A Multigrid Tutorial

By

William L. Briggs

Presented by

Van Emden Henson

Center for Applied Scientific Computing

Lawrence Livermore National Laboratory

This work was performed, in part, under the auspices of the United States Department of Energy by University of California Lawrence Livermore National Laboratory under contract number W-7405-Eng-48.

Outline

- Model Problems
- Basic Iterative Schemes
  - Convergence; experiments
  - Convergence; analysis
- Development of Multigrid
  - Coarse-grid correction
  - Nested Iteration
  - Restriction and Interpolation
  - Standard cycles: MV, FMG
- Performance
  - Implementation
  - storage and computation costs
- Performance, (cont)
  - Convergence
  - Higher dimensions
  - Experiments
- Some Theory
  - The spectral picture
  - The subspace picture
  - The ***whole*** picture!
- Complications
  - Anisotropic operators and grids
  - Discontinuous or anisotropic coefficients
  - Nonlinear Problems: FAS

Suggested Reading

- Brandt, “Multi-level Adaptive Solutions to Boundary Value Problems,” Math Comp., 31, 1977, pp 333-390.
- Brandt, “1984 Guide to Multigrid Development, with applications to computational fluid dynamics.”
- Briggs, “A Multigrid Tutorial,” SIAM publications, 1987.
- Briggs, Henson, and McCormick, “A Multigrid Tutorial, 2nd Edition,” SIAM publications, 2000.
- Hackbusch, Multi-Grid Methods and Applications,” 1985.
- Hackbusch and Trottenburg, “Multigrid Methods, Springer-Verlag, 1982”
- Stüben and Trottenburg, “Multigrid Methods,” 1987.
- Wesseling, “An Introduction to Multigrid Methods,” Wylie, 1992

Multilevel methods have been developed for...

- Elliptic PDEs
- Purely algebraic problems, with no physical grid; for example, network and geodetic survey problems.
- Image reconstruction and tomography
- Optimization (e.g., the travelling salesman and long transportation problems)
- Statistical mechanics, Ising spin models.
- Quantum chromodynamics.
- Quadrature and generalized FFTs.
- Integral equations.

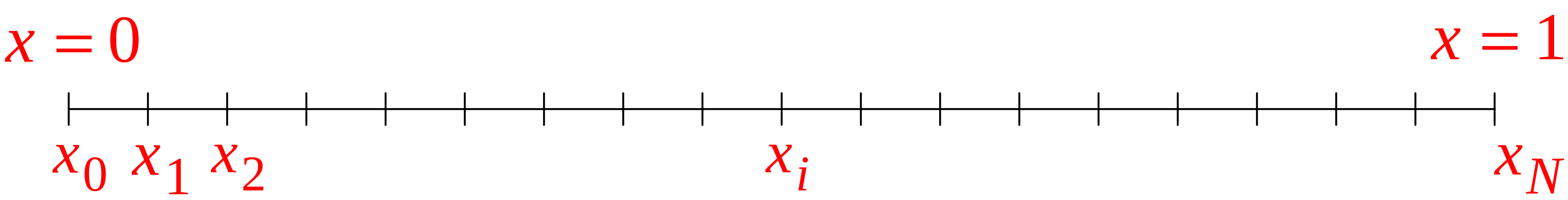
Model Problems

- One-dimensional boundary value problem:

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

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

- Grid: $h = \frac{1}{N}$, $x_i = ih$, $i = 0, 1, \ldots N$



- Let $v_i \approx u(x_i)$ and $f_i \approx f(x_i)$ for $i = 0, 1, \ldots N$

We use Taylor Series to derive an approximation to u''(x)

- We approximate the second 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 and solving,

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

We approximate the equation with a finite difference scheme

- We approximate the BVP

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

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

with the finite difference scheme:

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

$$v_0 = v_N = 0$$

The discrete model problem

- Letting $\boldsymbol{v} = (v_1, v_2, \ldots, v_{N-1})^T$ and

$$\boldsymbol{f} = (f_1, f_2, \ldots, f_{N-1})^T$$

we obtain the matrix equation $A\boldsymbol{v} = \boldsymbol{f}$ where A is (N-1) x (N-1), symmetric, positive definite, and

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

Solution Methods

- Direct
  - Gaussian elimination
  - Factorization

- Iterative
  - Jacobi
  - Gauss-Seidel
  - Conjugate Gradient, etc.

- Note: This simple model problem can be solved very efficiently in several ways. Pretend it can't, and that it is very hard, because it shares many characteristics with some very hard problems.

A two-dimensional boundary value problem

- Consider the problem:

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

$$u = 0 \ , \ x = 0, \ x = 1, \ y = 0, \ y = 1; \qquad \sigma > 0$$

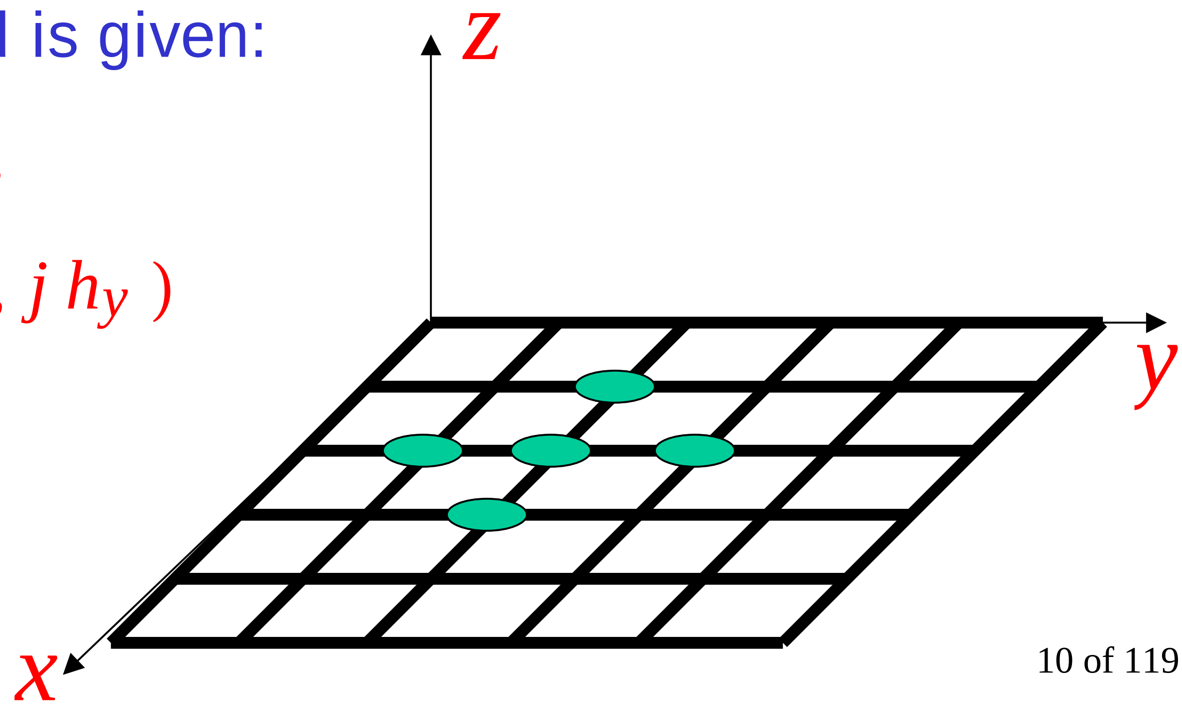
- Where the grid is given:

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

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

$$0 \le i \le M$$

$$0 \le j \le N$$



Discretizing the 2D problem

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

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

$$v_{ij} = 0 , \quad i = 0, \ \ i = M, \ \ j = 0, \ \ j = M$$

- Ordering the unknowns (and also the vector $\boldsymbol{f}$) lexicographically by y-lines:

$$\boldsymbol{v} = (v_{1,1}, v_{1,2}, \ldots, v_{1,N-1}, v_{2,1}, v_{2,2}, \ldots v_{2,N-1}, \ldots, v_{N-1,1}, v_{N-1,2}, \ldots, v_{N-1,N-1})^T$$

Yields the linear system

- We obtain a block-tridiagonal system $Av = f$:

$$\begin{pmatrix} A_1 & -I_y & & & & \\ -I_y & A_2 & -I_y & & & \\ & -I_y & A_3 & -I_y & & \\ & & & & & \\ & & & -I_y & A_{N-2} & -I_y \\ & & & & -I_y & A_{N-1} \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ \\ v_{N-1} \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ \\ f_{N-1} \end{pmatrix}$$

where I_y is a diagonal matrix with $\frac{1}{h_y^2}$ on the diagonal and

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

Iterative Methods for Linear Systems

- Consider $A\boldsymbol{u} = \boldsymbol{f}$ where A is NxN and let $\boldsymbol{v}$ be an approximation to $\boldsymbol{u}$.

- Two important measures:
  - The Error: $e = u - v$, with norms

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

  - The Residual: $r = f - Av$ with

$$\| r \|_\infty \qquad \| r \|_2$$

Residual correction

- Since $e = u - v$, and $r = f - Av$, we can write $Au = f$ as
$$A(v + e) = f$$
which means that $Ae = f - Av$, which is the Residual Equation:
$$\boxed{Ae = r}$$

- Residual Correction:
$$\boxed{u = v + e}$$

Relaxation Schemes

- Consider the 1D model problem

$$-u_{i-1} + 2u_i - u_{i+1} = h^2 f_i \qquad 1 \le i \le N-1 \qquad u_0 = u_N = 0$$

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

$$v_i^{(new)} = \frac{1}{2}\left(v_{i-1}^{(old)} + v_{i+1}^{(old)} + h^2 f_i \right) \qquad 1 \le i \le N-1$$

In matrix form, the relaxation is

- Let $A = (D - L - U)$ where D is diagonal and L and U are the strictly lower and upper parts of A.
- Then $Au = f$ becomes

$$(D - L - U)\, u = f$$

$$D\, u = (L + U)\, u + f$$

$$u = D^{-1}(L + U)\, u + D^{-1} f$$

- Let $R_J = D^{-1}(L + U)$, then the iteration is:

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

The iteration matrix and the error

- From the derivation,

$$u = D^{-1}(L+U)u + D^{-1}f$$
$$u = R_J u + D^{-1}f$$

- 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

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

Weighted Jacobi Relaxation

- Consider the iteration:

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

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

$$v^{(new)} = \left[(1-\omega) I + \omega D^{-1}(L+U) \right] v^{(old)} + \omega h^2 D^{-1} f$$

$$= R_\omega v^{(old)} + \omega h^2 D^{-1} f \quad .$$

- Note that

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

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

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

Gauss-Seidel Relaxation (1D)

- Solve equation ***i*** for u_i and update immediately.
- Equivalently: set each component of ***r*** to zero.
- Component form: for $i = 1, 2, \ldots N-1,$ set

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

- Matrix form:

$$A = (D - L - U)$$

$$(D - L)\, u = U u + f$$

$$u = (D - L)^{-1} U u + (D - L)^{-1} f$$

- Let $R_G = (D - L)^{-1} U$
- Then iterate $v^{(new)} \leftarrow R_G v^{(old)} + (D - L)^{-1} f$

- Error propagation: $e^{(new)} \leftarrow R_G e^{(old)}$

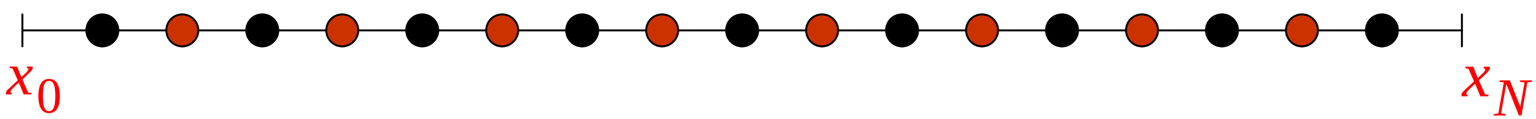
Red-Black Gauss-Seidel

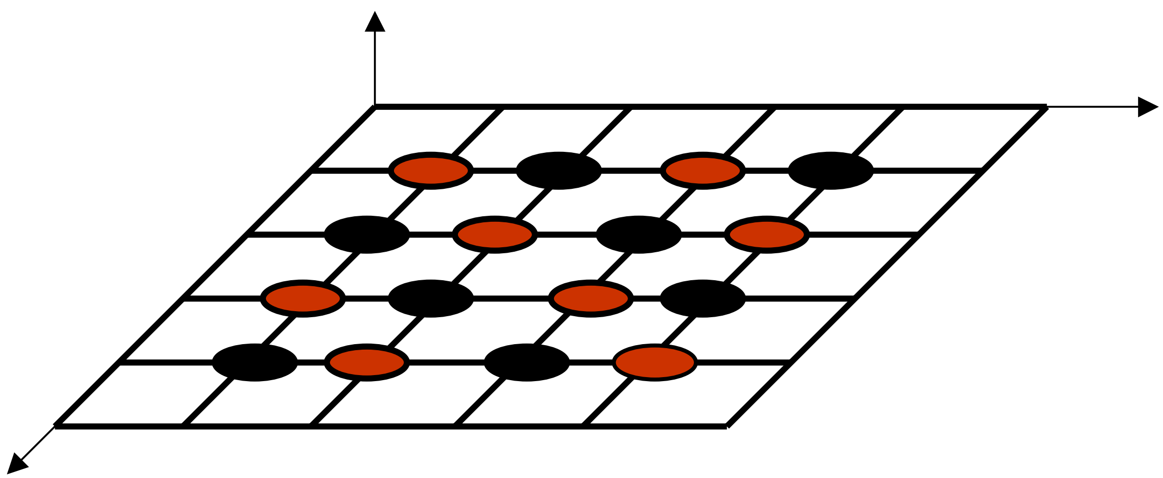
- Update the even (red) points

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

- Update the odd (black) points

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

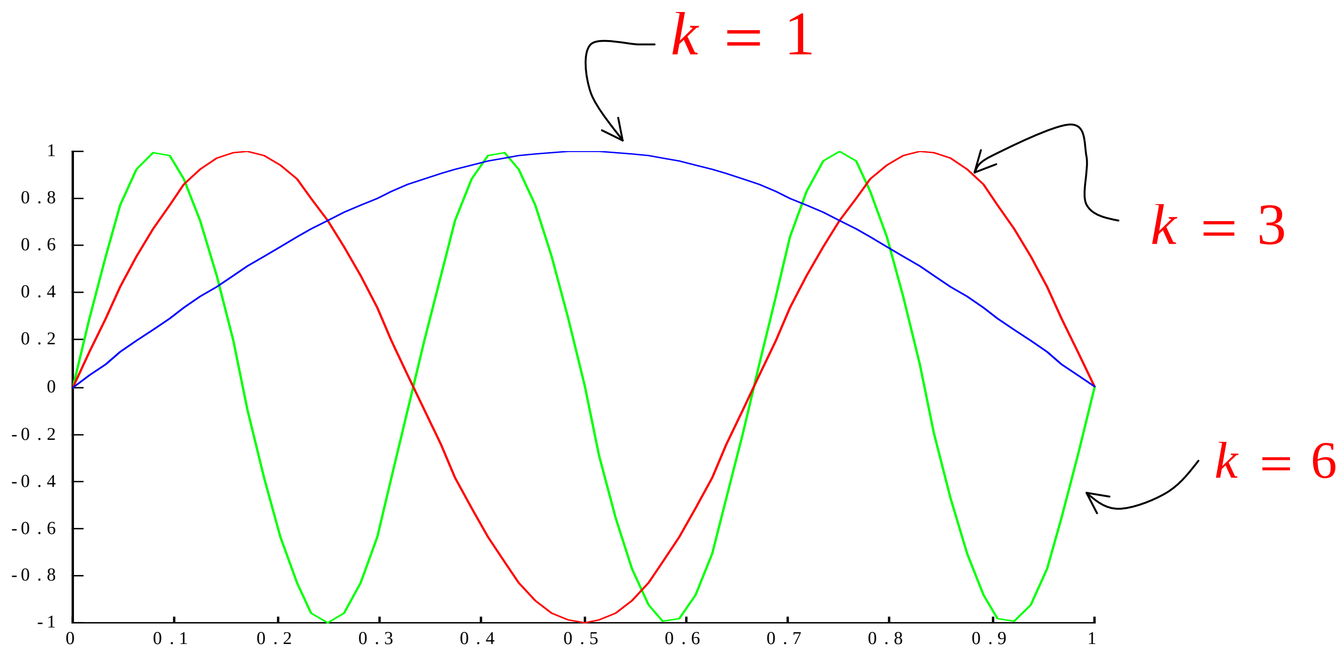




Numerical Experiments

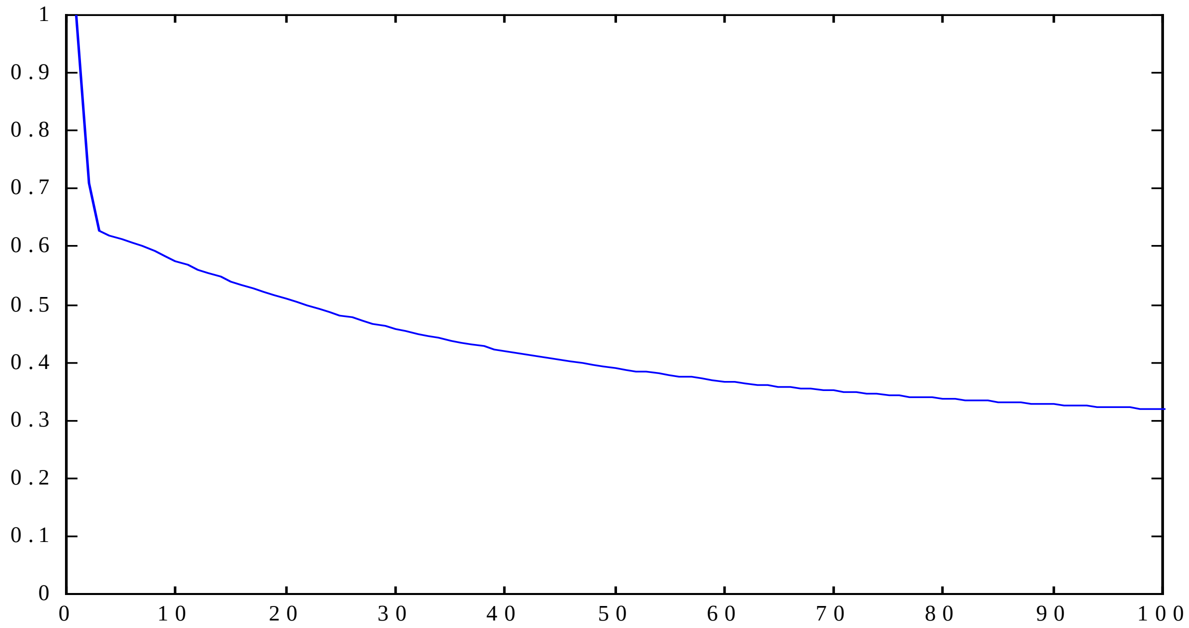
- Solve $Au = 0$, $-u_{i-1} + 2u_i - u_{i+1} = 0$
- Use Fourier modes as initial iterate, with N =64:

$$\vec{v}_k = (v_i)_k = \sin\left(\frac{ik\pi}{N}\right) \qquad \underbrace{1 \le i \le N-1}_{\text{component}}, \quad \underbrace{1 \le k \le N-1}_{\text{mode}}$$



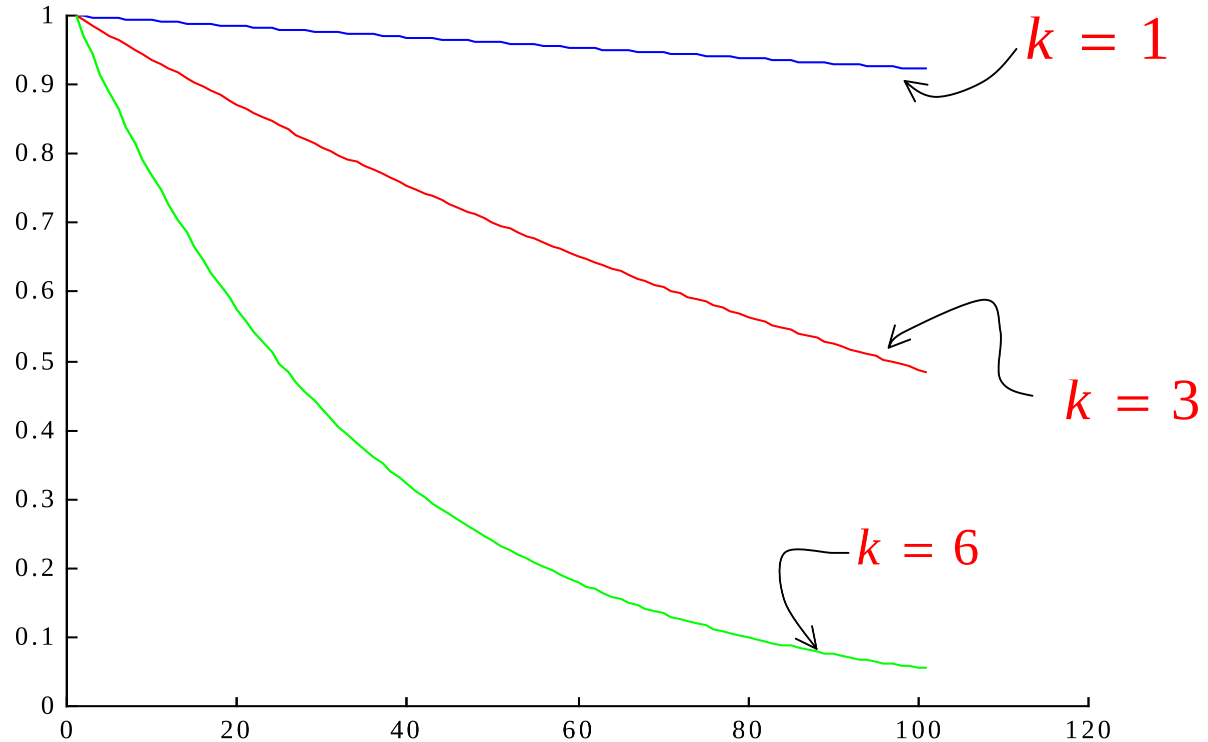
Error reduction stalls

- Weighted $\omega = \frac{2}{3}$ Jacobi on 1D problem.
- Initial guess: $v_0 = \frac{1}{3}\left(\sin\left(\frac{j\pi}{N}\right) + \sin\left(\frac{6j\pi}{N}\right) + \sin\left(\frac{32j\pi}{N}\right) \right)$
- Error $\|e\|_\infty$ plotted against iteration number:



Convergence rates differ for different error components

- Error, $||e||_\infty$, in weighted Jacobi on Au = 0 for 100 iterations using initial guesses of v_1, v_3, and v_6



Analysis of stationary iterations

- Let $v^{(new)} = R v^{(old)} + g$. The exact solution is unchanged by the iteration, i.e., $u = R u + g$
- Subtracting, we see that

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

- Letting $\boldsymbol{e^0}$ be the initial error and $\boldsymbol{e^i}$ be the error after the i^{th} iteration, we see that after $\boldsymbol{n}$ iterations we have

$$e^{(n)} = R^n e^{(0)}$$

A quick review of eigenvectors and eigenvalues

- The number λ is an eigenvalue of a matrix B, and w its associated eigenvector, if $Bw = \lambda w$.

- The eigenvalues and eigenvectors are characteristics of a given matrix.

- Eigenvectors are linearly independent, and if there is a complete set of N distinct eigenvectors for an NxN matrix, they form a basis; i.e., for any v, there exist unique c_k such that:

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

"Fundamental theorem of iteration"

- ***R*** is convergent (that is, $R^n \to 0$ as $n \to \infty$) if and only if $\rho(R) < 1$, where

$$\rho(R) = \max \{ |\lambda_1|, |\lambda_2|, \ldots, |\lambda_N| \}$$

therefore, for any initial vector $\boldsymbol{v}^{(0)}$, we see that $e^{(n)} \to 0$ as $n \to \infty$ if and only if $\rho(R) < 1$.

- $\rho(R) < 1$ assures the convergence of the iteration given by ***R*** and $\rho(R)$ is called the *convergence factor* for the iteration.

Convergence Rate

- How many iterations are needed to reduce the initial error by 10^{-d} ?

$$\frac{\| e^{(M)} \|}{\| e^{(0)} \|} \le \| R^M \| \sim (\rho(R))^M \sim 10^{-d}$$

- So, we have $M = \dfrac{d}{\log_{10}\left(\dfrac{1}{\rho(R)}\right)}$.

- The *convergence rate* is given:

$$\text{rate} = \log_{10}\left(\frac{1}{\rho(R)}\right) = -\log_{10}(\rho(R)) \ \frac{\text{digits}}{\text{iteration}}$$

Convergence analysis for weighted Jacobi on 1D model

$$R_\omega = (1-\omega)I + \omega D^{-1}(L+U)$$
$$= I - \omega D^{-1} A$$

$$R_\omega = I - \frac{\omega}{2}\begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & 2 & -1 & \\ & & \ddots & \ddots & \ddots \\ & & & -1 & 2 \end{pmatrix}$$

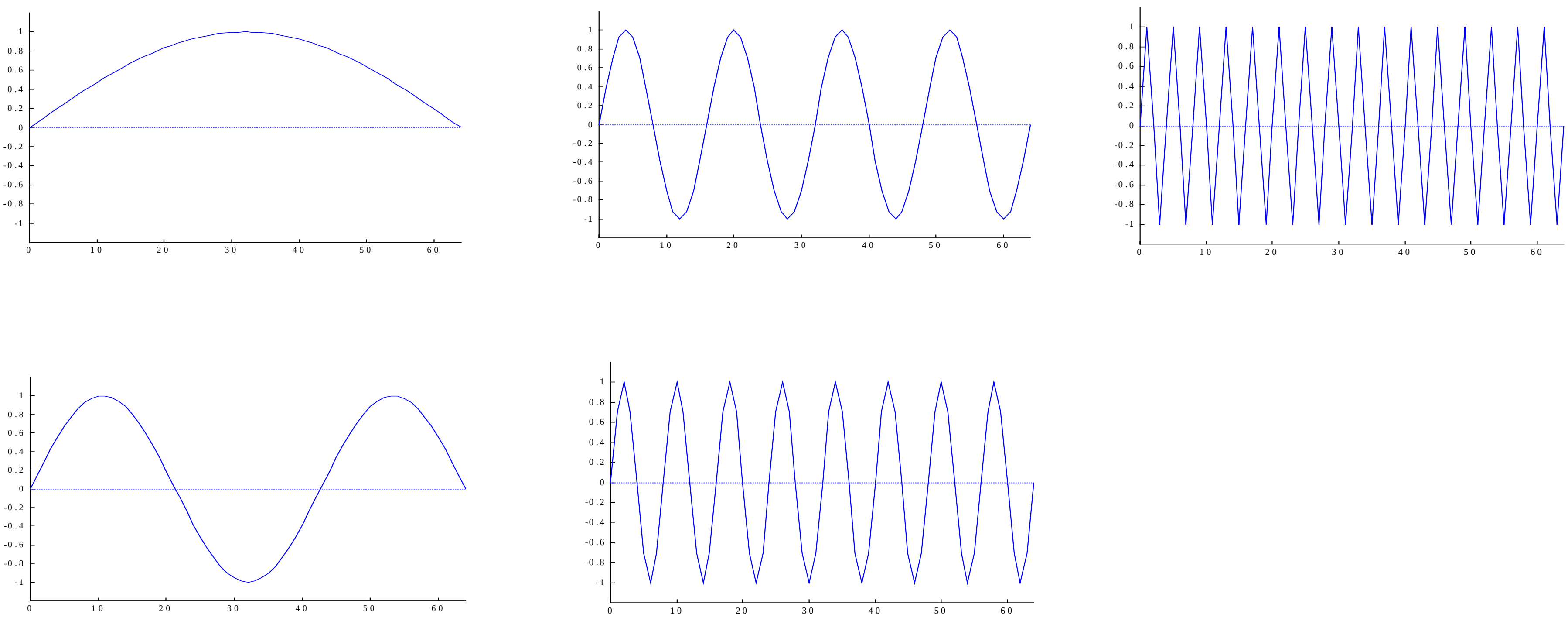
$$\lambda(R_\omega) = 1 - \frac{\omega}{2}\lambda(A)$$

For the 1D model problem, he eigenvectors of the weighted Jacobi iteration and the eigenvectors of the matrix A are the same! The eigenvalues are related as well.

Good exercise: Find the eigenvalues & eigenvectors of A

- Show that the eigenvectors of A are Fourier modes!

$$\lambda_k(A) = 4\sin^2\left(\frac{k\pi}{2N}\right), \qquad w_{k,j} = \sin\left(\frac{jk\pi}{N}\right)$$



Eigenvectors of R_ω and A are the same, the eigenvalues related

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

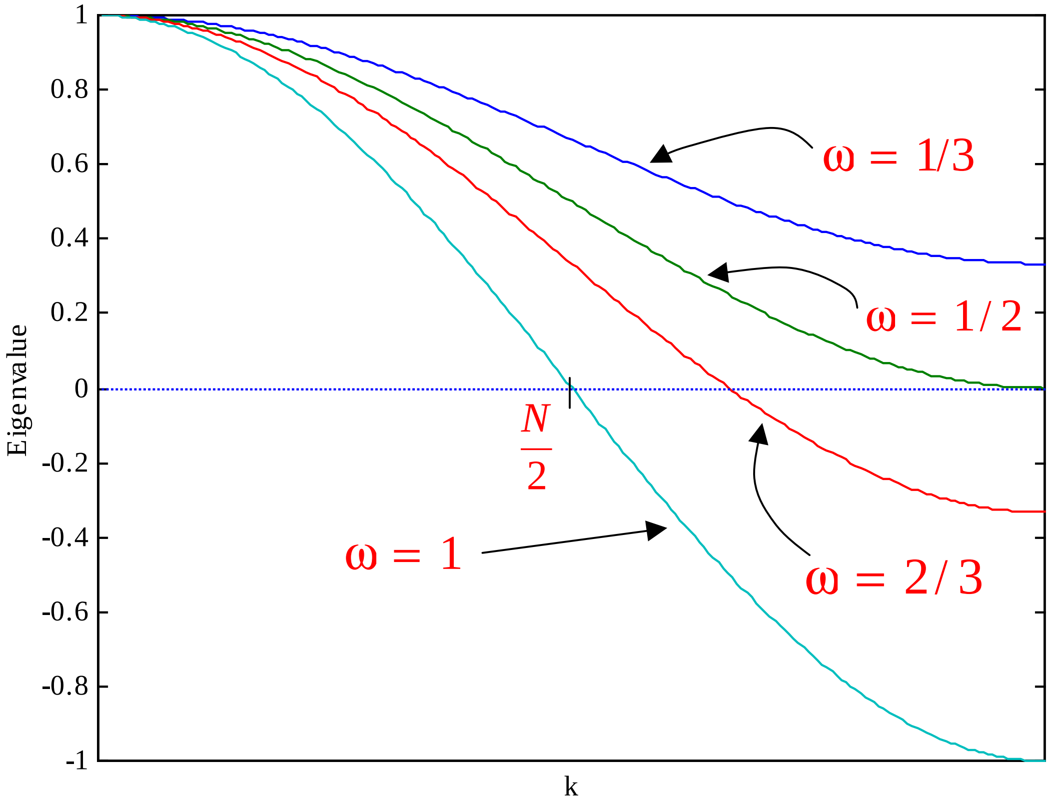
- Expand the initial error in terms of the eigenvectors:
$$e^{(0)} = \sum_{k=1}^{N-1} c_k\, w_k$$

- After M iterations,
$$R^M e^{(0)} = \sum_{k=1}^{N-1} c_k R^M w_k = \sum_{k=1}^{N-1} c_k \lambda_k^M w_k$$

- The k^{th} mode of the error is reduced by λ_k at each iteration

Relaxation suppresses eigenmodes unevenly

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



Note that if $0 \le \omega \le 1$ then $|\lambda_k(R_\omega)| < 1$ for $k = 1, 2, \ldots, N-1$

For $0 \le \omega \le 1$,

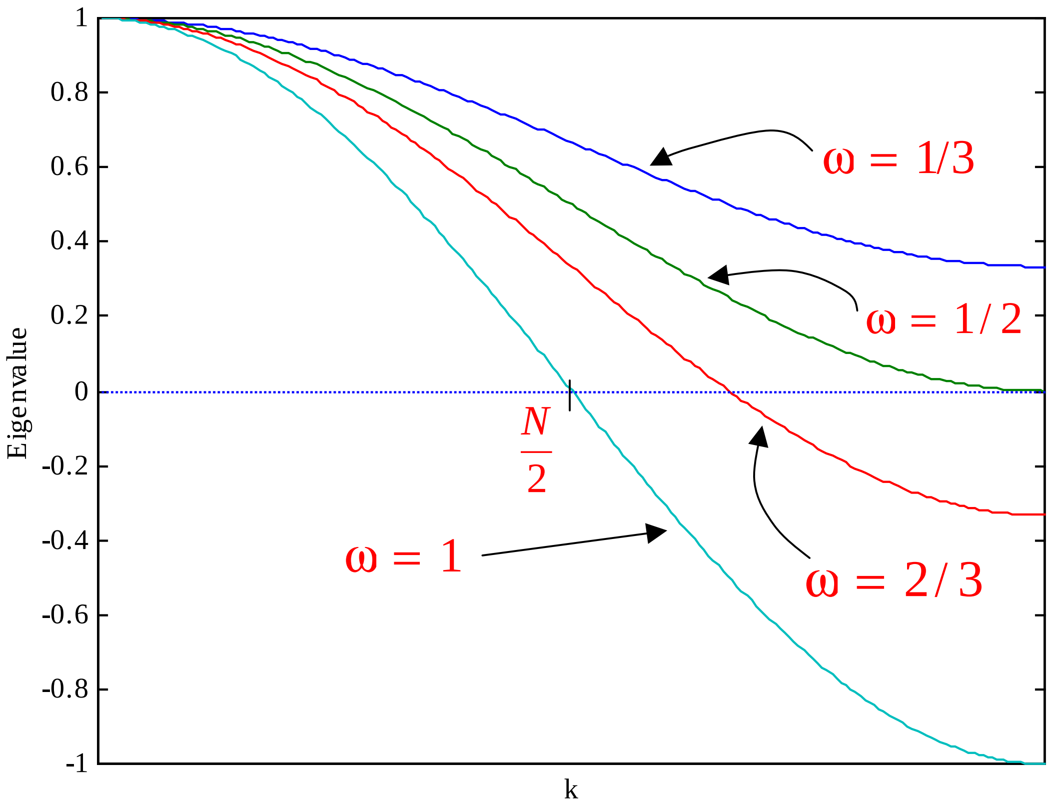
$$\lambda_1 = 1 - 2\omega\sin^2\left(\frac{\pi}{2N}\right)$$

$$= 1 - 2\omega\sin^2\left(\frac{\pi h}{2}\right)$$

$$= 1 - O(h^2) \approx 1$$

Low frequencies are undamped

- Notice that no value of ω will damp out the long (i.e., low frequency) waves.



What value of ω gives the best damping of the short waves ?

$$\frac{N}{2} \le k \le N$$

Choose ω such that

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

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

The Smoothing factor

- The smoothing factor is the largest absolute value among the eigenvalues in the upper half of the spectrum of the iteration matrix

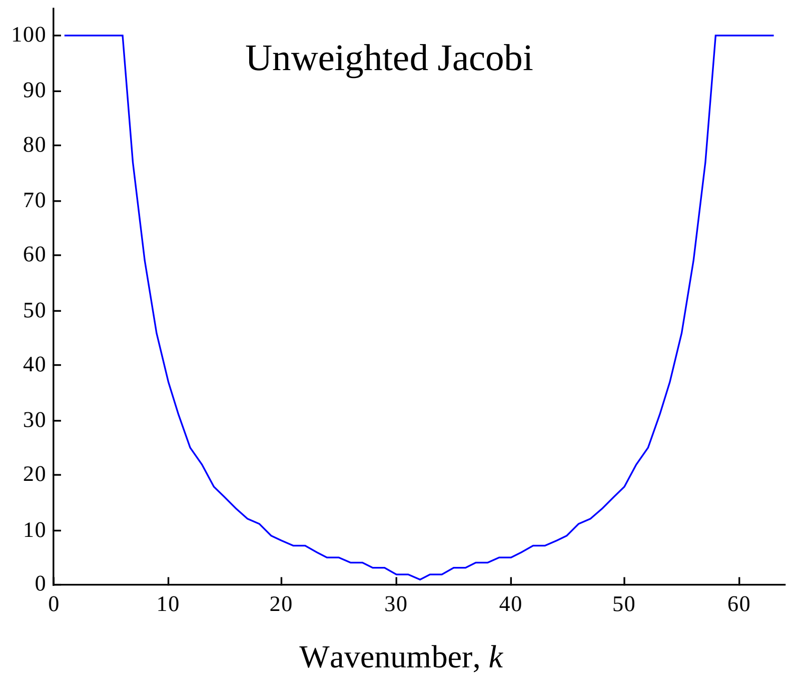
$$\text{smoothing factor} = \max \left| \lambda_k(R) \right| \quad \text{for} \quad \frac{N}{2} \le k \le N$$

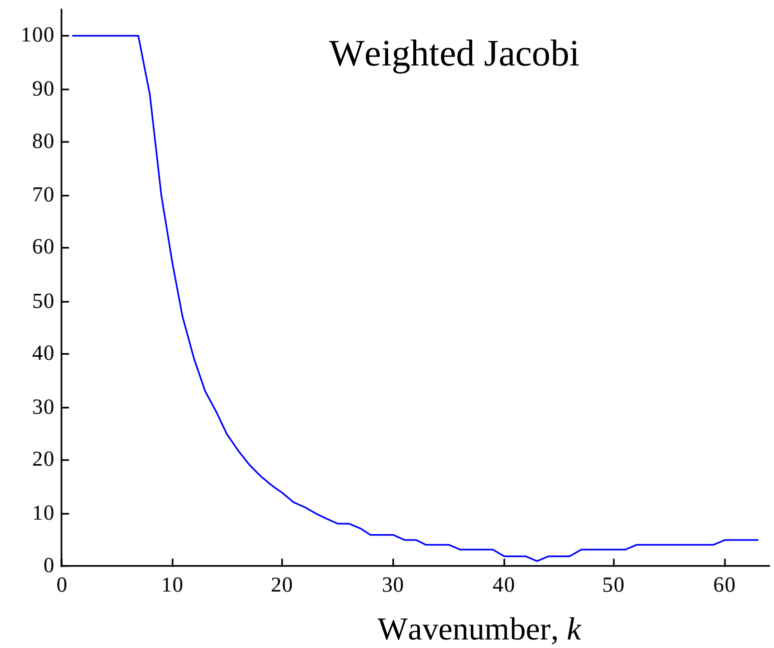
- For R_ω, with $\omega = \frac{2}{3}$, the smoothing factor is $\frac{1}{3}$, since

$$\left| \lambda_{\frac{N}{2}} \right| = \left| \lambda_N \right| = \frac{1}{3} \text{ and } \left| \lambda_k \right| < \frac{1}{3} \quad \text{for} \quad \frac{N}{2} < k < N.$$

- But, $\left| \lambda_k \right| \approx 1 - \frac{2}{3} k^2 \pi^2 h^2$ for long waves $\left(k \ll \frac{N}{2} \right)$.

Convergence of Jacobi on Au=0



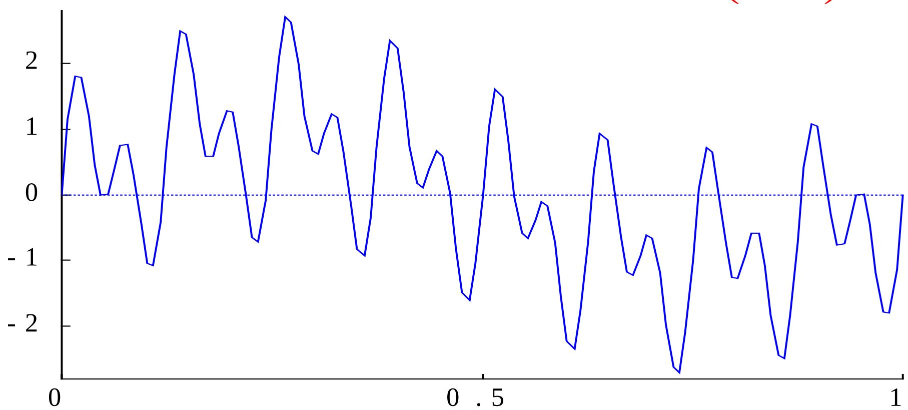


- Jacobi method on Au=0 with N=64. Number of iterations required to reduce to $||e||_\infty < .01$

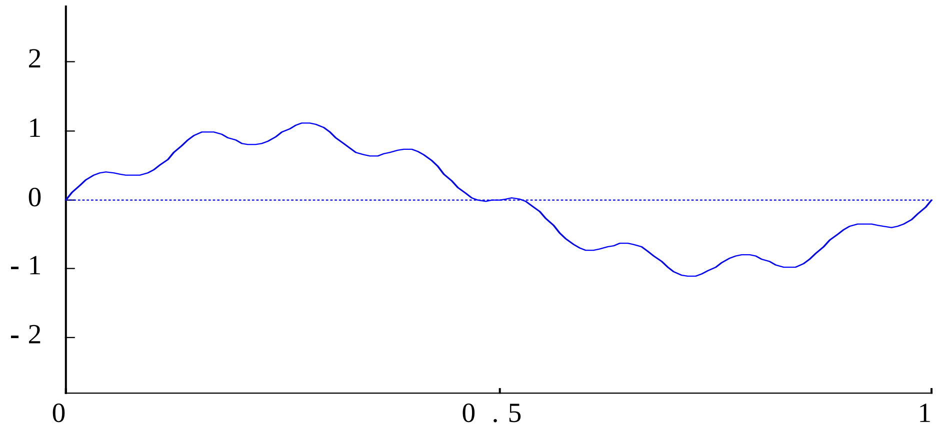
- Initial guess : $v_{kj} = \sin\left(\frac{jk\pi}{N}\right)$

Weighted Jacobi Relaxation Smooths the Error

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



- Error after 35 iteration sweeps:



Many relaxation schemes have the smoothing property, where oscillatory modes of the error are eliminated effectively, but smooth modes are damped very slowly.

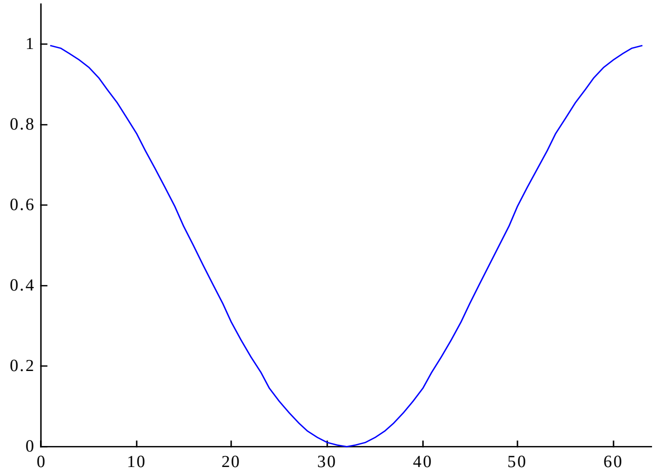
Other relaxations schemes may be analyzed similarly

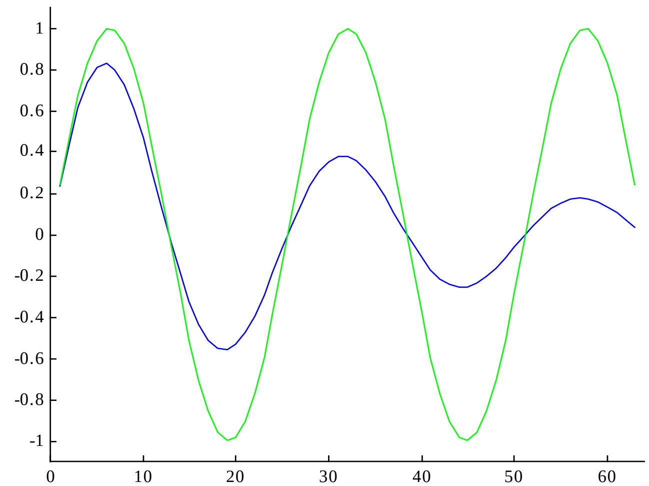
- Gauss-Seidel relaxation applied to the 3-point difference matrix A (1D model problem):

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

- Good exercise: Show that

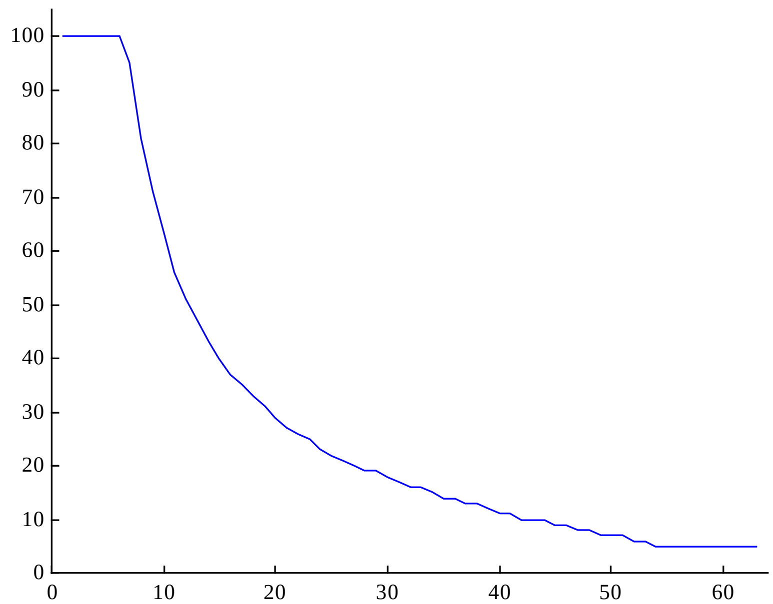
$$\lambda_k(R_G) = \cos^2\left(\frac{k\pi}{N}\right) \qquad w_{k,j} = (\lambda_k)^{j/2} \sin\left(\frac{jk\pi}{N}\right)$$





Convergence of Gauss-Seidel on Au=0

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



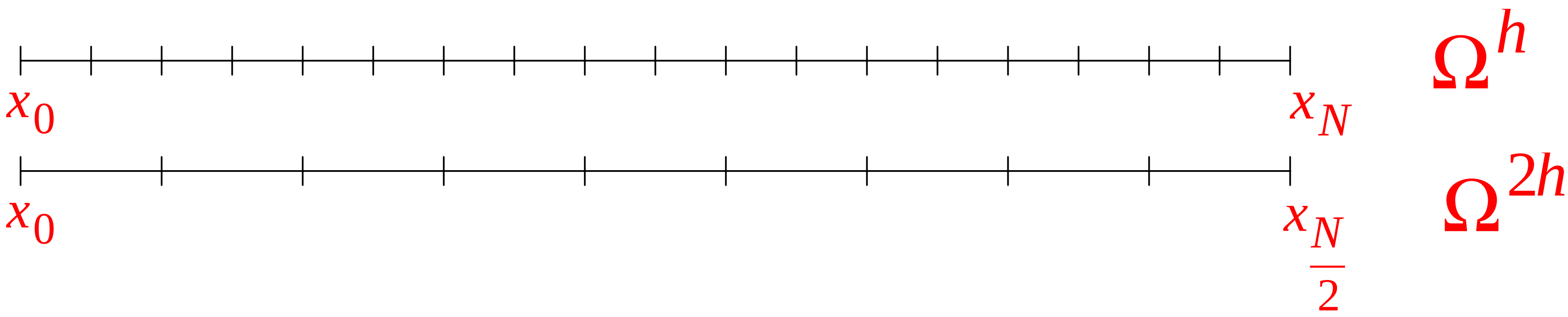
Jacobi method on Au=0 with N=64. Number of iterations required to reduce to $||e||_\infty < .01$

Initial guess : (Modes of A)

$$v_{kj} = \sin\left(\frac{jk\pi}{N}\right)$$

First observation toward multigrid

- **Many relaxation schemes have the smoothing property, where oscillatory modes of the error are eliminated effectively, but smooth modes are damped very slowly.**
- This might seem like a limitation, but by using coarse grids we can use the smoothing property to good advantage.



- Why use coarse grids??

Reason #1 for using 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 (1/2 as many points in 1D, 1/4 in 2D, 1/8 in 3D)

  – Relaxation on the coarse grid has a marginally better convergence rate, for example

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

Idea! Nested Iteration

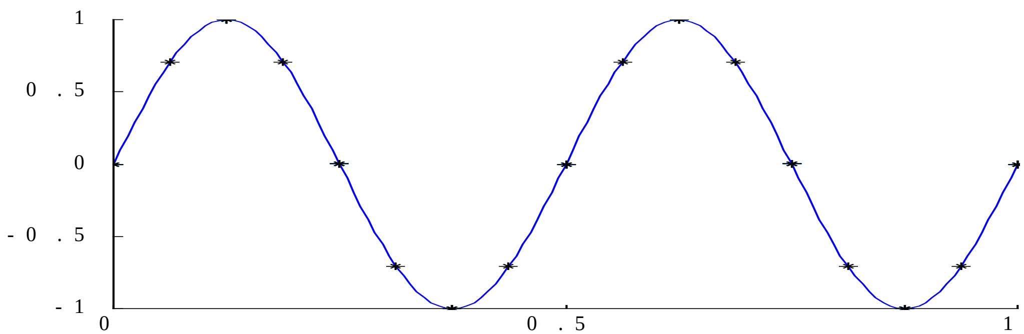
- ...
- Relax on Au=f on Ω^{4h} to obtain initial guess v^{2h}
- Relax on Au=f on Ω^{2h} to obtain initial guess v^{h}
- Relax on Au=f on Ω^{h} to obtain ... final solution???

- But, what is Au=f on Ω^{2h}, Ω^{4h} , ... ?

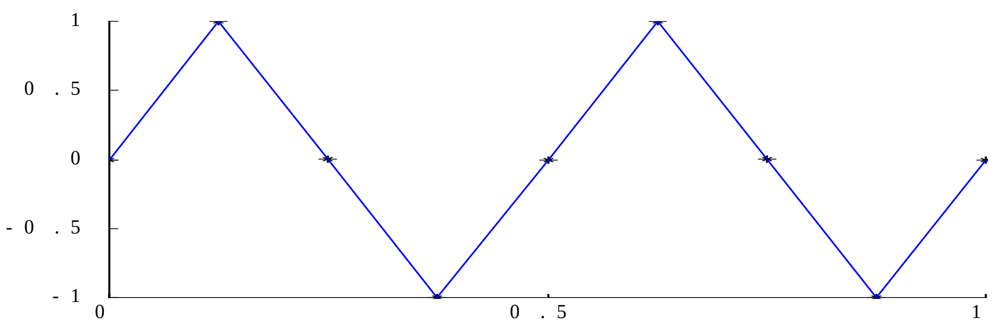
- What if the error still has smooth components when we get to the fine grid Ω^{h} ?

Reason #2 for using a coarse grid: smooth error is (relatively) more oscillatory there!

- A smooth function:



- Can be represented by linear interpolation from a coarser grid:

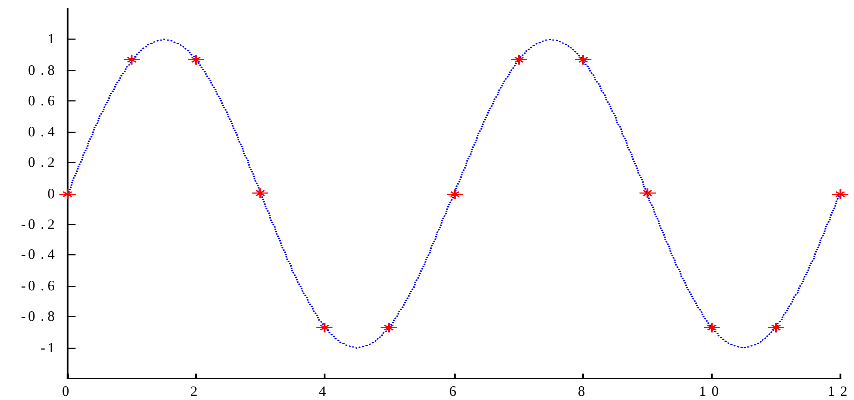


On the coarse grid, the smooth error appears to be relatively higher in frequency: in the example it is the 4-mode, out of a possible 16, on the fine grid, 1/4 the way up the spectrum. On the coarse grid, it is the 4-mode out of a possible 8, hence it is 1/2 the way up the spectrum.

Relaxation will be more effective on this mode if done on the coarser grid!!

For k=1,2,…N/2, the k^{th} mode is preserved on the coarse grid

$$w_{k,2j}^{h} = \sin\left(\frac{2jk\pi}{N}\right) = \sin\left(\frac{jk\pi}{N/2}\right) = w_{k,j}^{2h}$$

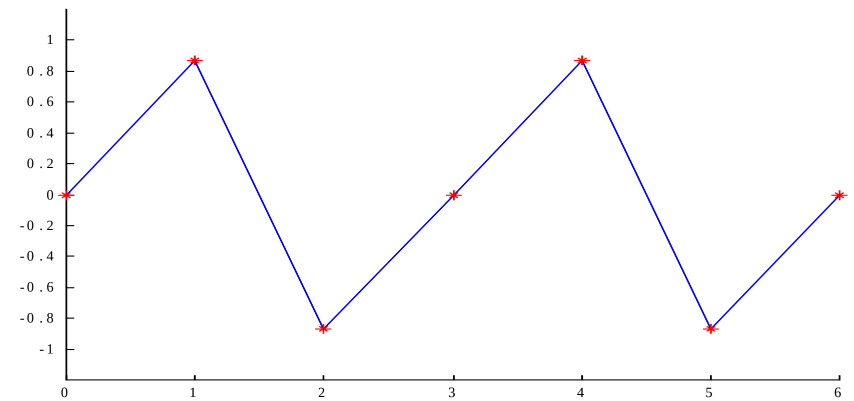


k=4 mode, N=12 grid

Also, note that

$$w_{N/2}^{h} \rightarrow 0$$

on the coarse grid



k=4 mode, N=6 grid

For $k > N/2$, w_k^h is invisible on the coarse grid: aliasing!!

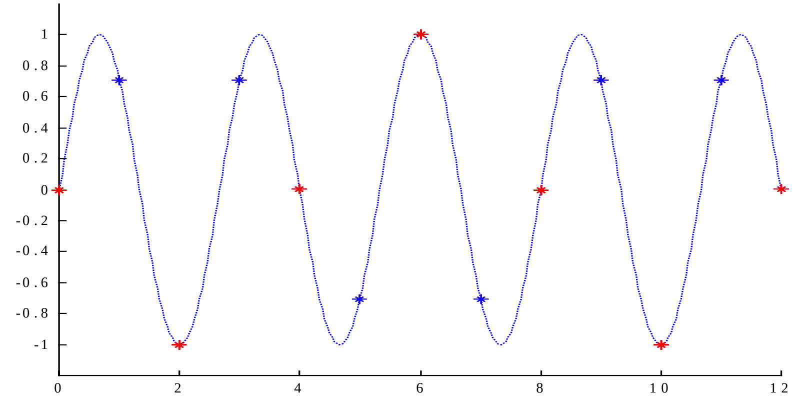
- For $k > N/2$, the k^{th} mode on the fine grid is aliased and appears as the $(N-k)^{th}$ mode on the coarse grid:

$$(w_k^h)_{2j} = \sin\left(\frac{(2j)\pi k}{N}\right)$$

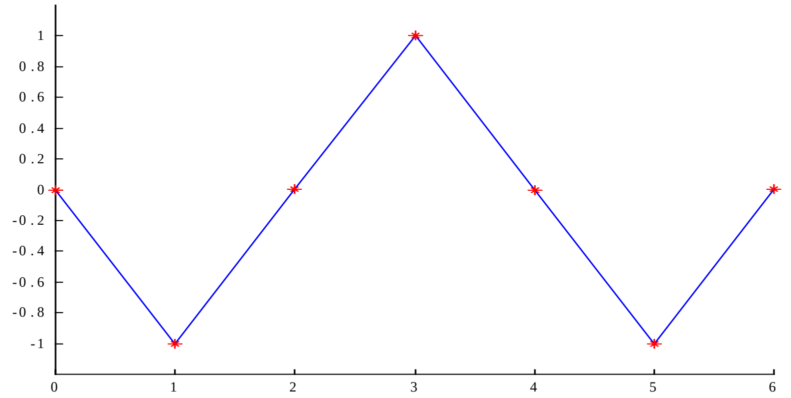
$$= -\sin\left(\frac{2\pi j(N-k)}{N}\right)$$

$$= -\sin\left(\frac{\pi(N-k)j}{N/2}\right)$$

$$= -(w_{N-k}^{2h})$$



k=9 mode, N=12 grid



k=3 mode, N=12 grid

Second observation toward multigrid:

- Recall the residual correction idea: Let v be an approximation to the solution of $Au=f$, where the residual $r=f-Av$. The the error $e=u-v$ satisfies $Ae=r$.

- After relaxing on $Au=f$ on the fine grid, the error will be smooth. On the coarse grid, however, this error appears more oscillatory, and relaxation will be more effective.

- Therefore we go to a coarse grid and relax on the residual equation $Ae=r$, with an initial guess of $e=0$.

Idea! Coarse-grid correction

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

- Clearly, we need methods for the mappings

$\Omega^h \Rightarrow \Omega^{2h}$ and $\Omega^{2h} \Rightarrow \Omega^h$

1D Interpolation (Prolongation)

- Mapping from the coarse grid to the fine grid:

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

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

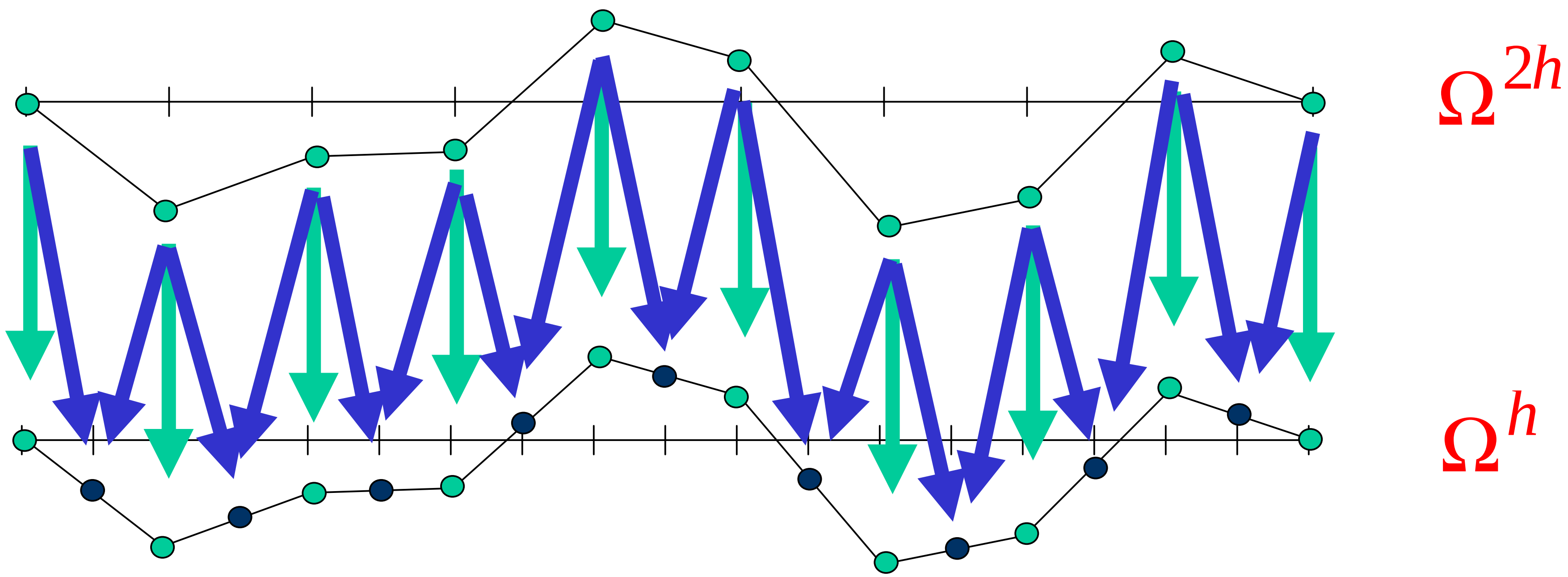
$$I_{2h}^{h} v^{2h} = v^{h}$$

where

$$\left.\begin{aligned} v_{2i}^{h} &= v_{i}^{2h} \\ v_{2i+1}^{h} &= \frac{1}{2}\left(v_{i}^{2h} + v_{i+1}^{2h} \right) \end{aligned}\right\} \text{ for } \quad 0 \le i \le \frac{N}{2} - 1$$

1D 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



The prolongation operator (1D)

- We may regard I_{2h}^{h} as a linear operator from $\Re^{N/2-1} \longrightarrow \Re^{N-1}$

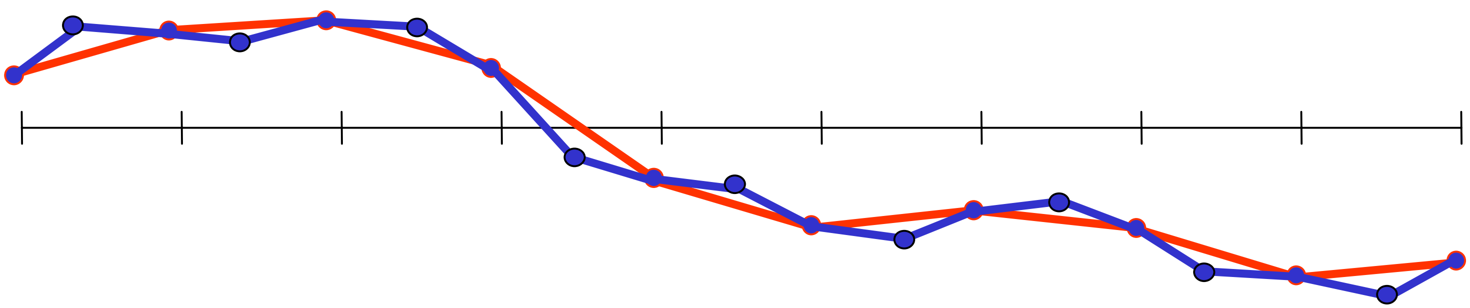
- e.g., for N=8,

$$\begin{pmatrix} 1/2 & & \\ 1 & & \\ 1/2 & 1/2 & \\ & 1 & \\ & 1/2 & 1/2 \\ & & 1 \\ & & 1/2 \end{pmatrix}_{7x3} \begin{pmatrix} v_1^{2h} \\ v_2^{2h} \\ v_3^{2h} \end{pmatrix}_{3x1} = \begin{pmatrix} v_1^{h} \\ v_2^{h} \\ v_3^{h} \\ v_4^{h} \\ v_5^{h} \\ v_6^{h} \\ v_7^{h} \end{pmatrix}_{7x1}$$

- I_{2h}^{h} has full rank, and thus null space $\{\phi\}$

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

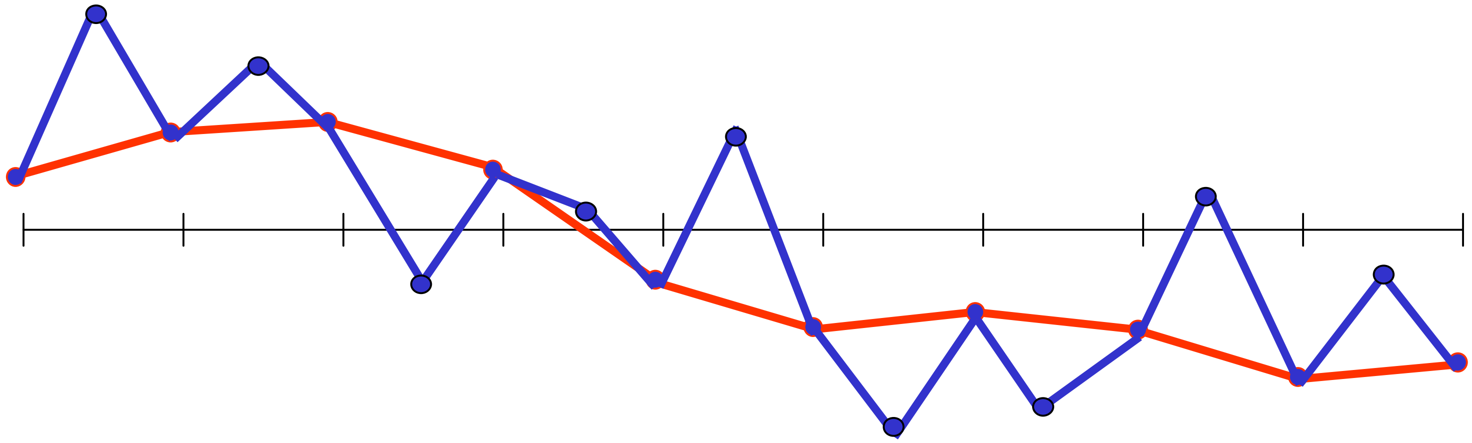
- Imagine that a coarse-grid approximation v^{2h} has been found. How well does it approximate the exact solution u ? Think u $\rightarrow$ error!!



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

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

- Imagine that a coarse-grid approximation v^{2h} has been found. How well does it approximate the exact solution u ? Think u → error!!



- If u is oscillatory, a coarse-grid interpolant of v^{2h} may ***<u>not</u>*** work well.

Moral of this story:

- If u is smooth, a coarse-grid interpolant of v^{2h} may do very well.
- If u is oscillatory, a coarse-grid interpolant of v^{2h} may ***not*** work well.
- Therefore, nested iteration is most effective when the error is smooth!

1D Restriction by injection

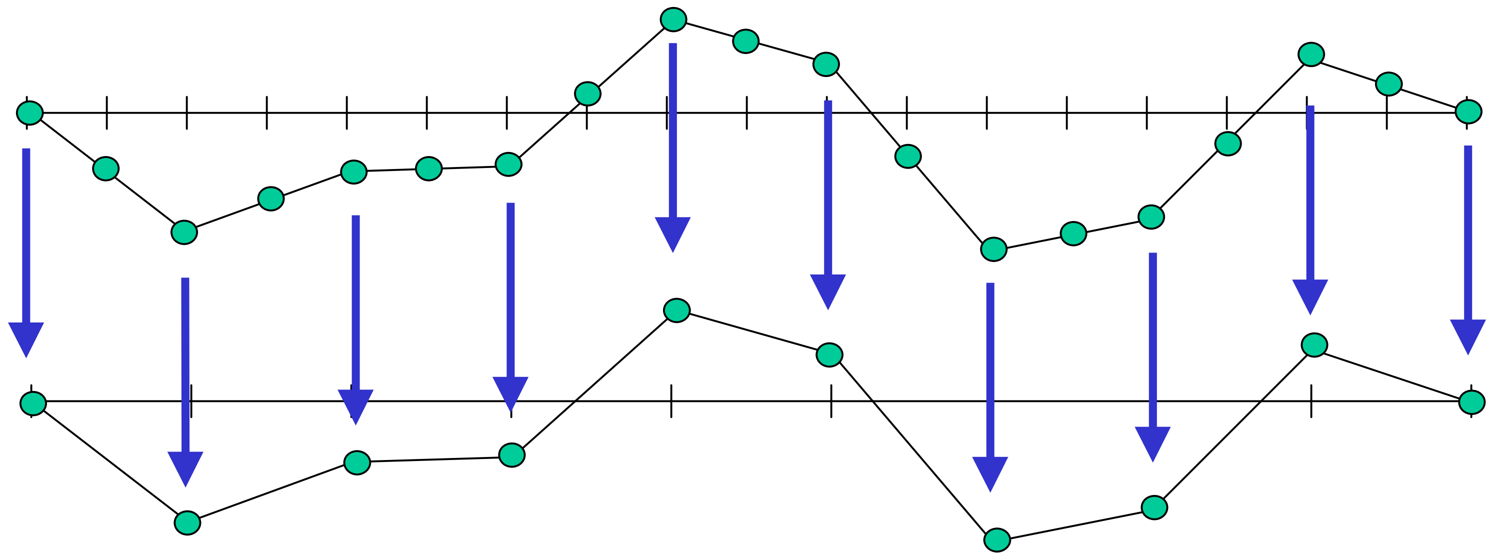
- Mapping from the fine grid to the coarse grid:

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

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

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

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



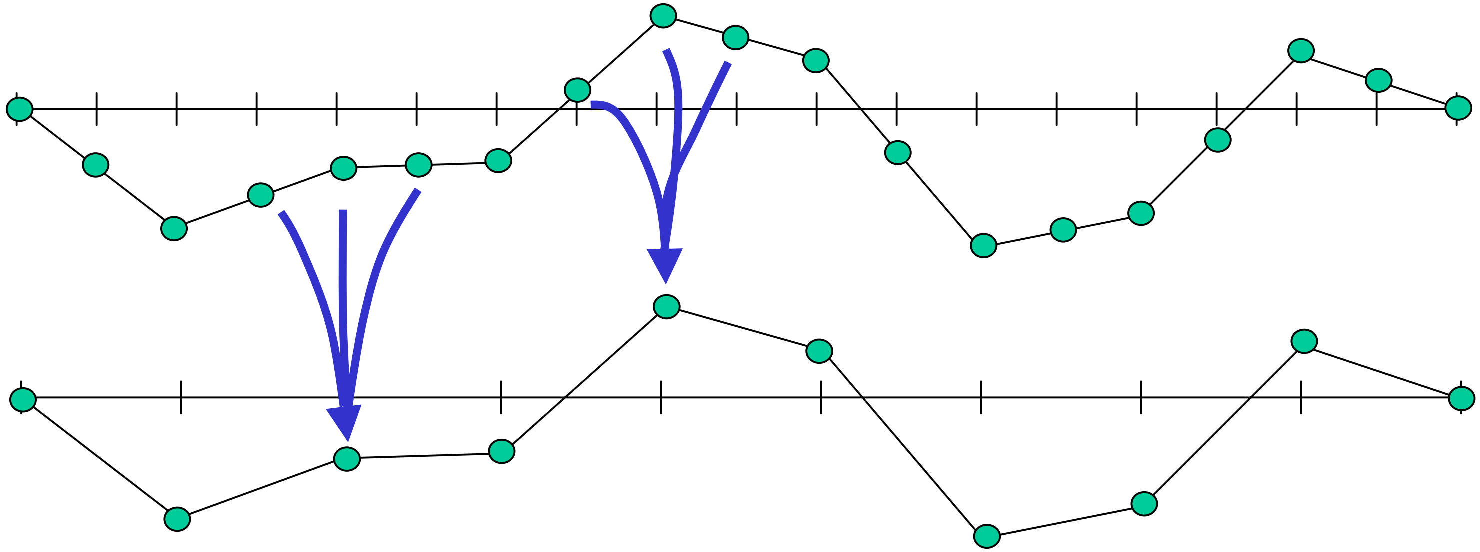
1D Restriction by full-weighting

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

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

where

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



The restriction operator R (1D)

- We may regard I_h^{2h} as a linear operator from $\Re^{N-1} \longrightarrow \Re^{N/2-1}$

- e.g., for N=8,

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

- I_h^{2h} has rank $\sim \frac{N}{2}$, and thus dim(NS(R)) $\sim \frac{N}{2}$

Prolongation and restriction are often nicely related

- For the 1D examples, linear interpolation and full-weighting are related by:

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

- A commonly used, and highly useful, requirement is that

$$\boxed{I_{2h}^{h} = c\,\left(I_{h}^{2h}\right)^{T}} \quad \text{for } c \text{ in } \Re$$

2D Prolongation

$$v^h_{2i,2j} = v^{2h}_{ij}$$

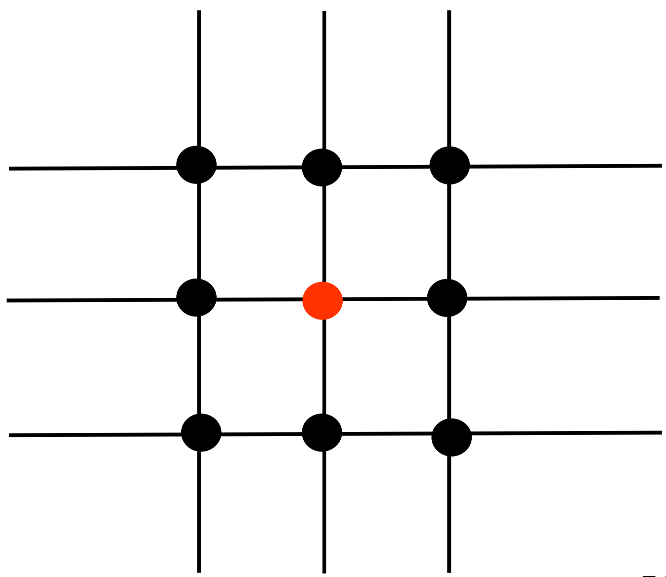
$$v^h_{2i+1,2j} = \frac{1}{2}\left(v^{2h}_{ij} + v^h_{i+1,j} \right)$$

$$v^h_{2i,2j+1} = \frac{1}{2}\left(v^{2h}_{ij} + v^h_{i,j+1} \right)$$

$$v^h_{2i+1,2j+1} = \frac{1}{4}\left(v^{2h}_{ij} + v^h_{i+1,j} + v^h_{i,j+1} + v^h_{i+1,j+1} \right)$$

$$\left] \begin{matrix} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & 1 & \frac{1}{2} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \end{matrix} \right[$$

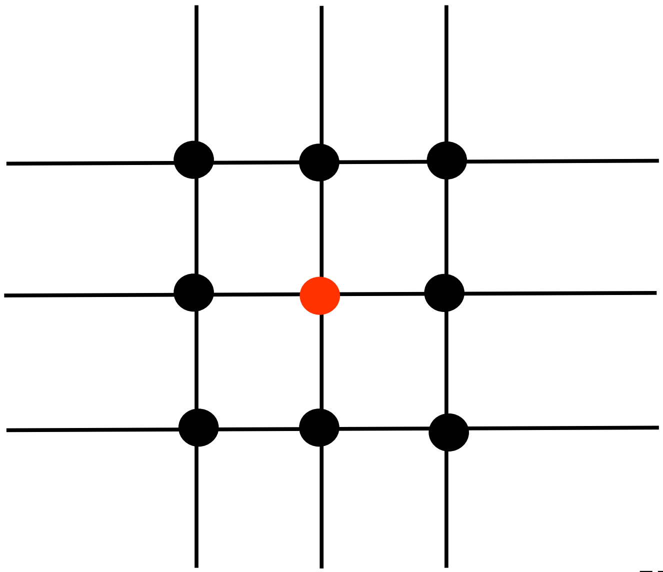
We denote the operator by using a “give to” stencil,] [. Centered over a c-point, ●, it shows what fraction of the c-point’s value is contributed to neighboring f-points, ●.



2D 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 "give to" stencil, [ ]. Centered over a c-point, ●, it shows what fractions of the neighboring (●) f-points' value is contributed to the value at the c-point.



Now, let's put all these ideas together

- Nested Iteration (effective on smooth error modes)

- Relaxation (effective on oscillatory error modes)

- Residual equation (i.e., residual correction)

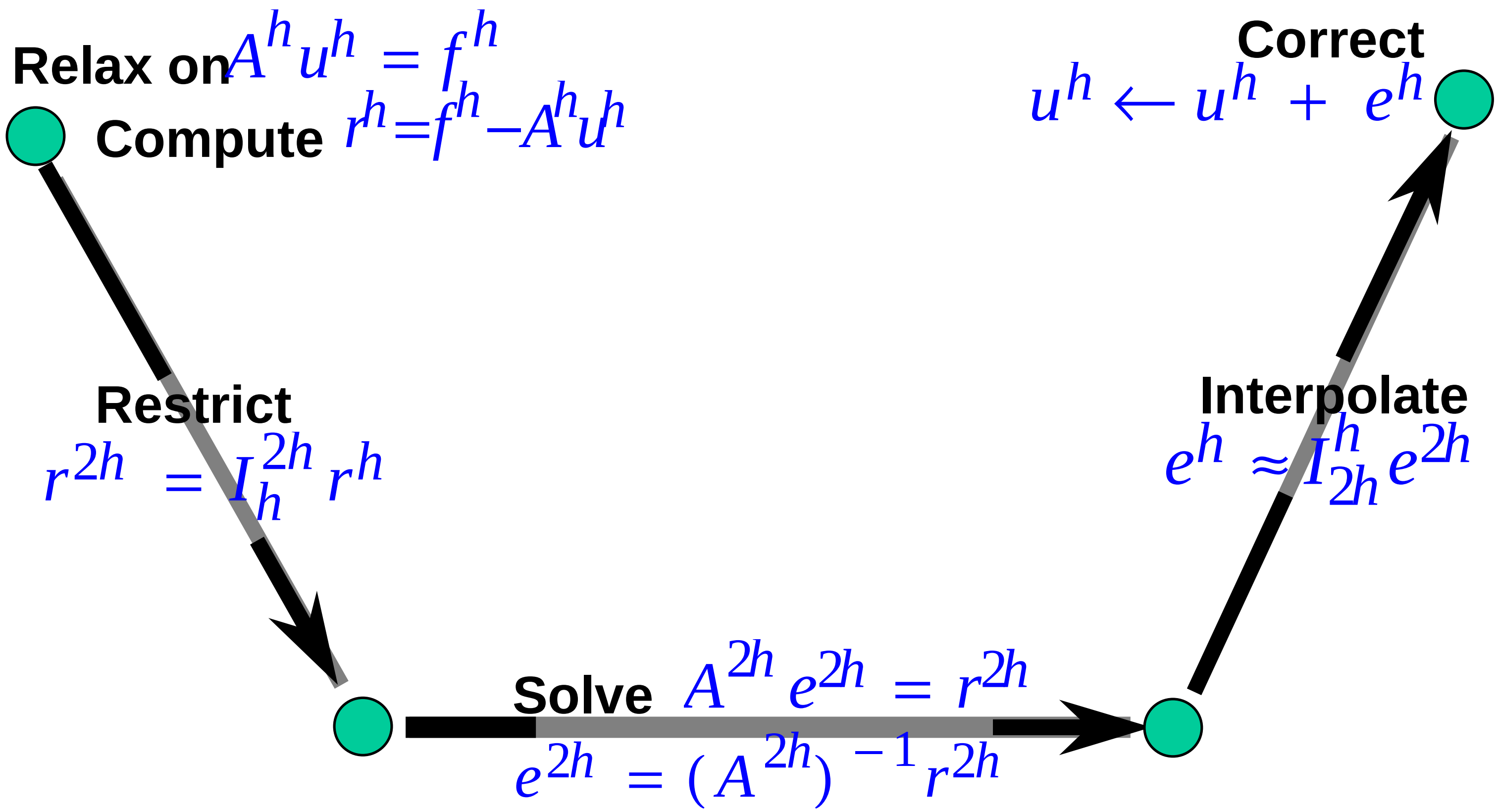
- Prolongation and Restriction

Coarse Grid Correction Scheme

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

- 1) Relax α_1 times on $A^h u^h = f^h$ on Ω^h with arbitrary initial guess v^h.
- 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 $A^h u^h = f^h$ on Ω^h with initial guess v^h.

Coarse-grid Correction



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.

- We will return to the question of constructing A^{2h} later.

How do we “solve” the coarse-grid residual equation? *Recursion!*

$$u^h \leftarrow G^{\nu}(A^h, f^h)$$

$$f^{2h} \leftarrow I_h^{2h}(f^h - A^h u^h)$$

$$u^{2h} \leftarrow G^{\nu}(A^{2h}, f^{2h})$$

$$f^{4h} \leftarrow I_{2h}^{4h}(f^{2h} - A^{2h} u^{2h})$$

$$u^{4h} \leftarrow G^{\nu}(A^{4h}, f^{4h})$$

$$f^{8h} \leftarrow I_{4h}^{8h}(f^{4h} - A^{4h} u^{4h})$$

$$u^{8h} \leftarrow G^{\nu}(A^{8h}, f^{8h})$$

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

$$u^{8h} \leftarrow u^{8h} + e^{8h}$$

$$e^{4h} \leftarrow I_{8h}^{4h} u^{8h}$$

$$u^{4h} \leftarrow u^{4h} + e^{4h}$$

$$e^{2h} \leftarrow I_{4h}^{2h} u^{4h}$$

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

$$e^h \leftarrow I_{2h}^h u^{2h}$$

$$u^h \leftarrow u^h + e^h$$

V-cycle (recursive form)

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

1) Relax α_1 times on $A^h u^h = f^h$, initial v^h arbitrary

2) If Ω^h is the coarsest grid, go to 4)

Else:

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

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

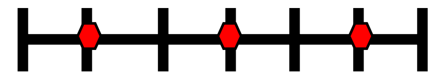
$$v^{2h} \leftarrow MV^{2h}(v^{2h}, f^{2h})$$

3) Correct $v^h \leftarrow v^h + I_{2h}^h v^{2h}$

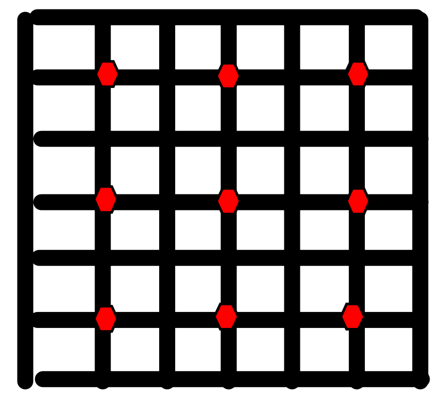
4) Relax α_2 times on $A^h u^h = f^h$, initial guess v^h

Storage Costs: v^h and f^h must be stored on each level

- In 1-d, each coarse grid has about half the number of points as the finer grid.



- In 2-d, each coarse grid has about one-fourth the number of points as the finer grid.



- In d-dimensions, each coarse grid has about 2^{-d} the number of points as the finer grid.
- Total storage cost: $2N^d\left(1+2^{-d}+2^{-2d}+2^{-3d}+\cdots+2^{-Md}\right) < \dfrac{2N^d}{1-2^{-d}}$ less than 2, 4/3, 8/7 the cost of storage on the fine grid for 1, 2, and 3-d problems, respectively.

Computation Costs

- Let 1 Work Unit (WU) be the cost of one relaxation sweep on the fine-grid.
- Ignore the cost of restriction and interpolation (typically about 20% of the total cost).
- Consider a V-cycle with 1 pre-Coarse-Grid correction relaxation sweep and 1 post-Coarse-Grid correction relaxation sweep.
- Cost of V-cycle (in WU):

$$2\left(1+2^{-d}+2^{-2d}+2^{-3d}+\cdots+2^{-Md}\right) < \frac{2}{1-2^{-d}}$$

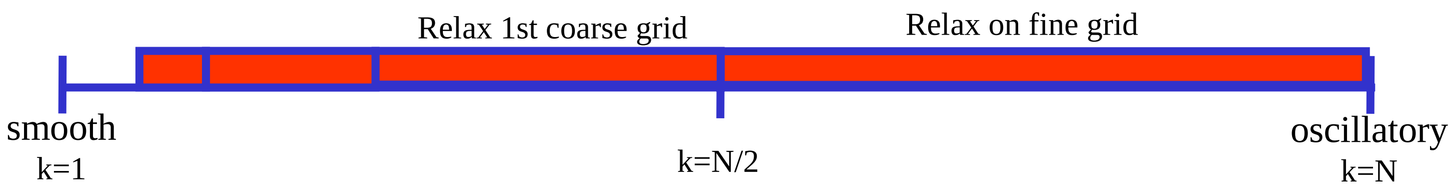
- Cost is about 4, 8/3, 16/7 WU per V-cycle in 1, 2, and 3 dimensions.

Convergence Analysis

- First, a heuristic argument:
  - The convergence factor for the oscillatory modes of the error (e.g., the smoothing factor) is small and independent of the grid spacing h.

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

  - The multigrid cycling schemes focus the relaxation on the oscillatory components on each level.



The overall convergence factor for multigrid methods is small and independent of h!

Convergence analysis, a little more precisely...

- Continuous problem: $Au = f\,, \quad u_i = u(x_i)$
- Discrete problem: $A^h u^h = f^h\,, \quad v^h \approx u^h$

- Global error: $E_i = u(x_i) - u_i^h$

  $\| E \| \leq K h^p$ (p=2 for model problem)

- Algebraic error: $e_i = u_i^h - v_i^h$

- Suppose a tolerance ε is specified such that v^h must satisfy $\| u - v^h \| \leq \varepsilon$

- Then this is guaranteed if

$$\| u - u^h \| + \| u^h - v^h \| = \| E \| + \| e \| \leq \varepsilon$$

We can satisfy the requirement by imposing two conditions

1) $\| E \| \leq \frac{\varepsilon}{2}$. We use this requirement to determine a grid spacing h^* from

$$h^* \leq \left(\frac{\varepsilon}{2K} \right)^{1/p}$$

2) $\| e \| \leq \frac{\varepsilon}{2}$, which determines the number of iterations required.

- If we iterate until $\| e \| \leq \frac{\varepsilon}{2} = K(h^*)^p$ on Ω^{h^*} then we have ***converged to the level of truncation.***

Converging to the level of truncation

- Use a MV scheme with convergence rate $\gamma < 1$ independent of h (fixed α_1 and α_2).

- Assume a d-dimensional problem on an NxNx...xN grid with $h = N^{-1}$.

- The V-cycle must reduce the error from $\| e \| \sim O(1)$ to

$$\| e \| \sim O(h^p) \sim O(N^{-p})$$

- We can determine θ, the number of V-cycles required to accomplish this.

Work needed to converge to the level of truncation

- Since θ V-cycles at convergence rate γ are required, we see that

$$\gamma^{\theta} \sim O(N^{-p})$$

implying that $\theta \sim O(\log N)$.

- Since one V-cycle costs O(1) WU and one WU is $O(N^d)$, we see that the cost of converging to the level of truncation using the MV method is

$$O(N^d \log N)$$

- which is comparable to fast direct methods (FFT based).

A numerical example

- Consider the two-dimensional model problem (with σ=0), given by

$$-u_{xx} - u_{yy} = 2\left[(1-6x^2)\, y^2 (1-y^2) + (1-6y^2)\, x^2 (1-x^2) \right]$$

inside the unit square, with u=0 on the boundary.

- The solution to this problem is

$$u(x,y) = (x^2 - x^4)\,(y^4 - y^2)$$

- We examine the effectiveness of MV cycling to solve this problem on NxN grids [(N-1) x (N-1) interior] for N=16, 32, 64, 128.

Numerical Results, MV cycling

	$n = 16$				$n = 32$			
V-cycle	$\|\mathbf{r}^h\|_h$	ratio	$\|\mathbf{e}\|_h$	ratio	$\|\mathbf{r}^h\|_h$	ratio	$\|\mathbf{e}\|_h$	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	1.03e−04	0.98*	9.79e−04	0.05	2.59e−05	0.59
5	7.45e−06	0.05	1.03e−04	1.00*	5.20e−05	0.05	2.58e−05	1.00*
6	3.75e−07	0.05	1.03e−04	1.00*	2.96e−06	0.06	2.58e−05	1.00*
7	2.08e−08	0.06	1.03e−04	1.00*	1.77e−07	0.06	2.58e−05	1.00*
8	1.24e−09	0.06	1.03e−04	1.00*	1.10e−08	0.06	2.58e−05	1.00*
9	7.74e−11	0.06	1.03e−04	1.00*	7.16e−10	0.06	2.58e−05	1.00*
10	4.99e−12	0.06	1.03e−04	1.00*	4.79e−11	0.07	2.58e−05	1.00*
11	3.27e−13	0.07	1.03e−04	1.00*	3.29e−12	0.07	2.58e−05	1.00*
12	2.18e−14	0.07	1.03e−04	1.00*	2.31e−13	0.07	2.58e−05	1.00*
13	2.33e−15	0.11	1.03e−04	1.00*	1.80e−14	0.08	2.58e−05	1.00*
14	1.04e−15	0.45	1.03e−04	1.00*	6.47e−15	0.36	2.58e−05	1.00*
15	6.61e−16	0.63	1.03e−04	1.00*	5.11e−15	0.79	2.58e−05	1.00*

	$n = 64$				$n = 128$			
V-cycle	$\|\mathbf{r}^h\|_h$	ratio	$\|\mathbf{e}\|_h$	ratio	$\|\mathbf{r}^h\|_h$	ratio	$\|\mathbf{e}\|_h$	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	6.45e−06	0.91*	8.99e−04	0.05	1.63e−06	0.50
6	1.39e−05	0.06	6.44e−06	1.00*	5.29e−05	0.06	1.61e−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	1.08e−13	0.07	6.44e−06	1.00*	3.64e−13	0.08	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*

Shown are the results of 16 V-cycles. We display, at the end of each cycle, the residual norm, the error norm, and the ratio of these norms to their values at the end of the previous cycle.

N=16, 32, 64, 128

Look again at nested iteration

- *Idea:* It's cheaper to solve a problem (i.e., takes fewer iterations) if the initial guess is good.

- How to get a good initial guess:
  - Interpolate coarse solution to fine grid.
  - "Solve" the problem on the coarse grid first.
  - Use interpolated coarse solution as initial guess on fine grid.

- Now, let's use the V-cycle as the solver on each grid level! This defines the Full Multigrid (FMG) cycle.

The Full Multigrid (FMG) cycle

$$v^h \leftarrow FMG(f^h)$$

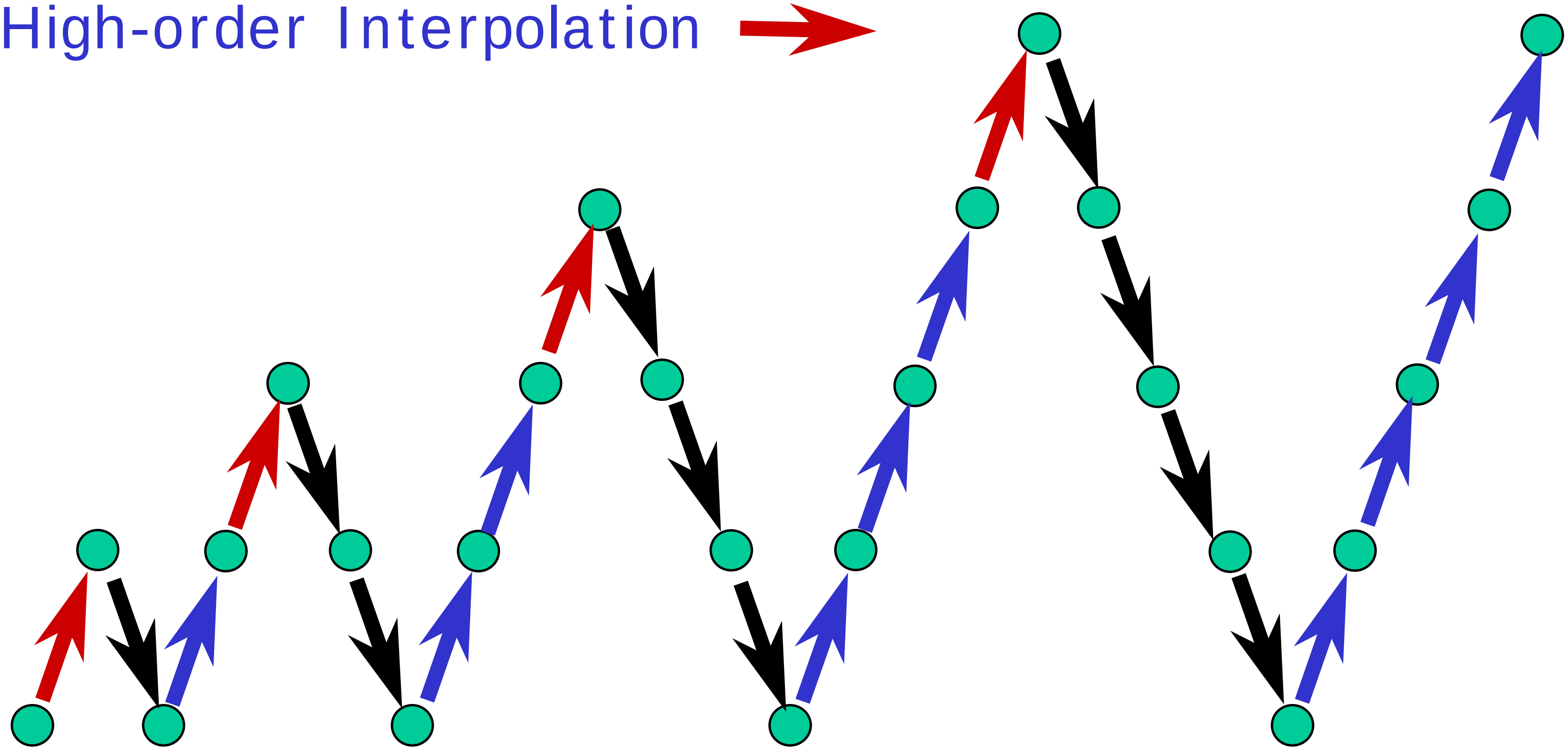
- Initialize $f^h, f^{2h}, f^{4h}, \ldots, f^H$
- Solve on coarsest grid $v^H = (A^H)^{-1} f^H$

$$\vdots$$

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

Full Multigrid (FMG)

- Restriction
- Interpolation
- High-order Interpolation



FMG-cycle (recursive form)

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

1) Initialize $f^h, f^{2h}, f^{4h}, \ldots, f^H$

2) If Ω^h is the coarsest grid, then solve exactly

Else: $v^{2h} \leftarrow FMG(f^{2h})$

3) Set initial guess $v^h \leftarrow I_{2h}^h v^{2h}$

4) $v^h \leftarrow MV(v^h, f^h)$, η times

Cost of an FMG cycle

- One V-cycle is performed per level, at a cost of $\left[\frac{1}{1-2^{-d}}\right]$ WU per grid (where the WU is for the size grid involved).
- The size of the WU for coarse-grid j is 2^{-jd} times the size of the WU on the fine grid (grid 0).
- Hence, cost of the FMG(1,1) cycle is less than

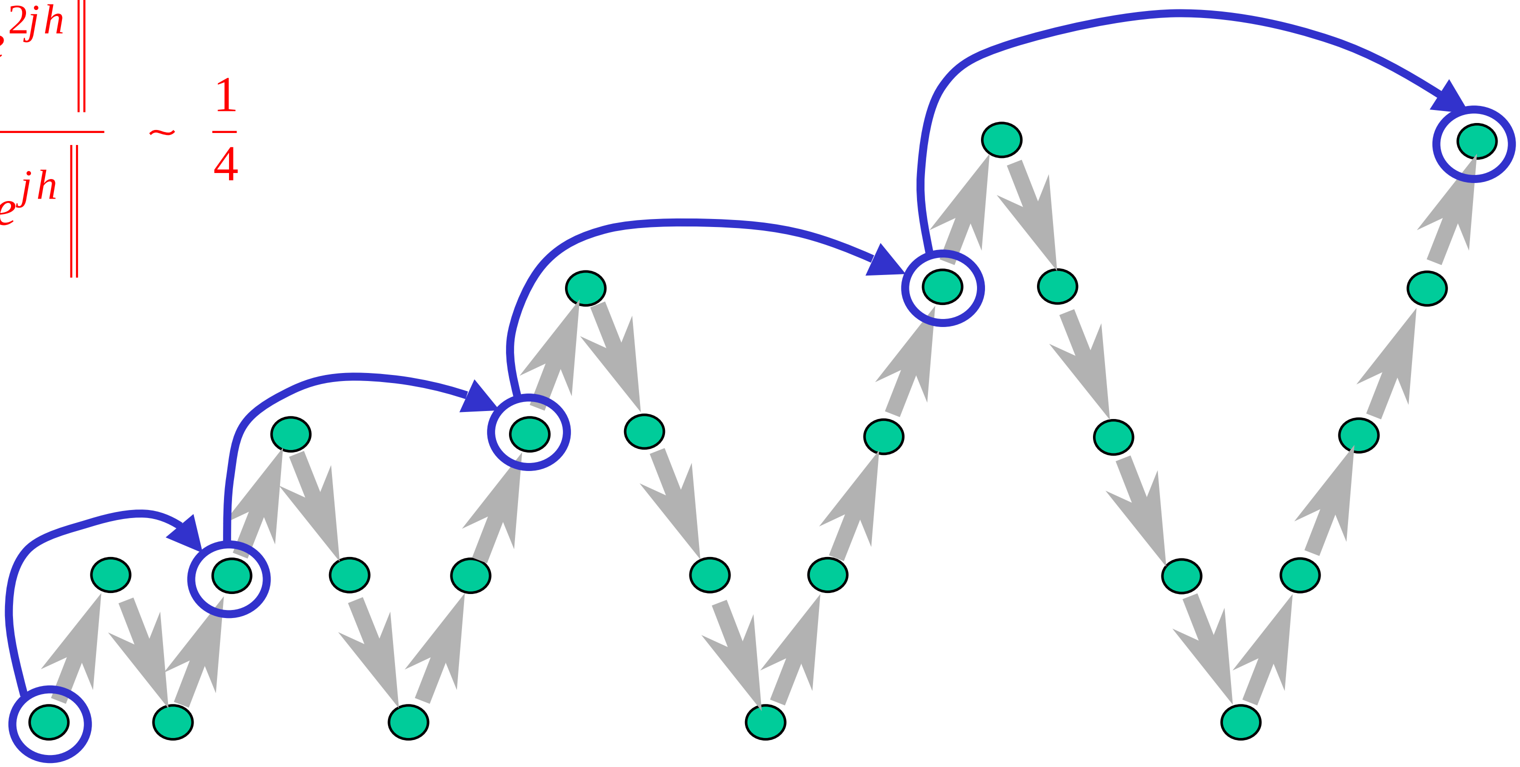
$$\left(\frac{2}{1-2^{-d}}\right)(1+2^{-d}+2^{-2d}+\dots) = \frac{2}{(1-2^{-d})^2}$$

- d=1: 8 WU; d=2: 7/2 WU d=3: 5/2 WU

How to tell if truncation error is reached with Full Multigrid (FMG)

- If truncation error is reached, $\|e\| \sim O(h^2)$ for each grid level h. The norms of the errors at the "solution" points in the cycle should form a Cauchy sequence and

$$\frac{\left\| e^{2jh} \right\|}{\left\| e^{jh} \right\|} \sim \frac{1}{4}$$



Cost to achieve convergence to truncation by the FMV method

- Consider using the FMV method, which solves the problem on Ω^{2h} to the level of truncation before going to Ω^{h}, i.e.,

$$\| e^{2h} \| = \| u^{2h} - v^{2h} \| \sim K(2h)^{p}$$

- We ask that $\| e^{h} \| \sim Kh^{p} = 2^{-p} \| e^{2h} \|$ which implies that the algebraic error must be reduced by 2^{-p} on Ω^{h}. Hence, θ_1 V-cycles are needed, where

$$\gamma^{\theta_1} \sim 2^{-p}$$

thus $\theta_1 \sim O(1)$ and computational cost of the FMV method $O(N^{d})$.

A numerical example

- Consider again the two-dimensional model problem (with σ=0), given by

$$-u_{xx} - u_{yy} = 2\left[(1-6x^2)\, y^2 (1-y^2) + (1-6y^2)\, x^2 (1-x^2) \right]$$

inside the unit square, with u=0 on the boundary.

- We examine the effectiveness of FMG cycling to solve this problem on NxN grids [(N-1) x (N-1) interior] for N=16, 32, 64, 128.

FMG results

- Shown are results for three FMG cycles, and a comparison to MV cycle results.

N	FMG(1,0) $\|\mathbf{e}\|_h$	FMG(1,0) ratio	FMG(1,1) $\|\mathbf{e}\|_h$	FMG(1,1) ratio	FMG(2,1) $\|\mathbf{e}\|_h$	FMG(2,1) 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	0.277	1.72e−04	0.257	7/2	4	16
32	4.70e−04	0.395	6.00e−05	0.238	4.00e−05	0.233	7/2	5	20
64	1.77e−04	0.377	1.36e−05	0.227	9.36e−06	0.234	7/2	5	20
128	6.49e−05	0.366	3.12e−06	0.229	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	3.44e−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

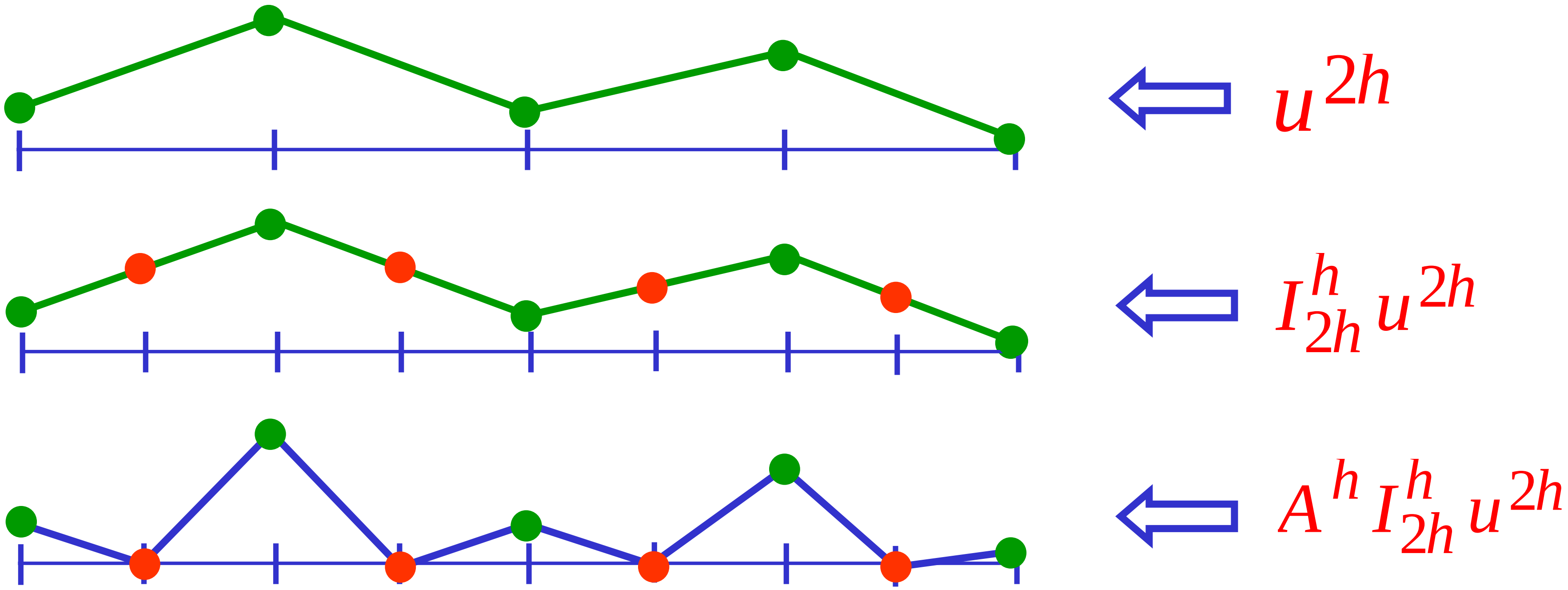
What is A^{2h} ?

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

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

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

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



- Therefore, the odd rows of $A^h I_{2h}^h$ are zero (in 1D only) and $r_{2i+1} = 0$. We therefore keep the even rows of $A^h I_{2h}^h$ for the residual equations on Ω^{2h} . These rows can be picked out by applying restriction!

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

Building A^{2h}.

- The residual equation on the coarse grid is:

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

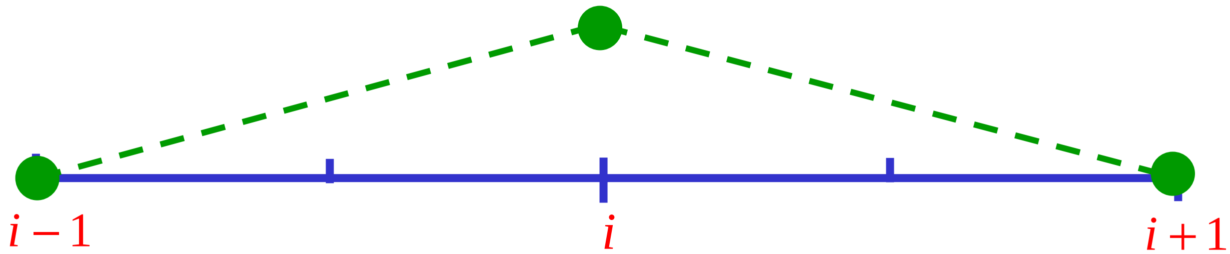
- Therefore, we identify the coarse-grid operator A^{2h} as

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

- Next, we determine how to compute the entries of the coarse-grid matrix.

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

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



0		1		0	$\hat{e}_i^{2h}$
0	$\frac{1}{2}$	1	$\frac{1}{2}$	0	$I_{2h}^{h}\,\hat{e}_i^{2h}$
$-\frac{1}{2h^2}$	0	$\frac{1}{h^2}$	0	$-\frac{1}{2h^2}$	$A^{h}\,I_{2h}^{h}\,\hat{e}_i^{2h}$
$-\frac{1}{4h^2}$		$\frac{1}{2h^2}$		$-\frac{1}{4h^2}$	$I_{h}^{2h}\,A^{h}\,I_{2h}^{h}\,\hat{e}_i^{2h}$

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

- The i^{th} row of A^{2h} is $\frac{1}{(2h)^2}\begin{bmatrix} -1 & 2 & -1 \end{bmatrix}$,

  which is the Ω^{2h} version of A^h.

- Therefore, ***IF*** relaxation on Ω^h leaves only error in the range of interpolation, then solving

  determines the error exactly!

- In general, this will not be the case, but this argument certainly leads to a plausible representation for A^{2h}.

The variational properties

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

$$\boxed{\begin{aligned} A^{2h} &= I_h^{2h} A^h I_{2h}^h \\ I_{2h}^h &= c\,(I_h^{2h})^T \end{aligned}}$$

(Galerkin Condition)

for c in $\Re$

Properties of the Grid Transfer Operators: Restriction

- Full Weighting: $I_h^{2h} : \Omega^h \to \Omega^{2h}$ or
  $I_h^{2h} : \Re^{N-1} \longrightarrow \Re^{N/2-1}$

- For N=8,

$$I_h^{2h} = \frac{1}{4}\begin{bmatrix} 1 & 2 & 1 & & & & \\ & & 1 & 2 & 1 & & \\ & & & & 1 & 2 & 1 \end{bmatrix}_{3\times 7}$$

- I_h^{2h} has rank $\frac{N}{2} - 1$ and a null space $N(I_h^{2h})$ with dimension $\frac{N}{2}$.

Spectral properties of I_h^{2h} .

- How does I_h^{2h} act on the eigenvectors of A^h ?
- Consider $w_{k,j}^h = \sin\left(\frac{jk\pi}{N}\right)$, $1 \leq k \leq N-1$, $0 \leq j \leq N$
- Good Exercise: show that

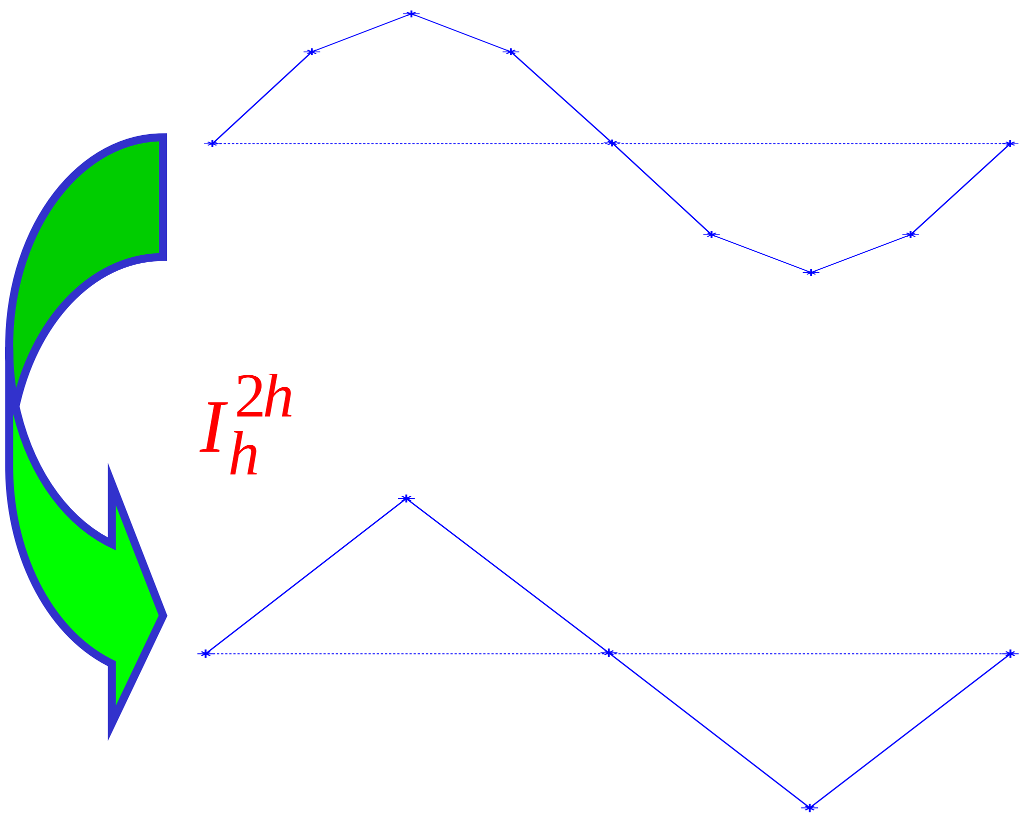
$$(I_h^{2h} w_k^h)_j = \cos^2\left(\frac{k\pi}{2N}\right) w_{k,j}^{2h}$$

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

for $1 \leq k \leq N/2$.

Spectral properties of I_h^{2h} (cont.).

- i.e., I_h^{2h} [kth mode on Ω^h] = c_k [kth mode on Ω^{2h}]



Ω^h : N=8, k=2

Ω^{2h} : N=4, k=2

Spectral properties of I_h^{2h} (cont.).

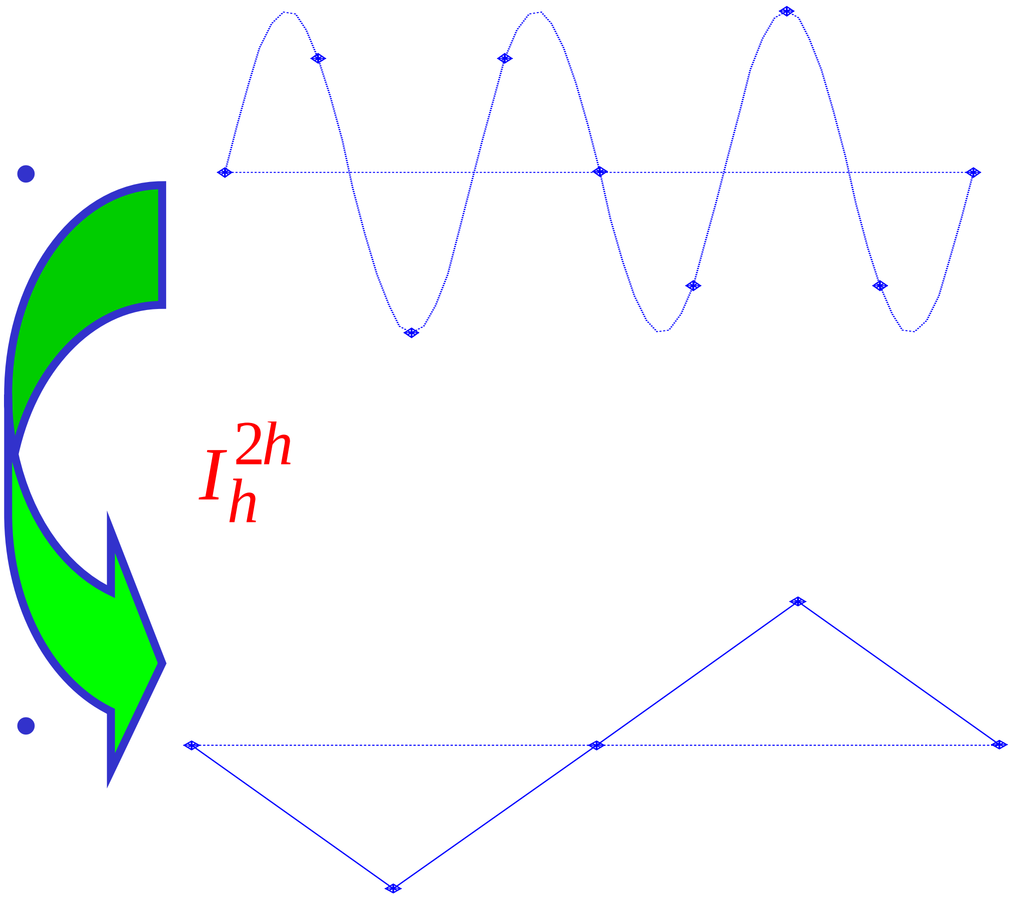
- Let $k' = N - k$ for $1 \leq k < \frac{N}{2}$, so that $\frac{N}{2} < k' < N$
- Then (another good exercise!)

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

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

Spectral properties of I_h^{2h} (cont.).

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



Ω^h : N=8, k=6

Ω^{2h} : N=4, k=2

Spectral properties of I_h^{2h} (cont.).

- Summarizing:

$$\left.\begin{aligned} I_h^{2h} w_k^h &= c_k\, w_k^{2h} \\ I_h^{2h} w_{k'}^h &= -s_k\, w_k^{2h} \end{aligned}\right\} \quad \left\{\begin{aligned} &1 < k < \frac{N}{2} \\ &k' = N - k \end{aligned}\right.$$

$$I_h^{2h} w_{N/2}^h = 0$$

- Complementary modes:

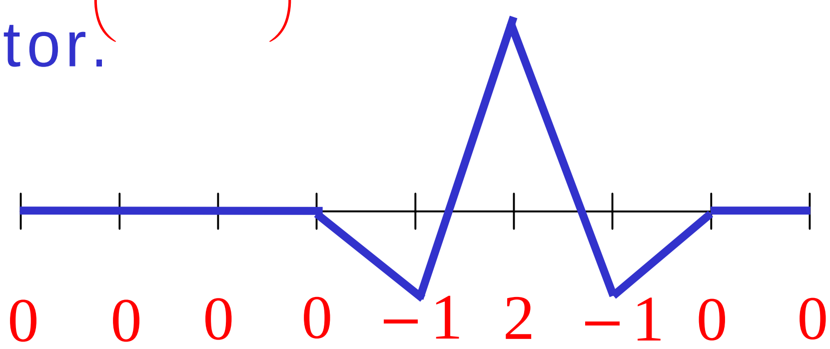
$$W_k = \operatorname{span}\{w_k^h, w_{k'}^h\}$$

$$I_h^{2h} W_k \to w_k^{2h}$$

Null Space of I_h^{2h}.

- Observe that $N(I_h^{2h}) = \text{span}\left(A^h \hat{e}_i^h \right)$ where i is odd and $\hat{e}_i^h$ is the i[th] unit vector.

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



- While the η_i look oscillatory, they contain <u>all</u> of the Fourier modes of A^h, i.e.,

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

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

Properties of the Grid Transfer Operators: Interpolation

- Interpolation: $I_{2h}^{h} : \Omega^{2h} \rightarrow \Omega^{h}$ or
  $I_{2h}^{h} : \Re^{N/2-1} \longrightarrow \Re^{N-1}$
- For N=8,

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

- I_{2h}^{h} has full rank and null space $\{\phi\}$.

Spectral properties of I_{2h}^{h}.

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

$$I_{2h}^{h} w_k^{2h} = \cos^2\left(\frac{k\pi}{2N}\right) w_k^h - \sin^2\left(\frac{k\pi}{2N}\right) w_{k'}^h$$

$$= c_k\, w_k^h - s_k\, w_{k'}^h$$

Spectral properties of I_{2h}^{h} (cont.).

$$\boxed{I_{2h}^{h} = c_k\, w_k^h - s_k\, w_{k'}^h}$$

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

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

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

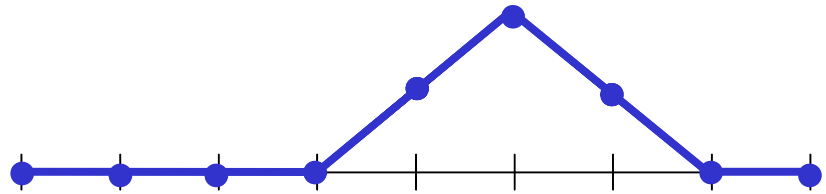
$$\approx w_k^h$$

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

The Range of I_{2h}^{h} .

- 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} .

ξ_i :



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

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

Use all the facts to analyze the coarse-grid correction scheme

1) Relax α times on Ω^h: $v^h \leftarrow R^{\alpha_1} v^h$

2) Compute and restrict residual $f^{2h} \leftarrow I_h^{2h}(f^h - A^h v^h)$

3) Solve residual equation $v^{2h} = (A^{2h})^{-1} f^{2h}$

4) Correct fine-grid solution $v^h \leftarrow v^h + I_{2h}^h v^{2h}$.

- The entire process appears as

$$v^h \leftarrow R^{\alpha} v^h + I_{2h}^h (A^{2h})^{-1} I_h^{2h} (f^h - A^h R^{\alpha} v^h)$$

- The exact solution satisfies

$$u^h \leftarrow R^{\alpha} u^h + I_{2h}^h (A^{2h})^{-1} I_h^{2h} (f^h - A^h R^{\alpha} u^h)$$

Error propagation of the coarse-grid correction scheme

- Subtracting the previous two expressions, we get

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

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

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

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

Error propagation of CG

- For now, assume no relaxation $\alpha = 0$. 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

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

CG error propagation for k << N

- Consider the case k << N (extremely smooth and 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!

Now consider CG with relaxation

- Next, include α relaxation sweeps. 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 << N/2,

$$w_k \rightarrow \lambda_k^\alpha s_k w_k + \lambda_k^\alpha s_k w_{k'}$$ Small!

$$w_{k'} \rightarrow \lambda_{k'}^\alpha c_k w_k + \lambda_{k'}^\alpha c_k w_{k'}$$ Small!

The crucial observation:

- Between relaxation and the coarse-grid correction, both smooth and oscillatory components of the error are effectively damped.

- This is essentially 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:

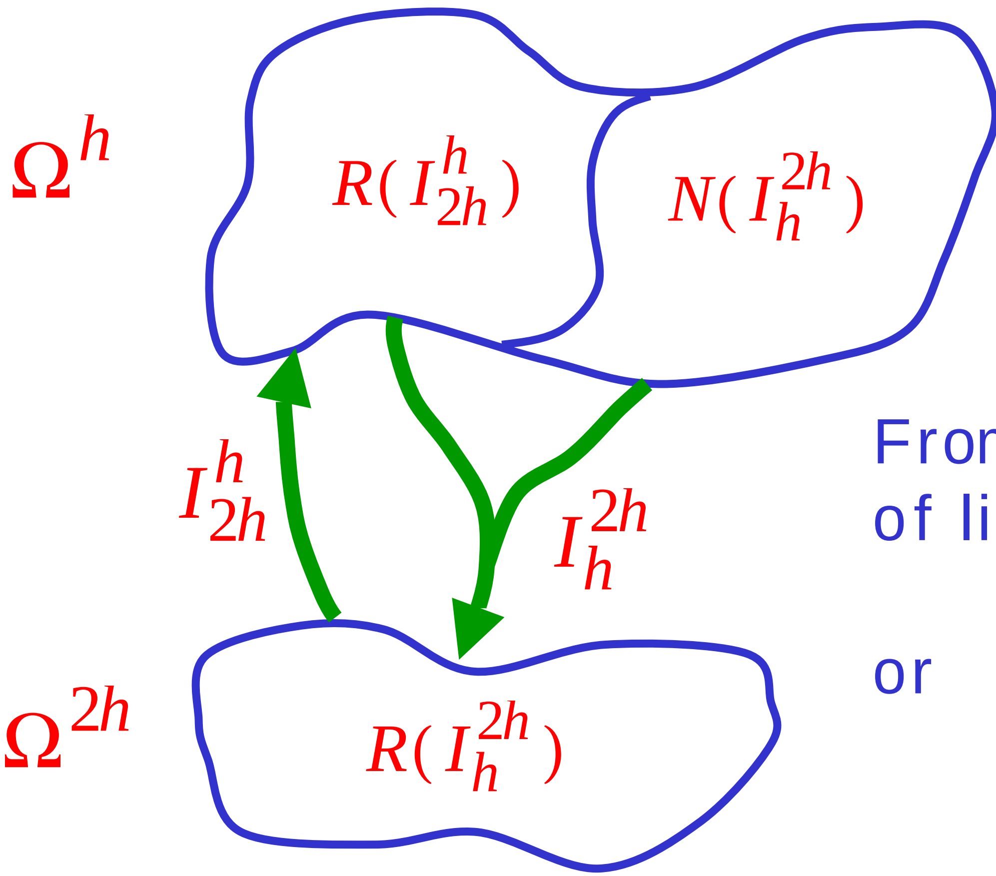
$$\boxed{\begin{aligned} A^{2h} &= I_h^{2h} A^h I_{2h}^h \\ I_{2h}^h &= c\,(I_h^{2h})^T \end{aligned}}$$

(Galerkin Condition)

for c in $\Re$

Algebraic interpretation of coarse-grid correction

- Consider the subspaces that make up Ω^h and Ω^{2h}



For the rest of this talk, ‘R( )’ refers to the Range of a linear operator.

From the fundamental theorem of linear algebra:

$$N(I_h^{2h}) \perp R(\,(I_h^{2h})^T\,)$$

or

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

Subspace decomposition of Ω^h.

- Since A^h has full rank, we can say, equivalently,

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

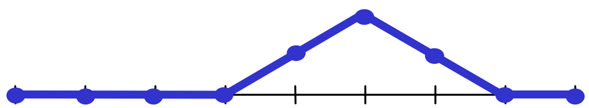
(where $x \perp_{A^h} y$ means that $\langle A^h x, y \rangle = 0$).

- Therefore, any e^h can be written as $e^h = s^h + t^h$ where $s^h \in R(I_{2h}^{h})$ and $t^h \in N(I_h^{2h} A^h)$.

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

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 $\eta_i = A^h \hat{e}_i$ therefore $N(I_h^{2h} A^h)$ is spanned by the unit vectors $\hat{e}_i^h = (0, 0, \ldots, 0, 1, 0, \ldots, 0)^T$ for odd i, which "look" oscillatory.

Algebraic analysis of coarse-grid correction

- 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 $q^{2h} \in \Omega^{2h}$
  and therefore

$$CG\, s^h = \left[I - I_{2h}^{h}(A^{2h})^{-1} \underbrace{I_{h}^{2h} A^{h} \Big] I_{2h}^{h}}_{A^{2h}} q^{2h} = 0$$

by Galerkin property

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

More algebraic analysis of coarse-grid correction

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

$$CG\, t^h = \left[I - I_{2h}^h (A^{2h})^{-1} \underbrace{I_h^{2h} A^h \Big] t^h}_{0}$$

- Therefore $CG\, t^h = t^h$.

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

How does the algebraic picture fit with the spectral view?

- We may view Ω^h in two ways:

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

that is,

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

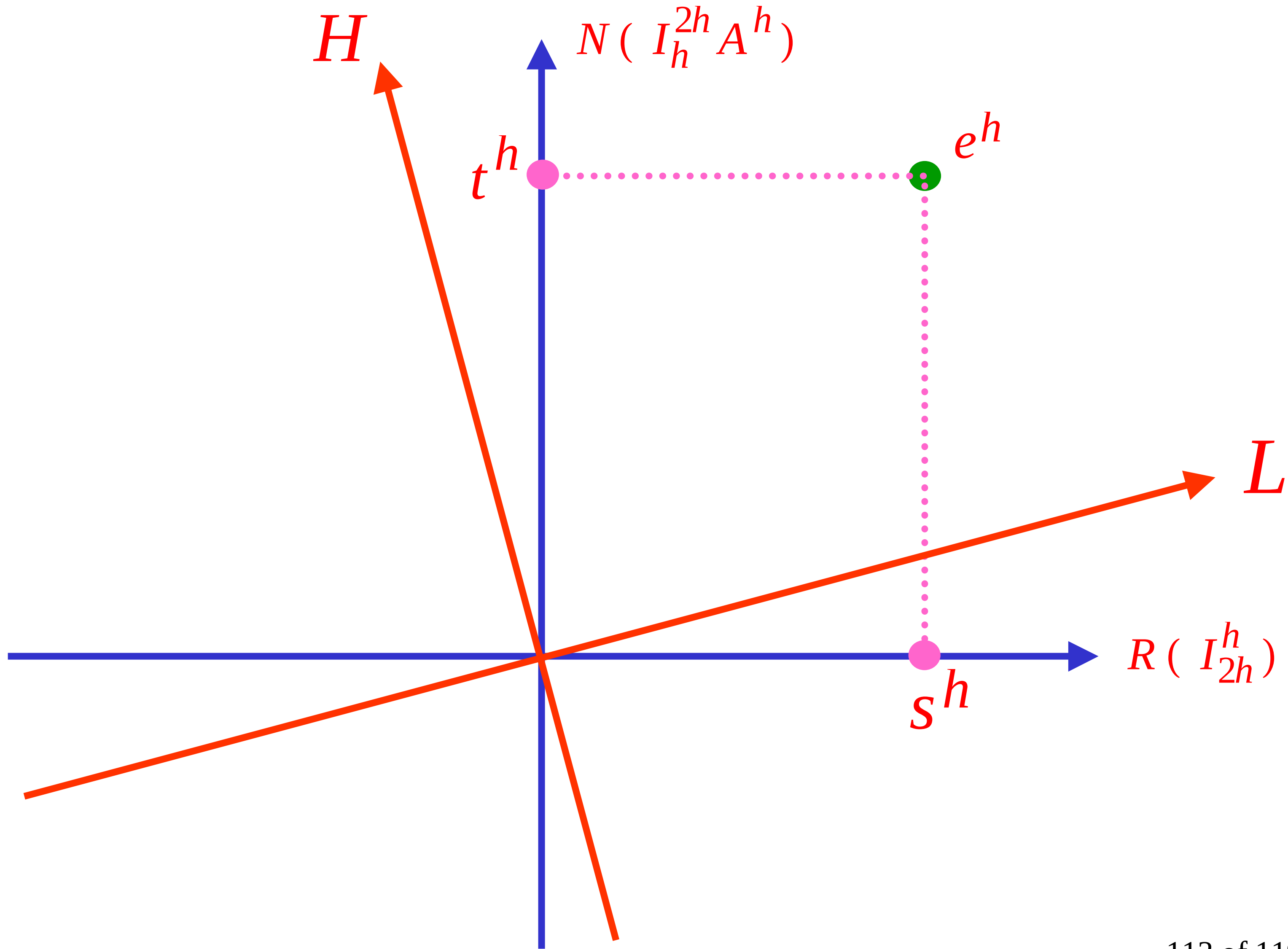
or as

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

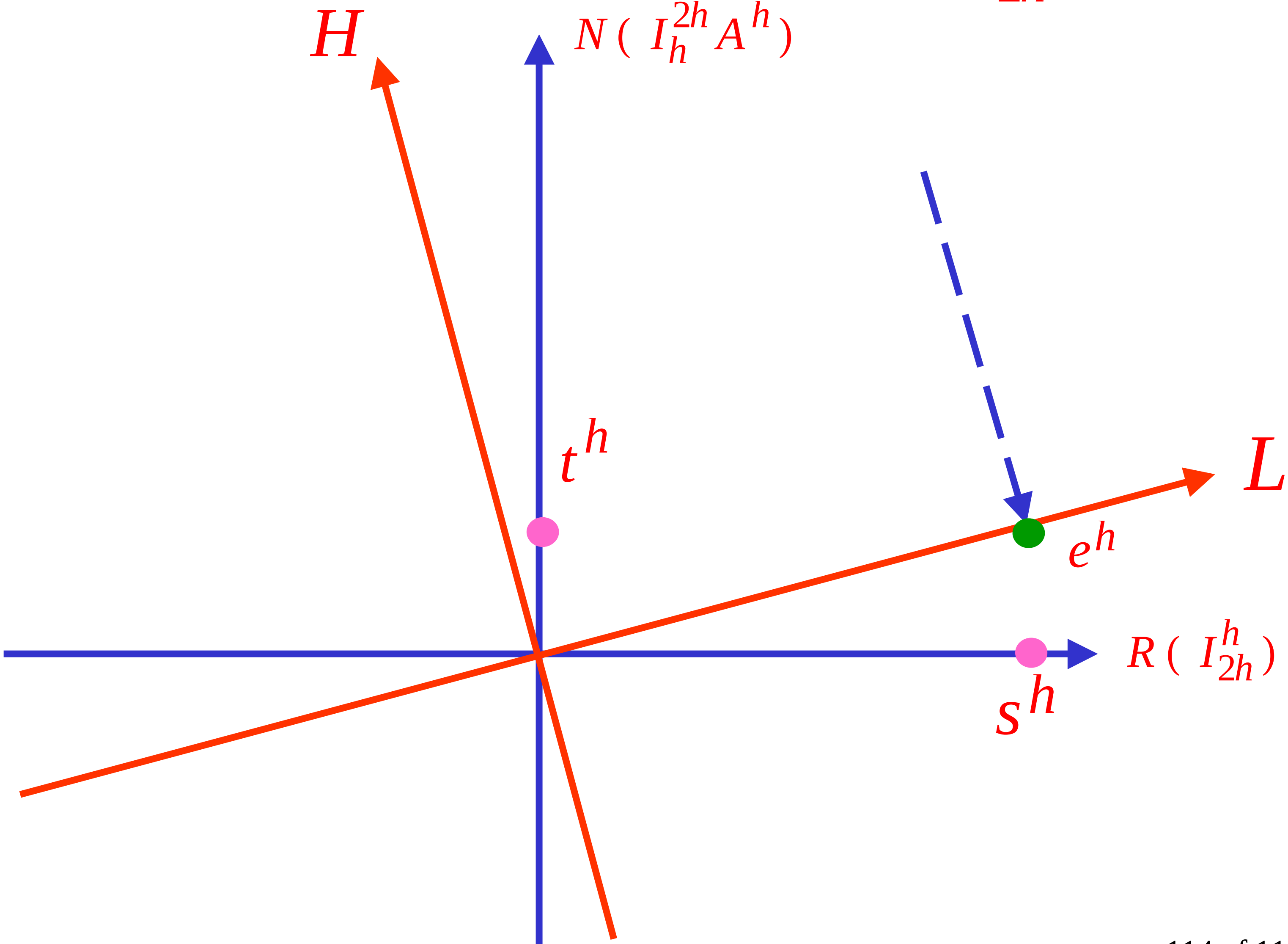
Actually, each view is just part of the picture

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

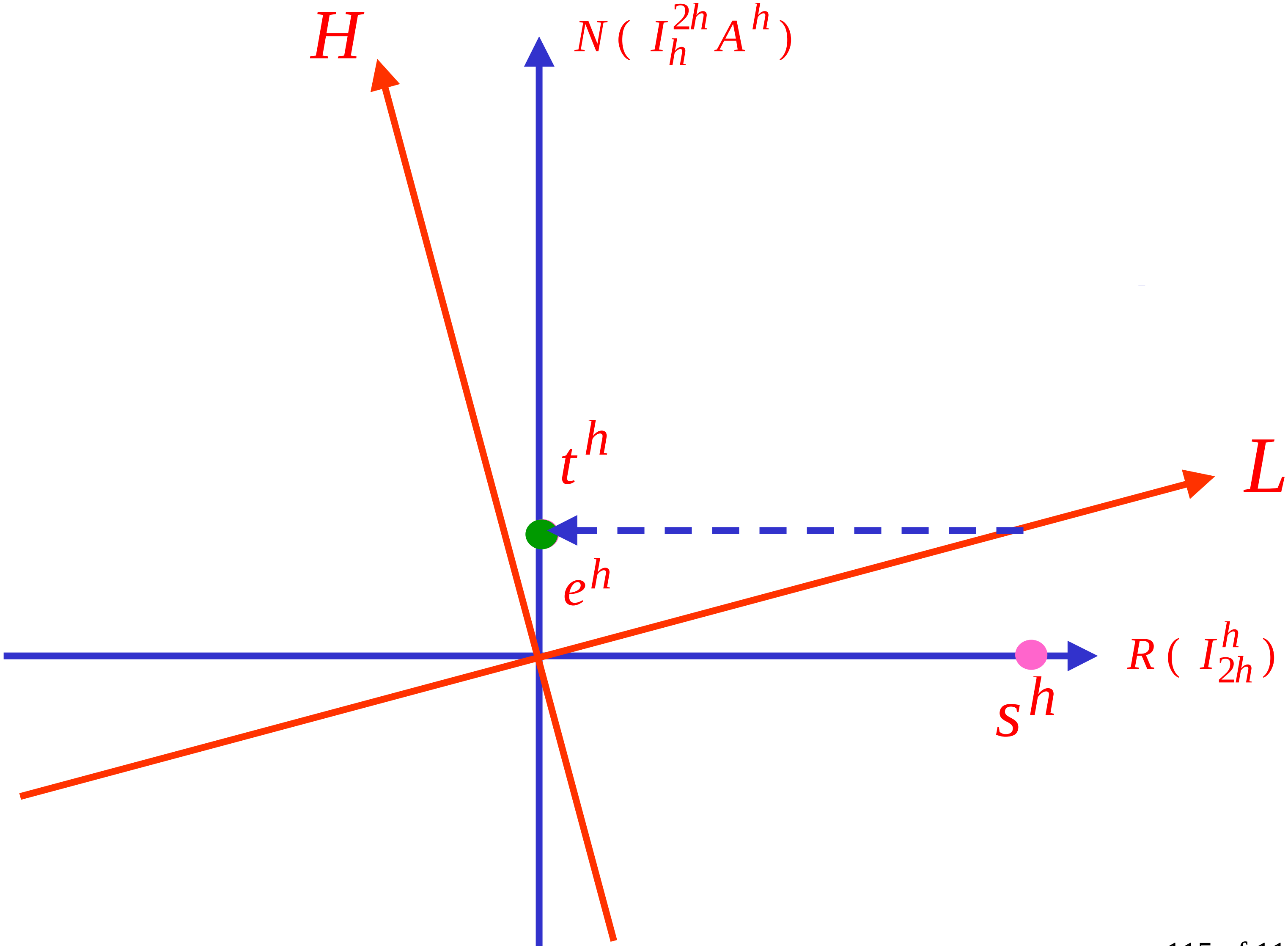
How it actually works (cartoon)



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



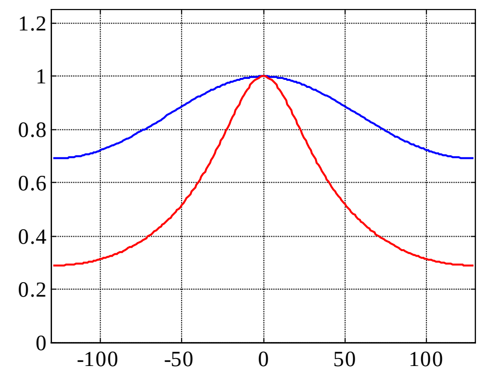
CG eliminates $R(I_{2h}^{h})$ but increases the error in H.



Difficulties: anisotropic operators and grids

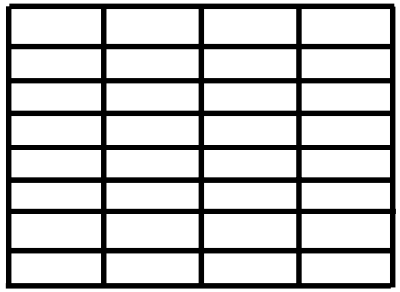
- Consider the operator : $-\alpha\frac{\partial^2 u}{\partial x^2} - \beta\frac{\partial^2 u}{\partial y^2} = f(x,y)$

- If $\alpha \ll \beta$ then the GS-smoothing factors in the *x*- and *y*-directions are shown at right.



Note that GS relaxation does not damp oscillatory components in the *x*-direction.

- The same phenomenon occurs for grids with much larger spacing in one direction than the other:



Difficulties: discontinuous or anisotropic coefficients

- Consider the operator : $-\nabla \bullet (D(x,y)\nabla u)$
  where
  $$D(x,y) = \begin{pmatrix} d_{11}(x,y) & d_{12}(x,y) \\ d_{21}(x,y) & d_{22}(x,y) \end{pmatrix}$$

- Again, GS-smoothing factors in the *x*- and *y*-directions can be highly variable, and very often, GS relaxation does not damp oscillatory components in the one or both directions.

- Solutions: line-relaxation (where whole gridlines of values are found simultaneously), and/or semi-coarsening (coarsening only in the strongly coupled direction).

For nonlinear problems, use FAS (Full Approximation Scheme)

- Suppose we wish to solve: $A(u) = f$ where the operator is non-linear. Then the linear residual equation $Ae = r$ does not apply.
- Instead, we write the non-linear residual equation:

$$A(u+e) - A(u) = r$$

- This is transferred to the coarse grid as:

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

- We solve for $w^{2h} \equiv u^{2h} + e^{2h}$ and transfer the error (only!) to the fine grid:

$$u^h \leftarrow u^h + I_{2h}^h(w^{2h} - I_h^{2h} u^h)$$

Multigrid: increasingly, the right tool!

- Multigrid has been proven on a wide variety of problems, especially elliptic PDEs, but has also found application among parabolic & hyperbolic PDEs, integral equations, evolution problems, geodesic problems, etc.

- *With the right setup, multigrid is frequently an optimal (i.e., O(N)) solver.*

- Multigrid is of great interest because it is one of the very few scalable algorithms, and can be parallelized readily and efficiently!