- Midterm result will be announced tomorrow.
- Final : December 15 (F) 6:30pm – 9:30pm
- HW4 : due date Dec. 18 → Dec. 20
- Supplementary lectures will be given after Wednesday.
- Final day for lecture may be Dec. 11.

5.10.4 Successive over Relaxation (SOR) scheme

$\nabla^2 \phi = f$

Let $d = \phi^{k+1} - \phi^k$

$\phi^{k+1} = \phi^k + d \cdot \omega$ ω: relaxation factor ($\omega > 1$)

($\omega < 1$: under relaxation for nonlinear eq.)

GS :

$$(D-L)\phi^{k+1} = U\phi^k + b$$

$$D\phi^{k+1} = L\phi^{k+1} + U\phi^k + b$$

$$\phi^{k+1} = D^{-1}L\phi^{k+1} + D^{-1}U\phi^k + D^{-1}b$$

SOR: $\phi^{k+1} = \phi^k + [D^{-1}L\phi^{k+1} + D^{-1}U\phi^k + D^{-1}b - \phi^k]\omega$

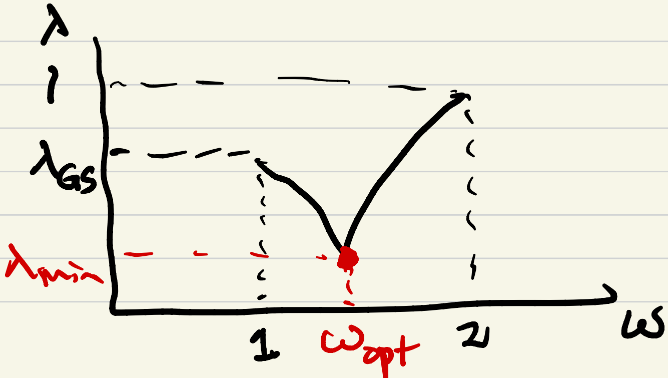
$\rightarrow (I - \omega D^{-1}L)\phi^{k+1} = [(1-\omega)I + \omega D^{-1}U]\phi^k + \omega D^{-1}b$

$$\rightarrow \boxed{\phi^{k+1} = \underbrace{(I-\omega D^{-1}L)[(1-\omega)I + \omega D^{-1}U]}_{G_{SOR} = A_1^{-1}A_2}\phi^k + (I-\omega D^{-1}L)\omega D^{-1}b}$$

for Poisson operator, $\lambda_{SOR} = \frac{1}{4}\left(\mu\omega + \sqrt{\mu^2\omega^2 - 4(\omega-1)}\right)^2$

where $\mu = \lambda_J \qquad \lambda_J = \frac{1}{2}\left(\cos\frac{l\pi}{M} + \cos\frac{j\pi}{N}\right)$

optimum ω? $\underline{\text{minimize } \lambda} \Rightarrow \frac{d\lambda}{d\omega} = 0 \rightarrow$ no sol.



$\frac{d\lambda}{d\omega} \rightarrow \infty \Rightarrow \omega_{opt} = \frac{2}{1+\sqrt{1-\mu_{max}^2}}$

As M & $N \uparrow$, $\mu_{max} \rightarrow 1$

$\Rightarrow \omega_{opt} \rightarrow 2$

$\omega = 1$: $\lambda_{SOR} = \mu^2 = \lambda_J^2 = \lambda_{GS}$

$\omega = 2$: $\lambda_{SOR} = \frac{1}{4}\left(2\mu + \sqrt{4\mu^2 - 4}\right)^2 = \frac{1}{4}\left(2\mu + i\sqrt{4(1-\mu^2)}\right)^2$

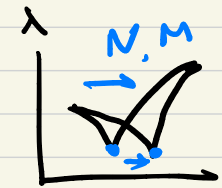
$\rightarrow |\lambda_{SOR}|^2 = \cdots = 1.$

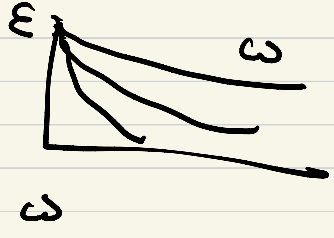
$\boxed{1 < \omega < 2 \text{ for SOR}}$ $\rightarrow \omega_{opt} = 1.7 \sim 1.9$

For problems w/ irregular geometries and non-uniform mesh, ω_{opt} cannot be obtained analytically, and thus ω_{opt} must be found by numerical experiments.

$\boxed{\nabla^2\phi = f}$

M	N	μ_{max}	ω_{opt}
10	10	↓ increases	1.2
100	100		1.7





ε after 100 iterations :

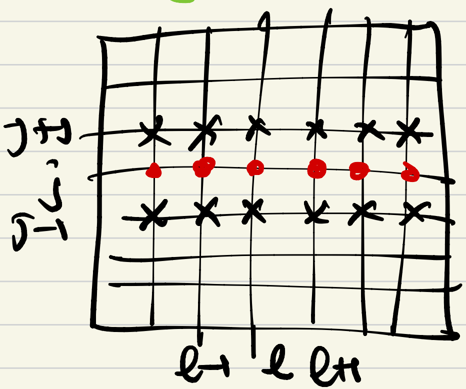
Jacobi	0.887
GS	0.786
SOR w/ $\omega = 1.75$	5×10^{-7}

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



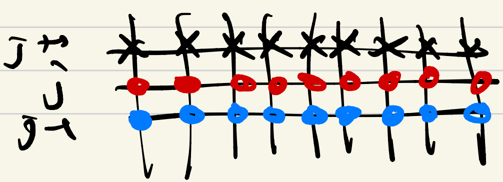
× : k-th iteration

● : k+1 th iteration

faster than point Jacobi

⑤ line GS method

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



● : recently updated ones.

faster than point GS

⑥ ADI method using artifical derivative term

$\nabla^2 \phi = b$

$\frac{\partial \phi}{\partial t} = \nabla^2 \phi - b$

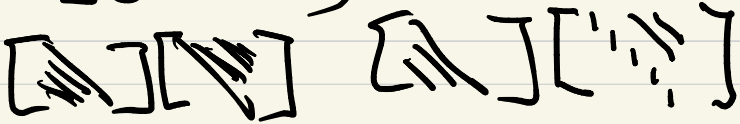
EE = Jacobi ×
IE − ○
CN − △
} not very popular

⑦ strongly implicit procedure (SIP)

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

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

$\underline{A} \parallel LU$ $O(M^3 N^3)$



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

incomplete
LU decomposition
(ILU)

⑧ Conjugate Gradient Solver (CGS)

Hesteness & Stiefel (1952) Nat. Bur. Stand J. Res. <u>49</u>, 409.

Kershaw (1978) J. Comp. Phys. <u>26</u>, 43.

— simple gradient method (steepest descent method)

— conjugate gradient method

For a symmetric matrix $\underline{A}$ $(a_{ij} = a_{ji})$

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

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

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

$$\rightarrow \Sigma a_{kj} x_j = b_k \Rightarrow Ax = b \Rightarrow x = A^{-1} b$$

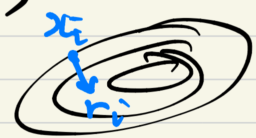
∴ minimizing ϕ & solving $Ax = b$ are equivalent problems.

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.

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

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



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

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

$$= \alpha\sum a_{ij} r_i r_j + \sum a_{ij} x_j r_i - \sum b_i r_i$$

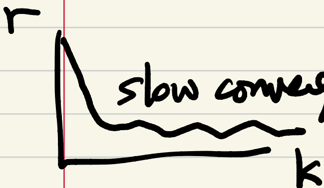
$$= \alpha\sum a_{ij} r_i r_j + \sum r_i (\underbrace{a_{ij} x_j - b_i}_{= -r_i}) = 0$$

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

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

$r_k \neq 0$. $k = k+1$ iteration

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



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

steepest descent method

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

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

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

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

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,

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

$$\rightarrow \alpha_1 = - \sum P_{1i} \underbrace{(a_{ij} x_j^0 - b_i)}_{= -r_i^0} / \sum a_{ij} P_{1i} P_{1j}$$
$$= + P_1^T r^0 / P_1^T A P_1$$

Also, from ②, $\alpha_2 = - \sum P_{2i} r_i^0 / \sum a_{ij} P_{2i} P_{2j} = + P_2^T r^0 / P_2^T A P_2$

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

Multiplying $p_{k-1} A$ on both sides

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

Algorithm : Guess $x^{\wedge}$

$$r^k = -Ax^k + b$$

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

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

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

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

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

iterate

conjugate gradient method

⑨ Multigrid method (multigrid acceleration)

One of the most powerful acceleration scheme for the convergence of iterative methods in solving elliptic PDEs.

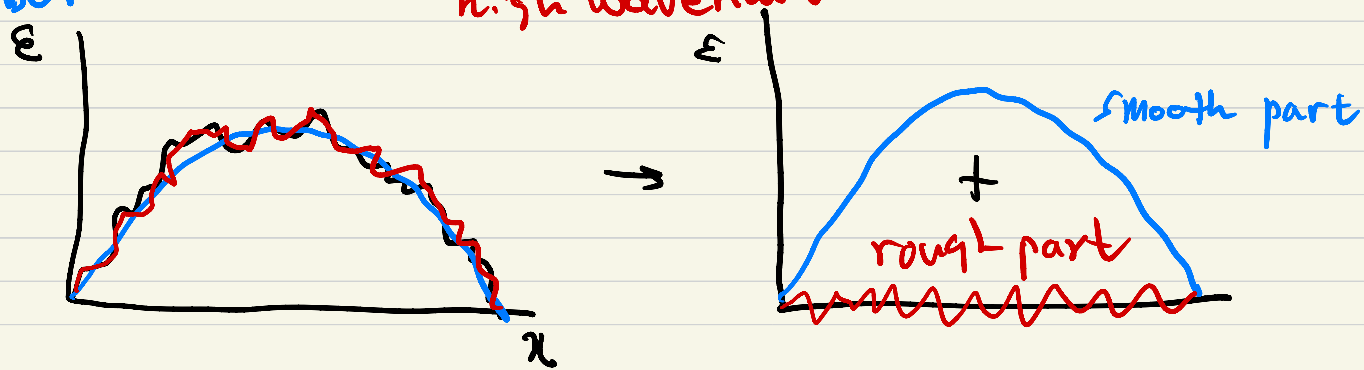
→ different components of the solution converge to the exact solution at different rates and thus should be treated differently.

$r = b - A\phi$

i.e. smooth components of the residual converge slowly to zero and the rough parts converge quickly.

low wavenumber

high wavenumber



$A\phi = b$

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

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

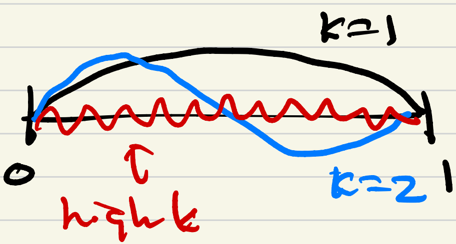
ε : error $\quad \epsilon = \phi - \psi$

$A\varepsilon = A(\phi - \psi) = b - A\psi = r \rightarrow \boxed{A\varepsilon = r}$ residual eq.

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

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

k : wavenumber



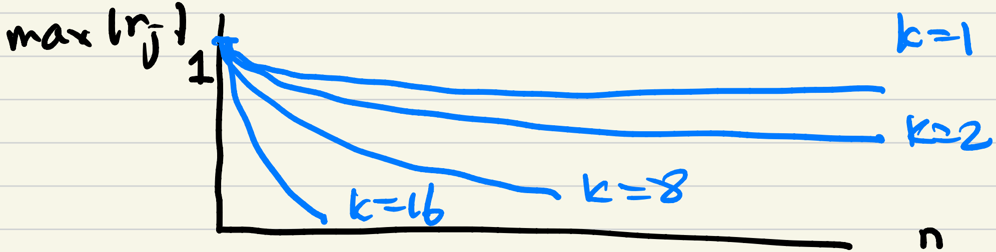
CD2 : $\frac{u_{j+1} - 2u_j + u_{j-1}}{\Delta x^2} = \sin k\pi x_j \quad , \; j = 1, 2, \cdots, N-1$

$u_0 = u_N = 0$

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

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

$r_j^n = \sin k\pi x_j - A u_j^n = \sin k\pi x_j - \dfrac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{\Delta x^2}$



initial guess :

$u^{(0)} = 0$

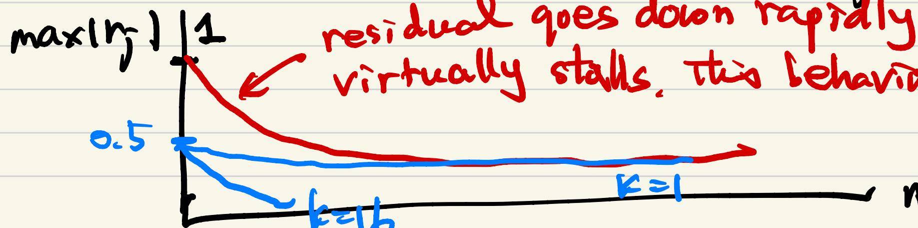
$r_j^{(0)} = \sin k\pi x_j$

convergence is faster for higher k.

ex) $\dfrac{d^2u}{dx^2} = \frac{1}{2}(\sin \pi x + \sin 16\pi x)$ k=1 & 16 $\left(\begin{array}{l} u^{(0)} = 0 \\ r_j^{(0)} = \frac{1}{2}(\sin\pi x + \sin 16\pi x) \end{array} \right.$

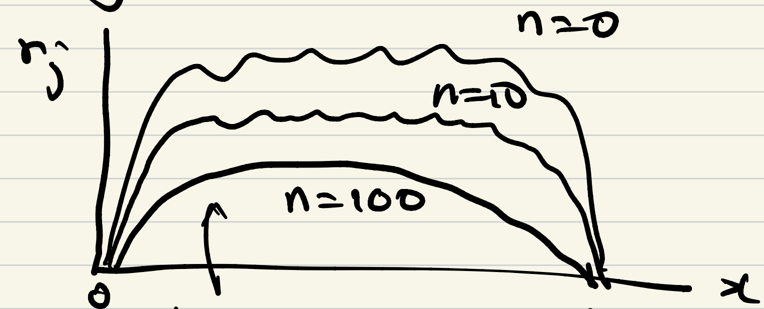
CD2

GS



residual goes down rapidly and then virtually stalls. this behavior is frequently observed.

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↑ (due to high k) $|\lambda| \to 1$ $\Rightarrow$ slow convergence
↑ due to low k.

reduce N to N/2 $\longrightarrow$ $|\lambda|$ becomes smaller → fast convergence
← this is fine for low k

→ this is the basic idea of multigrid method.

A. Brandt. Math. Comput. **31**, 333 (1977).

$A\phi = b \quad (A = A_1 - A_2)$

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

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

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

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

procedure : compute $r^n = b - A\phi^n$ $\quad$ iterate

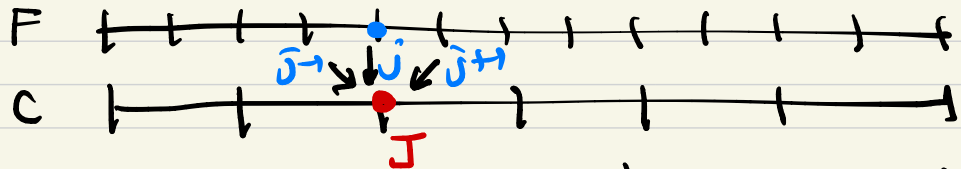
solve $A_1\delta\phi^{n+1} = r^n$ to get $\delta\phi^{n+1}$.

update $\phi^{n+1} = \phi^n + \delta\phi^{n+1}$

- <u>Multigrid algorithm</u>

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

② restrict (smoothen) residual to coarser grid



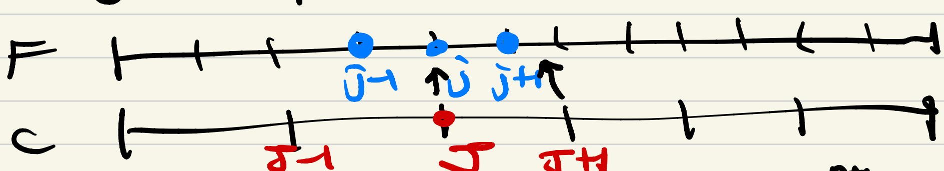
$r_J = r_j$ or $r_J = \frac{1}{4}(r_{j-1} + 2r_j + r_{j+1})$

③ iterate $\underline{A_1}\, \delta\phi^{n+1} = r^n$ 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.

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

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

⑤ update $\phi_j^{n+1} = \phi_j^{n} + \delta\phi_j^{n+1}$ on fine grid.

(2-level multigrid method.)

maximum α levels

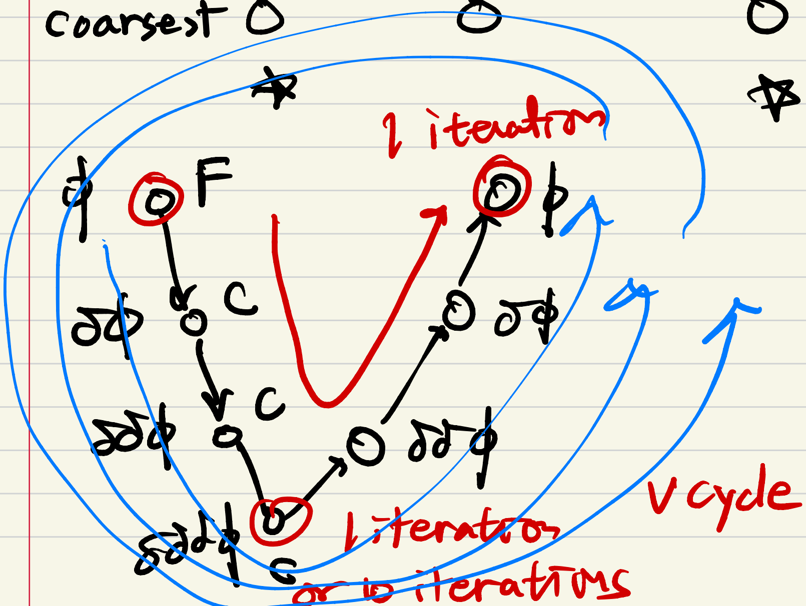
$N = 1 + 2^{\alpha} \times \square$

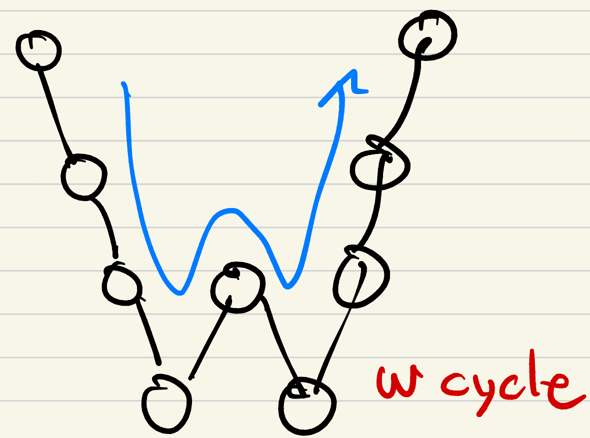
original • $r^n = b - A\phi^n$

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

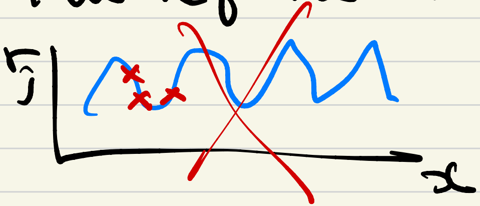
coarser □ □ □ □ □ □ □ □ □ $A_1 \delta\delta\phi^{n*} = r^*$

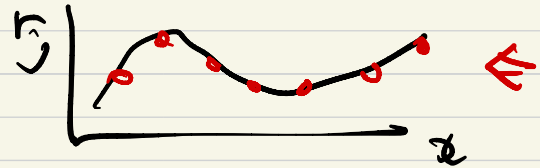
coarsest ○ ○ ○ ○ ○ $A_1 \delta\delta\delta\phi^{n+1} = r^*$





⇒ This requires 'smooth residual distribution' on the domain.

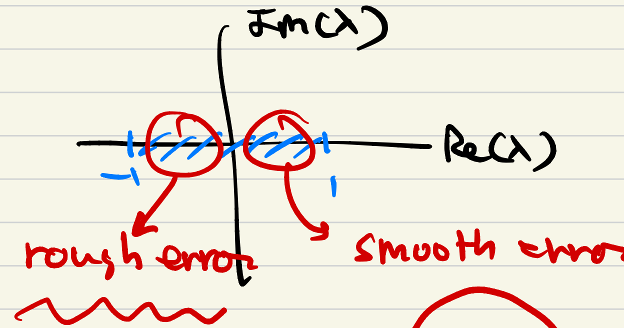




- error (or residual) from Jacobi:

$$\varepsilon^n = \underbrace{(A_1^{-1} A_2)^n}\,\varepsilon^0 \qquad \lambda^n$$

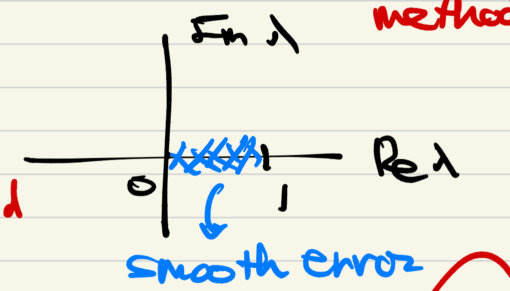
$$\lambda_J = \frac{1}{2}\left(\cos\frac{l\pi}{M} + \cos\frac{j\pi}{N}\right)$$



not good for multigrid method.

- error from GS

$$\lambda_{GS} = \lambda_J^2 = \frac{1}{4}\left(\cos\frac{l\pi}{M} + \cos\frac{j\pi}{N}\right)^2 > 0$$

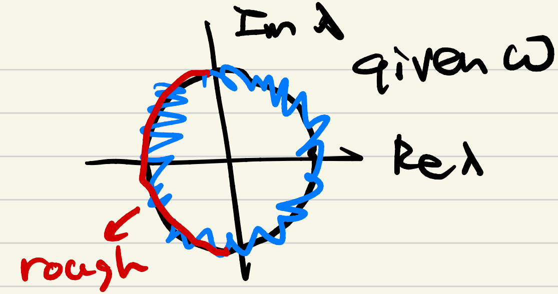


good for multigrid method

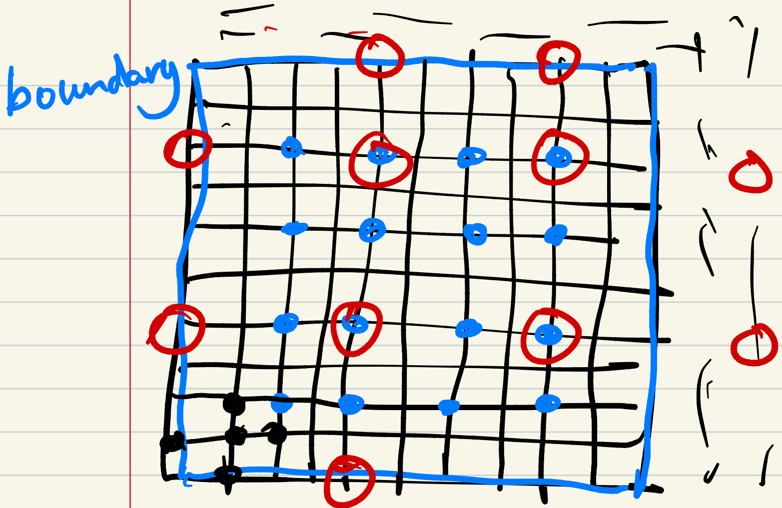
- error from SOR

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

Not good for multigrid



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



- fine grid
- 2nd level
- third level

$A\phi = b$

$\phi_{i,j}$ $\phi_{i+1,j}$ $\phi_{i-1,j}$ $\phi_{i,j+1}$, $\phi_{i,j-1}$

single grid transfers the information to an adjacent grid per iteration,

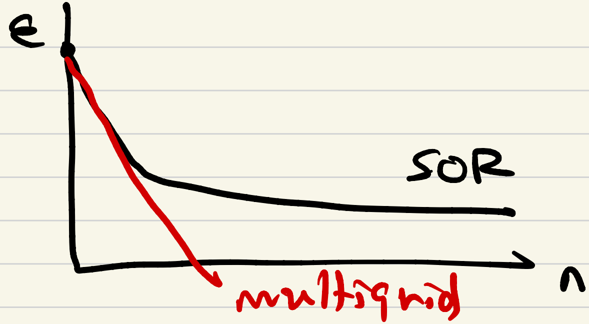
while multigrid " " " " to all grids per iteration.

As $N \uparrow$, $|\lambda| \to 1$ $\Rightarrow$ slow convergence

in multigrid, $N \to N/2 \to N/4 \to N/8 \to N/16 \to \cdots$

$\Downarrow$

$|\lambda| \downarrow$ $(|\lambda| \ll 1)$



- Convergence rate depends on smoother, cycle, ...
- Number of cycle iteration for convergence is in principle indep. of N.