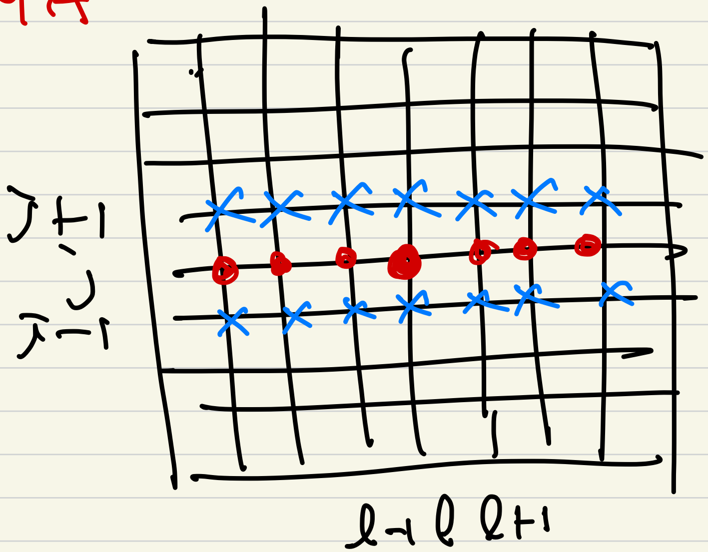
① Jacobi, ② GS ③ SOR

④ Line Jacobi method

$$\phi_{l,j}^{k+1} = \frac{1}{4}\left(\phi_{l-1,j}^{k+1} + \phi_{l+1,j}^{k+1} + \phi_{l,j-1}^{k} + \phi_{l,j+1}^{k}\right) - \frac{\Delta x^2}{4} f_{l,j}$$

TDMA



x : k-th iteration

• : k+1 th iteration

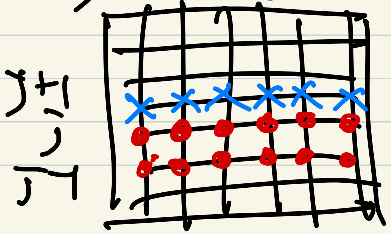
faster than point Jacobi.

⑤ Line GS method

$$\phi_{l,j}^{k+1} = \frac{1}{4}\left(\phi_{l-1,j}^{k+1} + \phi_{l+1,j}^{k+1} + \phi_{l,j-1}^{k+1} + \phi_{l,j+1}^{k}\right) - \frac{\Delta x^2}{4} f_{l,j}$$

TDMA

faster than point GS



• k+1 th iteration

x kth iteration

⑥ ADI method using artificial derivative term

$$\nabla^2\phi = b$$

$$\frac{\partial\phi}{\partial t} = \frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial y^2} - b$$

EE → Jacobi ✗
IE → ○
CN → △

not very popular

⑦ strongly implicit procedure (SIP)

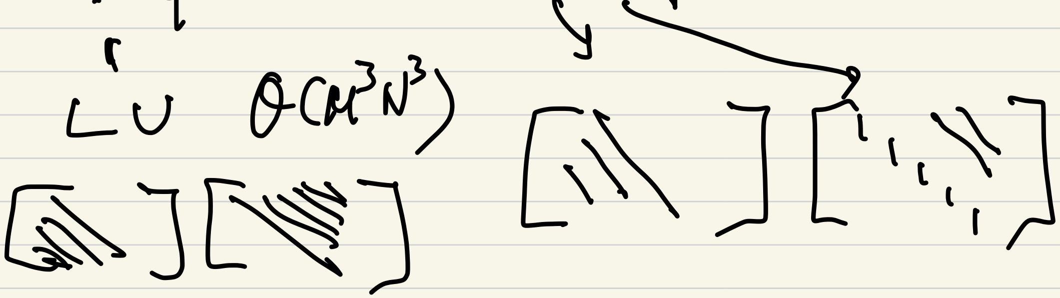
Stone (1968) SIAM J. Num. Anal. 5, 530.

$$A\phi = b \rightarrow LU\phi^{k+1} = (LU - A)\phi^k + b$$

LU $O(M^3N^3)$

find L & U s.t. $LU \simeq A$

incomplete LU decomp. (ILU)



⑧ Conjugate Gradient Solver (CGS)

Hesteness & Stiefel (1952) Nat. Bur. Stand. J. Res. 49, 409

Kershaw (1978) J. Comp. Phys. 26, 43.

— : simple gradient method (steepest descent method)

- - - : conjugate gradient method

for a symmetric matrix A ($a_{ij} = a_{ji}$)

$$\underline{\phi \equiv \frac{1}{2} x^T A x - x^T b} = \frac{1}{2} \sum a_{ij} x_i x_j - \sum x_i b_i$$

$$\frac{\partial \phi}{\partial x_k} = \frac{1}{2} \sum a_{ij} \delta_{ik} x_j + \frac{1}{2} \sum a_{ij} x_i \delta_{jk} - \sum \delta_{ik} b_i$$

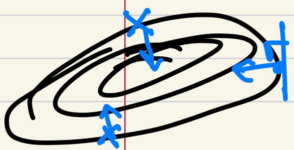
$$= \frac{1}{2} \sum a_{kj} x_j + \frac{1}{2} \sum a_{ik} x_i - b_k = 0 \quad \boxed{\text{for min } \phi}$$

$$\to \sum a_{kj} x_j = b_k \quad (\Rightarrow Ax = b) \to \underline{x = A^{-1} b}$$

∴ minimizing ϕ & solving $Ax=b$ are equivilent problems.

One of the simplest strategies for minimizing ϕ is the method of steepest descent.

At a current point x_c, the ft. ϕ decreases most rapidly in the direction of the negative gradient.



$$-\nabla\phi : \underline{-\frac{\partial \phi}{\partial x_i}} = -\sum a_{ij} x_j + b_i \equiv \underline{r_i} \quad (\text{residual of } Ax=b)$$

If residual is non-zero, $\exists$ an α s.t. $\phi(x_i + \alpha r_i) < \phi(x_i)$

$$\phi(x_i + \alpha r_i) = \frac{1}{2} \sum a_{ij} (x_i + \alpha r_i)(x_j + \alpha r_j) - \sum b_i (x_i + \alpha r_i)$$

$$\frac{\partial \phi}{\partial \alpha} = \frac{1}{2} \sum a_{ij} r_i (x_j + \alpha r_j) + \frac{1}{2} \sum a_{ij} (x_i + \alpha r_i) r_j - \sum b_i r_i$$

$$(a_{ij} = a_{ji}) \quad = \alpha \sum a_{ij} r_i r_j + \sum a_{ij} x_j r_i - \sum b_i r_i$$

$$= \alpha \sum a_{ij} r_i r_j + \sum r_i \big(\underbrace{a_{ij} x_j - b_i}_{= -r_i}\big) = 0$$

$$\Rightarrow \quad \alpha = \sum r_i r_i \Big/ \sum a_{ij} r_i r_j$$

$k = 0$: $x_0 = 0$, $r_0 = -\sum a_{ij} x_{j0} + b_i = b_i$

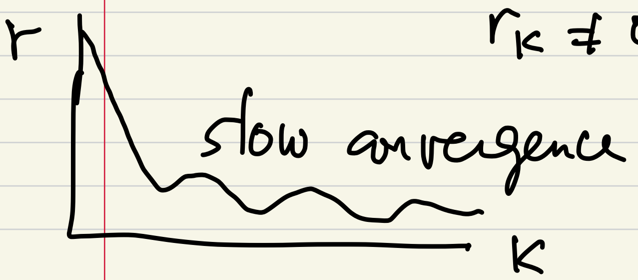
$r_k \neq 0$, $k = k+1$

$$\alpha_k = \sum r_{k-1}^T r_{k-1} \Big/ r_{k-1}^T A r_{k-1}$$

$$x_k = x_{k-1} + \alpha_k r_{k-1} \rightarrow r_k = b - A x_k$$

iteration

steepest descent method



$\phi(x_i + \alpha p_i) < \phi(x_i)$ $\quad p_i$ arbitrary

$$\phi(\alpha) = \frac{1}{2}\sum a_{ij}(x_i + \alpha p_i)(x_j + \alpha p_j) - \sum b_i(x_i + \alpha p_i)$$

$$\frac{\partial \phi}{\partial \alpha} = \cdots = \alpha \sum a_{ij} p_i p_j + \sum p_i (\underbrace{a_{ij} x_j - b_i}_{= -r_i}) = 0$$

$$\Rightarrow \alpha = \sum p_i r_i \Big/ \sum a_{ij} p_i p_j = p^T r / p^T A p$$

Now, we pick two directions p_{1i} & p_{2i}

$$x_i = x_i^0 + \alpha_1 p_{1i} + \alpha_2 p_{2i}$$

$$\phi(\alpha_1, \alpha_2) = \frac{1}{2}\sum a_{ij}(x_i^0 + \alpha_1 p_{1i} + \alpha_2 p_{2i})(x_j^0 + \alpha_1 p_{1j} + \alpha_2 p_{2j})$$
$$- \sum b_i (x_i^0 + \alpha_1 p_{1i} + \alpha_2 p_{2i})$$

$$\frac{\partial \phi}{\partial \alpha_1} = \sum a_{ij} p_{1j} x_i^0 + \alpha_1 \sum a_{ij} p_{1i} p_{1j} + \alpha_2 \sum a_{ij} p_{2i} p_{1j} - \sum b_i p_{1i} = 0 \quad (1)$$

$$\frac{\partial \phi}{\partial \alpha_2} = \sum a_{ij} p_{2j} x_i^0 + \alpha_1 \sum a_{ij} p_{1i} p_{2j} + \alpha_2 \sum a_{ij} p_{2i} p_{2j} - \sum b_i p_{2i} = 0 \quad (2)$$

Set $\sum a_{ij} p_{1i} p_{2j} = 0$ $\quad p_1^T A p_2 = 0 \rightarrow p_1$ & p_2 are conjugate.

Then, from ①

$$\sum a_{ij} p_{1j} x_i^0 + \alpha_1 \sum a_{ij} p_{1i} p_{1j} - \sum b_i p_{1i} = 0$$

$$\rightarrow \alpha_1 = -\sum p_{1i} (\underbrace{a_{ij} x_j^0 - b_i}_{-r_{0i}}) / \sum a_{ij} p_{1i} p_{1j}$$

$$= p_1^T r^0 / p_1^T A p_1$$

from ② $\quad \alpha_2 = \sum p_{2i} r_i^1 / \sum a_{ij} p_{2i} p_{2j} = p_2^T r^1 / p_2^T A p_2$

requiring $\quad p_{k-1} A p_k = 0, \quad p_k = r^{k-1} + \beta_k p_{k-1}$

$$\Rightarrow \beta_k = - p_{k-1} A r^{k-1} / p_{k-1} A p_{k-1}$$

Algorithm: $\quad$ Guess x^0 $\qquad$ Conjugat gradient method

$r^k = -A x^k + b$

if $k=0$, $\quad p_k = r^k$

else, $\quad \beta_k = - p_{k-1} A r^{k-1} / p_{k-1} A p_{k-1}$

$p_k = r^{k-1} + \beta_k p_{k-1}$

$\alpha_k = r^{k-1} r^{k-1} / p_k A p_k$

$x^k = x^{k-1} + \alpha_k p_k$

rate of convergence depends on the condition number of A.

Final exam

Dec. 9 (Wed) 6:30pm

~~Dec. 11~~ ~~10~~

⑨ multigrid method

HW4 on eTL

Final exam | Dec. 14 (Monday) 9:30 am – 11:30 am

HW4 : I changed from multigrid method to CGS for the last prob.
You have to download today.

⑨ Multigrid method (multigrid acceleration)

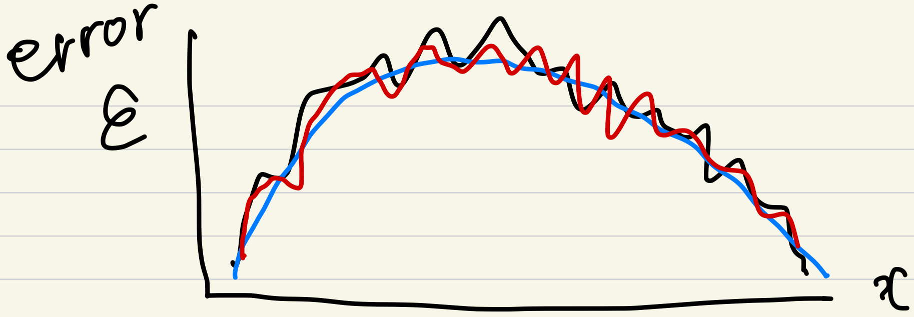
One of the most powerful acceleration schemes for the convergence of iterative methods in solving elliptic probs.

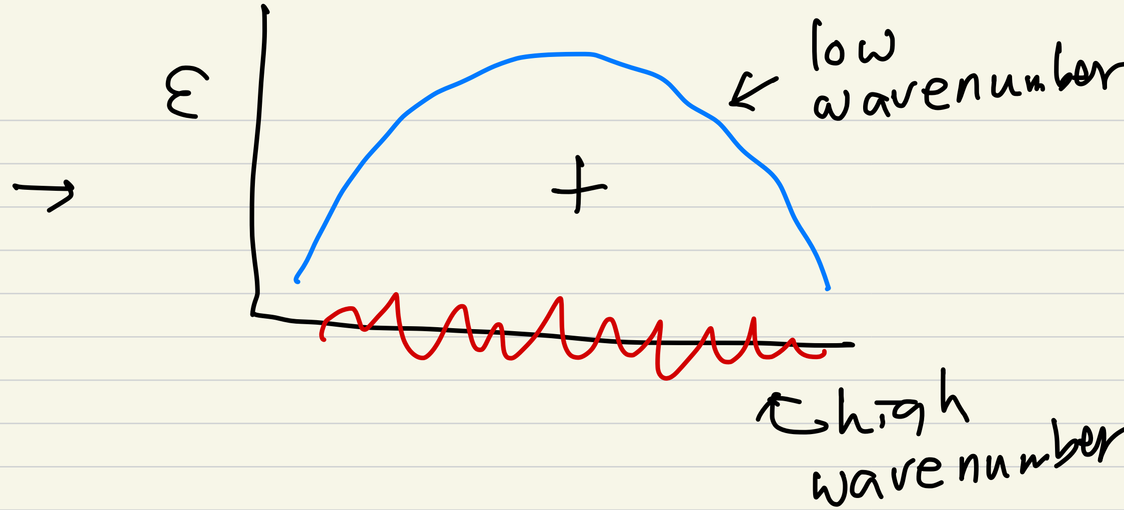
→ different components of the solution converges to the exact solution at different rates and thus should be treated differently.

i.e. smooth component of the residual converges slowly to zero and the rough part converges quickly.

low wavenumber

high wavenumber





$A\phi = b$

ψ : $\psi = \phi^k$ is an approx. to the sol. ϕ after k-th iterations.

r : residual, $r = b - A\psi$

ε : error, $\varepsilon = \phi - \psi$

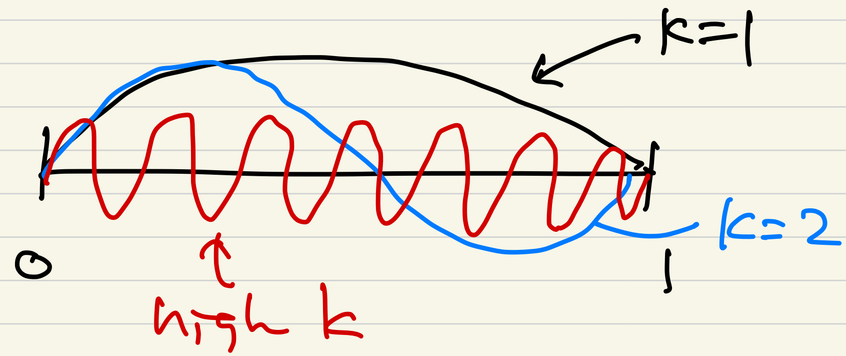
$$A\varepsilon = A(\phi - \psi) = b - A\psi = r \Rightarrow \boxed{A\varepsilon = r}$$

residual eq.

ex) $\dfrac{d^2u}{dx^2} = \sin k\pi x \quad (0 \le x \le 1)$

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

k : wavenumber



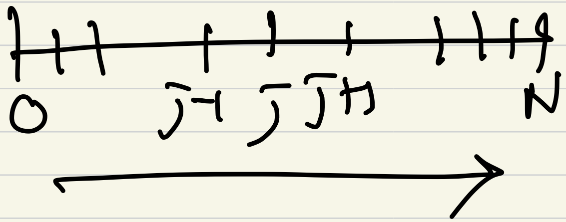
CD2
$$\begin{cases} \dfrac{u_{j+1} - 2u_j + u_{j-1}}{\Delta x^2} = \sin k\pi x_j\,, & j = 1, 2, \cdots, N-1 \\ u_0 = u_N = 0 \end{cases}$$

initial guess : $u^{(0)} = 0\,,\quad r_j^{(0)} = \sin k\pi x_j$

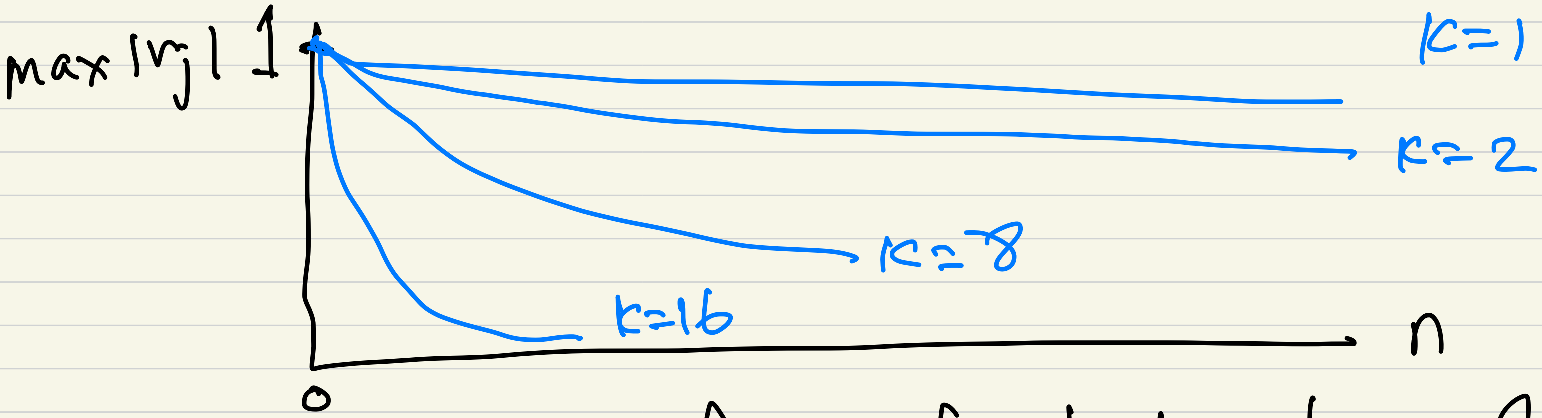
GS : $$\frac{u_{j+1}^{n} - 2u_j^{n+1} + u_{j-1}^{n+1}}{\Delta x^2} = \sin k\pi x_j$$

n: iteration index

$$\rightarrow\ u_j^{n+1} = \frac{1}{2}\left(u_{j+1}^{n} + u_{j-1}^{n+1} - \Delta x^2 \sin(k\pi x_j)\right)$$



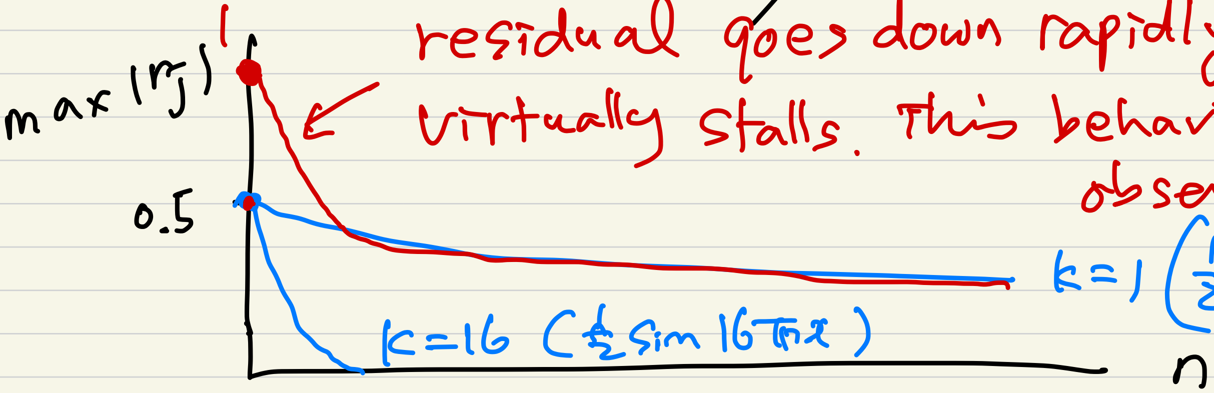
$$r_j^{n} = \sin k\pi x_j - A u_j^{n}$$



convergence is faster for high values of k.

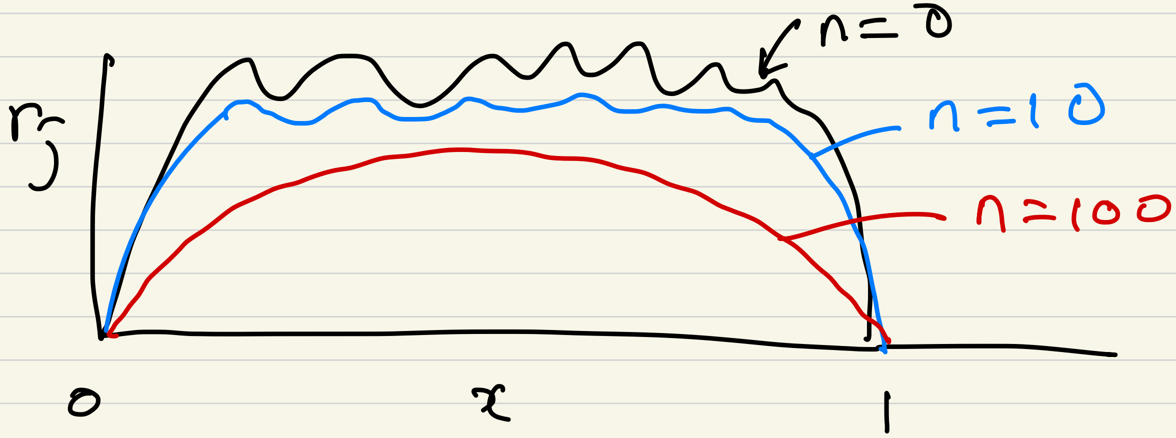
ex) $\dfrac{d^2u}{dx^2} = \dfrac{1}{2}(\sin \pi x + \sin 16\pi x)$ $\quad k = 1$ & 16

CD2



residual goes down rapidly and then virtually stalls. This behavior is frequently observable.

The reason is that the rapidly varying part of the residual goes to zero quickly and the smooth part of it remains.



only smooth part remains after 100 iterations.

many grids are required for high k's, but the convergence is fast for high k's and slow for low k's.

As $N \uparrow$, (due to high k) $|\lambda| \to 1 \Rightarrow$ slow convergence
comes from low k.

reduce N to $N/2 \to |\lambda|$ gets smaller.
for low k, $N/2$ is fine for resolution $\Rightarrow$ fast convergence.

$\Rightarrow$ This is the basic idea of multigrid method.

A. Brandt. Math. Comput. 31. 233 (1977)

$$A\phi = b \qquad A = A_1 - A_2$$

$$A_1\phi^{n+1} = A_2\phi^n + b \qquad n: \text{iteration index}$$

$$-)\ A_1\phi^n = A_1\phi^n$$

$$A_1(\underbrace{\phi^{n+1} - \phi^n}_{\equiv\, \delta\phi^{n+1}}) = (A_2 - A_1)\phi^n + b = -A\phi^n + b = r^n$$

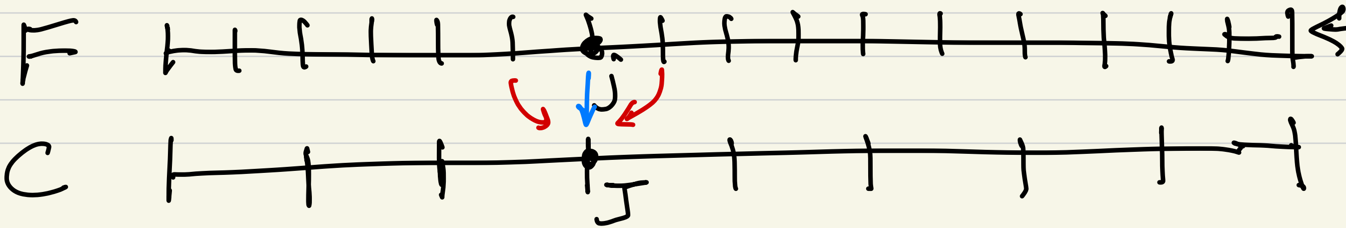
$$\Rightarrow \boxed{A_1 \delta\phi^{n+1} = r^n}$$

procedure :
- compute $r^n = b - A\phi^n$
- solve $A_1\delta\phi^{n+1} = r^n$ to get $\delta\phi^{n+1}$
- update $\phi^{n+1} = \phi^n + \delta\phi^{n+1}$

* Multigrid algorithm

① compute residual $r^n = b - A\phi^n$ on fine (original) grid.

② restrict (smoothen) residual to coarser grid.



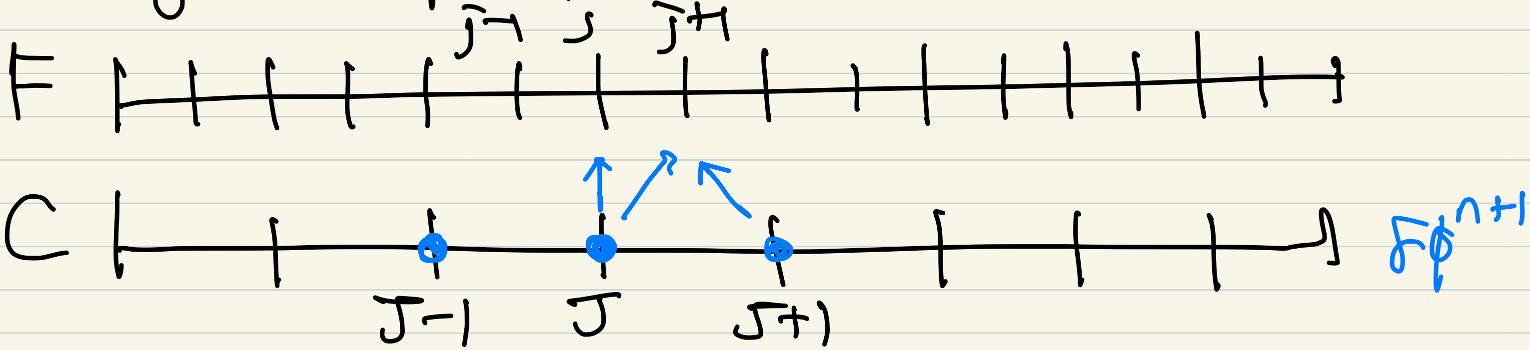
$$r_J = r_j \quad \text{or} \quad r_J = \frac{1}{4}\left(r_{j-1} + 2r_j + r_{j+1}\right)$$

③ iterate $A_1 \delta\phi^{n+1} = r^n$ on coarser grid.

↳ should be reconstructed on coarser grid.

obtain $\delta\phi^{n+1}$ on coarser grid.

④ prolong (interpolate) $\delta\phi^{n+1}$ to fine grid.



j even, $\delta\phi_j^{n+1} = \delta\phi_J^{n+1}$

, $\delta\phi_{j+1}^{n+1} = \frac{1}{2}\left(\delta\phi_J^{n+1} + \delta\phi_{J+1}^{n+1}\right)$

⑤ update $\phi_j^{n+1} = \phi_j^n + \delta\phi_j^{n+1}$ on fine grid, $j = 1, 2, \cdots, N-1$

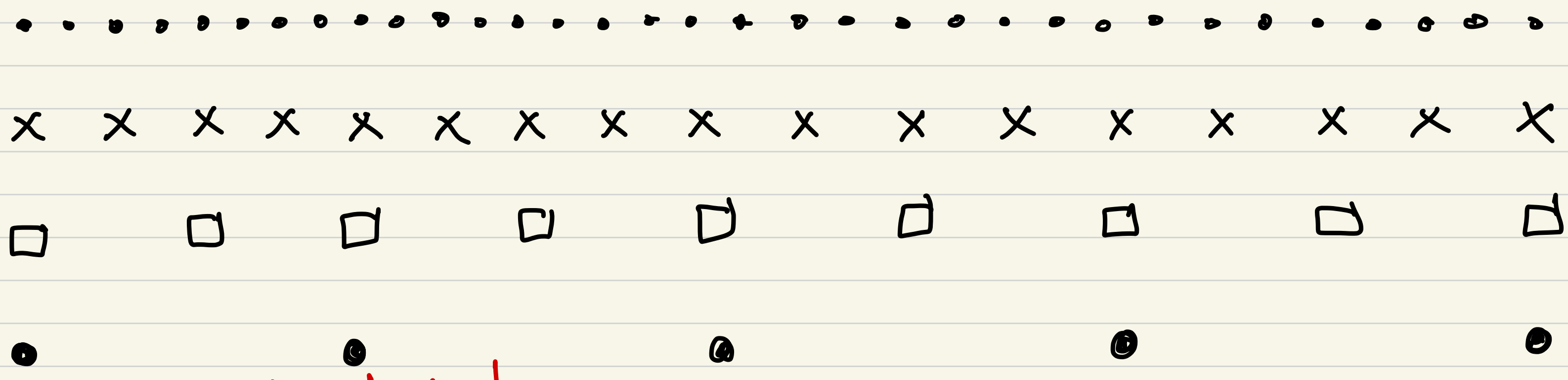
<u>2-level multigrid method.</u>

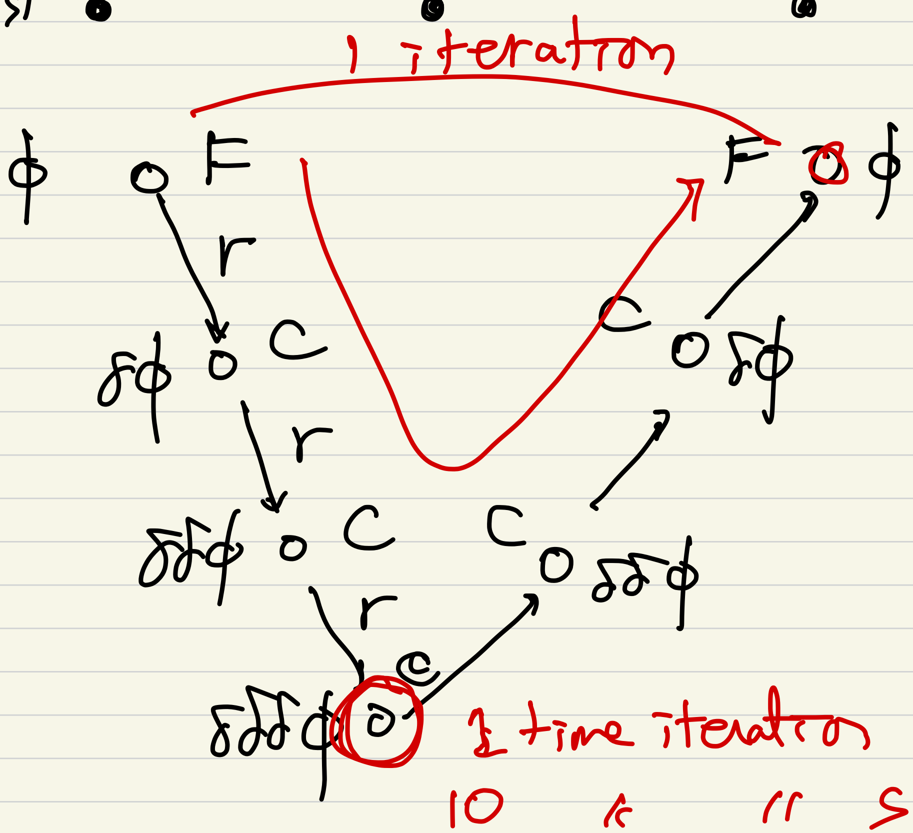
original grid $\quad r^n = b - A\phi^n$

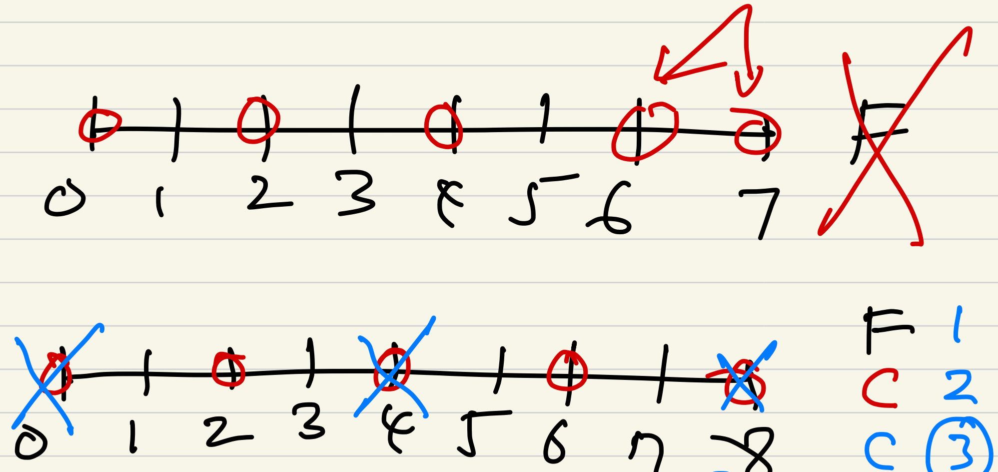
coarse $\quad A_1\delta\phi^{n+1} = r^{n*}$

coarser $\quad A_1\delta\delta\phi^{n+1} = r^{n**}$

coarsest $\quad A_1\delta\delta\delta\phi^{n+1} = r^{n+}$





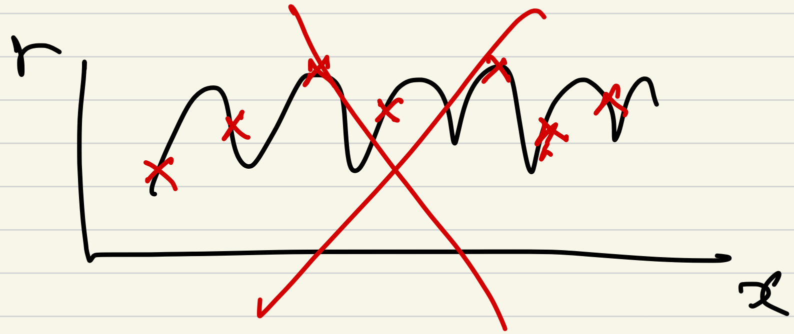


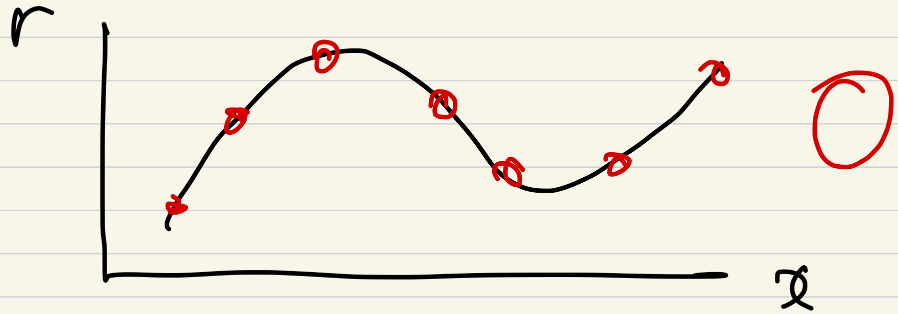
$\hookrightarrow N = 1 + 8 = 1 + 2^3$

$\therefore N = 1 + 2^{a} \times \boxed{}$ → a levels

max.

⇒ We need 'smooth residual distribution' in space.

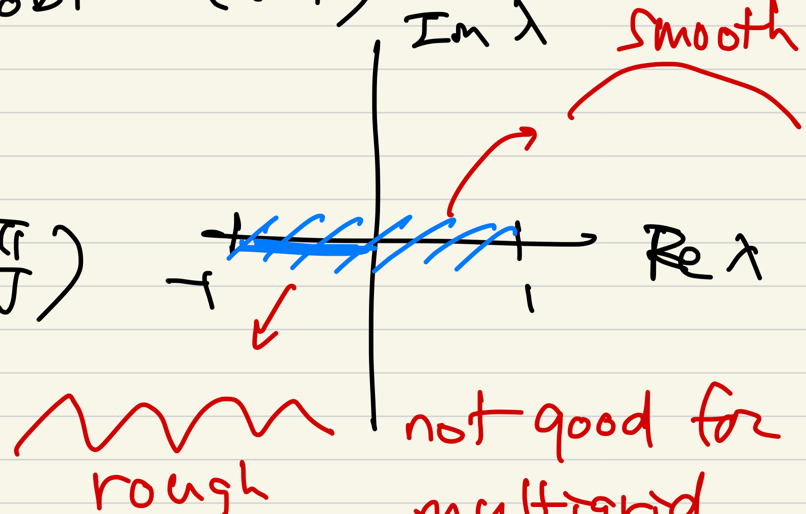




· error (or residual) from Jacobi (A_1)

$$\varepsilon^n = (A_1^{-1}A_2)^n \varepsilon^0$$

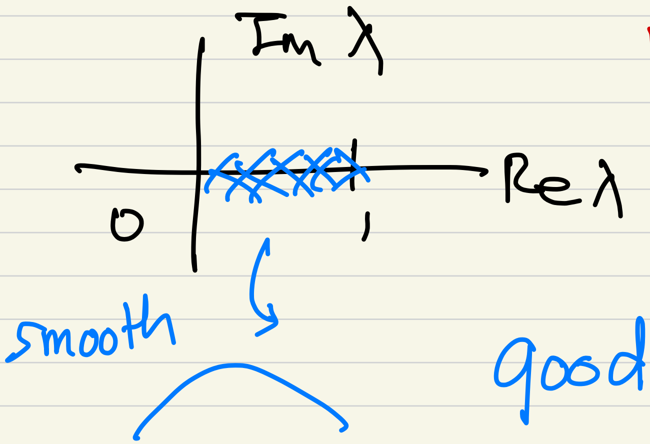
$$\lambda = \frac{1}{2}\left(\cos\frac{l\pi}{M} + \cos\frac{j\pi}{N}\right)$$



not good for multigrid method

· error from GS

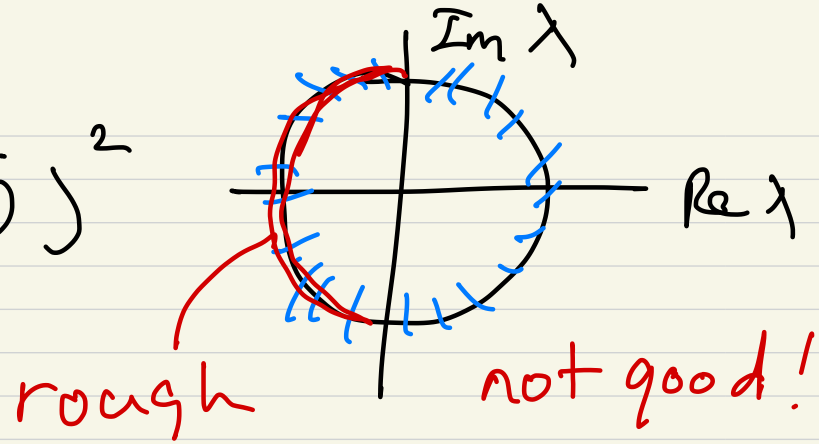
$$\lambda = \frac{1}{4}\left(\cos\frac{l\pi}{M} + \cos\frac{j\pi}{N}\right)^2$$



good!

- error from SOR

$$\lambda = \frac{1}{4}\left(\mu\omega + \sqrt{\mu^2\omega^2 - 4(\omega-1)}\right)^2$$



Method	Error	Solver	Multigrid
Jacobi	rough	bad	X
GS	smooth	bad	O
SOR	rough	good	X
SIP	smooth	good	O
ADI	rough	good	X

recommended.