

Multigrid method

노트 제목

2019-12-04

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

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

$$\underbrace{A_1 \phi^{n+1} = A_1 \phi^n}$$

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

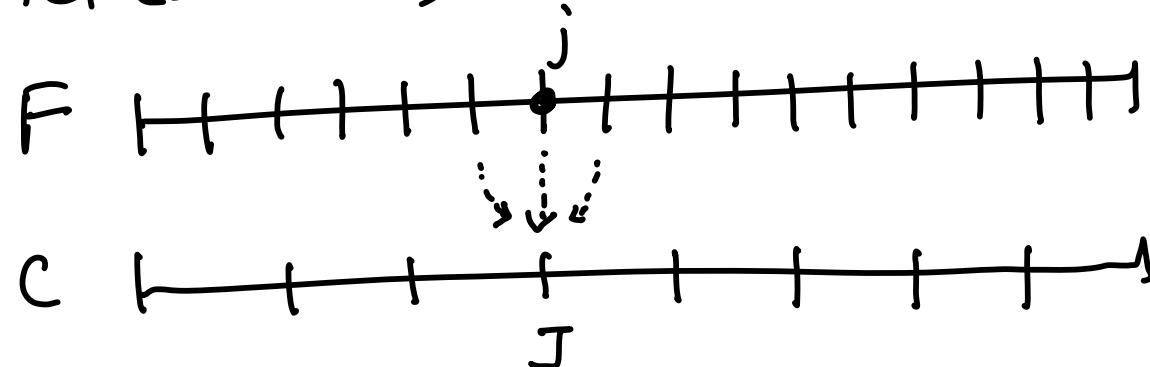
$$\boxed{A_1 \delta \phi^{n+1} = r^n}$$

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

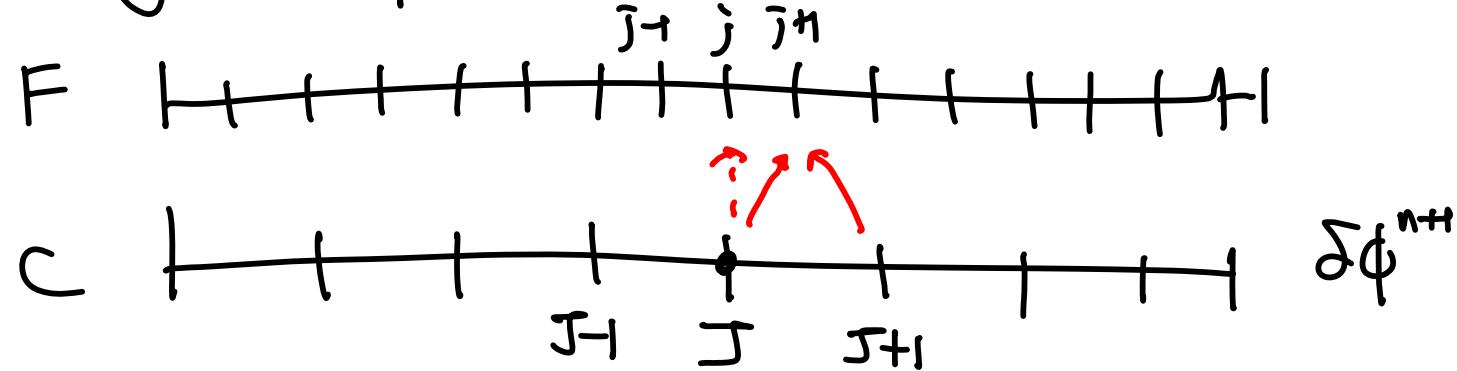


$$r_J = r_j \quad \text{or} \quad r_J = \frac{1}{k} (r_{j-1} + 2r_j + r_{j+1})$$

- ③ iterate $A_J \delta\phi^{n+1} = r^n$ on coarser grid
 $\tilde{\epsilon}$ should be reconstructed

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

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



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

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

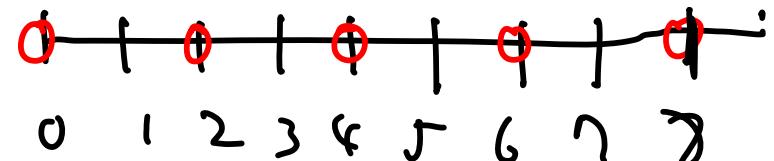
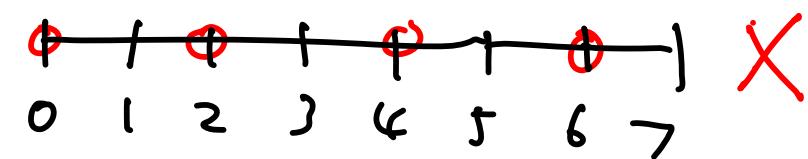
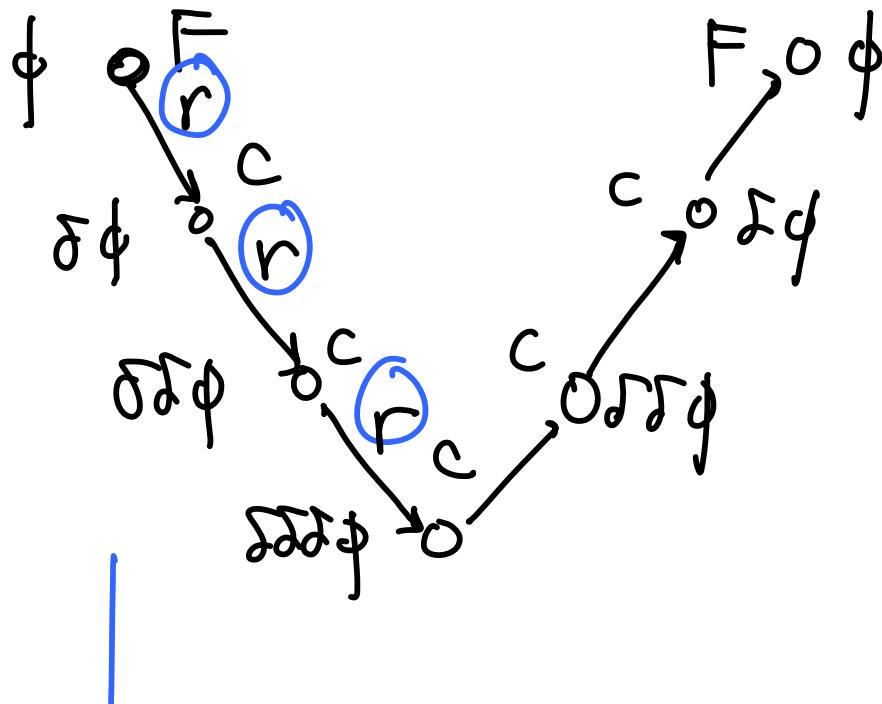
⑤ update $\phi_j^{n+1} = \phi_j^n + \delta\phi_j^{n+1}$ on fine grid. $j=1, 2, \dots, N_f$

$$r^n = b - A \phi^n$$

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

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

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

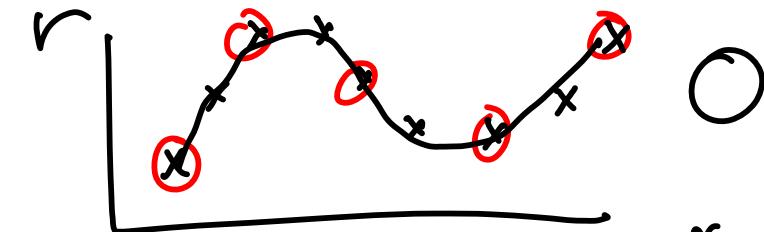
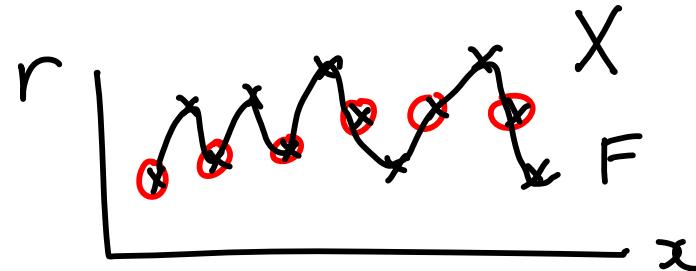


$$\hookrightarrow N = 1 + 2 = 1 + 2^3$$

$$N = 1 + [2^a \times 3^b] \times \dots$$

|

↳ We need "smooth residual distribution" in space.

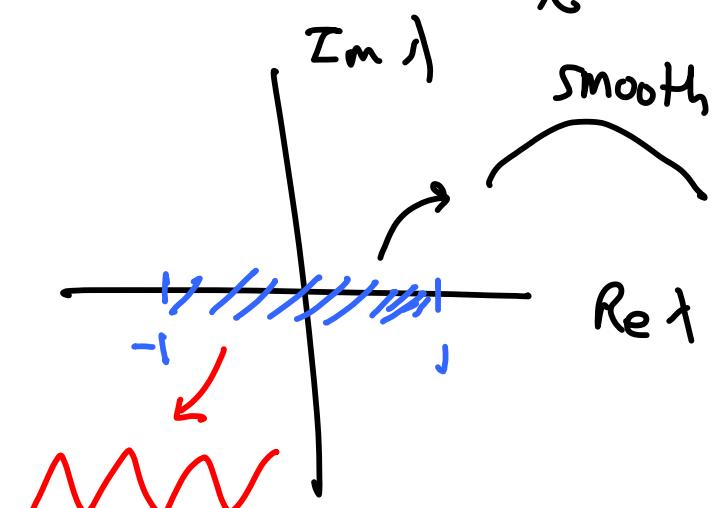


- error (or residual) from Jacobi:

$$\underbrace{\varepsilon^1}_{\text{wavy}} = (\underbrace{A_1^{-1} A_2}_\lambda)^1 \varepsilon^0$$

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

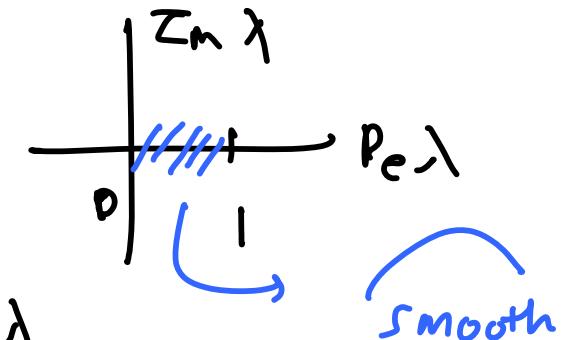
Jacobi is bad for multigrid method
due to this error behavior.



- error from GS

$$\lambda = \lambda_J^2 = \frac{1}{4} \left(\cos \frac{i\pi}{N} + \cos \frac{j\pi}{N} \right)^2 > 0$$

Good!

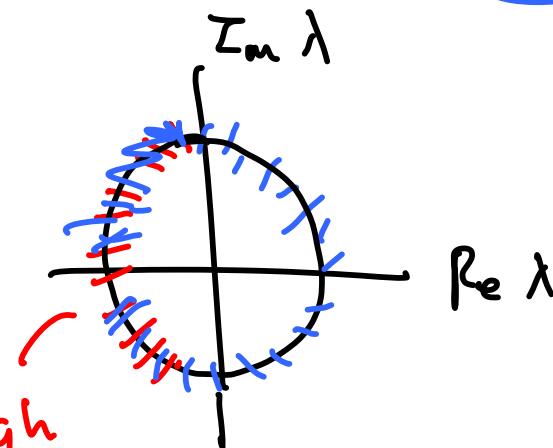


- error from SOR

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

bad!

rough



*	Method	Error	Solver	Multigrid
*	Jacobi	rough	bad	X
*	GS	smooth	bad	O

SOR

rough

good

X

SIP

smooth

good

O

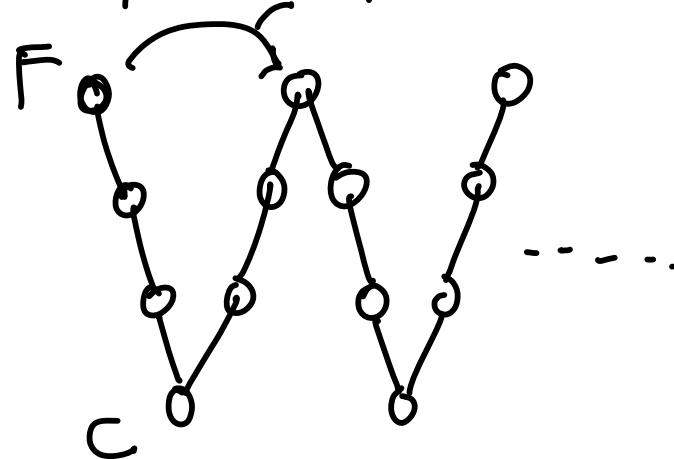
ADI

rough

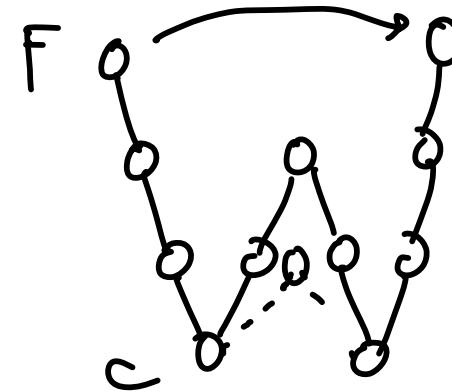
good

X

V cycle (cycle (iteration))



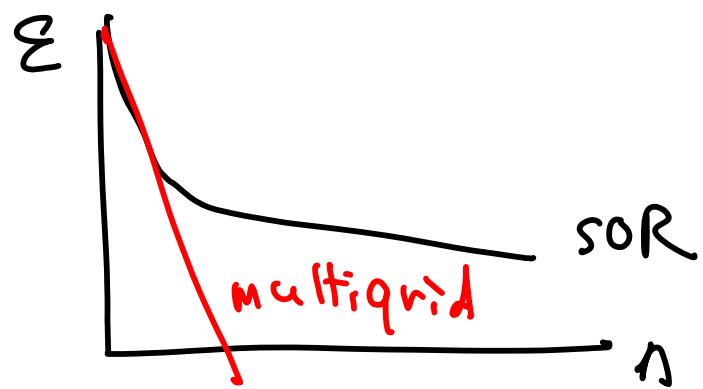
W cycle (cycle (iteration))



- Single grid transfers the information to an adjacent grid per iteration.
- Multigrid " " to all grids per iteration.

as $N \uparrow$, $\lambda \rightarrow 1 \Rightarrow$ slow convergence

in Multigrid method, $N \rightarrow N/2 \rightarrow N/4 \rightarrow N/8 \rightarrow N/16 \dots$
 $\Rightarrow \lambda \downarrow (\lambda \ll 1)$



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

(0)

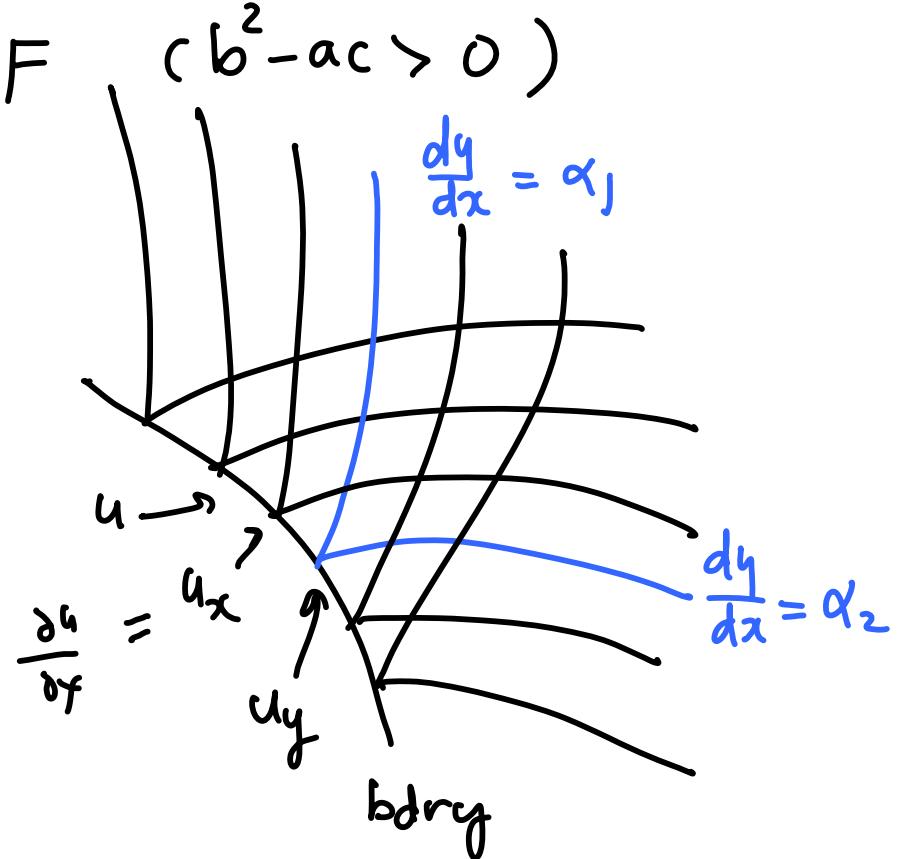
Hyperbolic PDE

$$a \frac{\partial^2 u}{\partial x^2} + 2b \frac{\partial^2 u}{\partial x \partial y} + c \frac{\partial^2 u}{\partial y^2} = F \quad (b^2 - ac > 0)$$

characteristic lines

$$\frac{dy}{dx} = \frac{b \pm \sqrt{b^2 - ac}}{a} = \alpha_1, \alpha_2$$

We need to know locations of char. lines (x, y) and variations of u_x and u_y on char. lines.



Along any diff'l line element (dx, dy)

$$\therefore d\left(\frac{\partial u}{\partial x}\right) \equiv du_x = \frac{\partial^2 u}{\partial x^2} dx + \frac{\partial^2 u}{\partial x \partial y} dy$$

$$d\left(\frac{\partial u}{\partial y}\right) \equiv du_y = \frac{\partial^2 u}{\partial x \partial y} dx + \frac{\partial^2 u}{\partial y^2} dy$$

\rightarrow GE $a u_{xx} + 2b u_{xy} + c u_{yy} = F$

$\dashrightarrow dx u_{xx} + dy u_{xy} = du_x$

$dx u_{xy} + dy u_{yy} = du_y$

a	2b	c
dx	dy	0
0	dx	dy

$\Rightarrow 0 \rightarrow$ eq. for char. lines

$\frac{dy}{dx} = \alpha_1, \alpha_2$

For the existence of u_{xx} , u_{xy} & u_{yy} ,

$$\begin{vmatrix} a & F & C \\ du_x & 0 & 0 \\ 0 & du_y & dy \end{vmatrix} = 0$$

$$\rightarrow dy(adu_x - Fdx) - du_y(-cdx) = 0$$

$$\rightarrow \frac{dy}{dx} adu_x + cd u_y - F dy = 0$$

char. lines $\frac{du}{dx} = \alpha_1, \alpha_2$

for $\frac{dy}{dx} = \alpha_1, \alpha_1 adu_x + cd u_y - F dy = 0$

$= \alpha_2, \alpha_2 adu_x + cd u_y - F dy = 0$

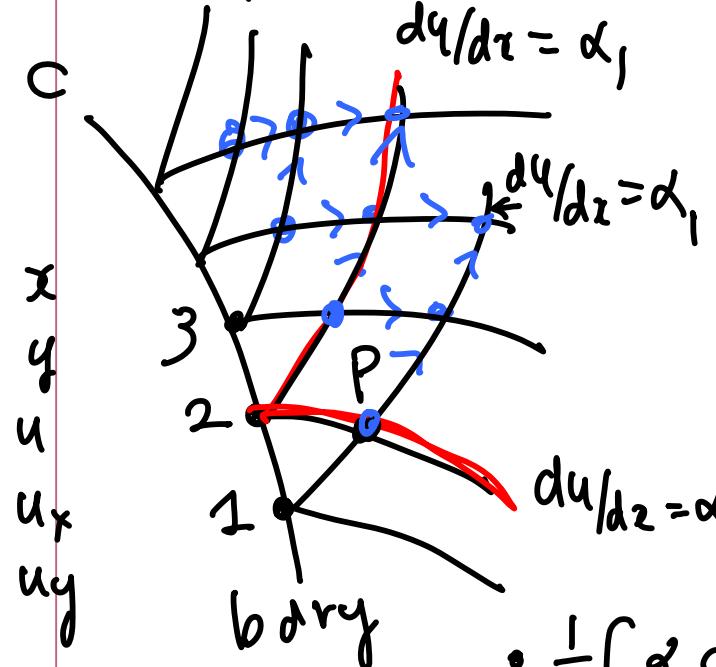
from chain rule, $du = u_x dx + u_y dy$

5 eqs.
for 5
unknowns
 $(\begin{matrix} x & y & u \\ u_x & u_y \end{matrix})$

PDE

→ ODEs : method of characteristics

(MOC)



$$\frac{dy}{dz} = \alpha_1 : \frac{y(P) - y(I)}{x(P) - x(I)} = \frac{1}{2} [\alpha_1(P) + \alpha_1(I)]$$

$$\frac{dy}{dz} = \alpha_2 : \frac{y(P) - y(2)}{x(P) - x(2)} = \frac{1}{2} [\alpha_2(P) + \alpha_2(2)]$$

$$+ \frac{1}{2} [\alpha_1(P) \alpha_1(P) + \alpha_1(I) \alpha_1(I)] (u_x(P) - u_x(I))$$

$$+ \frac{1}{2} [c(P) + c(I)] (u_y(P) - u_y(I)) - \frac{1}{2} [F(P) + F(I)] (y(P) - y(I)) = 0$$

- $\frac{1}{2} (\alpha_2(p) \alpha(p) + \alpha_2(2) \alpha(2)) (u_x(p) - u_x(2))$
 $+ \frac{1}{2} (c(p) + c(2)) (u_y(p) - u_y(2)) - \frac{1}{2} (f(p) + f(2)) (y(p) - y(2)) = 0$
- $(u(p)) - u(1) = \frac{1}{2} (u_2(p) + u_x(1)) (x(p) - x(1))$
 $+ \frac{1}{2} (u_y(p) + u_y(1)) (y(p) - y(1))$

→ In general, these are nonlinear → iterative method.

⑤ Advantage of MoC

- important properties of exact sol. are preserved in numerical sol.
- Method is easily adapted to the computation of problems

that contain discontinuities.

- ability to compute the sol. over a long span of indep. variables.

Disadvantages of MoC

- difficulties of keeping track of the locations of the characteristic lines and the values of variables in 3D.
- difficulties in handling mixed-type PDE.
(e.g. hyperbolic in one area and elliptic in other area)

① Explicit methods for hyperbolic eqs.

convection eq. $\frac{\partial \phi}{\partial t} + c \frac{\partial \phi}{\partial x} = 0 \quad x - ct = \text{const}$

wave eq $\frac{\partial^2 \phi}{\partial t^2} = c^2 \frac{\partial^2 \phi}{\partial x^2} \quad \begin{cases} x - ct = \text{const} \\ x + ct = \text{const} \end{cases}$

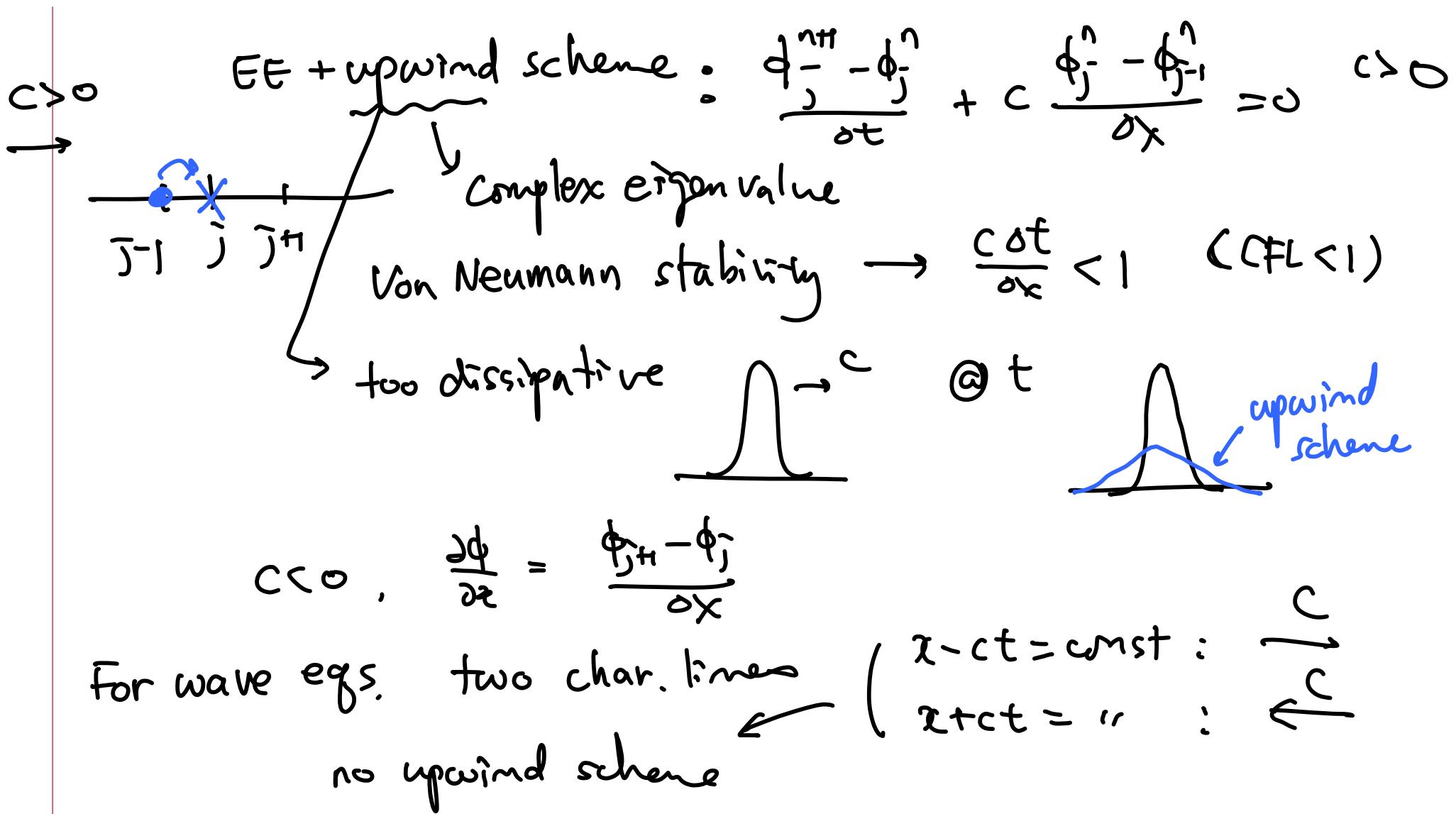
$$\begin{array}{c} u = \phi \\ w = u' \end{array} \rightarrow \begin{cases} \frac{\partial u}{\partial t} = c \frac{\partial w}{\partial x} \\ \frac{\partial w}{\partial t} = c \frac{\partial u}{\partial x} \end{cases}$$

For $\frac{\partial \phi}{\partial t} + c \frac{\partial \phi}{\partial x} = 0$

EE + CP2:



leapfrog: stable, no amp. error, spurious root
 $CCFL < 1$



CP scheme + E \in → unstable

+ leapfrog → stable for CFL < 1
spurious root

+ RK4 → conditionally stable