An Introduction of Multigrid Methods for Large-Scale Computation

Chin-Tien Wu

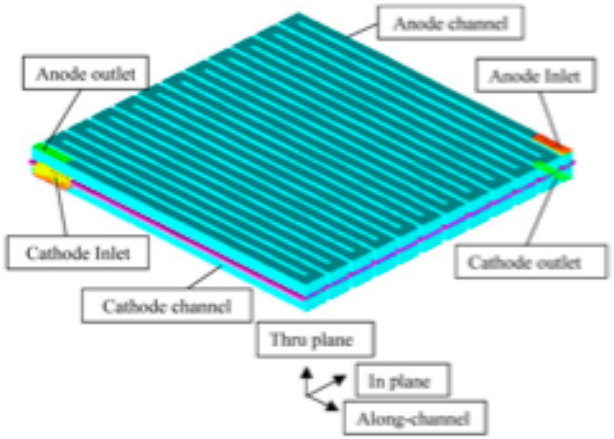
National Center for Theoretical Sciences

National Tsing-Hua University

01/24/2005

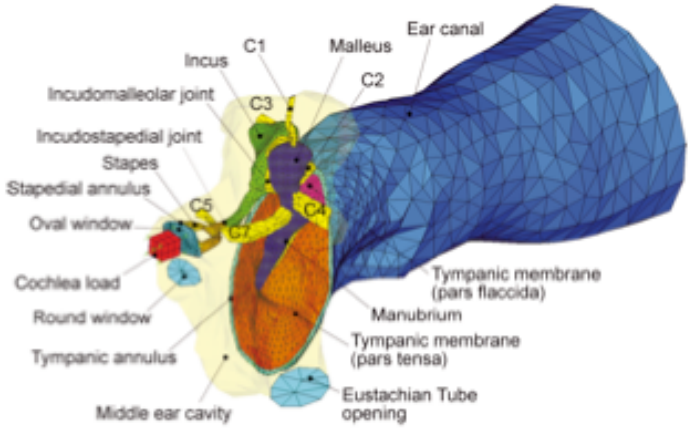
How Large the Real Simulations Are?

Large-scale Simulation of Polymer Electrolyte Fuel Cells by parallel Computing
(Hua Meng and Chao-Yang Wang, 2004)

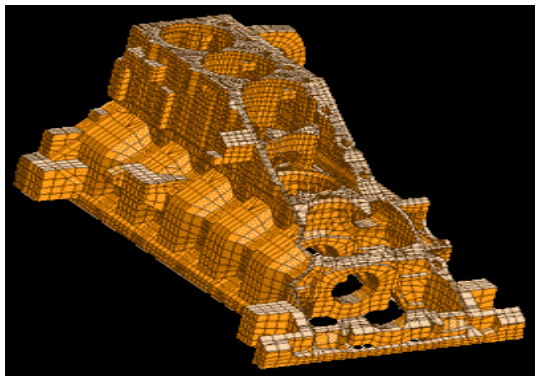


FEM model with $O(10^6)$ nodes

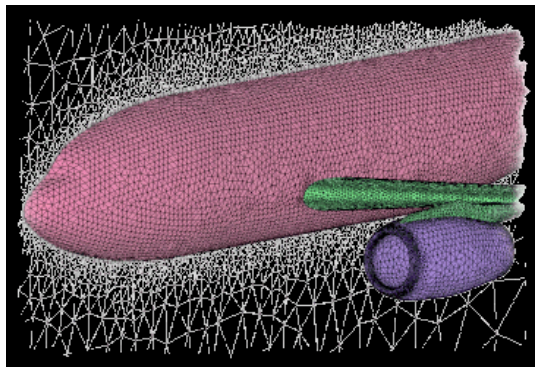
Three-Dimensional Finite Element Modeling of Human Ear for Sound Transmission
(R. Z. Gan, B. Feng and Q. Sun, 2004)



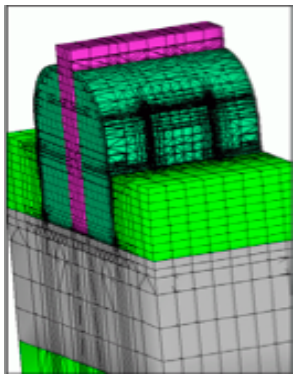
FEM model with $10^5 \sim 10^6$ nodes



Car engine with $O(10^5)$ nodes



Commercial Aircraft: 10^7 nodes



FINFET transistor: 10^5 nodes

What do we need in order to simulate?

- Deep understanding to physical problems
- Good mathematical models.
- Good computable mathematical models.
- Computation grids (not necessary but ...)
- Discretizations
- Solve linear systems
- Solve linear systems fast!

Our goal is to introduce multigrid methods for solving sparse linear systems.

Why multigrid?

1. Computation cost of multigrid is proportional to problem sizes.
2. Multigrid is “easy” to be parallelized.

Outlines

- Stationary Iterative Methods
- Some finite element error estimates
- Multigrid
- Algebraic Multigrid
- Nonlinear Multigrid (FAS)
- Multigrid Parallelization

Reference:

1. An introduction to multilevel methods (Jinchao Xu)
2. Multigrid Methods (Stephen F. McCormick)
3. A multigrid tutorial (William L. Briggs)
4. Matrix iterative analysis (Richard S. Varga)
5. The mathematical theory of finite element methods (Brenner and Scott)
6. Introduction to Algebraic Multigrid (Christian Wagner)

Solving Linear System Ax=b by Iterative Methods

Methods:

- Stationary Methods: Jacobi, Gauss Seidel (GS), SOR.
- Krylov Subspace Methods: Conjugate gradient, GMRES, by Saad and Schultz 1986, and MINRES, by Paige and Saunders 1975.
- Multigrid Methods: Geometric multigrid (MG), by Fedorenko 1961, and algebraic multigrid (AMG), by Ruge and Stüben 1985.

Basic questions and some definitions

Basic questions are

1. How do we iterate?
2. For what category of matrices A, the iteration converge?
3. What is the convergence rate?

Some definitions:

A is irreducible if there is no permutation P such that $P^T AP = \begin{bmatrix} A_{1,1} & A_{1,2} \\ 0 & A_{2,2} \end{bmatrix}$

A is non-negative (denoted as $A \geq 0$) if $a_{i,j} \geq 0$, for all $1 \leq i,j \leq n$

A is an M-matrix if A is nonsingular, $a_{i,j} \leq 0$ for $i \neq j$, and $A^{-1} \geq 0$

A is irreducibly diagonally dominant if A is irreducible,

diagonally dominant with $|a_{i,i}| > \sum_{j=1, j\neq i}^{n} a_{i,j}$ for some i.

A=M-N is a regular splitting of A if M is nonsingular and $M^{-1} \geq 0$

Stationary Iterative Methods

1. $r^{old} = f - Au^{old}$
2. Solve $e = B^{-1} r^{old}$
3. update $u^{new} = u^{old} + e$

$\Leftrightarrow$

$$e^{new} = e^{old} - B^{-1}\left(f - Au^{old}\right)$$
$$= e^{old} - B^{-1}A\left(u - u^{old}\right)$$
$$= \left(I - B^{-1}A\right)e^{old}$$

B is called an iterator or preconditioner of A.

$E_B = I - B^{-1}A$ is called the error reduction operator of the iterator B

Perron-Frobenius Theorem

Theorem: Let $A \geq 0$ be an irreducible matrix. Then

1. A has a positive real eigenvalue equal to its spetral radius
2. There is an eigenvector $x > 0$ corresponds to $\rho(A)$
3. $\rho(A)$ increases when any entry of A increases.
4. $\rho(A)$ is a simple eigenvalue of A.

Some Well Known Iterative Methods

Suppose $A = D - L - U$, where

D is the diagonal, L and U are lower and upper triangular parts, respectively.

Richardson: $B = \frac{1}{\omega}$, where $0<\omega<\frac{2}{\rho(\mathrm{A})}$.

Jacobi: $B = D$

Damped Jacobi: $B = \frac{1}{\omega} D$, where $0<\omega<\frac{2}{\rho(\mathrm{D}^{-1}\mathrm{A})}$.

Gauss-Seidel: $B = (D - L)$

SOR: $B = \frac{1}{\omega}(D - \omega L)$, where $0<\omega<2$.

Jacobi and Gauss-Seidel

Jacobi: $$x_i^{(m+1)} = -\sum_{\substack{j=1 \\ j\neq i}}^{n} \left(\frac{a_{i,j}}{a_{i,i}} \right) x_j^{(m)} + \frac{r_i}{a_{i,i}}$$

Gauss-Seidel: $$x_i^{(m+1)} = -\sum_{j=1}^{i-1} \left(\frac{a_{i,j}}{a_{i,i}} \right) x_j^{(m+1)} - \sum_{j=i+1}^{n} \left(\frac{a_{i,j}}{a_{i,i}} \right) x_j^{(m)} + \frac{r_i}{a_{i,i}}$$

HW1: Write down a formula for SOR

HW2:

Write a program to solve $$\begin{bmatrix} 1 & 0 & -\frac{1}{4} & -\frac{1}{4} \\ 0 & 1 & -\frac{1}{4} & -\frac{1}{4} \\ -\frac{1}{4} & -\frac{1}{4} & 1 & 0 \\ -\frac{1}{4} & -\frac{1}{4} & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = 0.5 \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$ by

Jacobi and Gauss-Seidel, starting with initial $x^{(0)} = [0,0,0,0]$.

Let $\mathrm{E_J} = \left(I - D^{-1}A\right)$ and $\mathrm{E_{GS}} = \left(I - (D-L)^{-1}A\right)$. Since the solution of HW2 is x=$[1,1,1,1]$ and $\mathrm{e}^0 = x - x^{(0)} = [1,1,1,1]$. Clearly, we have $\mathrm{e}_\mathrm{J}^\mathrm{m} = \left(\mathrm{E_J}\right)^\mathrm{m} \mathrm{e}^0$ and $\mathrm{e}_\mathrm{GS}^\mathrm{m} = \left(\mathrm{E_{GS}}\right)^\mathrm{m} \mathrm{e}^0$, One can easily check that

$$\mathrm{e}_\mathrm{J}^\mathrm{m} = \frac{-1}{2^m}\begin{bmatrix}1\\1\\1\\1\end{bmatrix} \text{ and } \mathrm{e}_\mathrm{GS}^\mathrm{m} = \frac{-1}{4^m}\begin{bmatrix}2\\2\\1\\1\end{bmatrix}. \text{ Thus, } \left\|\mathrm{e}_\mathrm{J}^\mathrm{m}\right\| = \frac{1}{2^{m-1}} > \left\|\mathrm{e}_\mathrm{GS}^\mathrm{m}\right\| = \frac{\sqrt{10}}{4^m}.$$

You might get a feeling that Gauss-Seidel method is faster than Jacobi method.

Stein-Rosenberg Theorem

Theorem: Let $\mathrm{B_J} = L + U$ be the Jacobi matrix and $\mathrm{B_{GS}} = \left(I - L\right)^{-1} U$ be the Gauss-Seidel matrix. Then one and only one of the following relations is vaild:

1) $\rho\left(\mathrm{B_J}\right) = \rho\left(\mathrm{B_{GS}}\right) = 0.$
2) $0 < \rho\left(\mathrm{B_{GS}}\right) < \rho\left(\mathrm{B_J}\right) < 1.$
3) $\rho\left(\mathrm{B_J}\right) = \rho\left(\mathrm{B_{GS}}\right) = 1.$
4) $1 < \rho\left(\mathrm{B_J}\right) < \rho\left(\mathrm{B_{GS}}\right).$

Convergence of Jacobi, Gauss-Seidel and SOR Iterative Methods

Lemma 1. If $A=\left(a_{i,j}\right)\geq 0$ is irreducible then either $\sum_{j=1}^{n} a_{i,j}=\rho(A)$ or

$$\min_{1\leq i\leq n}\left(\sum_{j=1}^{n} a_{i,j}\right)<\rho(A)<\max_{1\leq i\leq n}\left(\sum_{j=1}^{n} a_{i,j}\right) \quad \text{-------- (1).}$$

Proof: Case(1): All row sums of A are equal ($=\sigma$): Let $\zeta=[1,1,\cdots,1]$. Clearly, $A\zeta=\sigma\zeta$ and $\sigma\leq\rho(A)$.

However, the Gerchgorin's Theorem implies $\rho(A)\leq\sigma$. Hence, $\rho(A)=\sigma$.

Case(2): Not all row sums of A are equal:

Construct $B=\left(b_{i,j}\right)\geq 0$ and $C=\left(c_{i,j}\right)\geq 0$, by decreasing and increasing some entries of A, respectively, such that

$$\sum_{j=1}^{n} b_{\ell,j}=\alpha=\min_{1\leq i\leq n}\left(\sum_{j=1}^{n} a_{i,j}\right) \text{ and } \sum_{j=1}^{n} c_{\ell,j}=\beta=\max_{1\leq i\leq n}\left(\sum_{j=1}^{n} a_{i,j}\right), \text{ for all } 1\leq \ell\leq n.$$

By Perron-Frobenius theorem, we have $\rho(B)\leq\rho(A)\leq\rho(C)$.

Clearly, from the result of Case(1), the inequality (1) holds.

Lemma 2. Let A and B be two matrices with $0\leq|B|\leq A$. Then $\rho(B)\leq\rho(A)$

Theorem 1. Let $A=(a_{i,j})$ be a strictly or irreducibly diagonally dominant matrix then the Jacobi and Gauss-Seidel iterative methods converge.

Proof: Recall that $E_J = I-D^{-1}A = D^{-1}(L+U) = (b_{i,j})$, where $b_{i,j} = \begin{cases} 0 & i = j \\ \dfrac{-a_{i,j}}{a_{i,i}} & i \neq j \end{cases}$. From Lemma 2, it is clear that $\rho(B) \leq \rho(|B|)$. Since A is strictly diagonally dominant, clearly, we have $\sum_{j=1}^{n} |b_{i,j}| < 1$ for all $1 \leq i \leq n$. Therefore, Lemma 1 implies $\rho(|B|) < 1$. As a result, we have shown the Jacobi iterative method converge from $\rho(B) \leq \rho(|B|) < 1$.

Now, since $E_{GS} = I-(D-L)^{-1}A = (D-L)^{-1}U = (I-D^{-1}L)^{-1}D^{-1}U$. Let $\tilde{L}=D^{-1}L$ and $\tilde{U}=D^{-1}U$.

We have $|(I-\tilde{L})^{-1}\tilde{U}| \leq |(I-\tilde{L})^{-1}||\tilde{U}| \leq (I+|\tilde{L}|+|\tilde{L}|^2+\cdots+|\tilde{L}|^{n-1})|\tilde{U}| = |(I-|\tilde{L}|)^{-1}||\tilde{U}|$.

Now consider $\tilde{B}_J = |\tilde{L}| + |\tilde{U}|$ and $\tilde{B}_{GS} = |(I-|\tilde{L}|)^{-1}||\tilde{U}|$. Since we have already shown $\rho(\tilde{B}_J) < 1$, the Stein-Rosenberg theorem implies $\rho(\tilde{B}_{GS}) < 1$.

Therefore, we conclude the Gauss-Seidel iterative method converges.

Theorem 2. Let A=D-E-E^* and D be Hermitian matrices, where D is positive definite, and D-ωE is non-singular for $0 \le \omega \le 2$.

Let $E_{SOR} = I - \omega(D - \omega E)^{-1} A$. Then $\rho(E_{SOR}) < 1$ if only if A is positive definite and $0<\omega<2$.

Proof: First, assume e_0 is a nonzero vector, the SOR iteration can be written as

$$(D - \omega E)e_{m+1} = (\omega E^* + (1-\omega)D)e_m,\ m \ge 0 \quad \text{------- (2)}$$

Let $\delta_m = e_m - e_{m+1}$. Substracting $(D - \omega E)e_m$ and $(\omega E^* + (1-\omega)D)e_{m+1}$ from both side of (2), we have $(D - \omega E)\delta_m = \omega A e_m$ ---- (3) and $\omega A e_{m+1} = [(1-\omega)D + \omega E^*]\delta_m$ ----- (4).

From $e_m^* \times (3) - e_{m+1}^* \times (4)$ and "simplifying the expression" (HW), one has

$$(2-\omega)\delta_m^* D \delta_m = \omega\{e_m^* A e_m - e_{m+1}^* A e_{m+1}\} \quad \text{------ (5).}$$

Assume A is positive definite and $0<\omega<2$ and let e_0 be any eigenvector of E_{SOR}. We have $e_1 = \lambda e_0$ and $\delta_0 = (1-\lambda)e_0$ and (5) reduces to

$$\left(\frac{2-\omega}{\omega}\right)|1-\lambda|^2 e_0^* D e_0 = (1-|\lambda|^2)e_0^* A e_0 \quad \text{------- (6).}$$

Now, $\lambda \ne 1$. Otherwise, $\delta_0 = 0 \Rightarrow Ae_0 = 0$ (by (3)) $\Rightarrow$ $e_0 = 0 \Rightarrow$ contradiction!

Since A and D are positive definite and $0<\omega<2$, (6) implies $1 - |\lambda|^2 > 0$. Therefore, $\rho(E_{SOR}) < 1$.

Using similar arguments, one can show that the converse is also true.

- For SOR, it is possible to determined the optimal value of ω for special type of matrices (p-cyclic). The optimal value ω_b is precisely specified as the unique positive root

  (0<p/(p-1)) of the equation

$$\left(\rho\left(E_{\mathrm{J}}\right)\omega_b\right)^p=\left[\,p^p\left(p-1\right)^{1-p}\,\right]\left(\omega_b-1\right),\ \left(\text{Varga 1959}\right).$$

- For p=2, $\omega_b=1+\left(\dfrac{\rho\left(E_J\right)}{1+\sqrt{1-\rho^2\left(E_J\right)}}\right)$ (Young 1950)

- Semi-Iterative Method: $y_m=\sum_{j=0}^{m}v_j\left(m\right)x_j$ where $\sum_{j=0}^{m}v_j\left(m\right)=1.$

  We have $\tilde{\mathrm{e}}_{\mathrm{m}}=\sum_{j=0}^{m}v_j\left(m\right)\mathrm{e}_{\mathrm{m}}.$ In general, $\tilde{\mathrm{e}}_{\mathrm{m}}=P_m\left(E_S\right)\mathrm{e}_0$ where

  $\mathrm{P_m}$ is a polymonial and $\mathrm{E_S}$ is the error reduction of an iterater S

  This is so-called polynomial acceleration method.

  The most important one is the Chebyshev polynomials.

Chebyshev Semi-Iterative Method

Algorithm:

$y_{m+1} = \omega_{m+1}\left\{E_S y_m + f - y_{m-1}\right\} + y_{m-1}$, for m ≥ 1, where

$\omega_{m+1} = 1 + \dfrac{C_{m-1}(1/\rho)}{C_{m+1}(1/\rho)}$, $C_{\text{m-1}}$ and C_{m+1} are Chebyshev polynomials,

$\rho = \rho(E_S)$ and $y_0 = x_0$.

$\Downarrow$

$C_0 = 1,\ C_1 = x,\ C_{m+1} = 2xC_m - C_{m-1}$

Convergence:

$$\left\|\tilde{e}_m\right\| \leq \left(\frac{2(\omega_b - 1)^{m/2}}{1 + (\omega_b - 1)^m}\right)\left\|e_0\right\|, \text{ here } \omega_b = \frac{2}{1 + \sqrt{1 - \rho^2}}$$

The convergence rate is accelerated as $\rho \to 0$.

Remark: There are cases that the polynomial acceleration does not improve asymptotic rate of convergence.

Some PDE and Finite Element Analysis

Finite Element Solutions:

Let $\Im_h$ be a given triangulation, $V_h = \{ v \in H_0^1 : v|_T \in P_1(T),\ T \in \Im_h \}$ and

$\pi_h : H^1 \to V_h$ be the interpolation defined by $\pi_h(u)(N_i) = u(N_i)$.

Consider the weak solution $u \in H^1$ satisfying $a(u,v) = (f,v)$, for all $v \in H_0^1$,

a finite element solution $u_h \in V_h$ satisfies $a(u,v) = (f,v)$, for all $v \in V_h$.

Interpolation Errors:

$$\|v - \pi_h v\|_1 \le Ch^r |u|_{r+1} \text{ and } \|v - \pi_h v\|_0 \le Ch^{r+1} |u|_{r+1}.$$

H^2-Regularity: $a(\cdot,\cdot)$ is said to be H^2-Regular if there exists a constant C such that for all $f \in L^2$

$$|u|_2 \le C \|f\|_0$$

Finite Element Solution is Quasi-Optimal

Céa Theorem: $\|u-u_h\|_{\mathrm{H}^1} \le \frac{C}{\alpha} \min_{v\in V_h} \|u-v\|_{\mathrm{H}^1}$

where, C is the continuity constant and α is the coercivity constant of $\mathrm{a}(\cdot,\cdot)$.

Proof:

Step1: $a(u-u_h, v) = 0,$ for all $\mathrm{v} \in V_h$

Step2: $\alpha \|u-u_h\|_{H^1}^2 \le a(u-u_h, u-u_h) = a(u-u_h, u-v) + a(u-u_h, v-u_h)$

$$= a(u-u_h, u-v) \le C \|u-u_h\|_{H^1} \|u-v\|_{H^1}$$

HW4: If $\mathrm{a}(\cdot,\cdot)$ is self-adjoint, show that $\|u-u_h\|_{\mathrm{A}} = \min_{v\in V_h} \|u-v\|_A$,

where $\|v\|_A = \sqrt{a(v,v)}$ is the energy norm.

Remark: finite element solution is the orthogonal projection of the exact solution with respect to the energy norm.

FEM Error Estimation

Theorem 3: Assume the interpolation error estimations holds for the given $\mathfrak{I}_h$ and $a(\cdot,\cdot)$ has the H^2-Regularity. The following estimates hold.

$\|u-u_h\|_{H^1} \le Ch^r |u|_{r+1}$ ---- (i) and $\|u-u_h\|_{L^2} \le Ch^{r+1} |u|_{r+1}$ ---- (ii)

Proof: From the interpolation estimation and Céa Theorem, (i) is trivial.

To prove (ii), we use the duality argument. Let w be the solution to the adjoint problem, $a(v,w)=(u-u_h, v)$, for all $v \in H^1$. Choosing $v=u-u_h$, we have

$$(u-u_h, u-u_h) = a(u-u_h, w) = a(u-u_h, w-w_h), \text{ for any } w_h \in V_h$$

$$\le C\|u-u_h\|_{H^1}\|w-w_h\|_{H^1} \le Ch\|u-u_h\|_{H^1}|w|_2$$

$$\le Ch\|u-u_h\|_{H^1}\|u-u_h\|_{L^2}.$$

Therefore, $\|u-u_h\|_{L^2} \le Ch^{r+1}|u|_{r+1}$.

Definition: mesh dependent norm $\left\|\left|\mathrm{v}\right|\right\|_{k,h} = \sqrt{\left(A_h^k v, v\right)_h}$ for $\mathrm{v} \in \mathrm{V_h}$, k=0,1, where $\left(v,w\right)_h = \sum h^2 \left(v\left(N_i\right), w\left(N_i\right)\right)$. Clearly, $\left\|\left|\mathrm{v}\right|\right\|_{0,h} \equiv \|v\|_0$ and $\left\|\left|\mathrm{v}\right|\right\|_{1,h} \equiv \|v\|_A$.

Lemma 3: $\Lambda\left(\mathrm{A_h}\right) \le Ch^{-2}$

Proof: Let λ be an eigenvalue of $\mathrm{A_h}$ with eigenvector ϕ.

$$\mathrm{a}\left(\phi,\phi\right) = \left(\mathrm{A_h}\phi,\phi\right)_h = \lambda\left(\phi,\phi\right)_h = \lambda\left\|\left|\phi\right|\right\|_{0,h}^2 .$$

$$\lambda \le \frac{C\|\phi\|_A^2}{\left\|\left|\phi\right|\right\|_{0,h}^2} \le \frac{Ch^{-2}\|\phi\|_0^2}{\left\|\left|\phi\right|\right\|_{0,h}^2} \prec Ch^{-2}.$$

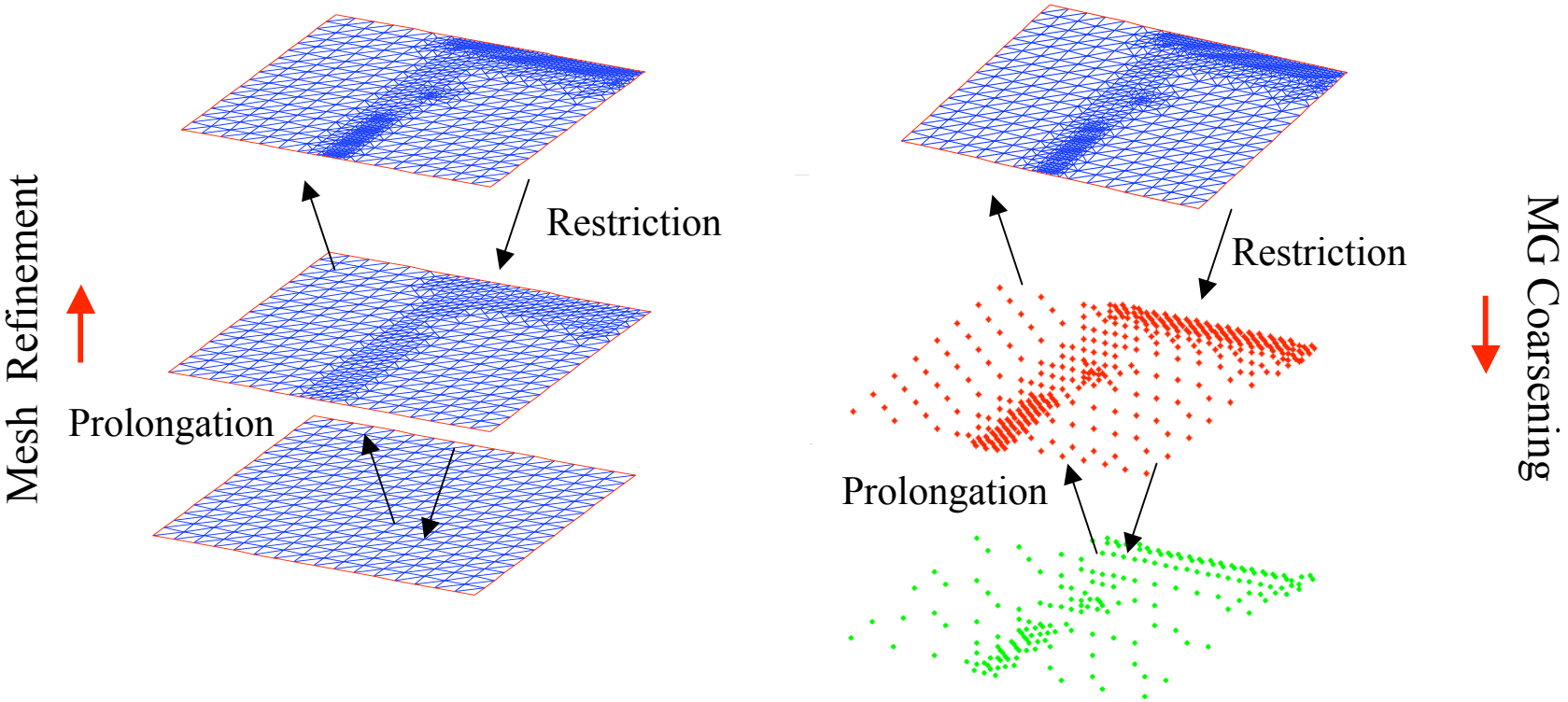
Lemma 4: (Generalized Cauchy-Schwarz Inequality)

$$\left|\mathrm{a}\left(v,w\right)\right| \le \left\|\left|v\right|\right\|_{1+t,h} \left\|\left|w\right|\right\|_{1-t,h} \quad \forall v, w \in V_h \text{ and } \mathrm{t} \in \mathrm{R}.$$

Multigrid Methods

Ideas:

- Approximate solutions on fine grid using iterative methods.
- Correct remaining errors from coarse grids.



MG V-cycle

Why Multigrid Works?

1. Relaxation methods converge slowly but smooth the error quickly.

Ex1: consider $Lu = -u'' = \lambda u \underset{\text{finite difference}}{\Longrightarrow} \frac{-u_{j-1} + 2u_j - u_{j+1}}{h^2} = \lambda u_j$

Eigenvalues $\lambda_k = \frac{4}{h^2}\sin^2\left(\frac{k\pi}{2(N+1)}\right)$ and eigenvectors $\phi_j^k = \sin\left(\frac{kj\pi}{N+1}\right)$

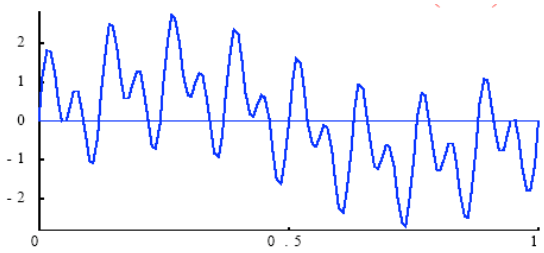
here, k=1⋯N is the wave number and j is the node number.

Richardson relaxation: $E_R = (I - \sigma^{-1}A)$ where $A_h = \frac{1}{h^2}$tridiag[-1 2 -1].

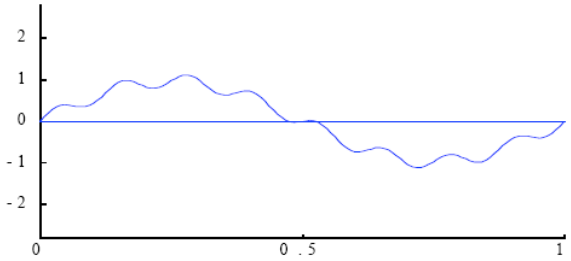
Fourier analysis: Choosing $\sigma = \frac{4}{h^2}$ (largest eigenvalue).

$$e_m = \sum_{k=1}^{N}\left(1 - \frac{\lambda_k}{\sigma}\right)^m \phi^k = \sum_{k=1}^{N}\left(1 - \sin^2\left(\frac{k\pi}{2(N+1)}\right)\right)^m \phi^k = \sum_{k=1}^{N}\alpha_k^m \phi^k$$

, after m relaxation. $\alpha_k^m \to 0$ more quickly for k close to N.



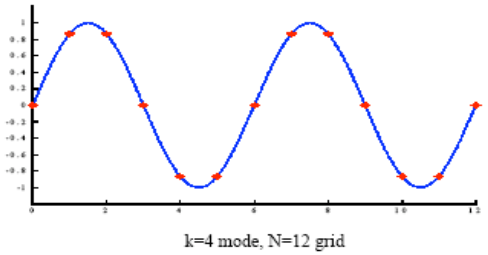
Damped Jacobi



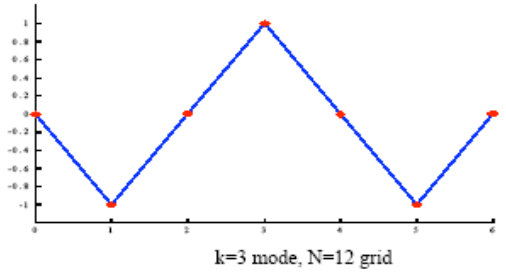
$$e = \sin\left(\frac{2j\pi}{N+1}\right) + \frac{1}{2}\sin\left(\frac{16j\pi}{N+1}\right) + \frac{1}{2}\sin\left(\frac{32j\pi}{N+1}\right)$$

$$e^m = \left(I - \omega D^{-1} A\right)^m e$$

2. Smooth error modes are more oscillatory on coarse grids. Smooth errors can be better corrected by relaxation on coarser grids.



Smooth error on fine grid



Smooth error on coarse grid

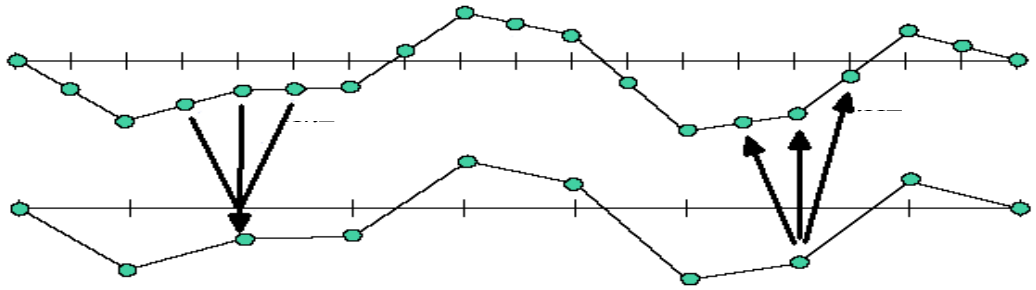
- Relaxation convergence rate on fine grid is 1-O(h^2)
- Relaxation convergence rate on coarse grid: 1-O($4h^2$)

Remember: $\alpha_1 \approx 1 - \left(\frac{\pi}{2(N+1)}\right)^2 = 1 - O\left(h^2\right)$ for $N \gg 1$

3. The smooth error is corrected by coarse grid correction operator:

$$E^c = \left(I - I_H^h A_H^{-1} I_h^H A_h\right) = \left(A_h^{-1} - I_H^h A_H^{-1} I_h^H\right) A_h,$$

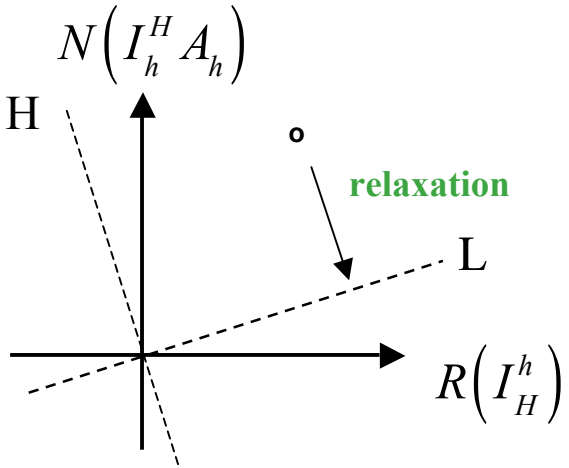
here I_h^H and I_H^h are called restriction and prolongation operator respectively.

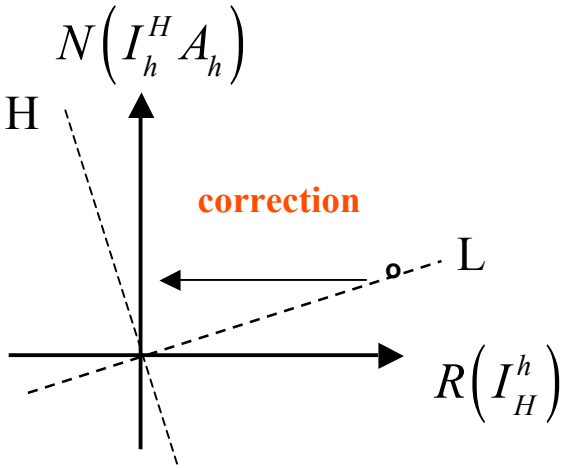


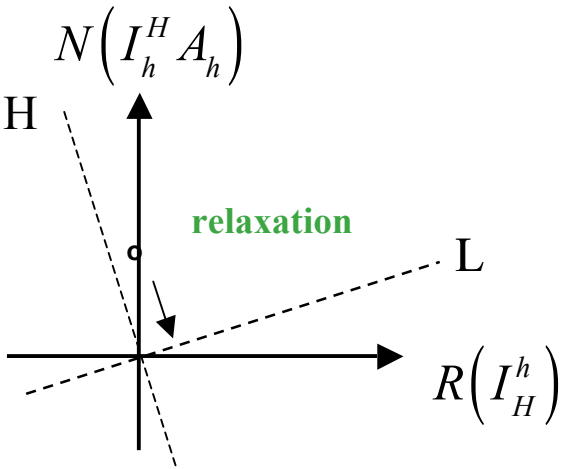
- A_H can be obtained from discretization on coarse grid
- $A_H = I_h^H A_h I_H^h$ and $I_h^H = \mathrm{c}\left(I_H^h\right)^{\mathrm{T}}$ (Galerkin formulation)

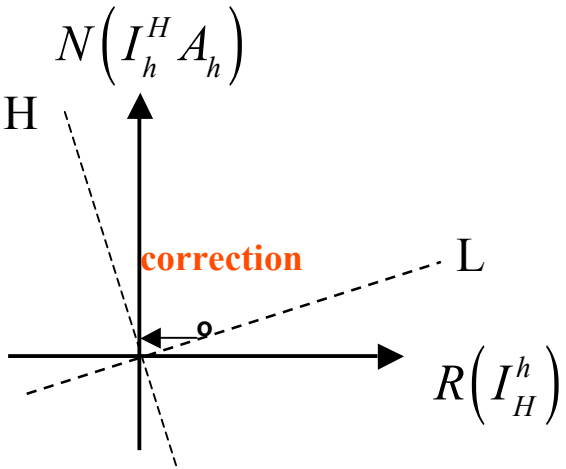
$$\Rightarrow \begin{cases} E^c \text{ is an } A\text{-orthogonal projection } \left\langle A_h E^c e, I_H^h e \right\rangle = 0 \\ N\left(E^c\right) = R\left(I_H^h\right) \\ R\left(E^c\right) = N\left(I_h^H A_h\right) \text{ and } E^c \text{ is identity on } N\left(I_h^H A_h\right) \end{cases}$$

A Picture That Show How Multigrid Works !









Consider $I_H^h = [\frac{1}{2}, 1, \frac{1}{2}]^T$ $\left(\text{linear interpolation}\right)$ and $I_h^H = \frac{1}{2}[\frac{1}{2}, 1, \frac{1}{2}]$.

It is easy to check that $\mathrm{A_H = I_h^H A_h I_H^h}$ is the discretization of L on $\Im_{\mathrm{H}}$.

Now, for any $\mathrm{v} \in \mathrm{V_h}$, let $f_v = A_h v$. One can consider v and $v_{\mathrm{H}} = I_H^h A_H^{-1} I_h^H f_v$ as finite element approximations of $\hat{v}$, the solution of $\mathrm{a}(\hat{v}, w) = (f_v, w)$. Then, from the FEM-error estimation and H^2-regularity, we have

$$\left\| \mathrm{E^c}(v) \right\|_k = \left\| \left(A_h^{-1} - I_H^h A_H^{-1} I_h^H \right) \left(A_h v \right) \right\|_k = \left\| \hat{v} - v_H - (\hat{v} - v) \right\|_k \quad \text{----- } (*)$$

$$\leq C h^{2-k} \left| \hat{v} \right|_2 \leq C h^{2-k} \left\| f_v \right\|_0 = C h^{2-k} \left\| A v \right\|_0$$

Consider the eigenfunction $\phi_{\mathrm{j}}^{\mathrm{k}}$, $\mathrm{k} \ll \frac{\mathrm{N}}{2}$. $\phi_{\mathrm{j}}^{\mathrm{k}}$ is also an eigenfunction of $\mathrm{A_H}$

We have $\left\| \mathrm{E^c} \left(\phi_{\mathrm{j}}^{\mathrm{k}} \right) \right\|_1 \leq C h \lambda_k = O(h)$. This concludes the coarse-grid correction fixes the low frequency errors. For $\mathrm{k} \approx N$, $\left\| \mathrm{E^c} \left(\phi_{\mathrm{j}}^{\mathrm{k}} \right) \right\|_1 \leq 4 \frac{C}{h}$, the high frequency errors can be amplified by coarse-grid correction.

Multigrid Algorithm

Multigrid (MG) Algorithm:

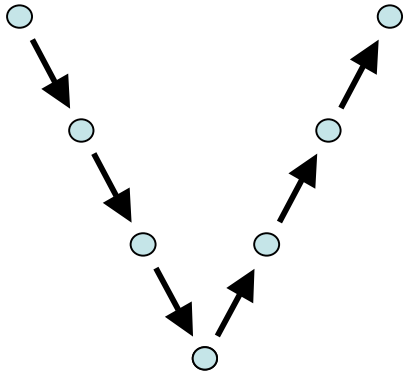
1. $x_k = w_k$
2. (pre-smoothing) $x_k = w_k + M_k^{-1}\left(g_k - A_k x_k\right)$
3. (restriction) $\tilde{g}_k = I_k^{k-1}\left(g_k - A_k x_k\right)$
4. (correction) $q_i = MG_{k-1}(q_{i-1}, \tilde{g}_k)$ for $1 \le i \le m$, $m=1$ or 2 and $q_0 = 0$
5. (prolongation) $\tilde{q}_m = I_{k-1}^{k} q_m$
6. set $x_k = x_k + \tilde{q}_m$
7. (post-smoothing) $x_k = x_k + M_k^{-1}\left(g_k - A_k x_k\right)$
8. set $MG_k(w_k, g_k) = x_k$

MG Error reduction operator:

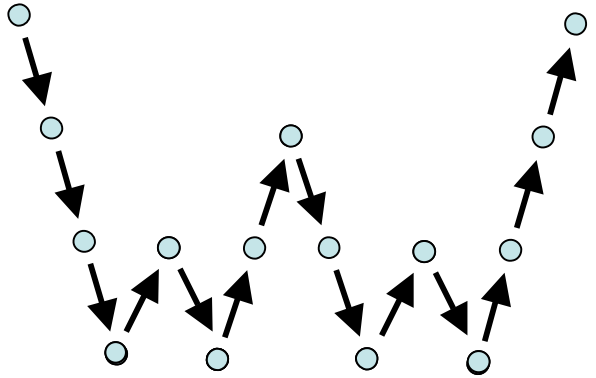
$$E_{mg} = \left(A_h^{-1} - I_H^h A_H^{-1} I_h^H\right)\left(A_h E^s\right) = E^s\left(I - I_H^h A_H^{-1} I_h^H A_h\right)$$

Pre-smoothing only Post-smoothing only

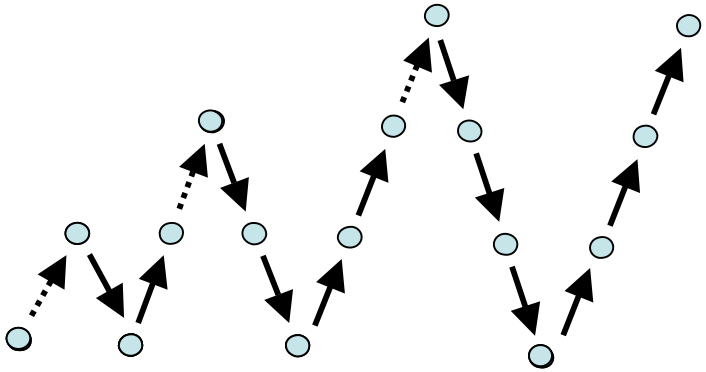
Multigrid Cycles



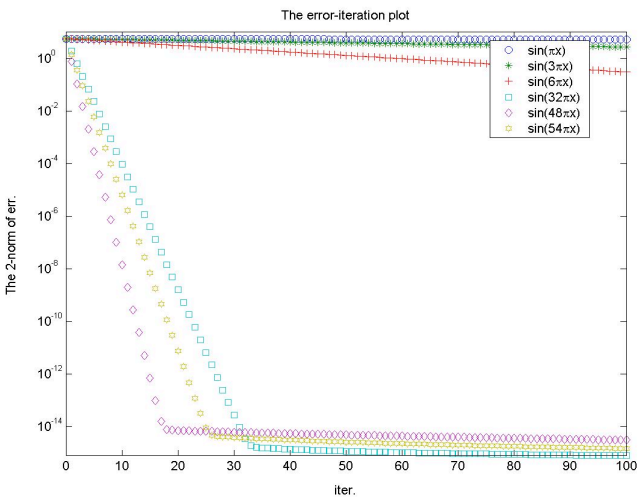
Multigrid V-cycle (m=1)

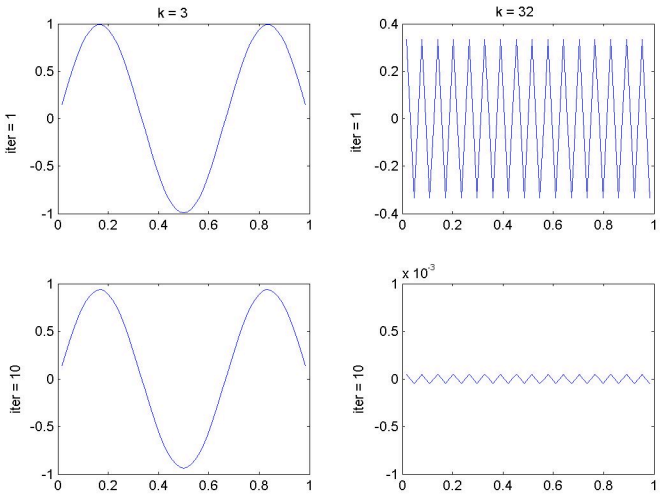


Multigrid V-cycle (m=2)

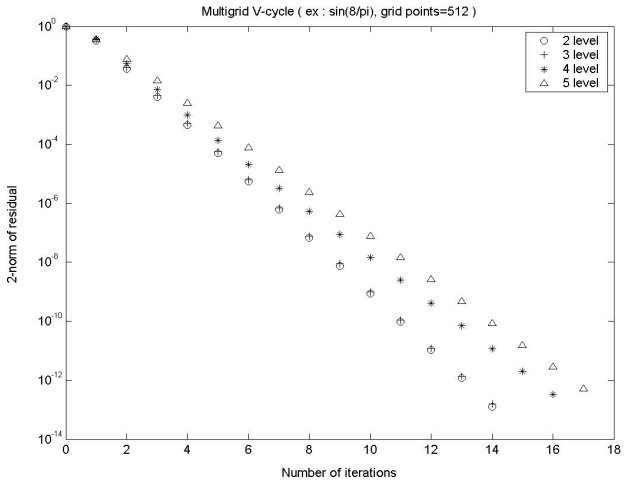


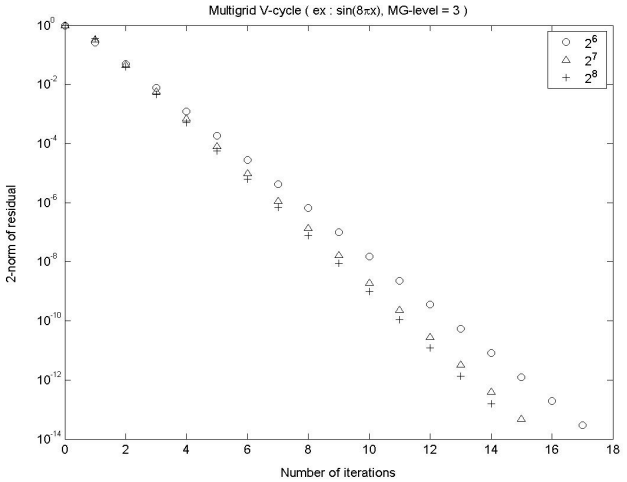
Full Multigrid cycle (FMG)





Damped Jacobi relaxation and smoothing effect





Multigrid V-cycle convergence for 1-D laplacian

Results provided by 曾昱豪 in NCTU

MG Convergence

Smoothing property: $\left\| A_l E^s \right\| \le \eta(m) \left\| A_l \right\|$, for all $0 \le m < \infty$ and $l > 0$.

Approximation property: $\left\| A_l^{-1} - I_h^H A_{l-1}^{-1} I_h^H \right\| \le C_A \left\| A_l \right\|^{-1}$, for all $l > 0$.

Ideas for proving the approximation property is shown in P.25 (*)

Proof of smoothing property:

Consider $E_S = E_R = \left(I - \frac{1}{\Lambda} A_h \right)$. Let $v \in V_h$ and $v_S^m = E_S^m (v)$. From Fourier expansion

$v = \sum v_k \phi_k$, we have $v_S^m = \left(1 - \frac{1}{\Lambda} A \right)^m v = \sum \left(1 - \frac{\lambda_k}{\Lambda} \right)^m v_k \phi_k$. Therefore,

$$\left\| A_h E_R (v) \right\|_0^2 = \left\| v_s \right\|_2^2 \le \sum \left(1 - \frac{\lambda_k}{\Lambda} \right)^{2m} \lambda_k^2 v_k^2 = \Lambda \sum \left(1 - \frac{\lambda_k}{\Lambda} \right)^{2m} \left(\frac{\lambda_k}{\Lambda} \right) \lambda_k v_k^2$$

$$\le \Lambda \sup_{0 \le x \le 1} \left\{ (1-x)^m x \right\} (v_s \cdot A_h v_s) \le C h^{-2} \frac{1}{m} \left\| v_s \right\|_0 \left\| v_s \right\|_2$$

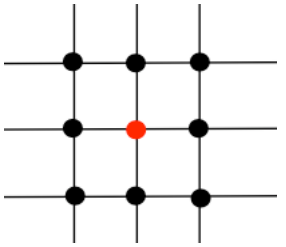
Since $\left\| v_s \right\|_0 \le \left\| v \right\|_0$ for Richarson iteration, clearly,

$$\left\| A_h E_R (v) \right\|_0 \le \frac{1}{m} \left\| A_h \right\| \left\| v \right\|_0 \Rightarrow \left\| A_h E_R \right\| \le \eta(m) \left\| A_h \right\|, \ \eta(m) \to 0 \text{ as } m \to \infty.$$

HW5: Prove MG with Richardson smoother is convergent in $\|\cdot\|_1$-norm

Choices of Interpolations and Coarse Grids

- Linear interpolation: $p = \frac{1}{4}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$ $r = \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$



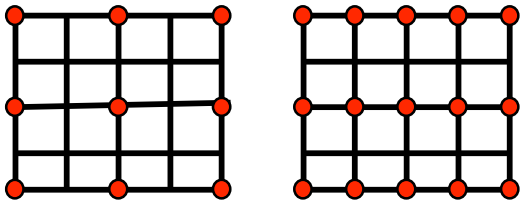
- Operator-dependent interpolation: De Zeeuw 1990

- The rule in chosing interpolation and restriction:

$$m_p + m_r > 2m \text{ (Brandt 1977)}$$

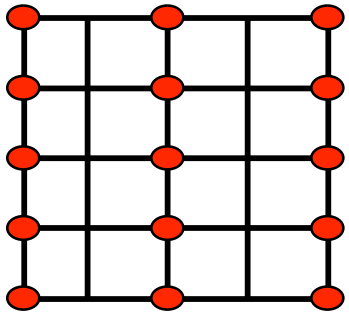
where 2m is the order of the PDE, m_p-1 is the degree of polynomials exactly interpolated by I_H^h and m_r-1 is the degree of ploynomials exactly interpolated by $\left(I_h^H\right)^T$.

- Coarse grid selection: regular coarsening, semi-coarsening, algebraic coarsening

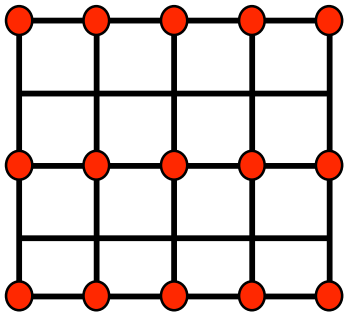


Choices of Smoothers

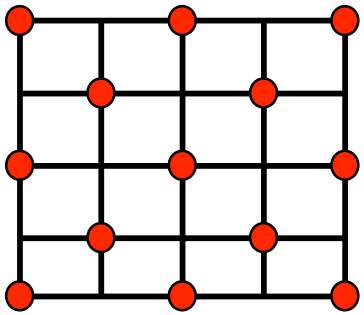
- Stationary iterative methods : Jacobi, Gauss-Seidel, SOR, …
- Block-type stationary iterative methods (blocks can be determined by the way we number the nodes)



Vertical line ordering



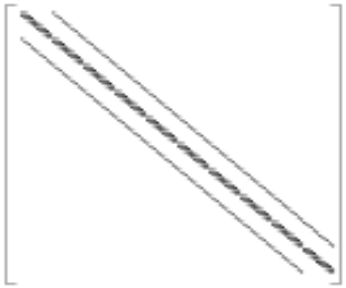
Horizontal line ordering



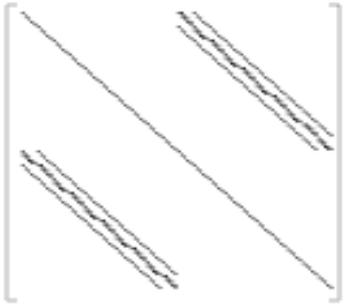
Red-Black ordering

$$A=\begin{bmatrix} T & -I & & \\ -I & T & -I & \\ & -I & T & -I \\ & & -I & T \end{bmatrix}, \quad T=[-1,4,-1]:$$

Matrix of 2-D Laplacian



Matrix Pattern for line ordering



Matrix Pattern for R-B ordering

HW6: Write a MG code for solving 1-D $\begin{cases} -u'' = f \\ u(0) = u(1) = 0 \end{cases}$

using linear interpolant I_H^h and $I_h^H = 0.5\left(I_H^h\right)^T$ and the

Red-Black Gauss-Seidel smoother 

Update even (Red) points first: $x_{2i} = 0.5\left(f_{2i} + x_{2i-1} + x_{2i+1}\right)$

Update odd (Black) points: $x_{2i} = 0.5\left(f_{2i+1} + x_{2i} + x_{2i+2}\right)$

Brandt's Local Mode Analysis

For analyzing the robustness of a smoother. Brandt's local mode analysis is a useful tool. Here, we demonstrate the method by considering the Jacobi and Gauss-Seidel relaxation for the 2-D laplace equation with periodic boundary condition.

Brandt's smoothing factor

Let ε be the error before relaxation. From discrete Fourier transform theory, ε can be written as

$$\varepsilon_{i,j} = \sum_{\theta \in \Theta_n} \hat{\varepsilon}_\theta \phi_{i,j}(\theta) \text{ ------(i), where } \theta = (\theta_1, \theta_2),\ \phi_{i,j}(\theta) = \mathrm{e}^{\mathrm{i}(i\theta_1 + j\theta_2)},$$

$$\hat{\varepsilon}_\theta = \frac{1}{(n+1)^2} \sum_{1 \le k,l \le n} \varepsilon_{k,l} \phi_{k,l}(-\theta), \text{ and}$$

$$\Theta_n = \left\{ \frac{2\pi}{n+1}(k,l) \middle| -\frac{n+1}{2} \le k,l \le \frac{n+3}{2},\ \text{n is odd} \right\}.$$

Similarly, the error $\tilde{\varepsilon}$ obtained after relaxation can be written as $\tilde{\varepsilon} = \sum_{\theta \in \Theta_n} \hat{\tilde{\varepsilon}}_\theta \phi_{i,j}(\theta)$ -----(ii). Let $\lambda(\theta) \equiv \frac{\hat{\tilde{\varepsilon}}_\theta}{\hat{\varepsilon}_\theta}$. Brandt's smoothing factor is defined as $\bar{\rho} = \sup\left\{ |\lambda(\theta)|, \frac{\pi}{2} \le |\theta_k| \le \pi, k = 1,2 \right\}$.

Smoothing Factor of Damped Jacobi Iteration

Recall that $\tilde{\varepsilon}_{i,j} = \varepsilon_{i,j} - \frac{\omega}{4}\left(4\varepsilon_{i,j} - \left(\varepsilon_{i+1,j} + \varepsilon_{i-1,j} + \varepsilon_{i,j+1} + \varepsilon_{i,j-1}\right)\right)$ Plug (i) and (ii) into it, we have

$$\sum_{\theta\in\Theta_n} \hat{\tilde{\varepsilon}}_\theta \phi_{i,j}(\theta) = \sum_{\theta\in\Theta_n} \left\{ \hat{\varepsilon}_\theta \phi_{i,j}(\theta) - \frac{\omega}{4}\left[4\hat{\varepsilon}_\theta \phi_{i,j}(\theta) - \left(\hat{\varepsilon}_\theta \phi_{i+1,j}(\theta) + \hat{\varepsilon}_\theta \phi_{i-1,j}(\theta) + \hat{\varepsilon}_\theta \phi_{i,j+1}(\theta) \right.\right.\right.$$

$$\left.\left.\left. + \hat{\varepsilon}_\theta \phi_{i+1,j}(\theta) \right) \right] \right\} = \sum_{\theta\in\Theta_n} \hat{\varepsilon}_\theta \left\{ \phi_{i,j}(\theta) - \frac{\omega}{4}\left[4\phi_{i,j}(\theta) - \phi_{i,j}(\theta)\mathrm{e}^{\mathrm{i}\theta_1} - \phi_{i,j}(\theta)\mathrm{e}^{-\mathrm{i}\theta_1} - \phi_{i,j}(\theta)\mathrm{e}^{\mathrm{i}\theta_2} \right.\right.$$

$$\left.\left. - \phi_{i,j}(\theta)\mathrm{e}^{-\mathrm{i}\theta_2} \right] \right\} = \sum_{\theta\in\Theta_n} \hat{\varepsilon}_\theta \left\{ 1 - \omega\left(1 - \frac{\cos(\theta_1) + \cos(\theta_2)}{2} \right) \right\} \phi_{i,j}(\theta)$$

Therefore, $\lambda(\theta) = 1 - \omega\left(1 - \frac{\cos(\theta_1) + \cos(\theta_2)}{2} \right)$. It is easy to see that $\bar{\rho} = \max\left\{ \left|1-\omega\right|, \left|1 - \frac{\omega}{2}\right|, \left|1 - \frac{3\omega}{2}\right| \right\}$. The optimal ω that minimize $\bar{\rho}$ is $\frac{4}{5}$ and the smoothing factor $\bar{\rho} = 0.6$ for such ω.

HW7: Show that the smoothing factor of the Gauss-Seidel iteration is 0.5

How Much Multigrid Costs?

Convergence:

- Stationary method ≈ 1-O(κ^{-1}) ≈ 1-h^2
- Conjugate gradient ≈ 1-O($\kappa^{-1/2}$) ≈ 1-h
- Multigrid ≈ O(1) independent with h

How much each MG step cost?

Ignore the cost associated with inter-grid transfer (typically within 10-20%). Computation cost of one MG V-cycle is

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

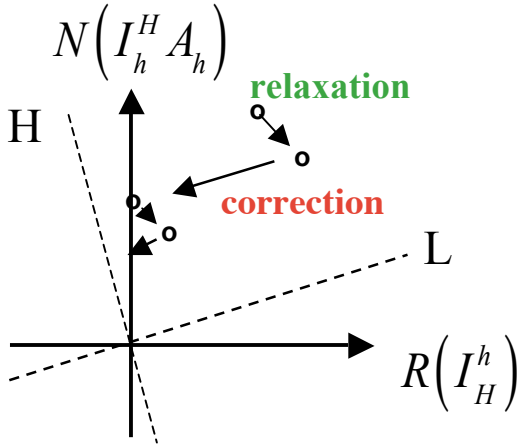
n^d : total number of points
d: dimension of the problem
c: cost for updating a single unknown
cn^d: cost per relaxation sweep.

Standard MG can fail!

- The original PDE has poor coercivity or regularity (for example, crack problems, convection-diffusion problems, etc.)

  - Relaxation may not smooth the error.
  - coarse grid correction can only capture a small portion of the error or even worse!

- The left figure is a sketch to illustrate why MG slow convergence



- Next, let's consider the following example:

$$\text{Ex2: } \begin{cases} -\dfrac{d}{dx}\left(c(x)\dfrac{du}{dx}\right) = f(x) \\ u(0) = u(1) = 0 \end{cases}, \text{ here } c(x) = \begin{cases} \varepsilon, & 0 \le x \le i_0 h \\ 1, & i_0 h < x \le i_1 h \\ \varepsilon, & i_1 h < x \le 1 \end{cases} \quad \text{---- (+)}$$

Discrete matrix of (+) $\Rightarrow$

$$A_h = \begin{bmatrix}
2\varepsilon & -\varepsilon & & & & & & & & \\
-\varepsilon & 2\varepsilon & -\varepsilon & & & & & & & \\
 & \ddots & \ddots & \ddots & & & & & & \\
 & & -\varepsilon & 1+\varepsilon & -1 & & & & & \\
 & & & -1 & 2 & -1 & & & & \\
 & & & & \ddots & \ddots & \ddots & & & \\
 & & & & & -1 & 1+\varepsilon & -\varepsilon & & \\
 & & & & & & -\varepsilon & 2\varepsilon & -\varepsilon & \\
 & & & & & & & \ddots & \ddots & \ddots \\
 & & & & & & & & -\varepsilon & 2\varepsilon
\end{bmatrix}$$

Damped Jacobi E_{DJ} matrix of A_h

$\Downarrow$

$$I - \omega D^{-1} A_h = I - \omega \begin{bmatrix}
1 & -1/2 & & & & & & & & \\
-1/2 & 1 & -1/2 & & & & & & & \\
 & \ddots & \ddots & \ddots & & & & & & \\
 & & -\varepsilon/(1+\varepsilon) & 1 & -1/(1+\varepsilon) & & & & & \\
 & & & -1/2 & 1 & -1/2 & & & & \\
 & & & & \ddots & \ddots & \ddots & & & \\
 & & & & & -1/(1+\varepsilon) & 1 & -\varepsilon/(1+\varepsilon) & & \\
 & & & & & & -1/2 & 1 & -1/2 & \\
 & & & & & & & \ddots & \ddots & \ddots \\
 & & & & & & & & -1/2 & 1
\end{bmatrix}$$

For $\varepsilon \to 0$, the eigen vector corresponding to the largest eigenvalue $\lambda^{(0)}$ of E_{DJ} converges toward to the vector $e^{(0)}$ while $\lambda^{(0)} \to 1$, where

$$\left(e^{(0)}\right)_i = \begin{cases} ih, & 0 \le x \le i_0 h \\ i_0 h, & i_0 h < x \le i_1 h \\ \dfrac{i_0}{i_1 - n - 1} ih - \dfrac{i_0 (n+1)}{i_1 - n - 1} h, & i_1 h < x \le 1 \end{cases} .$$

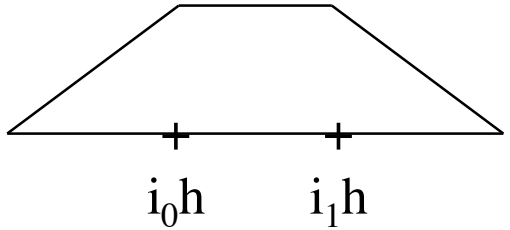


Figure of $e^{(0)}$

Damped Jacobi fails to smooth the high frequency error!
MG convergence is deteriorated as $\varepsilon \to 0$

A remedy of this is to use operator-dependent interpolation!
Construct such interpolation is not easy.
But, there is a "easier and better" way to do it!

Algebraic Multigrid

MG	AMG
1. A priori generated coarse grids are needed. Coarse grids need to be generated based on geometric information of the domain.	**1. A priori generated coarse grids are not needed! Coarse grids are generated by algebraic coarsening from matrix on fine grid.**
2. Interpolation operators are defined independent with coarsening process.	**2. Interpolation operators are defined dynamically in coarsening process.**
3. Smoother is not always fixed.	**3. Smoother is fixed.**

Ideas:

- Fix the smoothing operator.
- Carefully select coarse grids and define interpolation weights

AMG Convergence

Smoothing assumption: $\exists\ \alpha>0\ \ \ni\ \ \left\|E^s e\right\|_1^2 \le \left\|e\right\|_1^2 - \alpha\left\|e\right\|_2^2$ for all $e \in V_h$

Approximation assumption: $\min_{e_H}\left\|e - I_H^h e_H\right\|_0^2 \le \beta\left\|e\right\|_1^2$ where β is independent with e.

$$\left\|E^c e\right\|_1^2 = \left(AE^c e, E^c e - I_H^h e_H\right) \le \left\|E^c e\right\|_2 \left\|E^c e - I_H^h e_H\right\|_0 \le \beta\left\|E^c e\right\|_2 \left\|E^c e\right\|_1$$

$$\left\|E^s E^c\right\|_1^2 \le \left\|E^c e\right\|_1^2 - \alpha\left\|E^c e\right\|_2^2 \le \left(1 - \frac{\alpha}{\beta}\right)\left\|E^c e\right\|_1^2 \le \left(1 - \frac{\alpha}{\beta}\right)\left\|e\right\|_1^2$$

,here $\left\|\mathrm{v}\right\|_0 = \left\langle \mathrm{Dv},\mathrm{v}\right\rangle, \left\|\mathrm{v}\right\|_1 = \left\langle \mathrm{Av},\mathrm{v}\right\rangle, \left\|\mathrm{v}\right\|_2 = \left\langle \mathrm{D^{-1}Av},\mathrm{Av}\right\rangle,$ for $\mathrm{v} \in \mathrm{V_h}$

AMG works when A is a symmetric positive definite M-matrix.

In the following, we assume that A is also weakly diagonally dominate

What Does the Smooth Assumption Tell?

- Smooth error is characterized by $\left\| E_s \mathrm{e}_s \right\|_1 \approx \left\| \mathrm{e}_s \right\|_1$, $\left\| \mathrm{e}_s \right\|_2$ is very small

$$\|e\|_1^2 \le \left\| D^{-1/2} A e \right\| \left\| D^{1/2} e \right\| = \|e\|_2 \|e\|_0 \Rightarrow \|e\|_1 << \|e\|_0$$

$$(Ae, e) = \frac{1}{2} \sum_{i,j} -a_{i,j} (e_i - e_j)^2 + \sum_i \left(\sum_j a_{i,j} \right) e_i^2 << \sum_i a_{i,i} e_i^2$$

$$\frac{1}{2} \sum_j -a_{i,j} \left(e_i - e_j \right)^2 << a_{ii} e_i^2$$

$$\sum_{j \ne i} \frac{|a_{i,j}|}{a_{i,i}} \frac{\left(e_i - e_j \right)^2}{e_i^2} << 2$$

- **Smoother errors vary slowly in the direction of strong connection, from e_i to e_j**

  , where $|a_{i,j}| / a_{i,i}$ are large.

- **AMG coarsening should be done in the direction of the strong connections.**

- **In the coarsening process, interpolation weights are computed so that the approximation assumption is satisfied. (detail see Ruge and Stüben 1985)**

What Does the Approximation Assumption Tell?

$$\text{Approximation assumption} \equiv \min_{e_H} \left\| e - I_H^h e_H \right\|_0^2 \le \beta \left\| e \right\|_1^2$$

$$\sum_{i \in F} a_{ii} \left(e_i - \sum_{k \in C} w_{ik} e_k \right)^2 \le \beta \left(\frac{1}{2} \sum_{i,j} (-a_{ij})(e_i - e_j)^2 + \sum_i \left(\sum_j a_{ij} \right) e_i^2 \right)$$

$$\text{Since } \sum_{i \in F} a_{ii} \left(e_i - \sum_{k \in C} w_{ik} e_k \right)^2 = \sum_{i \in F} a_{ii} \left(\sum_{k \in C} w_{ik} (e_i - e_k) + (1 - s_i) e_i \right)^2$$

$$\le \sum_{i \in F} a_{ii} \left(\sum_{k \in C} w_{ik} (e_i - e_k)^2 + (1 - s_i) e_i^2 \right),$$

here $w_{ik} > 0$ is the interpolation weight from node k to node i, and $s_i = \sum_{k \in C} w_{ik} < 1$,

clearly, if (Θ)

$$\begin{cases} \sum_{i \in F} a_{ii} \sum_{k \in C} w_{ik} (e_i - e_k)^2 \le \frac{\beta}{2} \sum_{i,j} (-a_{ij})(e_i - e_j)^2 \\ \sum_{i \in F} a_{ii} (1 - s_i) e_i^2 \le \beta \sum_i \left(\sum_j a_{ij} \right) e_i^2, \end{cases}$$

the approximation assumption holds. For (Θ) to hold, we can simply require

$$(\Xi) \quad 0 \le a_{ii} w_{ik} \le \beta \left| a_{ik} \right| \text{ and } 0 \le a_{ii} \left(1 - s_i \right) \le \beta \sum_k a_{ik}.$$

Lemma 5: Given a $\beta \geq 1$, suppose the coarse grid C is selected such that

$$a_{i,i} + \sum_{\substack{j \notin C_i \\ j \neq i}} a_{i,j} = \sum_{j \notin C_i} a_{i,j} \geq \frac{1}{\beta} a_{i,i}$$

where $C_i = N_i \cap C$, C is the coarse grid and N_i = neighbors of i-th node

Then, the approximation assumption holds if the interpolation weights are defined as $w_{i,k} = \left|a_{i,k}\right| \Big/ \sum_{j \notin C_i} a_{i,j} \quad ---(\Phi)$.

$$a_{i,i}\omega_{i,k} = a_{i,i} \frac{\left|a_{i,k}\right|}{\sum_{j \notin C_i} a_{i,j}} = \frac{a_{i,i}}{\sum_{j \notin C_i} a_{i,j}} \left|a_{i,k}\right| \leq \beta \left|a_{i,k}\right|$$

$$a_{i,i}\left(1 - s_i\right) = a_{i,i}\left(1 - \sum_{k \in C_i} \omega_{i,k}\right) = a_{i,i}\left(1 - \sum_{k \in C_i} \frac{\left|a_{i,k}\right|}{\sum_{j \notin C_i} a_{i,j}}\right) = a_{i,i}\left(\frac{\sum_j a_{i,j}}{\sum_{j \notin C_i} a_{i,j}}\right)$$

$$\leq \beta \sum_j a_{i,j}$$

Therefore, (Ξ) holds. From the arguments in previous page, We can conclude the approximation holds.

Smoothing property holds for GS

Recall $\mathrm{E_S} = I - B^{-1}A$. We have

$$\begin{aligned}\left\|\mathrm{E_{GS}e}\right\|_1^2 &= \left(A\left(I-B^{-1}A\right)e, \left(I-B^{-1}A\right)e\right)\\ &= (Ae,e) - \left(AB^{-1}Ae,e\right) - \left(Ae,B^{-1}Ae\right) + \left(AB^{-1}Ae,B^{-1}Ae\right)\\ &= \|e\|_1^2 - \left(B^{-1}Ae,BB^{-1}Ae\right) - \left(BB^{-1}Ae,B^{-1}Ae\right) + \left(AB^{-1}Ae,B^{-1}Ae\right)\\ &= \|e\|_1^2 - \left(\left(B^T+B-A\right)B^{-1}Ae,B^{-1}Ae\right).\end{aligned}$$

The smooth assumption $\equiv \alpha\left\|\mathrm{e}\right\|_2^2 \le \left(\left(B^T+B-A\right)B^{-1}Ae,B^{-1}Ae\right)$ ---- (Θ)

Let ẽ=$B^{-1}Ae$. Since $\left\|\mathrm{e}\right\|_2^2 = \left(D^{-1}Ae,Ae\right) = \left(D^{-1}BB^{-1}Ae,BB^{-1}Ae\right)$,

clearly, (Θ) $\equiv \alpha\left(D^{-1}B\tilde{e},B\tilde{e}\right) \le \left(\left(B^T+B-A\right)\tilde{e},\tilde{e}\right)$ ---- (ΘΘ).

Now consider B=D-L. Since $B^T + B - A = D,$ we have

$$(\Theta\Theta) \equiv \alpha\frac{\left(B^TD^{-1}B\tilde{e},\tilde{e}\right)}{\left(D\tilde{e},\tilde{e}\right)} = \alpha\frac{\left(D^{-1}B^TD^{-1}BD\tilde{e},\tilde{e}\right)}{\left(D^{1/2}\tilde{e},D^{1/2}\tilde{e}\right)} = \alpha\frac{\left(D^{-1}B^TD^{-1}BD^{1/2}\tilde{e},D^{1/2}\tilde{e}\right)}{\left(D^{1/2}\tilde{e},D^{1/2}\tilde{e}\right)}.$$

$$= \alpha\rho\left(D^{-1}B^TD^{-1}B\right) \le 1. \equiv \alpha \le \frac{1}{\rho\left(D^{-1}B^TD^{-1}B\right)} \text{---- } (\Theta\Theta\Theta)$$

Therefore, the smooth assumption holds for Gauss-Seidel iteration.

If A is a diagonally dominant M-matrices, we can estimate α as follows:

Since $\rho\left(D^{-1}B^T D^{-1}B\right) \le \rho\left(I - D^{-1}L^T\right)\rho\left(I - D^{-1}L\right) \le \left(1+\rho\left(D^{-1}L^T\right)\right)\left(1+\rho\left(D^{-1}L\right)\right)$,

and $\rho\left(D^{-1}L\right) \le \max\limits_{1\le i\le n}\left\{\sum\limits_{j=1, j\ne i}^{n}\dfrac{\left|a_{i,j}\right|}{a_{i,i}}\right\} \le 1$, clearly, we have $\dfrac{1}{\rho\left(D^{-1}B^T D^{-1}B\right)} \ge \dfrac{1}{4}$.

Therefore, Gauss-Seidel iteration satisfies the smoothing property with $\alpha=\dfrac{1}{4}$.

If fact, for symmetric M-matrices, smooth assumption holds for both Gauss-Seidel and Jacobi iterations.

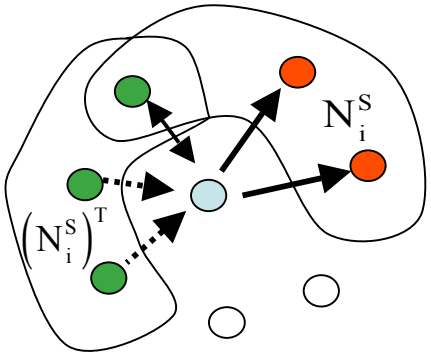
Furthermore, one can also show that the coarse grid matrix $A_H = \left(I_H^h\right)^T A_h I_H^h$ is also a diagonally dominant M-matrix when A_h is a diagonally dominant M-natrix and the interpolation weights satisfy (Ξ) and (Φ).

AMG Coarsening Criteria

First, let's define the following sets:

$$\mathrm{N}_i^S = \left\{ j : -a_{i,j} \geq \gamma \max_{m \neq i} \left(-a_{i,m} \right), 0 < \gamma < 1 \right\}$$

$$\left(\mathrm{N}_i^S \right)^T = \left\{ j : i \in \mathrm{N}_j^S \right\}$$



Here, N_i^S is the set of nodes that node i strongly connects to.
$\left(\mathrm{N}_i^S\right)^T$ is the set of nodes strongly connects to node i.

- C_i-nodes should be chosen from N_i^S

- From convergence result, we want β close to 1.
  This suggests we need a larger set C_i (we need to choose a small γ).

- We don't want C=ø but we want C as small as possible.

Criterion 1. For each node i in F, node j in N_i^s should be either in C or strongly connected to at least one node in C_i.

Criterion 2. C should be a maximal subset of all nodes with the property that no two C-nodes are strongly connected to each other.

AMG Coarsening (I)

$C = \emptyset; F = \emptyset; U = \{1, 2, \cdots, n\};$

For $(i = 1 : n)$, $z_i = |(N_i^S)^T|$;

while $(U \neq \emptyset)$ **do**

get $i \in U$ with maximal z_i then set $C = C \cup \{i\}$ and $U = U \setminus \{i\}$;

for $(j \in (N_i^S)^T \cap U)$ **do**

$F = F \cup \{j\}; U = U \setminus \{j\}$;

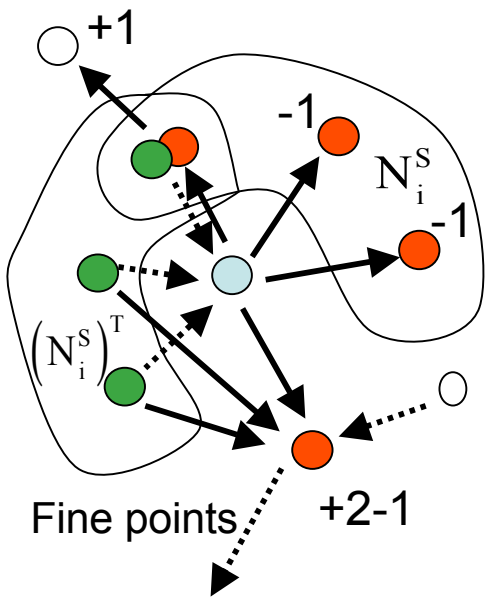
For $(k \in N_j^S)$, $z_k = z_k + 1$;

end for

For $(j \in N_i^S \cap U)$ $z_j = z_j - 1$;

end while

Algorithm 4.4.1: Preliminary C-point selection



AMG Coarsening (II)

$T = \emptyset$;

while $(F \setminus T \neq \emptyset)\{$

pick $i \in F \setminus T$; set $T = T \cup \{i\}$ and $done = 0$;

$C_i = N_i^S \cap C$; $D_i^S = N_i^S \setminus C_i$; $D_i^W = N_i \setminus N_i^S$; $\tilde{C}_i = \emptyset$;

while $(done == 0)\{$

$d_i = a_{i,i} + \sum_{k \in D_i^W} a_{i,k}$; $d_j = a_{i,j} \forall j \in C_i$

for $(k \in D_i^S)\{$

if $(N_k^S \cap C_i \neq \emptyset)$ $d_j = d_j + \frac{a_{i,k} a_{k,j}}{\sum_{m \in C_i} a_{k,m}} \forall j \in C_i$;

else {

if $(\tilde{C}_i \neq \emptyset)\{C = C \cup \{i\}; F = F \setminus \{i\};$ break; $\}$

else {

$\tilde{C}_i = \{k\}$; $C_i = C_i \cup \{k\}$; $D_i^S = D_i^S \setminus \{k\}$;

$done = 0$; break;

}

}

}

}

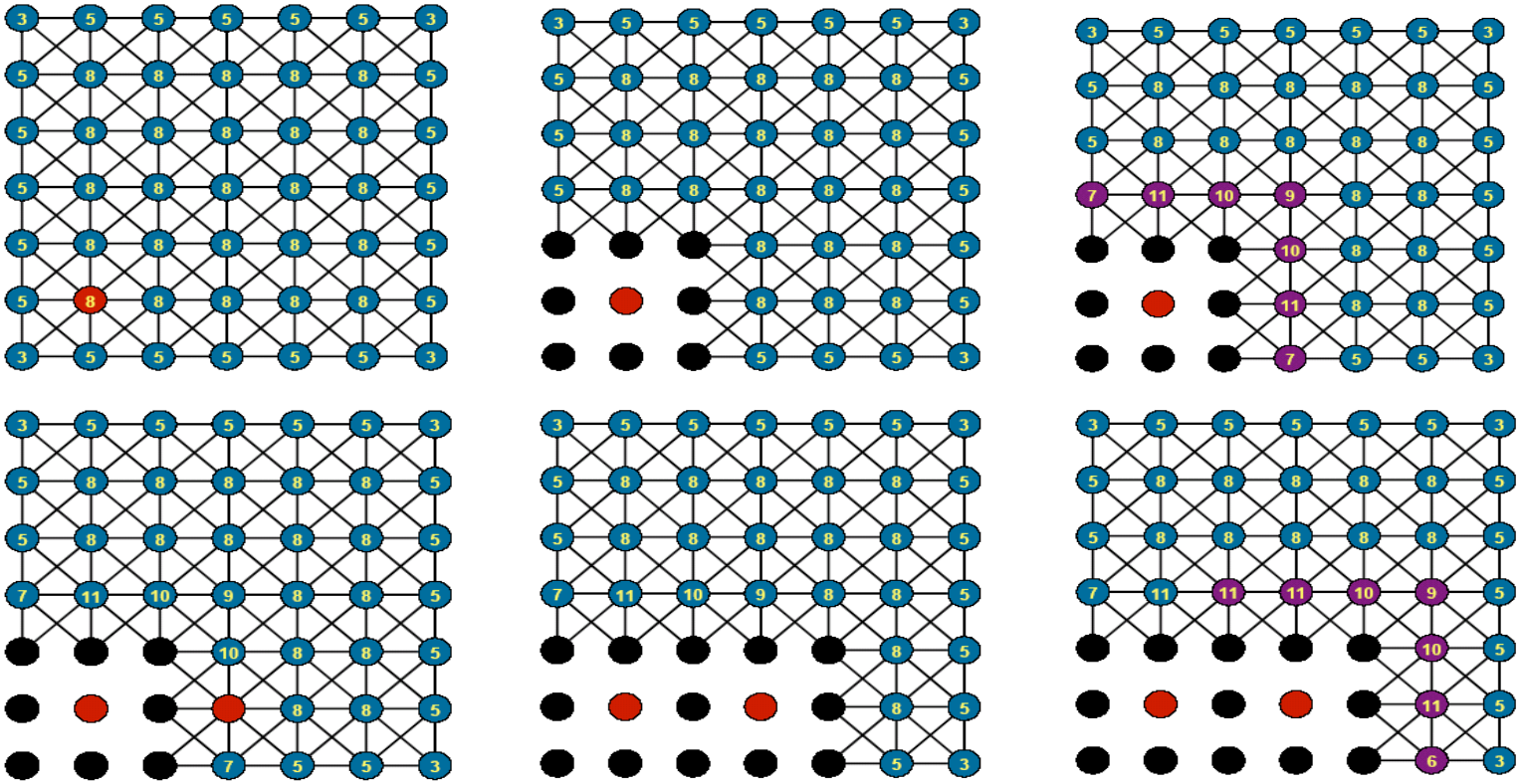
if $(i \in F)$ $\{C = C \cup \tilde{C}_i; F = F \setminus \tilde{C}_i; w_{i,j} = -d_j/d_i \forall j \in C_i\}$

}

Algorithm 4.4.2: Final C-point selection and definition of interpolation weights

AMG Coarsening: Example 1

Laplace operator from Galerkin FEM Discretization:



A very good MG and AMG tutorial resource (by Van Emden Henson) :

http://www.llnl.gov/CASC/people/henson

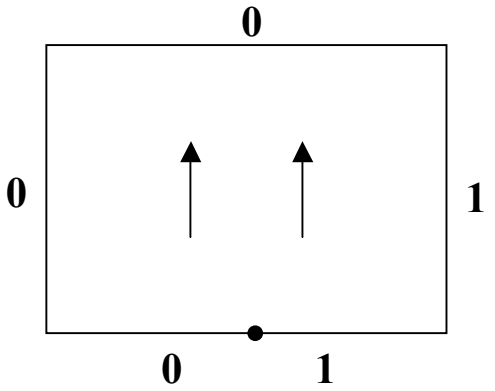
AMG Coarsening: Example 2

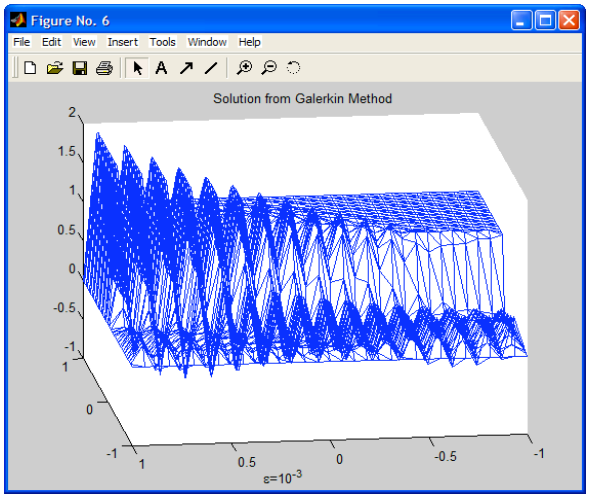
Convection-Diffusion with Characteristic and downstream layers

$$-\varepsilon\Delta u + \frac{\partial u}{\partial y} = 0$$

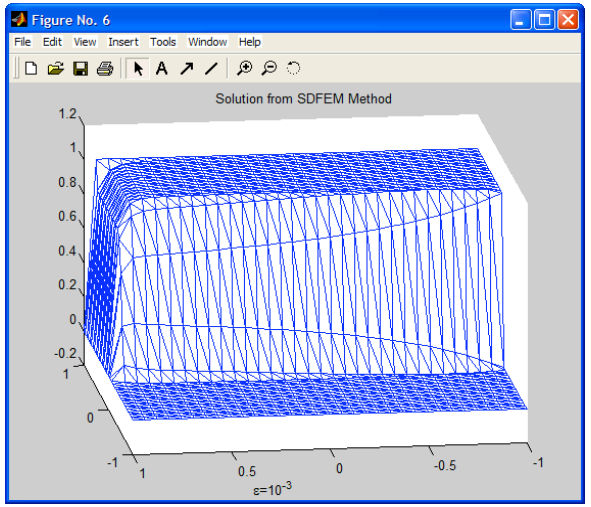
$$u\,|_{\partial\Omega} = \begin{cases} 1 & \text{if } (y=0 \cap x>0) \text{ or } x=1, \\ 0 & \text{otherwise,} \end{cases}$$

where $\Omega=[-1,1]\times[-1,1]$.





Solution from Galerkin discretization on 32x32 grid



Solution from SDFEM discretization on 32x32 grid

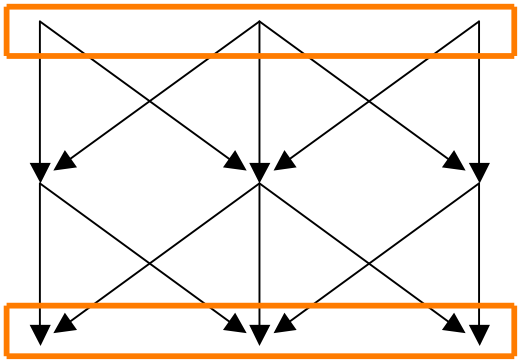
SDFEM discretization with $\delta_T = \frac{h}{2}$

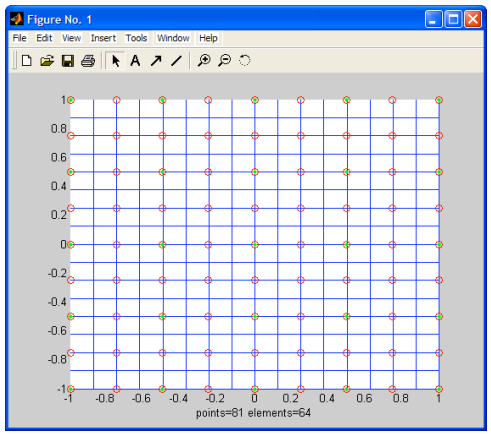
yields the left matrix stencil:

$$\begin{bmatrix} -\frac{\varepsilon}{3} & -\frac{\varepsilon}{3} & -\frac{\varepsilon}{3} \\ \frac{h}{6}-\frac{\varepsilon}{3} & \frac{2h}{3}+\frac{8\varepsilon}{3} & \frac{h}{6}-\frac{\varepsilon}{3} \\ -\frac{h}{6}-\frac{\varepsilon}{3} & -\frac{2h}{3}-\frac{\varepsilon}{3} & -\frac{h}{6}-\frac{\varepsilon}{3} \end{bmatrix}$$

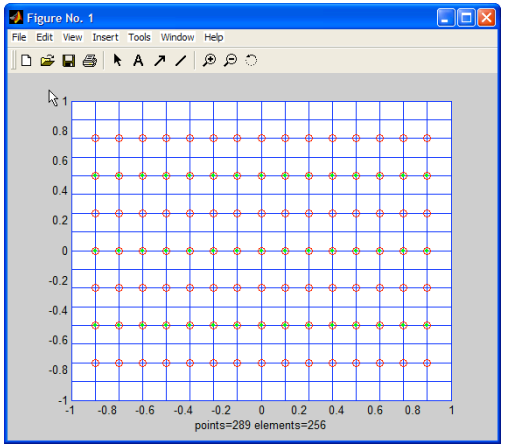
AMG coarsening with strong connection parameter

$\varepsilon/h << \beta << 0.25$





Coarse grids from GMG coarsening



Coarse grids from AMG coarsening

Example 2: GMG v.s. AMG

	GMG	AMG		
$\log_{10}\frac{1}{\epsilon}$	2, 3, 4	2	3	4
level=4	1089	1089	1089	1089
level=3	289	480	479	479
level=2	81	307	331	231
level=1	25	157	108	108

(a) On the uniform mesh

	GMG			AMG		
$\log_{10}\frac{1}{\epsilon}$	2	3	4	2	3	4
level=4	797	1275	2102	797	1275	2102
level=3	410	649	1047	348	580	996
level=2	215	320	528	159	304	523
level=1	122	176	239	88	166	281

(b) On the adaptive mesh

On the uniform mesh:

level	GMG	AMG
3	13	7
2	13	6
1	12	6

$\varepsilon=10^{-2}$

level	GMG	AMG
3	27	8
2	26	7
1	16	6

$\varepsilon=10^{-3}$

level	GMG	AMG
3	51	11
2	35	8
1	17	6

$\varepsilon=10^{-4}$

On the adaptive mesh:

Level	GMG	AMG
4	9	6
3	8	8
2	7	6
1	7	5

$\varepsilon=10^{-2}$

Level	GMG	AMG
4	22	8
3	24	9
2	18	8
1	17	7

$\varepsilon=10^{-3}$

Level	GMG	AMG
4	59	14
3	57	10
2	47	8
1	34	7

$\varepsilon=10^{-4}$

Nonlinear Multigrid

Nonlinear problems: $\mathrm{L}(\mathrm{u}) = f \underset{discretization}{\Rightarrow}$

$$\text{One needs to solve } \mathrm{A_h}(u_h) = f_h \equiv \begin{pmatrix} a_1(u_1, u_2, \cdots, u_n) \\ a_2(u_1, u_2, \cdots, u_n) \\ \vdots \\ a_n(u_1, u_2, \cdots, u_n) \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{pmatrix}.$$

Method 1: Linearize A_h using Newton method and solve the linear system by multigrid.

$$u_j \leftarrow u_j - \left[\frac{D}{Du}\mathrm{A_h}(u)\right]^{-1}\left(f - \mathrm{A_h}(u_j)\right)$$

Method 2: Nonlinear Multigrid , so called full approximation storage scheme (FAS)

- **Nonlinear relaxation**
- **Nonlinear defect correction**

Nonlinear Relaxation:

Jacobi: $a_i\left(u_1^{old},\cdots,u_{i-1}^{old},u_i^{new},u_{i+1}^{old},\cdots,u_n^{old}\right)$ for all i=1,2,⋯,n

Gauss-Seidel: $a_i\left(u_1^{new},\cdots,u_{i-1}^{new},u_i^{new},u_{i+1}^{old},\cdots,u_n^{old}\right)$ for all i=1,2,⋯,n

Solve local nonlinear problems iteratively.

Example (Nonlinear Gauss-Seidel) :

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

Discretiation:

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

Newton iteration for each j, starting from j=1

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

Nonlinear defect correction:

In linear case: $r_h^{(n)} = A_h\left(u_h\right) - A_h\left(u_h^{(n)}\right) = A_h\left(u_h - u_h^{(n)}\right)$

In nonlinear case: $r_h^{(n)} = A_h\left(u_h\right) - A_h\left(u_h^{(n)}\right) \neq A_h\left(u - u_h^{(n)}\right)$

Solving $A_H e_H = I_h^H r_h^{(n)}$ does not give an approximation to $e_h = u_h - u_h^{(n)}$.

Now consider $\begin{cases} e_h = u_h - u_h^{(n)} \text{ where } u_h \text{ satisfies } r_h^{(n)} = A_h\left(u_h\right) - A_h\left(u_h^{(n)}\right) \\ e_H = u_H - I_h^H u_h^{(n)} \text{ where } \mathrm{u_H} \text{ satisfies } I_h^H r_h^{(n)} = A_H\left(u_H\right) - A_H\left(I_h^H u_h^{(n)}\right) \end{cases}$

Observe that $u_h^{(n)} \to u_h \Rightarrow u_H \to I_h^H u_h \Rightarrow e_H \to I_h^H e_h$. (Here, I_h^H can simply be an injection)

In this point of view, $\mathrm{e_H}$ is a reasonable approximation of $\mathrm{e_h}$.

Now, we can write down the FAS algorithm:

FAS:

1. Nonlinear Relaxation
2. Restrict u_h^n and r_h^n by $r_H = I_h^H r_h^n$ and $v = I_h^H u_h^n$
3. Solve $A_H\left(u_H\right) = r_H + A_H\left(v\right)$
4. Compute $e_H = u_H - v$
5. Update $u_h^n \leftarrow u_h^n + I_H^h e_H$

$-\Delta u(x, y) + \gamma u(x, y)\, e^{u(x,y)} = f(x, y)$ in $[0,1]\times[0,1]$,

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

- Discretization: finite difference
- Interpolation and Restriction $p = \frac{1}{4}\begin{bmatrix}1 & 2 & 1\\ 2 & 4 & 2\\ 1 & 2 & 1\end{bmatrix}$ $r = \frac{1}{16}\begin{bmatrix}1 & 2 & 1\\ 2 & 4 & 2\\ 1 & 2 & 1\end{bmatrix}$
- Relaxation: Nonlinear Gauss-Seidel:

$$u_{i,j} = u_{i,j} - \frac{h^{-2}\left(4u_{i,j} - u_{i-1,j} - u_{i+1,j} - u_{i,j-1} - u_{i,j+1}\right) + u_{i,j}e^{u_{i,j}} - f_{i,j}}{4h^{-2} + \gamma\left(1+u_{i,j}\right)e^{u_{i,j}}}$$

starting from $(i,j) = (1,1),(1,2),\cdots,(2,1),(2,2),\cdots,(n,n).$

An example taking from Multigrid Tutorial (Briggs)

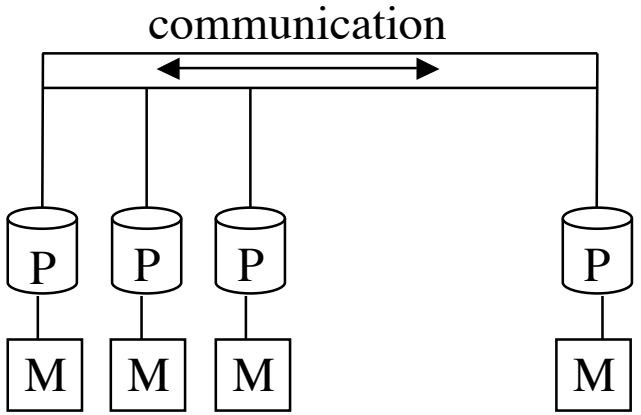
Method	Outer iterations	Inner iterations	Megaflops
Newton	3		1660.6
Newton-MG	3	20	56.4
Newton-MG	4	10	38.5
Newton-MG	5	5	25.1
Newton-MG	10	2	22.3
Newton-MG	19	1	24.6
FAS	11		27.1

Who is better Newton-MG or FAS?

Not so sure … but FAS is popular in CFD.

Multigrid Parallelization

Parallelization: Using multiple computers to do the job!



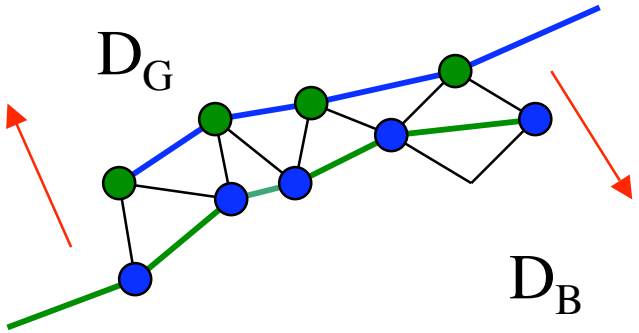
What need to be done?

1. Numerical algorithm need to be capable to do it.
2. Program has to distribute works to processors properly and dynamically. (load balancing)
3. Computers have to communicate each others. (Messaging passing interface, MPI)
4. Many others (grid topoloogy, scheduling, ….)

Multigrid is a scalable algorithm!

(Jim E. Jones, CASC, Lawrence Livermore National Laboratory)

Domain Decomposition:



- FEM assembling in domains D_G, D_B, … can be done simultaneously.
- Matrix-vector product $A \cdot x$ can be computed indenpendently in each domain. Pass x_G to D_2 and x_B to D_1.
- Jacobi and red-black Gauss-Seidel Relaxations can be done in parallel.
- Grid-Coarsening and refinement can be done in parallel (not quite easy … need to keep tracking grid topology).
- Interpolations can be parallelized too.

Scalability

$T(N,P)$: Time to solve a problems with N unknowns on P processors

Speedup S(N,P)=T(N,1)/T(N,P). Perfect if S(N,P)=P;

Scaled Efficiency: E(N,P)=T(N,1)/T(NP,P). Perfect if E(N,P)=1.

Assume 2D problem of size $(pN)^2$ is distributed to p^2 processors.
Number of unknowns in each processor N^2

5 point stencils

Time for relaxation on grid level k: $T_k = T_{comm}\left(\frac{N}{2^k}\right) + 5\left(\frac{N}{2^k}\right)^2 f,$

Time for a V-cycle = $T_v \approx \sum_k 2T_k \approx 8\alpha L + 16N\beta + \left(\frac{40}{3}\right)N^2 f,$

$\alpha =$ startup time,

$\beta =$ time to transfer a single double

$T_{comm}(n) = \alpha + \beta n =$ communication time for transmitting n doubles to one processor.

$f =$ one floating point operation time.

Since MG has O(1) convergence rate, we can analyze the scaled efficiency as follows:

$$T_v\left(N^2,1\right)\approx\left(\frac{40}{3}\right)N^2 f$$

$$T_v\left(\left(pN\right)^2,p\right)=8\alpha\log_2\left(pN\right)+16\beta\left(N\right)+\frac{40}{3}N^2 f$$

$$E\left(N,P\right)\approx O\left(1/\log_2 p\right)\text{ as p}\to\infty$$

$$E\left(N,P\right)\approx O\left(1\right)\text{ as N}\to\infty$$

We need to be careful. In IBM SP2,

$$\alpha=5\times10^{-5}$$
$$\beta=1\times10^{-6}\quad\Leftrightarrow\quad\text{Communication is expensive!}$$
$$f=8\times10^{-9}$$