Boulder

309 of 322

Continuity

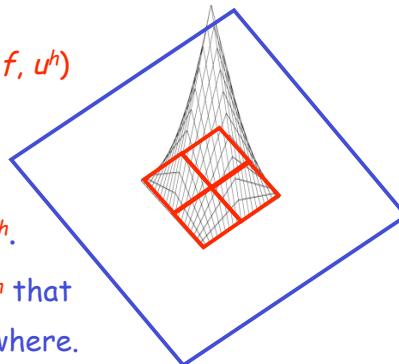


CU-Boulder

310 of 322

FE discretization

- Minimize $F(u^h) = (Lu^h, u^h)/2 - (f, u^h)$
over $u^h \in H^h$.
- Equivalent to solving
 $(Lu^h, v^h) = (f, v^h) \quad \forall v^h \in H^h$.
- Basis: $\epsilon_{(ij)}^h$ is the element of H^h that equals 1 @ node ij & 0 elsewhere.
- Expansion: $u^h(x,y) = \sum_{ij} u_{ij}^h \epsilon_{(ij)}^h(x,y)$.
- Problem: What is $Lu^h \equiv -u_{xx}^h - u_{yy}^h$???



CU-Boulder

311 of 322

Weak form

- The Gauss Divergence Theorem & homogeneous boundary conditions yield
 $(Lu, v) = (-u_{xx} - u_{yy}, v) = (-\nabla \cdot \nabla u, v) = (\nabla u, \nabla v)$.
- Note:
 $(\nabla u, \nabla v) = \int_{\Omega} (u_x v_x + u_y v_y) d\Omega$.
- So the problem becomes
 $(\nabla u^h, \nabla v^h) = (f, v^h) \quad \forall v^h \in H^h$
or
 $\int_{\Omega} (u_x^h v_x^h + u_y^h v_y^h) d\Omega = \int_{\Omega} f v^h d\Omega \quad \forall v^h \in H^h$.

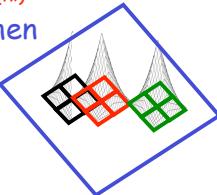
CU-Boulder

312 of 322

Towards the matrix equation for nodal values

$$(\nabla u^h, \nabla v^h) = (f, v^h) \quad \forall v^h \in H^h$$

- Using $u^h(x, y) = \sum_{ij} u_{ij}^h \varepsilon_{(ij)}^h(x, y)$ & choosing $v^h = \varepsilon_{(kh)}^h$:
 $\sum_{ij} u_{ij}^h (\nabla \varepsilon_{(ij)}^h, \nabla \varepsilon_{(kh)}^h) = (f, \varepsilon_{(kh)}^h) \quad \forall k, l.$
- Matrix terms $(\nabla \varepsilon_{(ij)}^h, \nabla \varepsilon_{(kl)}^h)$ are 0 when
 $|i - k|$ or $|j - l| > 1$.



- We compute
 $(\nabla \varepsilon_{(ij)}^h, \nabla \varepsilon_{(ij)}^h) = 8/3$ & $(\nabla \varepsilon_{(ij)}^h, \nabla \varepsilon_{(i \pm 1, j \pm 1)}^h) = -1/3$.
- Assume f is bilinear:
 $(f, \varepsilon_{(kh)}^h) = \int_{\Omega} f \varepsilon_{(kh)}^h d\Omega \approx h^2 f(x_{kl})$.

$$u^h = (u_{ij}^h) \quad \& \quad f^h = (h^2 f(x_{kl}))$$

actually,
 $B^h f^h$
(mass matrix)
313 of 322

Some matrix properties

$$A_{ij}^h = \frac{1}{3} \begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix}$$

- Symmetric! ij "reaches" to $i \pm 1, j \pm 1$ as $i \pm 1, j \pm 1$ to ij .
- Singular? $A^h \mathbf{1} = 0$?! Depends on boundaries!

Dirichlet boundary: $A_{ij}^h = \frac{1}{3} \begin{pmatrix} -1 & -1 \\ 8 & -1 \\ -1 & -1 \end{pmatrix}$.

- Positive definite!
- Diagonally dominant (strictly so @ boundaries).

The matrix equation

$$A^h u^h = f^h$$

where

$$u^h = (u_{ij}^h) \quad \& \quad f^h = (h^2 f(x_{kl}))$$

& the matrix is given by the stencil

$$A_{ij}^h = \frac{1}{3} \begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix}. \quad \text{stiffness matrix}$$

Abstract FE relaxation focus on functions, not nodal vectors

- Relaxation involves "local" changes: $u^h \leftarrow u^h - s \varepsilon^h$ for some scalar s & $\varepsilon^h = \varepsilon_{(ij)}^h$.
But how do we pick s ???
- Use FE principle of minimizing $F(u^h - s \varepsilon^h)$ over s :

$$\begin{aligned} F(u^h - s \varepsilon^h) &= (L(u^h - s \varepsilon^h), u^h - s \varepsilon^h)/2 - (f, u^h - s \varepsilon^h) \\ &= F(u^h) - s(Lu^h - f, \varepsilon^h) + (s^2/2)(L\varepsilon^h, \varepsilon^h). \end{aligned}$$
- Let s be the root of the derivative of this quadratic polynomial w.r.t. s : $s = (Lu^h - f, \varepsilon^h)/(L\varepsilon^h, \varepsilon^h) = r_{ij}/L_{ij,ij}^h$, so

$$F(u^h - s \varepsilon^h) - F(u^h) = - (Lu^h - f, \varepsilon^h)^2/(2(L\varepsilon^h, \varepsilon^h)) = - r_{ij}^2/(2L_{ij,ij}^h).$$

This is just Gauss-Seidel!

Abstract FE coarsening

again with focus on functions

- Coarsening involves a "global" change: $u^h \leftarrow u^h + w^{2h}$ for some coarse-grid function w^{2h} .
But how do we pick $w^{2h}???$
- Use FE principle of minimizing $F(u^h + w^{2h})$ over w^{2h} :

$$F(u^h + w^{2h}) = (L(u^h + w^{2h}), u^h + w^{2h})/2 - (f, u^h + w^{2h})$$

$$= F(u^h) + (Lu^h - f, w^{2h}) + (Lw^{2h}, w^{2h})/2.$$
- Let w^{2h} be the root of the gradient of this quadratic functional w.r.t. w^{2h} . This is tricky because you need to write the gradient as a function in the subspace H^{2h} . We go instead from abstract functions to nodal vectors...

Concrete FE coarsening (cont'd)

$$A^h u^h = f^h$$

- Solving this matrix equation is equivalent to minimizing
 $F^h(v^h) \equiv (A^h v^h, v^h)/2 - (f^h, v^h)$ (parens here mean Euclidean norm)
over $v^h \in H^h$. So how do we now correct $v^h???$
- We minimize $F^h(v^h + I_{2h}^h v^{2h})$ over $v^{2h} \in H^{2h}$:

$$F^h(v^h + I_{2h}^h v^{2h})$$

$$= (A^h(v^h + I_{2h}^h v^{2h}), v^h + I_{2h}^h v^{2h})/2 - (f^h, v^h + I_{2h}^h v^{2h})$$

$$= F^h(v^h) + (A^{2h} v^{2h}, v^{2h})/2 - (f^{2h}, v^{2h}) \quad \text{variational conditions}$$

$$\begin{aligned} f^{2h} &= I_{2h}^{h^T} (f^h - A^h v^h) \\ A^{2h} &= I_{2h}^{h^T} A^h I_{2h} \end{aligned}$$

$$\equiv F^h(v^h) + F^{2h}(v^{2h}).$$

Concrete FE coarsening: I_{2h}^h

now with focus on nodal vectors

- Adding nodal representations of v^h & v^{2h} :

$$v^{2h}(x, y) = \sum_{ij} v_{ij}^{2h} \varepsilon^{2h}_{(ij)}(x, y)$$

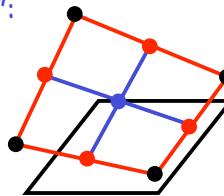
(sum over $2h$ indices
 \leftrightarrow even h indices)

$$= \sum_{ij} v_{ij}^h \varepsilon^h_{(ij)}(x, y) ???$$

(sum over h indices)

- We should be able to do this because $v^h \in H^h \subset H^{2h}$.

- Cell " $2i+2, 2j+2$ ":



$$\begin{aligned} v_{2i+2, 2j}^h &= v_{ij}^{2h} \\ v_{2i+1, 2j}^h &= (v_{ij}^{2h} + v_{i+1, j}^{2h})/2 \\ v_{2i+2, 2j+1}^h &= (v_{ij}^{2h} + v_{i+1, j+1}^{2h})/2 \\ v_{2i+1, 2j+1}^h &= (v_{ij}^{2h} + v_{i+1, j}^{2h} + v_{i+1, j+1}^{2h} + v_{i+1, j+1}^{2h})/4 \end{aligned}$$

Outline

Chapters 1-5:

- ✓ Model Problems
- ✓ Basic Iterative Methods
 - Convergence tests
 - Analysis
- ✓ Elements of Multigrid
 - Relaxation
 - Coarsening
- ✓ Implementation
 - Complexity
 - Diagnostics
- ✓ Some Theory
 - Spectral vs. algebraic

Homework Due
Projects Next Week

Chapters 6-10:

- ✓ Nonlinear Problems
 - Full approximation scheme
- ✓ Selected Applications
 - Neumann boundaries
 - Anisotropic problems
 - Variable meshes
 - Variable coefficients
- ✓ Algebraic Multigrid (AMG)
 - Matrix coarsening
- ✓ Multilevel Adaptive Methods
 - FAC
- ✓ Finite Elements
 - Variational methodology

Multigrid/multilevel/multiscale

an important methodology

Multigrid rules!

We conclude with a few observations:

- o *We have barely scratched the surface of the myriad ways that multigrid has been, & can be, employed.*
- o *With diligence & care, multigrid can be made to handle many types of complications in a robust, efficient manner.*
- o *Further extensions to multigrid methodology are being sought by many people working on many different problems.*

- Multigrid has proved successful on a wide variety of problems, especially elliptic PDEs, but has also found application in parabolic & hyperbolic PDEs, integral equations, evolution problems, geodesic problems, ...
- Multigrid can be optimal, often $O(N)$.
- Multigrid can be robust in a practical sense.
- Multigrid is of great interest because it is one of the very few scalable algorithms, & it can be parallelized readily & efficiently!