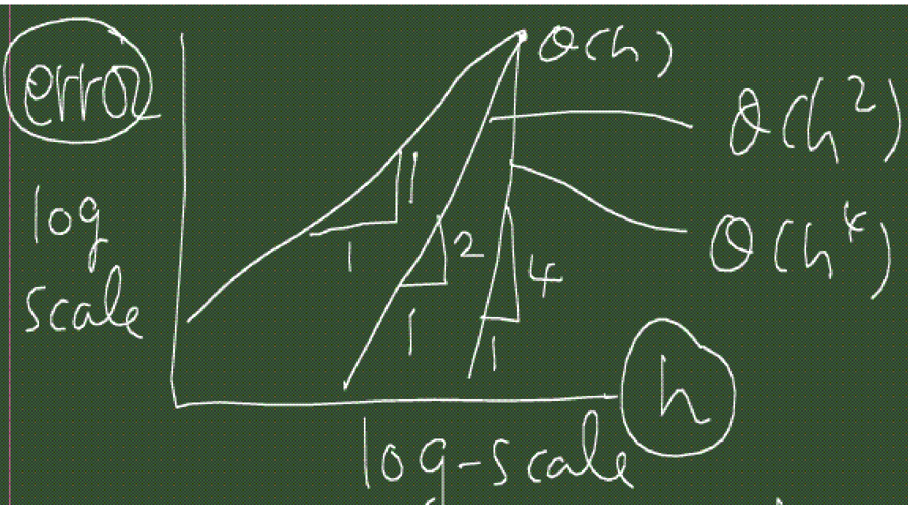
## 2.3 An alternative measure for the accuracy of FD.

As $h \to \frac{h}{2}$, $O(h^2)$

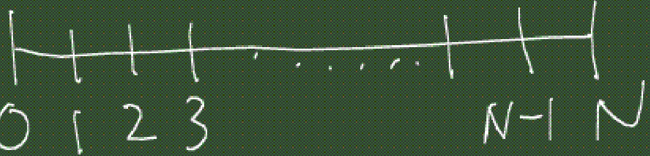
2nd order accuracy: error $\to \frac{1}{4}$

4th " " $\to \frac{1}{16}$

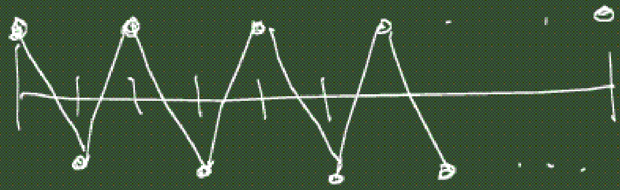



- Modified wavenumber approach for measuring the order of accuracy ex. $f(x) = e^{ikx} \to f' = ike^{ikx} = \boxed{ikf}$

$0 \quad L$




$\Delta x = L/N$



$K \cdot (2\Delta x) = 2\pi$

$e^{ikx}$

$K \cdot (2\Delta x) = 2\pi$

$$K = \frac{2\pi}{2\Delta x} = \frac{\pi}{\Delta x} = \frac{\pi N}{L}$$ : largest wavenumber

smallest wavenumber

$\Delta k \cdot L = 2\pi \to \Delta k = 2\pi / L$

$k = \frac{2\pi}{L} \cdot n, \quad n = 0, 1, 2, \dots, \frac{N}{2}$

Q: how accurately 2nd-order CD schemes computes the derivatives of f for different values of k?

CD2: $\left.\frac{\delta f}{\delta x}\right|_j = \frac{f_{j+1} - f_{j-1}}{2h}$, $h = \frac{L}{N}$

$f_j = e^{ikx_j}$

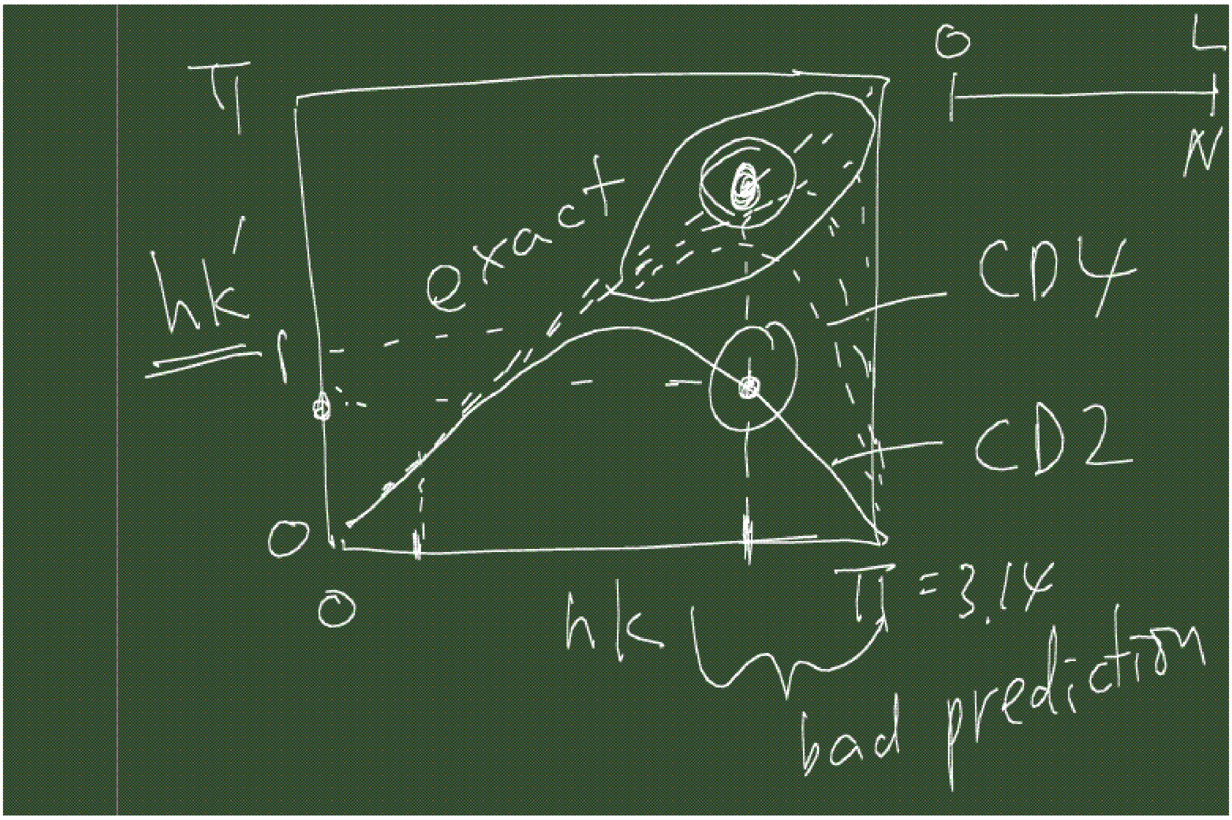
$$= \frac{1}{2h}\left(e^{ikx_{j+1}} - e^{ikx_{j-1}}\right)$$

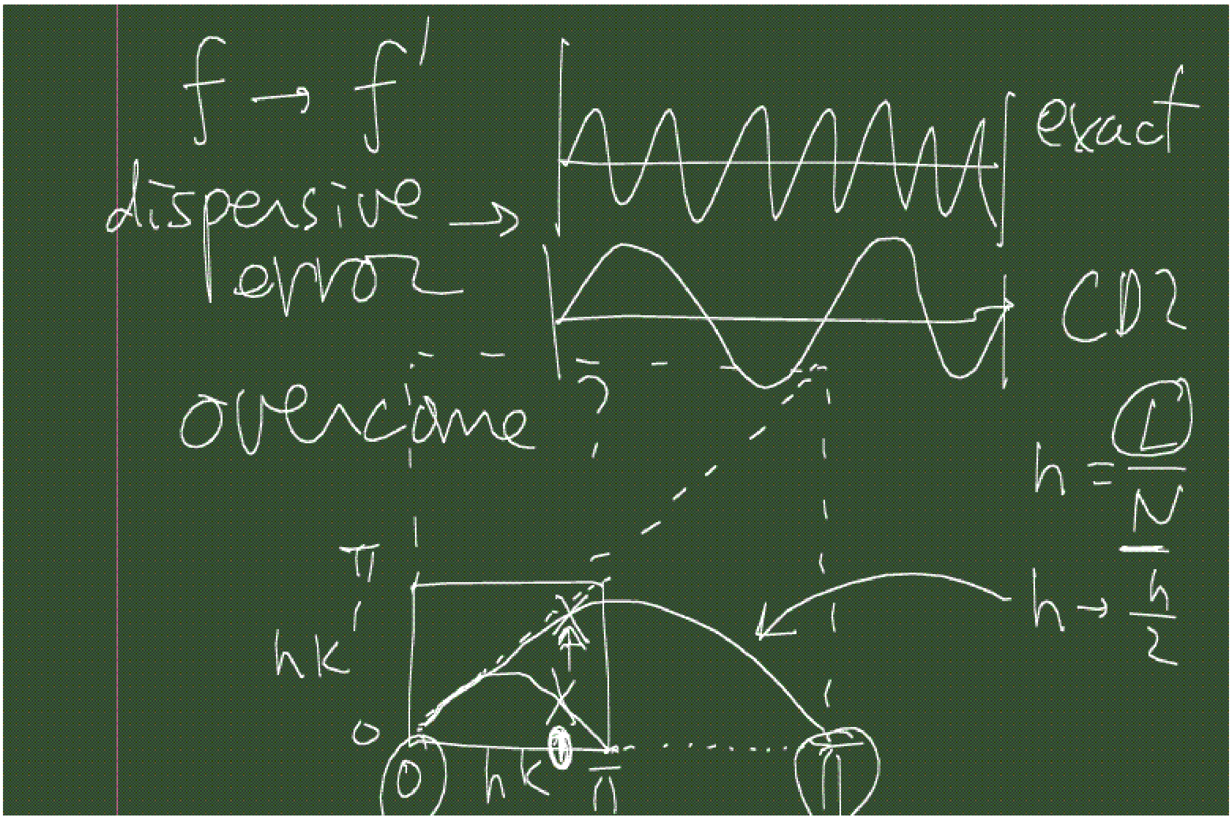
$$= \frac{1}{2h} e^{ikx_j}\left(e^{ikh} - e^{-ikh}\right)$$

$$= \frac{i}{h}\sin kh \cdot f_j = \boxed{i\frac{\sin kh}{h} f_j}$$

$$k' = \frac{\sin kh}{h}$$ : modified wavenumber

$\rightarrow hk' = \sin hk$


π
hk'
exact
CD4
CD2
0
0
hk
π = 3.14
bad prediction
0
L
N


f → f'
dispersive error
exact
CD2
overcome?
h = L/N
h → h/2
π
hk'
0
0
hk
π

## 2.4 Padé Approximations

Include the derivatives!

Ex. Find the most accurate formula that involves

$$f_{j+1},\ f_{j-1},\ f_j,\ f'_{j+1},\ f'_{j-1}$$

$$\underline{f'_j} + b_1 f'_{j+1} + b_2 f'_{j-1} + a_1 f_{j+1} + a_0 f_j + a_2 f_{j-1} = O(?)$$

Taylor table

| | $f_j$ | $f'_j$ | $f''_j$ | $f'''_j$ | $f^{iv}_j$ | $f^{v}_j$ |
|---|---|---|---|---|---|---|
| $b_1 f'_{j+1}$ | 0 | $b_1$ | $b_1 h$ | $b_1 \frac{h^2}{2}$ | $b_1 \frac{h^3}{6}$ | $b_1 \frac{h^4}{24}$ |
| $f'_j$ | 0 | 1 | 0 | 0 | 0 | 0 |
| $b_2 f'_{j-1}$ | 0 | $b_2$ | $-b_2 h$ | - | - | - |
| $a_1 f_{j+1}$ | - | - | | - | | - |
| $a_0 f_j$ | - | - | - | - | - | |
| $a_2 f_{j-1}$ | - | - | - | | - | |

Solve 5 eqs. (5 unknowns)

$$a_0 = 0,\quad a_1 = -\frac{3}{4h},\quad a_2 = \frac{3}{4h},$$

$$b_1 = \frac{1}{4},\quad b_2 = \frac{1}{4}$$

$$\rightarrow f'_{j+1} + 4f'_j + f'_{j-1} = \frac{3}{h}\left(f_{j+1} - f_{j-1}\right) + \frac{h^4}{30} f_j^{v} + \cdots,$$

$$O(h^4)\ ? \qquad j = 1, 2, 3, \ldots, n-1$$

n-1 eqs for n+1 unknowns

$(f'_0, f'_1, \ldots, f'_n)$

① $f'_0 = \frac{1}{2h}\left(-3f_0 + 4f_1 - f_2\right) + O(h^2)$

② or $f'_0 + 2f'_1 = \frac{1}{h}\left(-\frac{5}{2}f_0 + 2f_1 + \frac{1}{2}f_2\right) + O(h^3)$

$$\begin{pmatrix} 4 & 1 & 0 & & \phi \\ 1 & 4 & 1 & & \\ 0 & 1 & 4 & 1 & \\ \phi & & \ddots & \ddots & \ddots \end{pmatrix} \begin{pmatrix} f'_1 \\ f'_2 \\ \vdots \\ f'_{n-1} \end{pmatrix} = \begin{pmatrix} \vdots \\ \vdots \end{pmatrix}$$

✓ tri-diagonal matrix system. :)

✓ compact scheme! even though higher order
$(j, j-1, j+1)$

$$\frac{1}{12} f''_{j-1} + \frac{10}{12} f''_j + \frac{1}{12} f''_{j+1} = \frac{1}{h^2}\left(f_{j+1} - 2f_j + f_{j-1}\right) + O(h^4)$$

---

$j=1$: $f'_2 + 4f'_1 + f'_0 = \frac{3}{h}(f_2 - f_0)$

$$f'_2 + 4f'_1 = -f'_0 + \frac{h}{3}(f_2 - f_0)$$

$$= -\frac{1}{2h}\left(-3f_0 + 4f_1 - f_2\right) + \frac{h}{3}(f_2 - f_0)$$

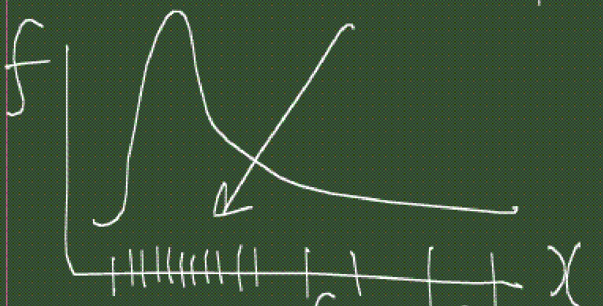
$$\begin{pmatrix} 4 & 1 & 0 & \cdot & 0 \\ & & & & \\ & & & & \end{pmatrix} \begin{pmatrix} f'_1 \\ f'_2 \\ \vdots \end{pmatrix} = \begin{pmatrix} \checkmark \\ \\ \\ \end{pmatrix}$$

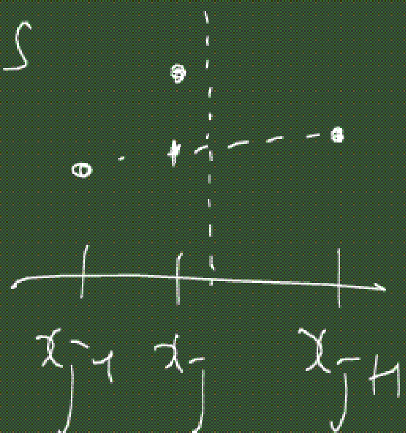
② $f_0' + 2f_1' = \frac{1}{h}\left( \star \right) \leftarrow j=0$

$j=1$: $f_2' + 4f_1' + f_0' = \frac{3}{h}\left(f_2 - f_0\right)$

$$\begin{pmatrix} 1 & 2 & & & \\ 1 & 4 & 1 & & \phi \\ & \ddots & \ddots & \ddots & \\ \phi & & \ddots & \ddots & \ddots \\ & & & \vdots & \vdots \end{pmatrix} \begin{pmatrix} f_0' \\ f_1' \\ f_2' \\ \vdots \\ \vdots \end{pmatrix} = \begin{pmatrix} \frac{1}{h}\star \\ \frac{3}{h}(f_2 - f_0) \\ \vdots \\ \vdots \\ \vdots \end{pmatrix}$$

## 2.5 Non-uniform grids




ex) $f_j' = \dfrac{f_{j+1} - f_{j-1}}{x_{j+1} - x_{j-1}}$ :

$$f_j'' = 2\left[ \frac{f_{j+1}}{h_j (h_j + h_{j+1})} - \frac{f_j}{h_j h_{j+1}} + \frac{f_{j-1}}{h_{j+1}(h_j + h_{j+1})} \right]$$

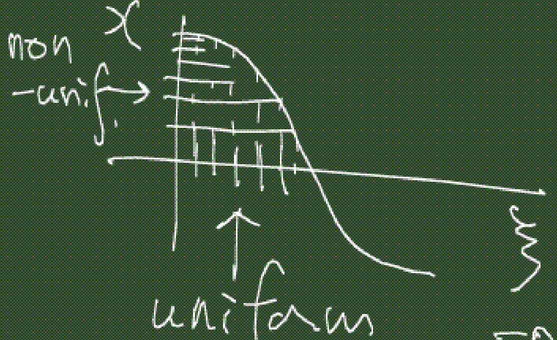
FD formulae for non-uniform meshes generally have a lower order of accuracy than their counterparts w/ the same stencil for uniform meshes.

ex) → strictly a first order

<u>or</u> use a coord. transformation.

$x \to \xi$

ex) $\xi = \cot^{-1} x$




In general,

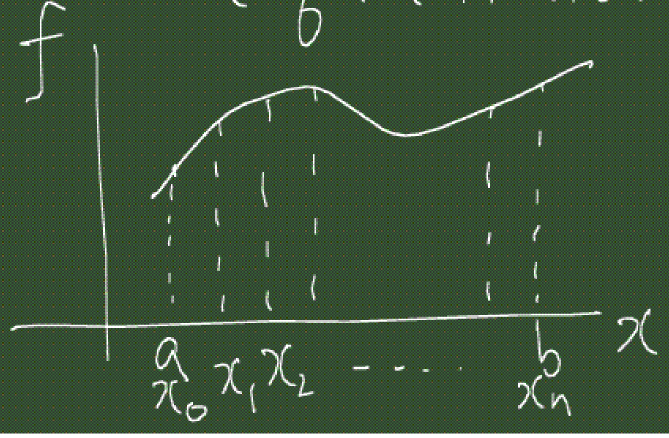
$\xi = g(x)$

$$\frac{df}{dx} = \frac{d\xi}{dx}\frac{df}{d\xi} = g'\frac{df}{d\xi}$$

$$\frac{d^2f}{dx^2} = \frac{d}{dx}\left(\frac{df}{dx}\right) = g''\frac{df}{d\xi} + g'^2\frac{d^2f}{d\xi^2}$$

FD w/ unif. mesh

# Ch. 3 Numerical Integration
(quadrature)




$$I = \int_a^b f(x)\,dx$$

$$= \sum_{j=0}^{n} f_j \underbrace{\omega_j}_{\text{weight}}$$

## §3.1 Trapezoidal and Simpson's rules

Lagrange polynomial

$$p(x) = \sum_{j=0}^{n} f_j L_j(x)$$

$$I = \int_a^b p(x)\,dx = \int_a^b \sum_{j=0}^{n} f_j L_j(x)\,dx$$

$$= \sum_{j=0}^{n} f_j \underbrace{\int_a^b L_j(x)\,dx}_{\omega_j}$$

$$I = (b-a)\sum_{j=0}^{n} C_j^n f_j$$

where $C_j^n = \frac{1}{b-a}\int_a^b L_j(x)\,dx$

Newton-Cotes formula

$C_j^n$ : Cotes number

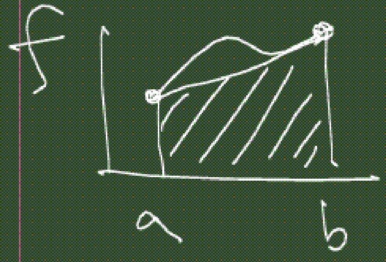
For $n=1$: $x_0 = a$, $x_1 = b$

$$L_0(x) = \frac{x-b}{a-b}, \quad L_1(x) = \frac{x-a}{b-a}$$

$$C_0^1 = \frac{1}{b-a}\int_a^b \frac{x-b}{a-b}\,dx = \frac{1}{2}$$

$$C_1^1 = \frac{1}{b-a}\int_a^b \frac{x-a}{b-a}\,dx = \frac{1}{2}$$

$$\therefore I = (b-a)\left[\frac{1}{2}f(a) + \frac{1}{2}f(b)\right]$$




Trapezoidal rule

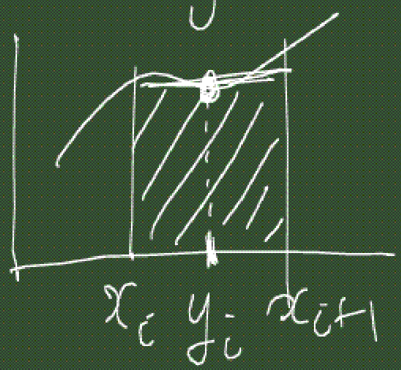
For $n=2$: $x_0=a$, $x_1=\frac{1}{2}(a+b)$, $x_2=b$

$$I=\int_a^b f(x)\,dx=\frac{b-a}{6}\left[f(a)+4f\left(\frac{a+b}{2}\right)+f(b)\right]$$

Simpson's rule

## 3.2 Error analysis

- Rectangle (Midpoint) rule




$$y_i=\frac{1}{2}(x_i+x_{i+1})$$

$$I=\int_{x_i}^{x_{i+1}} f(x)\,dx=h_i f(y_i)$$

where $h_i=x_{i+1}-x_i$

$$f(x) = f(y_i) + (x-y_i) f'(y_i) + \frac{1}{2}(x-y_i)^2 f''(y_i) + \cdots$$

$$I = \int_{x_i}^{x_{i+1}} f(x)\,dx = f(y_i) h_i + \frac{1}{2}(x-y_i)^2 f'(y_i) \Big|_{x_i}^{x_{i+1}} + \frac{1}{6}(x-y_i)^3 f''(y_i) \Big|_{x_i}^{x_{i+1}} + \cdots$$

even powers cancle (*)

$$\rightarrow \int_{x_i}^{x_{i+1}} f(x)\,dx = \underline{f(y_i) h_i} + \underbrace{\frac{1}{24} h_i^3 f''(y_i)}_{\text{leading error term}} + \frac{1}{1920} h_i^5 f^{(iv)}(y_i) + \cdots$$

!

For $\boxed{\text{one}}$ interval, the midpoint rule is <u>3rd order</u> accurate.

- Trapezoidal rule

$$\int_{x_i}^{x_{i+1}} f(x)\,dx = \frac{h_i}{2}\left[f(x_i) + f(x_{i+1})\right]$$

$$f(x_i) = f(y_i) - \frac{1}{2} h_i f'(y_i) + \frac{1}{8} h_i^2 f''(y_i) - \frac{1}{48} h_i^3 f'''(y_i) + \cdots$$

$$f(x_{i+1}) = \text{''} + \text{''} + \text{''} + \text{''} + \cdots$$

$$\frac{1}{2}\left(f(x_i) + f(x_{i+1})\right) = f(y_i) + \frac{1}{8} h_i^2 f''(y_i) + \frac{1}{384} h_i^4 f^{(iv)}(y_i) + \cdots$$

Solve for $f(y_i)$ and substitute in (*)

$$\int_{x_i}^{x_{i+1}} f(x)\,dx = h_i \frac{f(x_i) + f(x_{i+1})}{2} \underbrace{- \frac{1}{12} h_i^3 f''(y_i)}_{\text{leading error term}} - \frac{1}{480} h_i^5 f^{(iv)}(y_i) + \cdots \quad \cancel{**}$$

$\therefore$ TR has twice bigger error than midpoint rule.

How about global interval $[a,b]$?

$$I = \int_a^b f(x)\,dx$$

Assume uniform spacing $h_i = h$

$$I = \sum_{i=0}^{n-1} \int_{x_i}^{x_{i+1}} f(x)\,dx$$

$$\stackrel{TR}{=\!=} \frac{h}{2}\left[ f(a) + f(b) + 2\sum_{j=1}^{n-1} f(x_j) \right]$$

$$- \frac{h^3}{12} \sum_0^{n-1} f''(y_j) - \frac{1}{480} h^5 \sum_0^{n-1} f^{(iv)}(y_j) + \cdots$$

Mean Value Theorem

$$\left(\begin{array}{ll} \sum_{0}^{n-1} f''(y_j) = n f''(\bar{x}) & \text{where } a<\bar{x}<b \\ \sum_{0}^{n-1} f^{(iv)}(y_j) = n f^{(iv)}(\xi) & a<\xi<b \end{array}\right.$$

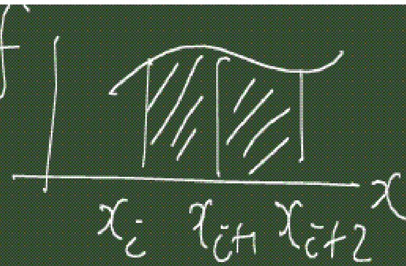
$$n = \frac{b-a}{h}$$

$$I = \frac{h}{2}\left(f(a)+f(b)+2\sum_{1}^{n-1} f_j\right)$$

$$\underline{-(b-a)\frac{h^2}{12} f''(\bar{x})} - \frac{b-a}{480} h^4 f^{(iv)}(\xi) + \cdots$$

Hence, for the <u>entire</u> interval $[a,b]$, TR is <u>2nd order</u> accurate.

- Simpson's rule




$$\int_{x_i}^{x_{i+2}} f(x)\,dx = \frac{h}{3}\left[ f(x_i) + 4f(x_{i+1}) + f(x_{i+2}) \right] \equiv S(f)$$

$$\int_{x_i}^{x_{i+2}} f(x)\,dx = h\left[ f(x_i) + f(x_{i+2}) \right] \equiv T(f) \quad \text{TR}$$

$$\int_{x_i}^{x_{i+2}} f(x)\,dx = 2h \cdot f(x_{i+1}) \equiv R(f) \quad \text{midpoint}$$

$$S(f) = \frac{2}{3} R(f) + \frac{1}{3} T(f)$$

Recall that truncation error of TR is 2 times of that of midpoint rule w/ opposite sign.

$\Rightarrow$ Simpson's rule is <u>4th order</u>!

fifth order in one interval.

leading error $\Rightarrow -\frac{1}{90} \underline{\underline{h^5}} f^{(iv)}(x_{i+1})$

$$I = \int_a^b f(x)\,dx =$$

$$= \frac{h}{3}\left[ f(a) + f(b) + 4\sum_{\substack{1 \\ \text{odd}}}^{n-1} f_j + 2\sum_{\substack{2 \\ \text{even}}}^{n-2} f_j \right]$$

$$- \frac{h^4}{180}(b-a) f^{(iv)}(\bar{x}) + \cdots$$

3.3 TR w/ end correction

$$\int_{x_i}^{x_{i+1}} f(x)\,dx = h_i \cdot \frac{f_i + f_{i+1}}{2} - \frac{1}{12} h_i^3 f''(y_i) + O(h^5)$$

$$\left( f''(y_i) = \frac{f'_{i+1} - f'_i}{h_i} + O(h_i^2) \right)$$

$$= h_i \frac{f_i + f_{i+1}}{2} - \frac{1}{12} h_i^3 \frac{f'_{i+1} - f'_i}{h_i} + O(h^5)$$

Sum for the entire domain

$$I = \frac{h}{2}\sum_{0}^{n-1}\left(f_i + f_{i+1}\right) - \frac{h^2}{12}\sum_{0}^{n-1}\left(f'_{i+1} - f'_i\right) + O(h^4)$$

$$= \quad '' \quad \underbrace{-\frac{h^2}{12}\left(f'(b) - f'(a)\right)}_{\text{end correction}} + O(h^4)$$

---

ex) $f(x) = e^x$

$$\int_0^4 e^x\,dx = e^4 - 1 = 53.59815$$

9pts. $I_T = 54.71015$ error $-1.112$

$I_S = 53.61622$ " $-0.01807$

$I_{TC} = 53.59352$ " $+0.00463$

# 3.4 Romberg Integration and Richardson Extrapolation

< technique for obtaining an accurate sol. by combining two or more less accurate sols.

Romberg Integration

→ Integration method

+ Richardson Extrapolation

- Trapezoidal rule

$$I = \int_a^b f(x)\,dx = \underbrace{\frac{h}{2}\left[f(a) + f(b) + 2\sum_{1}^{n-1} f_j\right]}_{= I_1} + c_1 h^2 + c_2 h^4 + \cdots$$

$$\tilde{I}_1 = I - c_1 h^2 - c_2 h^4 - c_3 h^6 - \cdots$$

Apply TR w/ $h_1 = h/2$.

Call this as $\tilde{I}_2$

$$\tilde{I}_2 = I - c_1 \frac{h^2}{4} - c_2 \frac{h^4}{16} - c_3 \frac{h^6}{64} - \cdots$$

Idea:

$$4\tilde{I}_2 - \tilde{I}_1 = 3I + \frac{3}{4} c_2 h^4 + \cdots$$

$$\frac{4\tilde{I}_2 - \tilde{I}_1}{3} = I + \underbrace{\frac{1}{4} c_2 h^4}_{} + \cdots \quad (*)$$

Combine two $2^{nd}$-order estimates of $I \longrightarrow 4^{th}$ order estimate.

Evaluate $I$ w/ $h_2 = h/4$

$$\tilde{I}_3 = I - c_1 \frac{h^2}{16} - c_2 \frac{h^4}{256} - \cdots$$

$$\Rightarrow \frac{4\tilde{I}_3 - \tilde{I}_2}{3} = I + \frac{h^4}{64} c_2 + \frac{5}{1024} h^6 c_3 + \cdots \quad (**)$$

(*) & (**)

$$\Rightarrow \frac{16}{15}\left(\frac{4\tilde{I}_3 - \tilde{I}_2}{3}\right) - \frac{1}{15}\left(\frac{4\tilde{I}_2 - \tilde{I}_1}{3}\right)$$

$$= I + O(h^6).$$

$$\begin{array}{cccc}
h & \tilde{I}_1 & & \\
\frac{h}{2} & \tilde{I}_2 & (*) & \\
\frac{h}{4} & \tilde{I}_3 & (**) & \\
 & O(h^2) & O(h^4) & O(h^6)
\end{array}$$

Extrapolation

Romberg integration

3.5 Adaptive quadrature

Romberg I. $\xrightarrow[\text{restriction}]{}$ · points are evenly spaced throughout the interval of integration.

inefficient

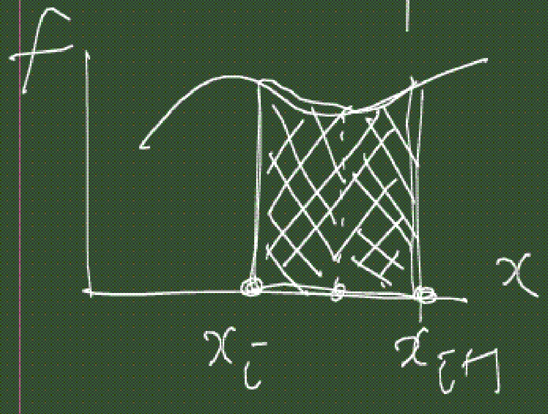
· Adaptive quadrature

user provides error tolerance.

→ the program automatically subdivides the interval to achieve the prescribed accuracy.

Error $\varepsilon$, $\left| I - \int_a^b f(x)\,dx \right| \le \varepsilon$

Base: Simpson's rule




For $(x_i, x_{i+1})$,

$$S_i = \frac{h_i}{6}\left[f(x_i) + 4f(x_{i+\frac{1}{2}}) + f(x_{i+1})\right]$$

$h_i = x_{i+1} - x_i$, 2 panels.

Subdivide into 4 panels.

$$S_i^{(2)} = \frac{h_i}{12}\left[f(x_i) + 4f(x_{i+\frac{1}{4}}) + 2f(x_{i+\frac{1}{2}}) + 4f(x_{i+\frac{3}{4}}) + f(x_{i+1})\right]$$

Let $I_i$ be the exact integral in $(x_i, x_{i+1})$.

$$I_i - S_i = c h_i^5 f^{(iv)}(x_{i+\frac{1}{2}}) + \cdots \quad (*)$$

$$I_i - S_i^{(2)} = c\left(\frac{h_i}{2}\right)^5 \left[ \underbrace{f^{(iv)}(x_{i+\frac{1}{4}})} + \underbrace{f^{(iv)}(x_{i+\frac{3}{4}})} \right] + \cdots$$

$$\left[\begin{array}{l} f^{(iv)}(x_{i+\frac{1}{4}}) = f^{(iv)}(x_{i+\frac{1}{2}}) - \frac{h_i}{4} f^{(v)}(x_{i+\frac{1}{2}}) + \cdots \\ f^{(iv)}(x_{i+\frac{3}{4}}) = \quad '' \quad + \quad '' \quad + \cdots \end{array}\right.$$

$$\Rightarrow I_i - S_i^{(2)} = c\left(\frac{h_i}{2}\right)^5 \cdot 2 f^{(iv)}(x_{i+\frac{1}{2}}) + \cdots \quad (**)$$

Subtract (**) from (*):

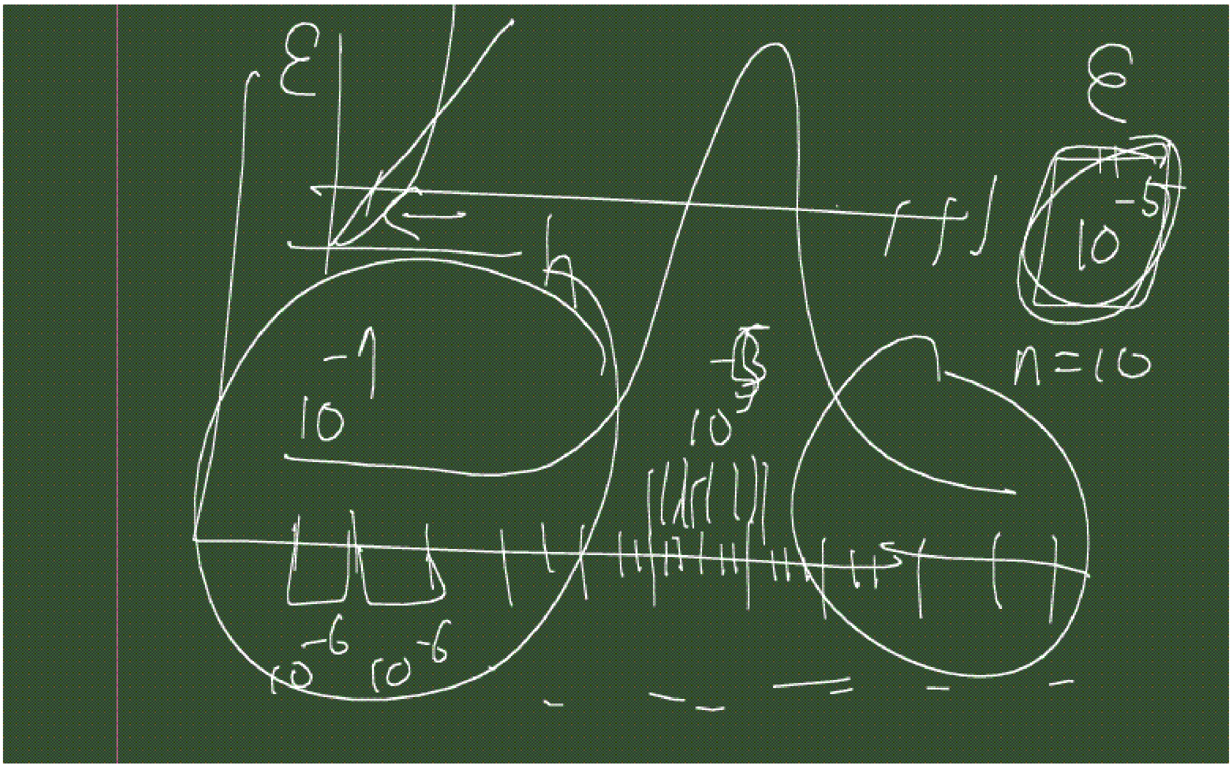
$$S_i^{(2)} - S_i = \frac{15}{16} c h_i^5 f^{(iv)}(x_{i+\frac{1}{2}}) + \cdots$$

$$\Rightarrow \boxed{I_i - S_i^{(2)} = \frac{1}{15}(S_i^{(2)} - S_i) + \cdots}$$

So, the error in $S_i^{(2)}$ is $\frac{1}{15}$ of $S_i^{(2)} - S_i$.

$$n = \frac{b-a}{h_i}$$

$$\frac{1}{15}\left|S_i^{(2)} - S_i\right| \begin{cases} \le \dfrac{\varepsilon}{n} = \dfrac{h_i}{b-a}\varepsilon & \text{ok} \\ > \dfrac{h_i}{b-a}\varepsilon & \text{subdivide further.} \end{cases}$$



## 3.6 Gauss quadrature

Method is optimum in the sense of maximum accuracy for a given number of function evaluations.

price for this → lack of a simple method for systematic error reduction

$$\int_a^b f(x)\,dx = \sum_{i=0}^{n} f_i \omega_i$$

$\left.\begin{array}{l} n+1 \text{ weights} \\ n+1 \text{ points} \end{array}\right\}$ $2n+2$ adjustable parameters.

⇒ a polynomial of degree $2n+1$.

Choose $x_i$ and $w_i$ for highest accuracy.

[ Mid term - October 21 (Wed) ]
18:00 - 20:00

노트 제목 2009-10-05

Gauss quadrature

$$\int_a^b f(x)\,dx = \sum_{i=0}^{n} f_i \omega_i$$

$n+1$ weights $\left.\right\}$ $2n+2$
$n+1$ pts $\left.\right\}$ adj. parameters

a polynomial of degree "$2n+1$".

Choose $x_i$ and $\omega_i$ for highest accuracy.

Let $f$ be a polynomial of degree $2n+1$.

Select pts. $x_0, x_1, x_2, \cdots, x_n$

$f(x_0), f(x_1), f(x_2), \cdots, f(x_n)$

Interpolate w/ Lagrange poly. of degree $n$.

$$p(x) = \sum_{j=0}^{n} f_j L_j(x)$$

$f-p$: ✓ poly. of degree $2n+1$

✓ $f-p = 0$ @ $x_0, x_1, \cdots, x_n$

So, define $F(x) = (x-x_0)(x-x_1)\cdots(x-x_n)$

$$f(x) - p(x) = F(x) \underline{\sum_{l=0}^{n} q_l x^l}$$ poly. deg $2n+1$

$$\int f - \int p = \int F \sum_{l=0}^{n} q_l x^l$$

$$= \sum_{l=0}^{n} q_l \int F(x)\, x^l\, dx$$

Demand $\int F(x)\, x^l\, dx = 0$ for $l = 0, 1, \cdots, n$

$\rightarrow$ $F(x)$ orthogonal to all polynomials of degree $n$ or less

($F$ is a poly. of degree $n+1$)

Then, $\int f(x)\,dx = \int P(x)\,dx$

$$= \sum_{j=0}^{n} f(x_j) \underbrace{\int L_j(x)\,dx}_{\omega_j}$$

Here, F belongs to class of Legendre polynomials

$$\int_{-1}^{1} F_n F_m\,dx = \delta_{mn} \qquad -1 \le x \le 1$$

$$= \begin{cases} 0 & m \ne n \\ 1 & m = n \end{cases}$$

$x_j$ are the zeros of Legendre polynomials.

Transform $a \le x \le b$ to $-1 \le \xi \le 1$.

w/ $x = \frac{1}{2}(b+a) + \frac{1}{2}(b-a)\xi$.

$$P_n(x) = \sum_{m=0}^{M} (-1)^m \frac{(2n-2m)!}{2^n m!(n-m)!(n-2m)!} \cdot x^{n-2m}$$

where $M = \frac{n}{2}$ or $\frac{n-1}{2}$ (integer)

$P_0(x) = 1$

$P_1(x) = x$

$P_2(x) = \frac{1}{2}(3x^2 - 1)$

$P_3(x) = \frac{1}{2}(5x^3 - 3x)$

$P_4(x) = \frac{1}{8}(35x^4 - 30x^2 + 3)$, ....

ex $I = \int_0^4 e^x dx = 53.59815003$

5 pts, $\int_a^b f\,dx = \frac{b-a}{2}\int_{-1}^{1} f(\xi)\,d\xi$

$P_5(x) = \frac{1}{8}(63x^5 - 70x^3 + 15x)$

| $\xi_K$ | $\omega_K$ |
|---|---|
| -0.9061 | 0.2369 |
| -0.5384 | 0.4786 |
| 0 | 0.5688 |
| +0.5384 | 0.4786 |
| +0.9061 | 0.2369 |

$I = \sum f_i \omega_i \times 2$

$= 53.59813663$

$\varepsilon = 0.0000134$

cf. $\varepsilon = 0.018$ w/ Simpson's rule with 9pts.

- $\int_0^\infty e^{-x} f(x)\, dx = \sum_j \omega_j f_j$

  Gauss-Laguerre formula

  Orthogonality $\int_0^\infty L_n L_m e^{-x} dx = \delta_{nm}$

- $\int_{-\infty}^{\infty} e^{-x^2} f(x)\, dx$

  Gauss-Hermite formula

  $$\int_{-\infty}^{\infty} H_n H_m e^{-x^2} dx = \delta_{nm}$$

- $\int_{-1}^{1} \frac{f(x)}{\sqrt{1-x^2}}\, dx$ Chebyshev-Gauss quad.

  $$\int_{-1}^{1} T_n T_m \frac{1}{\sqrt{1-x^2}}\, dx = \delta_{nm}.$$

The only disadvantage of Gauss quadrature is the difficulty of accuracy improvement.

- Singularity? $\int_a^b f(x)\,dx$

① substitution

$$I = \int_0^1 \frac{e^x}{\sqrt{x}}\,dx \qquad \text{let } x = t^2, \quad dx = 2t\,dt$$

$$\rightarrow I = 2\int_0^1 e^{t^2}\,dt$$

② Integration by parts

$$u = e^x, \quad dv = \frac{dx}{\sqrt{x}} \rightarrow v = 2\sqrt{x}$$

$$I = 2\sqrt{x}e^x\Big|_0^1 - 2\int_0^1 \sqrt{x}\,e^x\,dx$$

③ Singularity subtraction ← not really good.

$$I = \int_0^1 \frac{1}{\sqrt{x}}\,dx + \int_0^1 \frac{e^x - 1}{\sqrt{x}}\,dx.$$

## Integrals w/ $\infty$

$$I = \int_0^\infty e^{-x^2} dx$$

1. Gauss quadrature
2. change indep. variable

$t = \frac{1}{1+x}$ maps $[0, \infty) \to [0, 1]$

# Ch. 4 Numerical Sol. of ODE

$$y'' + \omega^2 y = f(x)$$

$\begin{cases} y(0) = y_0 \\ \left.\frac{dy}{dx}\right|_0 = v \end{cases}$ or $\begin{cases} y(0) = y_0 \\ y(L) = y_L \end{cases}$

$\Uparrow$ Initial value problem

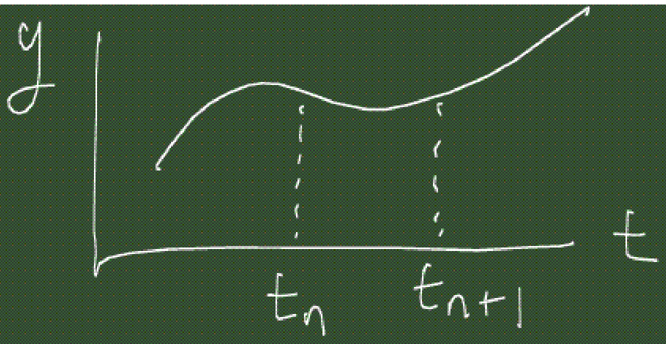
$\Uparrow$ Boundary value problem

4.1 Initial value probs.

$$\frac{dy}{dt} = \boxed{y' = f(y,t)}$$

$$y(0) = y_0$$

Higher order ODE's can be converted to a system of first order ODE's.




All methods assume that sol. is known at $0 \le t \le t_n$ and use it to get the sol. at $t = t_{n+1} = t_n + \underline{\Delta t}$

computational time step

Taylor series $y' = f(y,t)$

$y_{n+1} = y(t_n + \Delta t) \qquad \Delta t = h$

$= y_n + h y_n' + \frac{h^2}{2} y_n'' + \frac{h^3}{6} y_n''' + \cdots$

$y_n' = f(y_n, t_n)$

$y'' = \frac{d}{dt} y' = \frac{d}{dt}[f(y,t)]$

$= \frac{\partial f}{\partial t} + \frac{\partial f}{\partial y}\frac{\partial y}{\partial t} = f_t + f f_y$

$y''' = \frac{d}{dt}(f_t + f f_y)$

$= f_{tt} + f_t f_y + 2 f f_{yt} + f_y^2 f + f^2 f_{yy}$

# of items increases rapidly.
Hence, it is not very practical to include higher order terms than third order.

- <u>Euler method</u> (based on first two terms)

$$y_{n+1} = y_n + h f(y_n, t_n)$$

or, $$\frac{y_{n+1} - y_n}{h} = f(y_n, t_n)$$

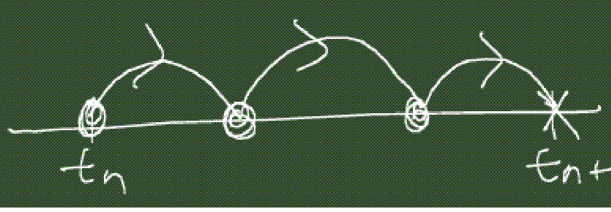
2nd order accurate for one time step

Globally ($t_0 \to t_f$), it is <u>1st order</u>.

Explicit Euler method (EE)

- <u>Runge-Kutta method</u> (RK)

$y_{n+1}$ is obtained in terms of $y_n$, $f(y_n, t_n)$ and values of $f$ at intermediate times $t_n \leq t \leq t_{n+1}$.

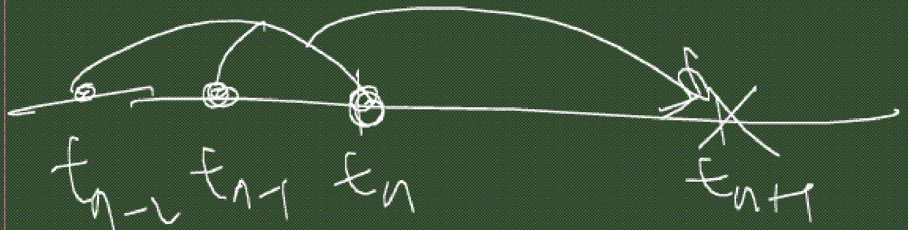


- **Multi-step methods**

use information from $t \le t_n$

$y_{n-1}, y_{n-2}, \ldots$ not

$f_{n-1}, f_{n-2}, \ldots$ self starting.




- **Explicit and implicit methods**

formula involves $f(y_{n+1}, t_{n+1})$

e.g. $y_{n+1} = y_n + h f(y_{n+1}, t_{n+1})$

involves solving a nonlinear algebraic eqn

but has better numerical stability.

## 4.2 Numerical Stability

It is possible for num. sol. of a diff'l eq. to blow up (grow unbounded) even though the exact sol. is well-behaved.

We seek parameters of the num. method (such as h) so that the num. sol. is well behaved.

- Stable num. method

Num. sol. does not blow up with any choice of the parameters.

- **Unstable num. method**

  Num. sol. always blows up irrespective of the choice of parameters.

- **Conditionally stable num. method**

  Sol. is well-behaved with some choice of parameters.

⊙ Model problem

$$y' = f(y, t)$$

Linear stability analysis

$$f(y,t) = f(y_0, t_0) + (t - t_0)\left.\frac{\partial f}{\partial t}\right|_{t_0, y_0} + (y - y_0)\left.\frac{\partial f}{\partial y}\right|_{t_0, y_0} + \cdots$$

$$y' = \lambda y + \underbrace{\alpha_1 + \alpha_2 t}_{\text{particular sol.}} + \text{high order terms}$$

!

$y = \gamma y^2 \rightarrow y = -\dfrac{1}{\gamma t + c}$

decaying sol.

$$\boxed{\begin{array}{l}\text{Model problem} \\ y' = \underline{\underline{\lambda}} y, \quad y(0).\end{array}}$$

---

$\lambda$ : complex number $(\lambda_R + i\lambda_I)$

$\lambda_R \le 0$ for stability

exact sol. $y = y_0 e^{\lambda t} = y_0 e^{(\lambda_R + i\lambda_I)t}$

?

$$\underline{y'' + \omega^2 y = 0}$$

$y_1 \equiv y$

$y_2 \equiv y_1' \rightarrow y_2' = y_1'' = y'' = -\omega^2 y_1$

$$\therefore \begin{pmatrix} y_1' \\ y_2' \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\omega^2 & 0 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$$

eigenvalues $\lambda = \pm i\omega$

$$\underline{u} \equiv \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \rightarrow \underline{u}' = A\underline{u} = S^{-1}\Lambda S\underline{u}$$

$$\Lambda = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}$$

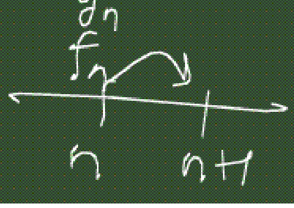
$$S\underline{u}' = \Lambda \underbrace{S\underline{u}}_{\equiv \underline{z}} \Rightarrow \underline{z}' = \underbrace{\Lambda}_{\begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}} \underline{z}$$

$$\rightarrow \quad z_1 = c_1 e^{\lambda_1 t}, \quad z_2 = c_2 e^{\lambda_2 t}$$

## 4.3 Stability analysis for the Euler method

Forward Euler

Explicit Euler



$$y_{n+1} = y_n + h f(y_n, t_n)$$

Apply to model prob. ($f = \lambda y$)

$$y_{n+1} = y_n + h\lambda y_n$$

$$= (1 + \lambda h) y_n$$

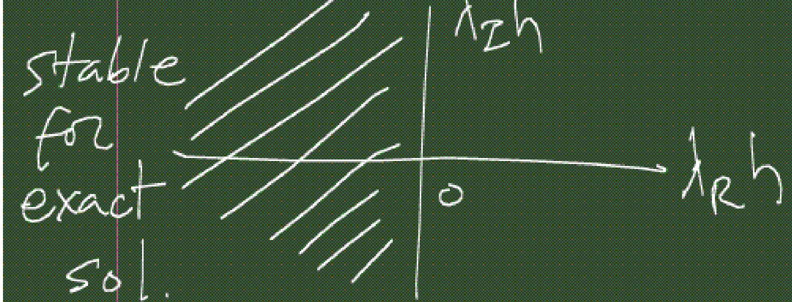
$$\rightarrow y_n = (1+\lambda h)^n y_0 , \quad t = nh$$

$$= (1 + \lambda_R h + i\lambda_I h)^n y_0$$

Whether the sol. remains bounded depends on $\lambda_R h$, $\lambda_I h$.

exact sol. $y = y_0 e^{\lambda t} = y_0 e^{\lambda_R t} e^{i\lambda_I t}$

For the exact sol. to be well behaved, $\lambda_R \leq 0$

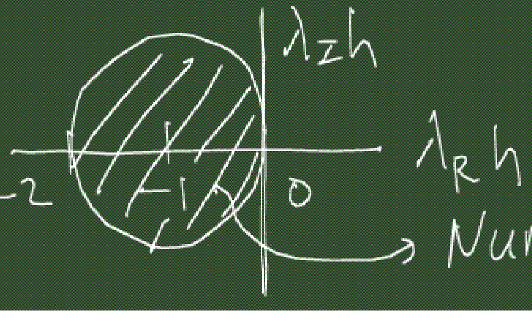


$$y_n = (1+\lambda h)^n y_0 \quad EE$$
$$= \sigma^n y_0$$

require $|\sigma| \leq 1$ $\quad \sigma = 1 + \lambda_R h + i \lambda_I h$

$$|\sigma|^2 = (1+\lambda_R h)^2 + (\lambda_I h)^2 \leq 1$$

to be stable.




- When $\lambda$ is real and negative,

$$|\lambda h_{max}| = 2 \rightarrow h_{max} = \frac{2}{|\lambda|}$$

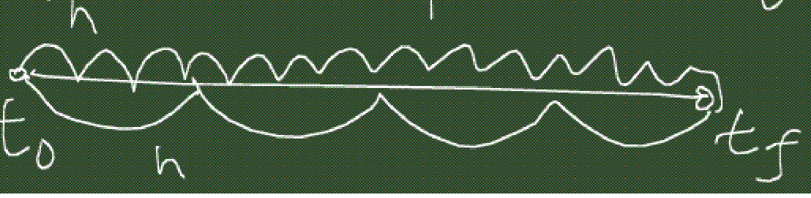
- When $\lambda$ is purely imaginary, $(\lambda = i\omega)$

$$|\sigma|^2 = 1 + \omega^2 h^2 > 1$$

EE is unstable for $\lambda = i\omega$.

Stability places a restriction on the magnitude of the time step that you can choose.

So, if want to integrate to $t = t_f$, more steps are req.




- Note on Accuracy

$y' = \lambda y$

exact sol. $y = y_0 e^{\lambda t} = y_0 e^{\lambda h n}$

$$= y_0 \left( \underline{1 + \lambda h} + \tfrac{1}{2}\lambda^2 h^2 + \cdots \right)^n$$

EE: $y_n = y_0 \left( \underline{1 + \lambda h} \right)^n$ $\quad \tfrac{1}{2}\lambda^2 h^2 n = \frac{T}{h}$

EE recovers only up to $\lambda h$ term

$\rightarrow$ first order accurate.

$$(1+x)^n = 1 + nx$$

$$= 1 + \underline{n} \cdot \frac{1}{2}\lambda^2 h^2$$

$$= 1 + \frac{T}{\cancel{h}} \cdot \frac{1}{2}\lambda^2 h^{\cancel{2}}$$

$$= 1 + \underbrace{\frac{1}{2} T \lambda^2 h}_{O(h)}$$

Signal for num. instability

$\lambda$ real & negative

$$y_n = (1+\lambda h)^n y_0$$

if unstable, $|\underbrace{1+\lambda h}_{\text{negative}}| > 1$

## 4.4 Implicit Euler (Backward Euler)

IE

$$y' = f(y, t)$$

$$\frac{y_{n+1} - y_n}{h} = f(y_{n+1}, t_{n+1})$$

Or, $\boxed{y_{n+1} = y_n + h\, f(y_{n+1}, t_{n+1})}$

cost/timestep is higher

$$y' = \lambda y$$

IE : $y_{n+1} = y_n + h\lambda y_{n+1}$

$\rightarrow y_{n+1} = \left(\frac{1}{1-\lambda h}\right) y_n$

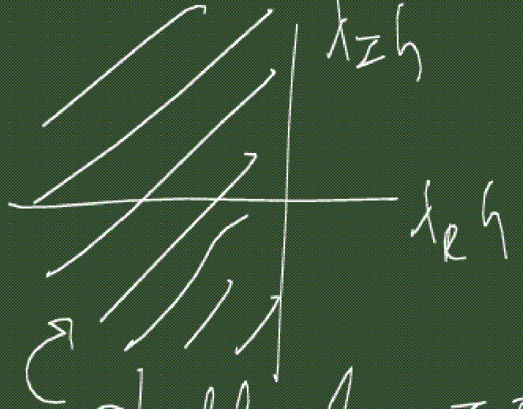
$$\sigma = \frac{1}{1 - \lambda_R h - i\lambda_I h}$$

$\rightarrow y_n = \sigma^n y_0$

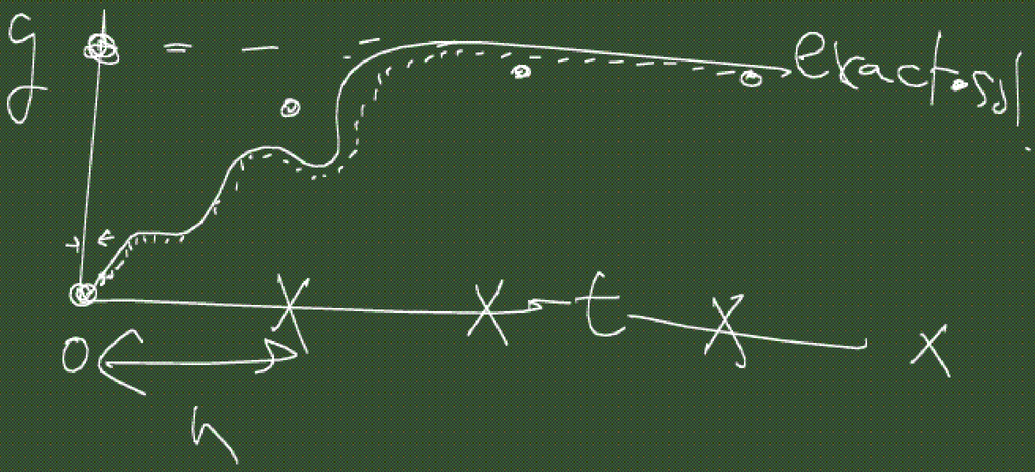
$$|\sigma|^2 = \frac{1}{(1-\lambda_R h)^2 + (\lambda_I h)^2}$$

$\rightarrow |\sigma|^2 < 1$ for $\lambda_R < 0$

$\rightarrow$ unconditionally stable!




stable for IE



# 4.5 Numerical accuracy revisited

Accuracy (IE) $\sigma = \dfrac{1}{1-\lambda h} = 1 + \lambda h + \lambda^2 h^2 + \lambda^3 h^3 + \cdots$

$y_n = y_0 \sigma^n$ $\quad e^{\lambda h} = \boxed{1 + \lambda h} + \frac{1}{2}\lambda^2 h^2 + \cdots$

$\rightarrow$ So, IE is first-order accurate

$$e^{\lambda h} - \frac{1}{1-\lambda h} = -\frac{1}{2}\lambda^2 h^2 + \cdots \quad \text{IE}$$

$$e^{\lambda h} - (1+\lambda h) = \frac{1}{2}\lambda^2 h^2 + \cdots \quad \text{EE}$$

$\rightarrow$ Stability does not say anything about accuracy.

- Accuracy $y' = \lambda y$

$\lambda = i\omega \rightarrow y' = i\omega y$

exact sol. $y = y_0 e^{i\omega t}$

$= y_0 (\cos\omega t + i \sin\omega t)$

$\rightarrow |e^{i\omega t}| = 1$

EE $\rightarrow y_{n+1} = y_n + i\omega h y_n$

$= (1 + i\omega h) y_n$

$\rightarrow y_n = (1 + i\omega h)^n y_0 = \sigma^n y_0$

$|\sigma| = \sqrt{1 + \omega^2 h^2} > 1$ amplitude of sol.

$\rightarrow$ sol. will diverge

(IE? $\sigma = \dfrac{1}{1 - i\omega h} \Rightarrow |\sigma| = \dfrac{1}{\sqrt{1 + \omega^2 h^2}} < 1$)

sol. will decay

$$\sigma = |\sigma| e^{i\theta}$$

$$\theta = \tan^{-1}\frac{Im(\sigma)}{Real(\sigma)} = \tan^{-1}\omega h$$

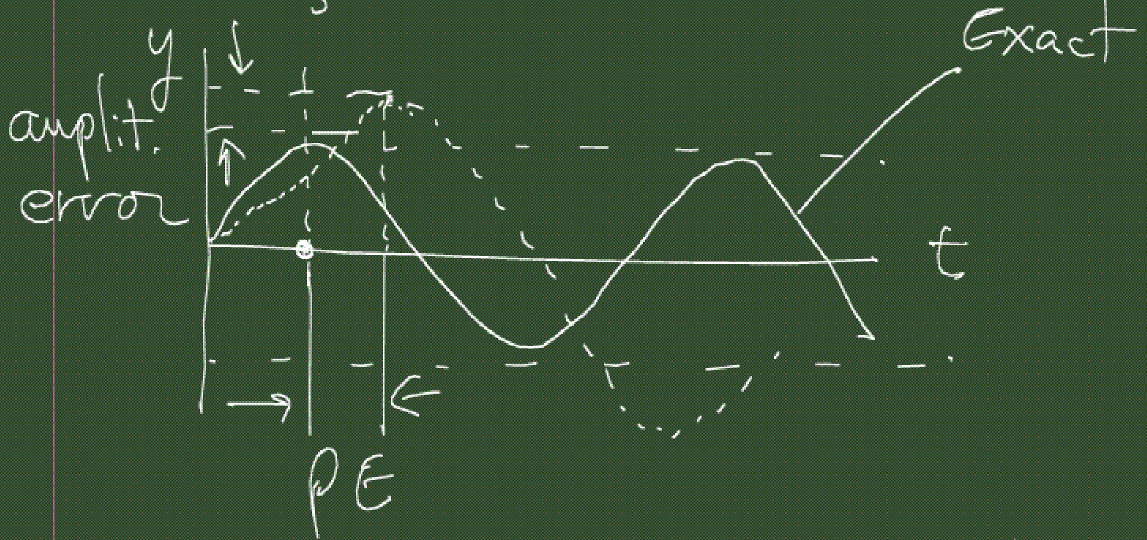
num. sol. $y_n = \sigma^n y_0 = y_0 |\sigma|^n e^{i\theta n}$

exact. sol. $y = y_0 e^{i\omega t} = y_0 e^{(i\omega h) n}$

phase error $= \omega h - \theta = \omega h - \tan^{-1}\omega h$
(PE)

$$\tan^{-1}\omega h = \omega h - \frac{1}{3}(\omega h)^3 + \frac{1}{5}(\omega h)^5 - \dots$$

$$PE = \frac{1}{3}(\omega h)^3 + \dots$$



4.6 Trapezoidal method

$$y' = f(y,t) \rightarrow \frac{y_{n+1} - y_n}{h} = \frac{1}{2}(f_{n+1} + f_n)$$

$$\rightarrow \boxed{y_{n+1} = y_n + \frac{h}{2}\left[f(y_{n+1}, t_{n+1}) + f(y_n, t_n)\right]}$$

$$\left.\begin{array}{l}\frac{dy}{dt} = f(y,t) \\ y(t_n) = y_n\end{array}\right] \rightarrow y_{n+1} = y_n + \underbrace{\int_{t_n}^{t_{n+1}} f(y,t)\,dt}_{\frac{h}{2}[f_{n+1} + f_n]}$$

- implicit method
- 2nd-order accurate method

model prob. $y' = \lambda y$

$$\rightarrow y_{n+1} = y_n + \frac{h}{2}(\lambda y_{n+1} + \lambda y_n)$$

$$\rightarrow y_{n+1} = \frac{1 + \lambda h/2}{1 - \lambda h/2} y_n = \sigma y_n$$

$$\sigma = \frac{1 + \lambda h/2}{1 - \lambda h/2} = 1 + \lambda h + \frac{1}{2}\lambda^2 h^2 + \frac{1}{4}\lambda^3 h^3 + \cdots$$

$$e^{\lambda h} = \underline{1 + \lambda h + \frac{1}{2}\lambda^2 h^2} + \frac{1}{3!}\lambda^3 h^3 + \cdots$$

$\therefore$ TR is second order

Stability ?

$$\lambda = \lambda_R + i\lambda_I$$

$$\sigma = \frac{1 + \frac{1}{2}\lambda_R h + i\frac{1}{2}\lambda_I h}{1 - \frac{1}{2}\lambda_R h - i\frac{1}{2}\lambda_I h} = \frac{Ae^{i\theta}}{Be^{i\alpha}}$$

$$\theta = \tan^{-1}\frac{\frac{1}{2}\lambda_I h}{1 + \frac{1}{2}\lambda_R h}, \quad \alpha = \tan^{-1}\frac{-\frac{1}{2}\lambda_I h}{1 - \frac{1}{2}\lambda_R h}$$

$$|\sigma| = \frac{|A|}{|B|} = \frac{\sqrt{(1 + \frac{1}{2}\lambda_R h)^2 + (\frac{1}{2}\lambda_I h)^2}}{\sqrt{(1 - \frac{1}{2}\lambda_R h)^2 + (\frac{1}{2}\lambda_I h)^2}}$$

Phase $= \theta - \alpha$

For $\lambda_R < 0$, $|A| < |B| \rightarrow |\sigma| < 1$

$\therefore$ TR is unconditionally stable.

For $\lambda_R = 0$, $\lambda = i\omega$

$\rightarrow |\sigma| = 1$

$\therefore$ TR does not have amp. error.

- phase error? $\lambda = i\omega$

$$\text{phase} = 2\theta = 2\tan^{-1}\frac{\omega h}{2}$$

$$PE = \omega h - 2\tan^{-1}\frac{\omega h}{2}$$

$$= \frac{1}{12}(\omega h)^3 + \cdots$$

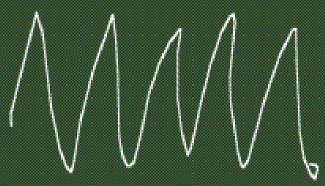
four times better than EE.

TR $\rightarrow$ ODE

PDE : Crank-Nicolson method

• Suppose $\lambda$ is real and negative
$(\lambda_i = 0, \ \lambda_R < 0)$

for very large $h$, $\sigma \to -1$.

Then, $y_n = y_0 \sigma^n$ 

oscillates bet. $-y_0$ and $+y_0$

but does not blow up.

Ex. $y'' + \omega^2 y = 0$, $y(0) = y_0$, $y'(0) = 0$

$\Rightarrow y_1' = y_2, \ y_2' = -\omega^2 y_1$

$$\begin{bmatrix} y_1' \\ y_2' \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\omega^2 & 0 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \to \lambda = \pm i\omega$$

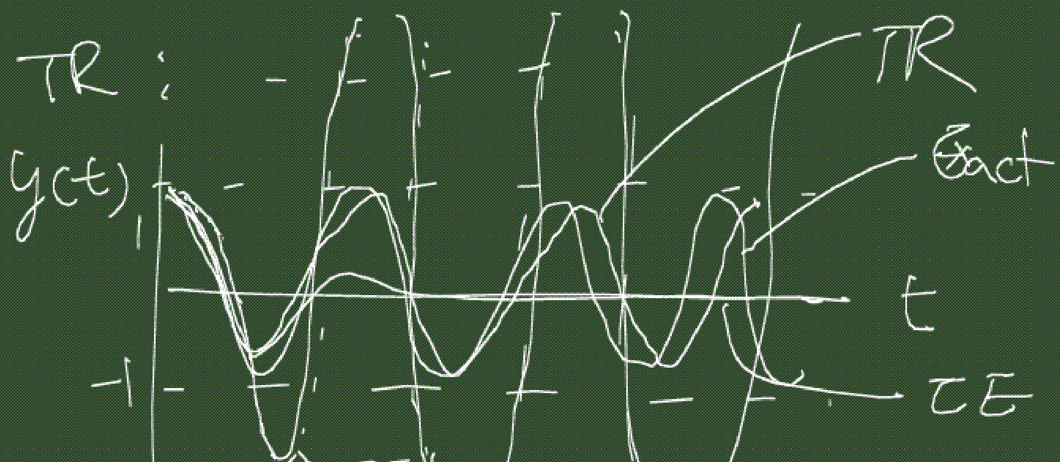
$$\Rightarrow \begin{bmatrix} z_1' \\ z_2' \end{bmatrix} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$$

EE :
$$\frac{y_{1\,n+1} - y_{1\,n}}{h} = y_{2\,n}$$
$$\frac{y_{2\,n+1} - y_{2\,n}}{h} = -\omega^2 y_{1\,n}$$

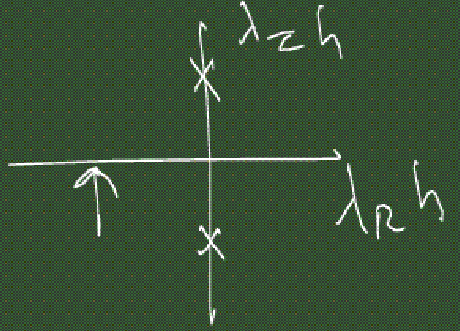
$$\rightarrow \left( \begin{array}{l} y_{1\,n+1} = y_{1\,n} + h\, y_{2\,n} \\ y_{2\,n+1} = y_{2\,n} - \omega^2 h\, y_{1\,n} \end{array} \right.$$

IE :
$$\left( \begin{array}{l} y_{1\,n+1} = y_{1\,n} + h\, y_{2\,n+1} \\ y_{2\,n+1} = y_{2\,n} - \omega^2 h\, y_{1\,n+1} \end{array} \right.$$

TR :



$\lambda = \pm i\omega \Rightarrow$ EE unstable




IE & TR stable

IE $\rightarrow$ decay

TR $\rightarrow$ $|\sigma| = 1$ ok.

## 4.7 Linearization for implicit methods

implicit method $\rightarrow$ solving nonlinear algebraic eq.

iterative sol. procedure $\leftarrow$

$$y' = \lambda y^5$$

IE: $\underbrace{y_{n+1} - y_n = h\lambda y_{n+1}^5}$

$\Downarrow$

can be avoided by linearization.

$$y' = f(y, t)$$

$$\text{TR}: \; y_{n+1} = y_n + \frac{h}{2}\left[f(y_{n+1}, t_{n+1}) + f(y_n, t_n)\right] + \underline{O(h^3)}$$

$$f(y_{n+1}, t_{n+1}) = f(y_n, t_{n+1}) + (y_{n+1} - y_n)\frac{\partial f}{\partial y}\bigg|_{y_n, t_{n+1}} + \frac{1}{2}(y_{n+1} - y_n)^2 \frac{\partial^2 f}{\partial y^2}\bigg|_{y_n, \ldots} + \cdots$$

$$y_{n+1} = y_n + O(h)$$

$$(y_{n+1} - y_n)^2 \sim O(h^2)$$

$$\Rightarrow y_{n+1} = y_n + \frac{h}{2}\left[f(y_n, t_{n+1}) + (y_{n+1} - y_n)\frac{\partial f}{\partial y}\bigg|_{y_n, t_{n+1}} + f(y_n, t_n)\right] + O(h^3)$$

$$\rightarrow y_{n+1} = y_n + \frac{h}{2} \frac{f(y_n, t_{n+1}) + f(y_n, t_n)}{1 - \frac{h}{2} \frac{\partial f}{\partial y}\Big|_{y_n, t_{n+1}}}$$

✓ this formula does not require iteration while retaining the global second-order accuracy.

✓ linear stability analysis
→ linearized scheme is also unconditionally stable.

✓ Sometimes, linearization may lead to some loss of total stability for nonlinear f.

노트 제목 2009-10-14

Midterm — #301-105
Oct. 21 (Wed) 6pm-8pm

4.8 Runge-Kutta methods

- Predictor-corrector method

explicit method $\longleftrightarrow$ implicit m.

timestep limit ↓ iterative sol.

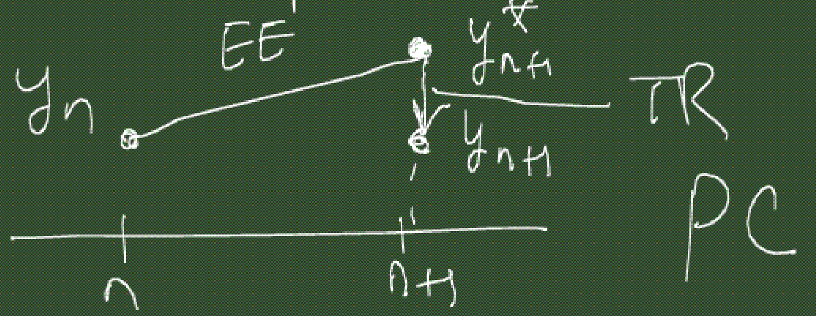
predictor-corrector

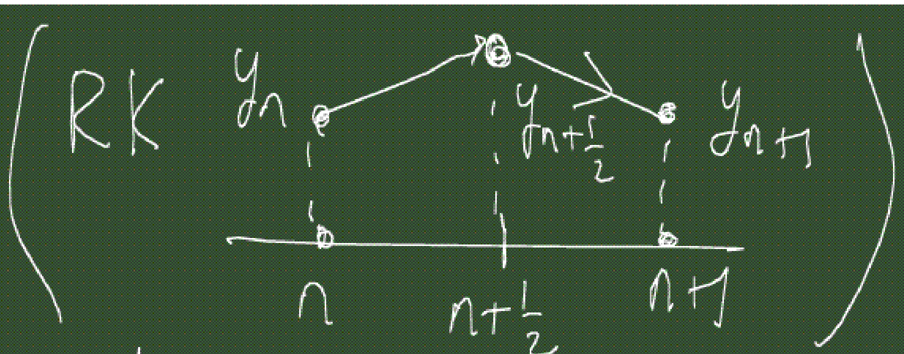
Predictor-corrector methods and Runge-Kutta methods provide 훨씬 better stability than explicit methods but require less work/time step compared to implicit method.

$$\begin{cases} y^*_{n+1} = y_n + hf(y_n, t_n) & \text{EE as predictor} \\ y_{n+1} = y_n + \frac{h}{2}\left[f(y^*_{n+1}, t_{n+1}) + f(y_n, t_n)\right] & \end{cases}$$

trapezoidal as corrector

explicit method




model prob. $y' = \lambda y$

PC $\quad y^*_{n+1} = y_n + h\lambda y_n = (1+\lambda h)\, y_n$

$$y_{n+1} = y_n + \frac{h}{2}\left[\lambda y^*_{n+1} + \lambda y_n\right]$$

$$\rightarrow y_{n+1} = y_n + \frac{h}{2}\left[\lambda(1+\lambda h)y_n + \lambda y_n\right]$$

$$= \left(1 + \lambda h + \frac{1}{2}\lambda^2 h^2\right) y_n$$

$\longrightarrow$ 2nd-order accurate.

<u>Stability</u>

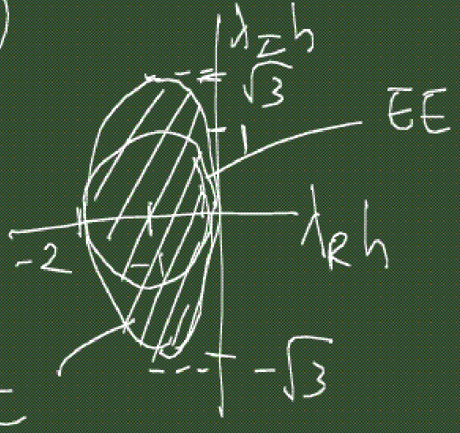
$$\sigma = 1 + \lambda h + \frac{1}{2}\lambda^2 h^2$$

$$|\sigma| \le 1 \rightarrow \left|1 + \lambda h + \frac{1}{2}\lambda^2 h^2\right| \le 1$$

complex polynomial root finder

$$1 + \lambda h + \frac{1}{2}\lambda^2 h^2 = e^{i\theta}$$

Find $\lambda h$ (complex) for different trials for $\theta$



Runge-Kutta method

Advantages

1) good stability properties

2) time-step can be changed

3) self-starting

2nd order Runge-Kutta method

(RK2)

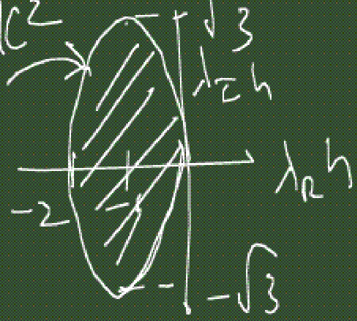
$$\begin{cases} y^*_{n+\frac{1}{2}} = y_n + \frac{h}{2} f(y_n, t_n) \\ y_{n+1} = y_n + h f(y^*_{n+\frac{1}{2}}, t_{n+\frac{1}{2}}) \end{cases}$$

model prob. $y' = \lambda y$

$$RK2 \rightarrow y_{n+1} = \underline{(1 + \lambda h + \tfrac{1}{2}\lambda^2 h^2)}\, y_n$$

∴ second order

Stability $|\sigma| \le 1$




• $\lambda = i\omega$

$$\sigma = 1 + i\omega h - \frac{1}{2}\omega^2 h^2$$

$$|\sigma|^2 = \left(1 - \frac{1}{2}\omega^2 h^2\right)^2 + \omega^2 h^2 = 1 + \frac{1}{4}\omega^4 h^4$$

$$|\sigma| = \sqrt{1 + \omega^4 h^4/4} > 1 \quad \text{unstable}$$

$$\sigma = |\sigma| e^{i\theta}$$

$$\theta = \tan^{-1}\frac{\omega h}{1 - \frac{1}{2}\omega^2 h^2}$$

$$PE = \omega h - \theta = -\frac{\omega^3 h^3}{6} + \cdots$$

(cf. PE for EE $= \frac{1}{3}\omega^3 h^3$)

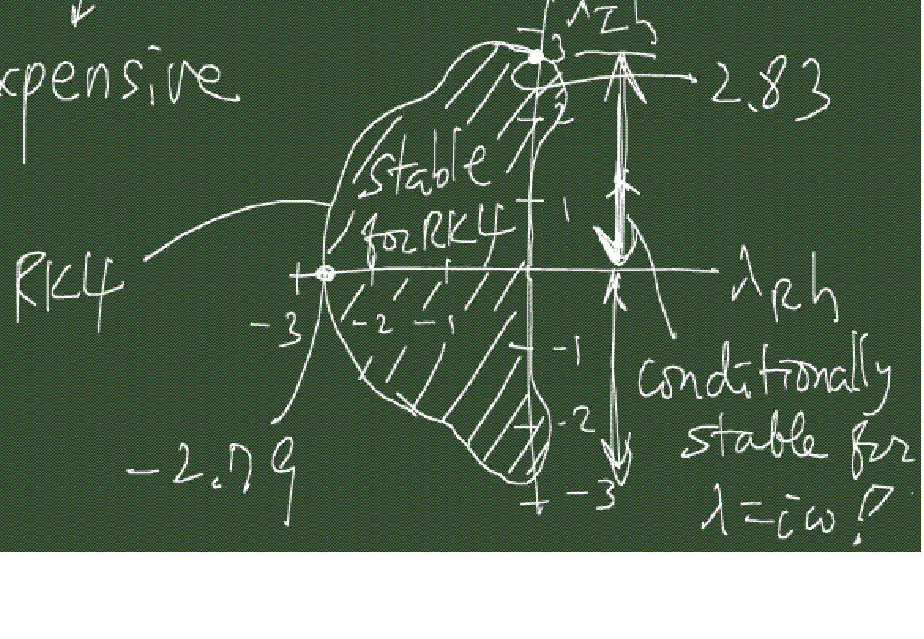
- 4th-order RK (most popular scheme) RK4

$$\begin{cases} y^{*}_{n+\frac{1}{2}} = y_n + \frac{h}{2} f(y_n, t_n) \\ y^{**}_{n+\frac{1}{2}} = y_n + \frac{h}{2} f(y^{*}_{n+\frac{1}{2}}, t_{n+\frac{1}{2}}) \\ y^{***}_{n+1} = y_n + h f(y^{**}_{n+\frac{1}{2}}, t_{n+\frac{1}{2}}) \\ y_{n+1} = y_n + h\left[\frac{1}{6} f(y_n, t_n) + \frac{1}{3} f(y^{*}_{n+\frac{1}{2}}, t_{n+\frac{1}{2}}) + \frac{1}{3} f(y^{**}_{n+\frac{1}{2}}, t_{n+\frac{1}{2}}) + \frac{1}{6} f(y^{***}_{n+1}, t_{n+1})\right] \end{cases}$$

( 4 function evaluations → RK4

RK5 ? → requires 6 ft. evaluations )

expensive



$$\sigma = 1 + \lambda h + \frac{1}{2}\lambda^2 h^2 + \frac{1}{6}\lambda^3 h^3 + \frac{1}{24}\lambda^4 h^4$$

RK4

- How to construct RK?

RK2? $\quad y' = f(y, t)$ two-stage

$$\begin{cases} k_1 = h f(y_n, t_n) \\ k_2 = h f(y_n + \beta k_1, t_n + \alpha h) \\ y_{n+1} = y_n + \gamma_1 k_1 + \gamma_2 k_2 \end{cases}$$

Find $\beta, \alpha, \gamma_1, \gamma_2$ to ensure the highest order of accuracy for RK2.

Taylor series of $k_2$

$$k_2 = h\left[ f(y_n, t_n) + \beta k_1 \frac{\partial f}{\partial y}\Big|_n + \alpha h \frac{\partial f}{\partial t}\Big|_n + \cdots \right]$$

$$y_{n+1} = y_n + \gamma_1 k_1 \quad k_1 = h f_n$$

$$+ \gamma_2 h\left[ f_n + \beta (h f_n) f_{y_n} + \alpha h f_{t_n} + \cdots \right]$$

$$\rightarrow y_{n+1} = y_n + (\gamma_1 + \gamma_2) h f_n$$
$$+ \gamma_2 \beta h^2 f_n f_{y_n} + \gamma_2 \alpha h^2 f_{t_n} + \cdots$$

Taylor series of $y$

$$y(t_{n+1}) = y(t_n) + h y'(t_n) + \frac{1}{2} h^2 y''(t_n) + \cdots$$
$$\left( y' = f, \quad y'' = f_t + f_y f \right)$$

Match the coeff.

$$\begin{cases} \gamma_1 + \gamma_2 = 1 \\ \gamma_2 \alpha = \frac{1}{2} \\ \gamma_2 \beta = \frac{1}{2} \end{cases}$$

3 eqs. & 4 unknowns

$\rightarrow$ $\alpha$ as free

$$\Rightarrow \gamma_2 = \frac{1}{2\alpha}, \quad \beta = \alpha, \quad \gamma_1 = 1 - \frac{1}{2\alpha}$$

$y' = f(y, t)$

$K_1 = h f(y_n, t_n)$

$K_2 = h f(y_n + \alpha k_1, t_n + \alpha h)$

$y_{n+1} = y_n + \left(1 - \frac{1}{2\alpha}\right) k_1 + \frac{1}{2\alpha} k_2$

model prob. $y' = \lambda y$

$k_1 = h \lambda y_n$

$k_2 = h \left[ \lambda (y_n + \alpha h \lambda y_n) \right]$

$y_{n+1} = y_n + \left(1 - \frac{1}{2\alpha}\right) h \lambda y_n$

$\quad + \frac{1}{2\alpha} h \left[ \lambda (1 + \alpha \lambda h) y_n \right]$

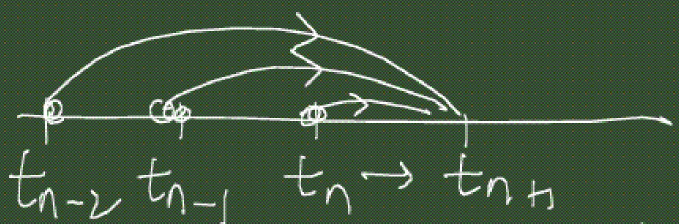
$\quad = y_n \left(1 + \lambda h + \frac{1}{2} \lambda^2 h^2\right)$ $\therefore$ 2nd order accurate

No Class on Wednesday
but we have mid-term on Wed.
6pm. here! (up to ch. 3)

## 4.9 Multi-step methods

Higher order accuracy is achieved by using data at previous time steps $t_{n-1}, t_{n-2}, \cdots$.

price: storage & memory
not self-starting




$$y' = f(y, t) \Rightarrow \frac{y_{n+1} - y_{n-1}}{2h} = f(y_n, t_n)$$

$$\boxed{y_{n+1} = y_{n-1} + 2h f(y_n, t_n)}$$

Leapfrog method (Central Diff.).

Not self-starting

e.g. get $y_1$ using EE.

Model eq: $y' = \lambda y$

$$y_{n+1} = y_{n-1} + 2h\lambda y_n$$

Assume $y_n = \sigma^n y_0$

$$\sigma^{n+1} = \sigma^{n-1} + 2\lambda h \sigma^n$$

$$\sigma^2 - 2\lambda h \sigma - 1 = 0$$

$$\to \sigma = \lambda h \pm \sqrt{\lambda^2 h^2 + 1}$$

Two roots

$$\sigma_1 = \lambda h + \sqrt{\lambda^2 h^2 + 1}$$
$$= 1 + \lambda h + \frac{1}{2}\lambda^2 h^2 - \frac{1}{8}\lambda^4 h^4 + \cdots$$

$\Rightarrow$ the method is 2nd-order accurate!

$$\sigma_2 = \lambda h - \sqrt{\lambda^2 h^2 + 1}$$

$$= -1 + \lambda h - \frac{1}{2}\lambda^2 h^2 + \frac{1}{8}\lambda^4 h^4 + \cdots$$

$\hookrightarrow$ "spurious root"

has no physical significance.

as $h \to 0$, $\sigma_2 = -1$.

For $\lambda$ real and negative, leapfrog leads to severe instability

since $|\sigma_2| > 1$.

the general sol. $y_n = c_1 \sigma_1^n + c_2 \sigma_2^n$

Find $c_1$ and $c_2$

$n=0$: $y_0 = c_1 + c_2$ — ①

Let $y_1$ be the sol. at step 1 obtained by some other method.

$n=1$: $y_1 = c_1 \sigma_1 + c_2 \sigma_2$ — ②

① & ② $\rightarrow$ $$C_1 = \frac{y_1 - y_0 \sigma_2}{\sigma_1 - \sigma_2}$$

$$C_2 = \frac{-y_1 + y_0 \sigma_1}{\sigma_1 - \sigma_2}$$

If we choose $y_1 = \sigma_1 y_0$, $\Rightarrow C_2 = 0$

spurious root is completely suppressed.

In general, the starting scheme plays a role in determing the level of contribution of the spurious root.

Even if the spurious root is initially suppressed, round-off-errors can restart it again.

For $\lambda = i\omega$, $\quad \sigma = \lambda h \pm \sqrt{1+\lambda^2 h^2}$

$$= i\omega h \pm \sqrt{1-\omega^2 h^2}$$

if $|\omega h| < 1$, $|\sigma|^2 = \omega^2 h^2 + 1 - \omega^2 h^2 = 1$

no amplitude error!

if $|\omega h| > 1$, $|\sigma|^2 = \left|\omega h \pm \sqrt{\omega^2 h^2 - 1}\right|^2 > 1$

unstable

- 2nd-order Adams-Bashforth method

(AB2) widely used

$$y_{n+1} = y_n + h \underbrace{y_n'}_{= f(y_n, t_n)} + \frac{1}{2} h^2 \underbrace{y_n''}_{= \frac{1}{h}(y_n' - y_{n-1}') + O(h)} + \underbrace{\frac{1}{6} h^3 y_n'''}_{} + \cdots$$

$y_n' = f(y_n, t_n)$

$y_{n-1}' = f(y_{n-1}, t_{n-1})$

$$y_{n+1} = y_n + h f(y_n, t_n) + \frac{1}{2} h \left[ f(y_n, t_n) - f(y_{n-1}, t_{n-1}) \right] + O(h^3)$$

$$\rightarrow \boxed{y_{n+1} = y_n + \frac{h}{2}\left[ 3 f(y_n, t_n) - f(y_{n-1}, t_{n-1}) \right]} + O(h^3)$$

AB2

globally second-order accurate

$$y' = \lambda y$$

$$y_{n+1} = y_n + \frac{h}{2}\left( 3\lambda y_n - \lambda y_{n-1} \right)$$

$$y_{n+1} - \left(1 + \frac{3}{2}\lambda h\right) y_n + \frac{1}{2}\lambda h\, y_{n-1} = 0$$

Assume $y_n = \sigma^n y_0$

$$\rightarrow \sigma^2 - \left(1 + \frac{3}{2}\lambda h\right)\sigma + \frac{1}{2}\lambda h = 0$$

$$\sigma = \frac{1}{2}\left(1+\frac{3}{2}\lambda h \pm \sqrt{1+\lambda h+\frac{9}{4}\lambda^2 h^2}\right)$$

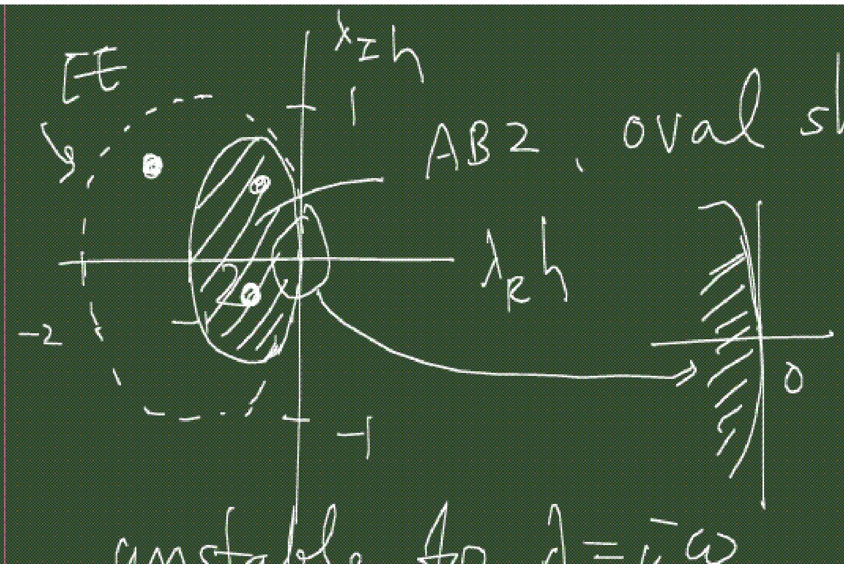
$$\left(\sqrt{1+x} = 1+\frac{1}{2}x-\frac{1}{8}x^2+\frac{3}{48}x^3+\cdots\right)$$

$$\rightarrow \quad \sigma_1 = 1+\lambda h+\frac{1}{2}\lambda^2 h^2+\cdots$$

$$\sigma_2 = \frac{1}{2}\lambda h-\frac{1}{2}\lambda^2 h^2+\cdots$$

$\sigma_1 \rightarrow$ second order accurate

$\sigma_2 \rightarrow 0$ as $h \rightarrow 0$ less dangerous.




unstable for $\lambda = i\omega$     mild

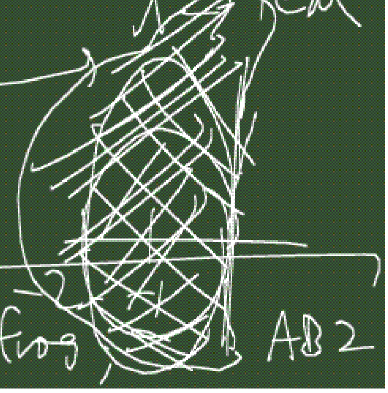
e.g. $\lambda = i\omega$   $|\sigma_1| = 1.000434$   instability

$\omega h = 0.2$   $|\sigma_1|^{100} = 1.02$

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0 \Rightarrow \lambda = i\omega$$

$$\frac{\partial u}{\partial t} + \boxed{c\frac{\partial u}{\partial x}} = \boxed{\nu\frac{\partial^2 u}{\partial x^2}}$$

$\lambda \rightarrow$ real



$\Rightarrow$ PDE F

EE, IE, TR, RK2, RK4, leapfrog, AB2

AB2 vs RK3

Kim, Moin & Moser (1987) JFM

$\rightarrow$ $\boxed{\text{AB2}}$ NASA Ames RC

$\boxed{\text{RK3}}$

AB2

$$y_{n+1} = y_n + \frac{h}{2}(3f_n - f_{n-1})$$

$y_0 \rightarrow y_1$ : EE $\Rightarrow$ $h'$

$t_n$ $h$ $t_{n+1}$ $t_{n+2}$

# 4.10 System of first order ODEs

ex. chemical reactions

$$\begin{cases} \dfrac{dy_1}{dt} = \alpha_{11} y_1^2 + \alpha_{12} y_1 \underline{y_2} + \cdots \\ \dfrac{dy_2}{dt} = \alpha_{21} \underline{y_1} y_2 + \alpha_{22} y_2^2 + \cdots \\ \cdots \end{cases}$$

Or, higher order ODE

$\rightarrow$ system of 1st order ODEs

ex. Blasius eq. $f''' + ff'' = 0$

$$\left.\begin{aligned} y_1 &\equiv f, \\ y_2 &\equiv f', \\ y_3 &\equiv f'' \end{aligned}\right\} \rightarrow \begin{cases} y_1' = y_2 \\ y_2' = y_3 \\ y_3' = -y_1 y_3 \end{cases}$$

From the conceptual pt. of view, there is only one fundamental difference bet. numerical sol. of one ODE and that of a system.

$\longrightarrow$ stiffness property

Single ODE

$$\frac{dy}{dt} = f(y,t)$$

System

$$\frac{dy_i}{dt} = f_i(t, y_1, y_2, \cdots, y_m)$$

$$i = 1, 2, \cdots, m$$

model prob.

$$\frac{dy}{dt} = \lambda y$$

$$\boxed{\frac{dy}{dt} = Ay}$$

$$\frac{dy}{dt} = Ay$$

Assume $A$ has a complete set of eigenvectors

$$A = S\Lambda S^{-1}$$

$$\frac{dy}{dt} = S\Lambda S^{-1} y$$

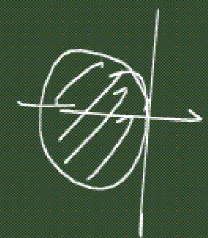
$$\frac{d}{dt}(S^{-1}y) = \Lambda S^{-1} y$$

$$(u \equiv S^{-1} y)$$

$$\rightarrow \frac{du}{dt} = \Lambda u \rightarrow \frac{du_i}{dt} = \lambda_i u_i$$

$$i = 1, 2, \ldots, m$$

EE: $u_i^n = (1+\lambda_i h)^n u_{0i}$



$$\Rightarrow |1+\lambda_i h| \le 1$$

largest eigenvalue → smallest h.

• $\frac{dy}{dt} = Ay \left( \Rightarrow \frac{y_{n+1} - y_n}{h} = Ay_n \right)$

EE: $y_{n+1} = y_n + hAy_n$

$= (I + hA)\, y_n$

$\to y_n = (I + hA)^n y_0 = B^n y_0$

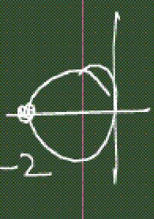
$\lim_{n \to \infty} B^n \to 0$ if eigenvalues of B have modulus < 1.

B

For stability, $|\alpha_i| \le 1$

$\alpha_i = 1 + h\lambda_i$ eigenvalues of

eigenvalues of A




$|1 + h\lambda_i| \le 1$

for $\lambda$ real, $h \le \frac{2}{|\lambda_{max}|}$.

Stiffness $\frac{|\lambda_{max}|}{|\lambda_{min}|} \gg 1$

→ system is stiff.

Stiffness can arise in physical systems w/ several degrees of freedom, but w/ widely different response times.

<u>ex</u>. a system composed of two springs, one very stiff and the other very flexible.

a mixture of chemical species w/ very different reaction rates.

Ex. $\begin{cases} u' = 998u - 1998v \\ v' = -999u - 1999v \end{cases}$

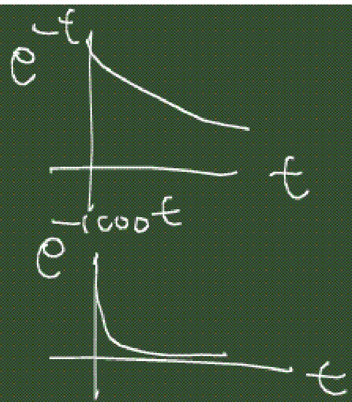
$u(0) = v(0) = 1$

$$\begin{pmatrix} u' \\ v' \end{pmatrix} = \begin{pmatrix} 998 & -1998 \\ -999 & -1999 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}$$

$\lambda_1 = -1, \quad \lambda_2 = -1000$

exact sol.

$$\begin{cases} u = 4\underline{e^{-t}} - 3\underline{e^{-1000t}} \\ v = -2e^{-t} + 3e^{-1000t} \end{cases}$$




EE: $h_{max} = \dfrac{2}{|\lambda_{max}|} = \dfrac{1}{500}$

Advance 5 timesteps

$t = 5 \times \dfrac{1}{500} = 0.01$.

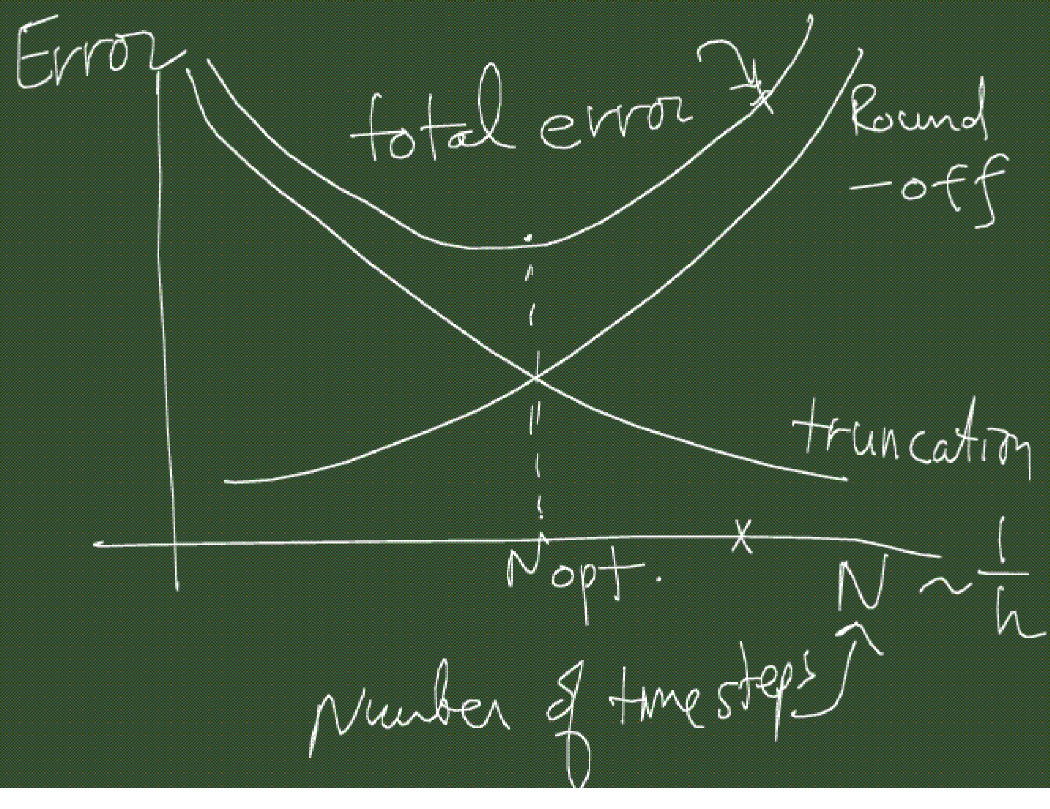
$$e^{-1000t} = 4.5 \times 10^{-5}$$

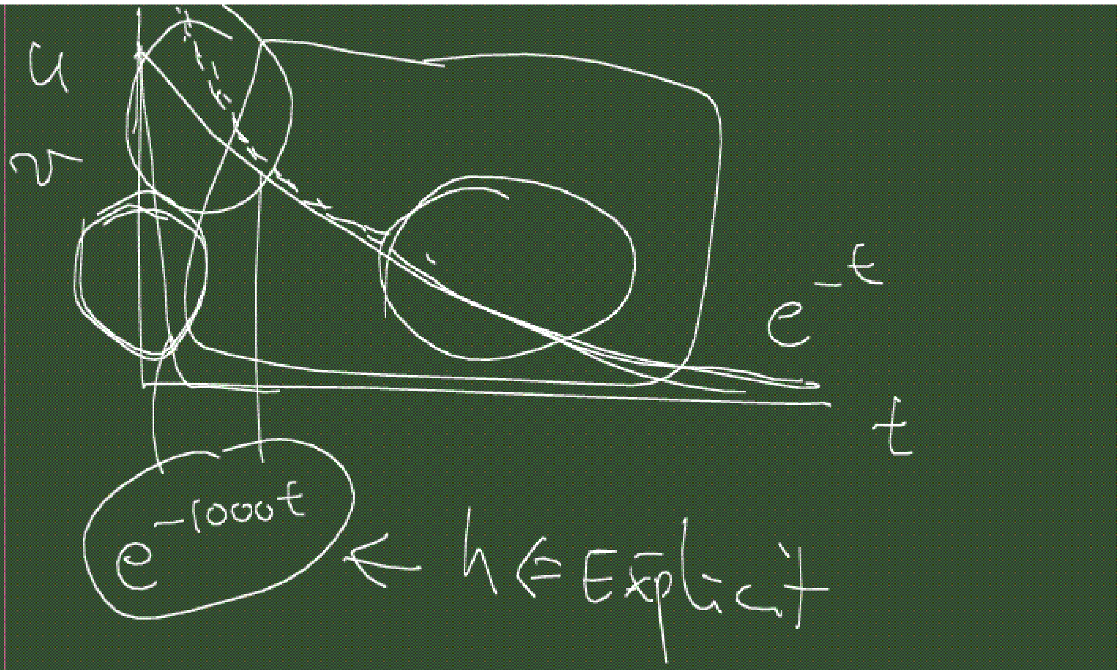
$$e^{-t} = 0.99$$

Take 3500 time steps to drive

$e^{-t}$ to $10^{-3}$.

$\longrightarrow$ Use implicit method






implicit method

→ nonlinear ODEs

① Linearization

$$\frac{d\underline{y}}{dt} = \underline{f}(y_1, y_2, \cdots, y_m) = \underline{f}(\underline{y})$$

TR: $\underline{y}_{n+1} = \underline{y}_n + \frac{h}{2}\left[\underline{f}(\underline{y}_{n+1}) + \underline{f}(\underline{y}_n)\right] + O(h^3)$

$$f_i(\underline{y}_{n+1}) = f_i(\underline{y}_n) + (y_{1\,n+1} - y_{1\,n})\frac{\partial f_i}{\partial y_1}$$

$$+ (y_{2\,n+1} - y_{2\,n})\frac{\partial f_i}{\partial y_2} + \dots + (y_{m\,n+1} - y_{m\,n})\frac{\partial f_i}{\partial y_m}$$

$$+ O(h^2)$$

$$= f_i(\underline{y}_n) + \sum_{j=1}^{m} (y_{j\,n+1} - y_{j\,n}) \left.\frac{\partial f_i}{\partial y_j}\right|_n + O(h^2)$$

$$= \underline{f}(\underline{y}_n) + A(\underline{y}_{n+1} - \underline{y}_n) + O(h^2)$$

$$A = \begin{bmatrix} \frac{\partial f_1}{\partial y_1} & \frac{\partial f_1}{\partial y_2} & \cdots & \frac{\partial f_1}{\partial y_m} \\ \frac{\partial f_2}{\partial y_1} & \ddots & & \vdots \\ \vdots & & & \vdots \\ \frac{\partial f_m}{\partial y_1} & \frac{\partial f_m}{\partial y_2} & \cdots & \frac{\partial f_m}{\partial y_m} \end{bmatrix}_{t=t_n}$$

$$\frac{\partial f_i}{\partial y_j}$$

Jacobian matrix

TR $\rightarrow$ $\underline{y}_{n+1} = \underline{y}_n + \frac{h}{2}\left[\underline{f}(\underline{y}_n) + A(\underline{y}_{n+1} - \underline{y}_n) + f(\underline{y}_n)\right] + O(h^3)$

$$\left(I - \frac{h}{2}A_n\right)\underline{y}_{n+1} = \left(I - \frac{h}{2}A_n\right)\underline{y}_n + h\,\underline{f}(\underline{y}_n).$$

Solve sys. of eqs at each time step.

$\rightarrow$ Linearization has an error for strongly nonlinear prob.

② Direct sol. of nonlinear eq.

$$\underline{F} = -\frac{h}{2}\underline{f}(\underline{y}_{n+1}) - \frac{h}{2}\underline{f}(\underline{y}_n) + (\underline{y}_{n+1} - \underline{y}_n) = 0$$

Newton-iterative method

$$F(x) = 0 \qquad k: \text{iteration index}$$

$$\frac{dF}{dx} = \frac{F^{k+1} - F^{k}}{x^{k+1} - x^{k}}, \qquad F^{k+1} = 0$$

$$\left.\frac{dF}{dx}\right|^{k} \underset{\sim}{(x^{k+1}} - x^{k}) = -F^{k}$$

$$\underline{F}(\underline{y}_{n+1}) = 0$$

$$\underbrace{\left.\frac{\partial F_i}{\partial y_j}\right|^{k}}_{A} \underbrace{\left(y_j^{k+1} - y_j^{k}\right)}_{I} = \underbrace{-F_i^{k}}$$

$$A = -\frac{h}{2}\left.\frac{\partial f_i}{\partial y_j}\right|^{k} + \delta_{ij}, \qquad \delta_{ij} = \left(\frac{\partial y_i}{\partial y_j}\right)$$

$$-F_i^{k} = +\frac{h}{2} f_i(\underline{y}^{k}) + \frac{h}{2} f_i(\underline{y}_n) - (\underline{y}^{k} - \underline{y}_n)$$

* Inherent instability

Consider the equation that might arise in a heat transfer prob.

$$y'' - k^2 y = 0$$

$$y(0) = y_0, \quad y'(0) = -k y_0$$

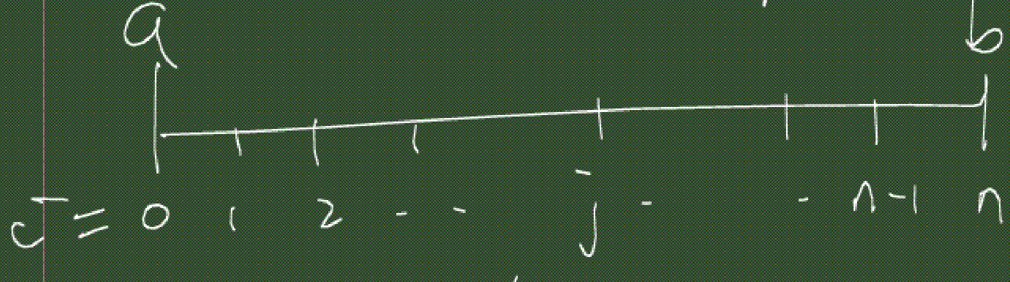
→ exact sol. $y = y_0 e^{-kx}$

Numerical sol. $y = c_1 e^{-kx} + \underline{\underline{c_2 e^{kx}}}$

truncation and round-off errors will be exponentially amplified and the calculation quickly diverges ⇒ inherently unstable

None of the standard method will produce the correct solution of it!

$$\phi(x) = f(x) + \int_a^b K(x,t)\,\phi(t)\,dt$$




$$\phi_j = f_j + \int_a^b k_j(t)\,\phi(t)\,dt$$

$$\int_a^b k_j(t)\,\phi(t)\,dt \qquad K(x,t)$$

$$= k_1(x)\,k_2(t)$$

$$= \int_0^1 + \int_1^2 + \cdots + \int_{n-1}^n$$

$$\int_i^{i+1} k_j(t)\,\phi(t)\,dt = \frac{1}{2}\Big[k_j(i+1)\,\underline{\phi(i+1)} + k_j(i)\,\underline{\phi(i)}\Big]$$

$$\boxed{[\quad][\phi] = b}$$

## 4.11 Boundary value problems

2nd-order ODE

$$y'' = f(y', y, x)$$

$$y(0) = y_0, \quad y(L) = y_L$$

① shooting method

② direct method

① shooting method

Convert the eq. to 2 1st order eqs.

$$\begin{matrix} u = y \\ v = y' \end{matrix} \rightarrow \begin{cases} u' = v \\ v' = f(v, u, x) \end{cases}$$

$$u(0) = y_0, \quad u(L) = y_L$$

we need $v(0)$!

Guess for $v(0)$ $(= E)$

$\rightarrow$ $u(0)$ & $v(0)$ to integrate to $x=L$.

$\rightarrow$ check if $\underbrace{u(L) = y_L}$

$\rightarrow$ If not, try a different $v(0)$.

$\rightarrow$ <u>Iterative process?</u>

consider a linear eq.

$$y''(x) + A(x)\,y'(x) + B(x)\,y(x) = f(x)$$

$$y(0) = y_0,\ y(L) = y_L$$

Start two sols, $y_1(x)$ and $y_2(x)$, obtained from 2 guesses for $\underset{\downarrow}{y'(0)}$.

$v(0)$

Since eq. is linear, the linear combination of $y_1$ and $y_2$ is also a sol.

$$\underline{y(x)} = c_1 y_1(x) + c_2 y_2(x)$$

$$x=0: \quad \underset{1}{\cancel{y}(0)} = c_1 \underset{1}{y_1\cancel{(0)}} + c_2 \underset{1}{y_2\cancel{(0)}}$$

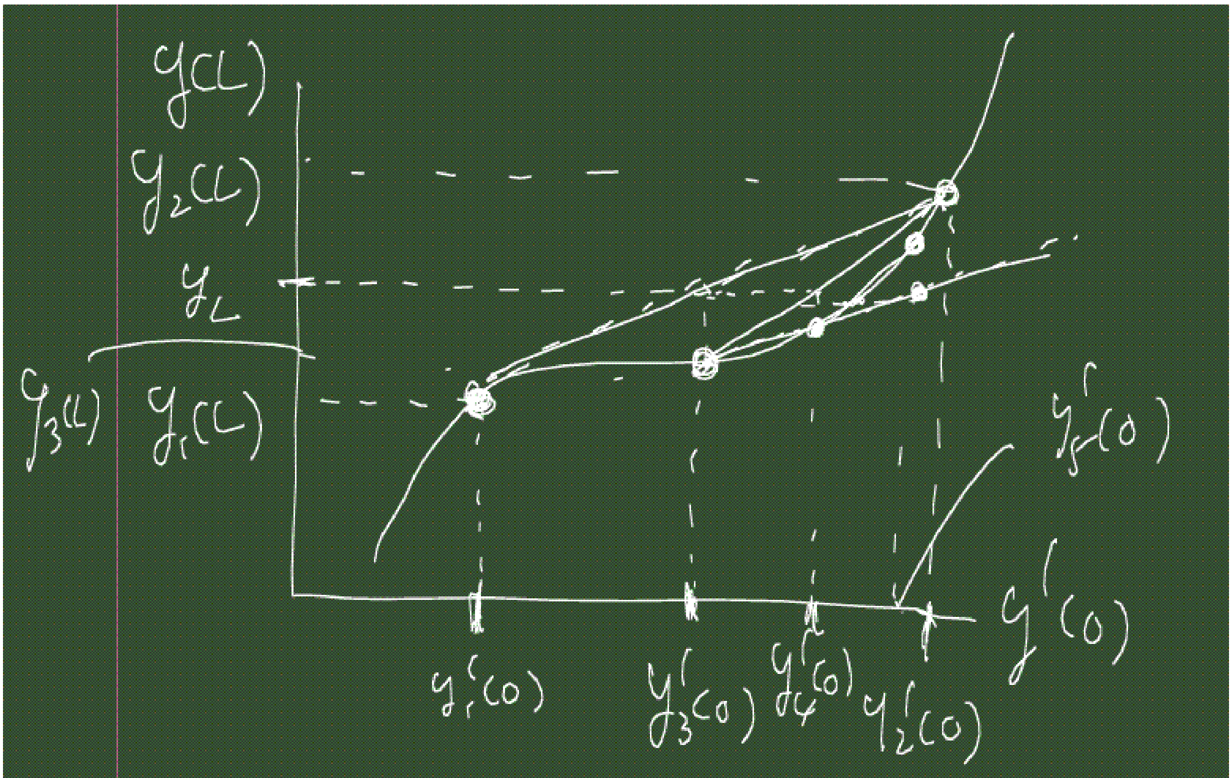
$$\rightarrow \quad c_1 + c_2 = 1 \quad \text{— (1)}$$

$$x=L: \quad y(L) = \underline{y_L} = c_1 \underline{y_1(L)} + c_2 \underline{y_2(L)} \quad \text{(2)}$$

$$\Rightarrow c_1 = \frac{y_L - y_2(L)}{y_1(L) - y_2(L)}, \quad c_2 = 1 - c_1$$

What about nonlinear eq.?

$y(L)$ is a nonlinear ft. of $y'(0)$,
$\rightarrow v(0)$




Secant method.

$$y_1'(0) \rightarrow y_1(L)$$

$$y_2'(0) \rightarrow y_2(L)$$

Form a straight line between

$(y_1'(0), y_1(L))$ & $(y_2'(0), y_2(L))$.

$$\frac{y - y_2(L)}{y' - y_2'(0)} = \frac{y_1(L) - y_2(L)}{y_1'(0) - y_2'(0)} = m$$

$\rightarrow \quad y' = y_2'(0) + \dfrac{y - y_2(L)}{m}$

Next guess $\quad y_3'(0) = y_2'(0) + \dfrac{y_L - y_2(L)}{m}$.

In general,

$$y_{n+1}'(0) = y_n'(0) + (y_L - y_n(L)) \frac{y_n'(0) - y_{n-1}'(0)}{y_n(L) - y_{n-1}(L)}$$

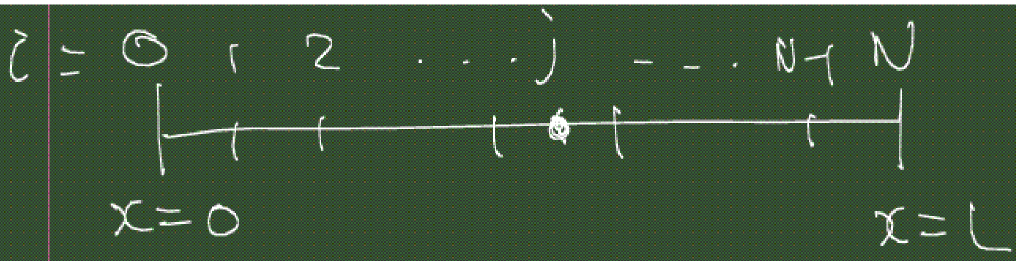
- Direct method

Approximate the derivative in the diff'l eq. with FD.

Incorporate the b.c. as required.

$$y''(x) + A(x)\, y'(x) + B(x)\, y(x) = C(x)$$

$$y(0) = y_0, \quad y(L) = y_L .$$




$$\frac{y_{j+1} - 2y_j + y_{j-1}}{h^2} + A_j \frac{y_{j+1} - y_{j-1}}{2h} + B_j y_j = C_j$$

$$j = 1, 2, \dots, N-1$$

$$\underbrace{\left(\frac{1}{h^2} + \frac{A_j}{2h}\right)}_{\alpha_j} y_{j+1} + \underbrace{\left(B_j - \frac{2}{h^2}\right) y_j}_{\beta_j} + \underbrace{\left(\frac{1}{h^2} - \frac{A_j}{2h}\right)}_{\gamma_j} y_{j-1} = C_j$$

$$\rightarrow \alpha_j y_{j+1} + \beta_j y_j + \gamma_j y_{j-1} = C_j$$

(a) $j=1$. $\alpha_1 y_2 + \beta_1 y_1 + \gamma_1 \boxed{y_0} = c_1$

$$\rightarrow \alpha_1 y_2 + \beta_2 y_1 = c_1 - \gamma_1 y_0$$

(b) $j=N-1$, $\beta_{N-1} y_{N-1} + \gamma_{N-1} y_{N-2}$

$$= c_{N-1} - \alpha_{N-1} y_N \quad (y_N = y_L)$$

$$\begin{bmatrix} \beta_1 & \alpha_1 & & \\ \gamma_2 & \beta_2 & \alpha_2 & 0 \\ & \ddots & \ddots & \ddots \\ 0 & & \ddots & \ddots \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{N-1} \end{bmatrix} = \begin{bmatrix} c_1 - \gamma_1 y_0 \\ c_2 \\ \vdots \\ c_{N-2} \\ c_{N-1} - \alpha_{N-1} y_L \end{bmatrix}$$

Mixed b.c

✓ $a y(0) + b y'(0) = g$

$$y'(0) = \frac{y_1 - y_0}{h}$$

$$y'(0) = \frac{-3y_0 + 4y_1 - y_2}{2h}$$

Solve for $y(0)$ in terms of $y_1$ and $y_2$.

Increase accuracy by using higher order difference.

$$y_j'' = \frac{-y_{j-2} + 16y_{j-1} - 30y_j + 16y_{j+1} - y_{j+2}}{12h^2}$$

$$\hat{j}, \hat{j}-1, \hat{j}-2, \hat{j}+1, \hat{j}+2$$

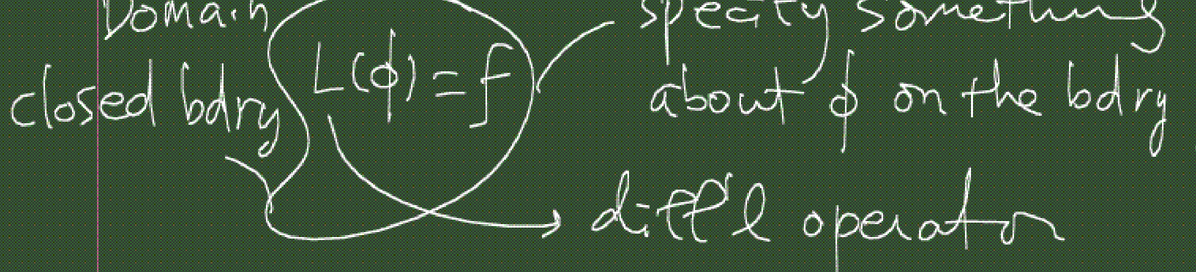
difficulties near boundary.

→ use lower order one-side difference scheme near bdry.

## 5. Numerical Sol. of PDE.

- physical classification

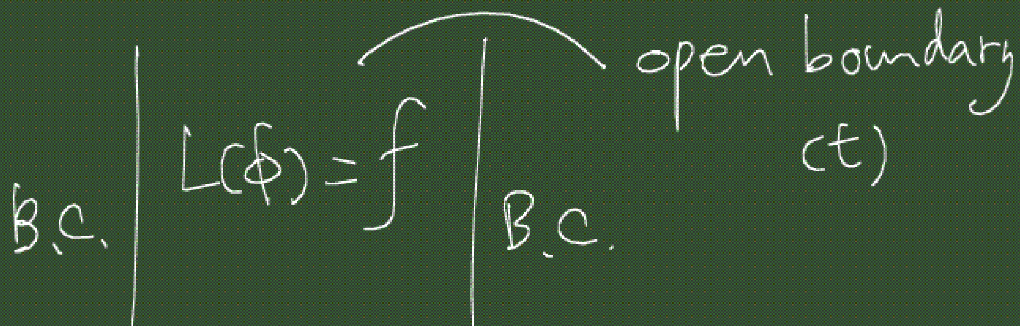
1) equilibrium probs. ⇒ Elliptic PDE.
   steady state probs.



2) propagation probs.

Initial value probs.

Transient nature




Initial data $I_i(\phi) = h_i$

- Mathematical classification

Quasi-linear second order eq.

$$\underline{a u_{xx} + b u_{xy} + c u_{yy} = f}$$

a, b, c, f can be fts. of $u$, $u_x$ & $u_y$.

<u>Hyperbolic PDE</u> if $b^2 - 4ac > 0$

two real characteristics

parabolic PDE if $b^2 - 4ac = 0$

one real characteristics

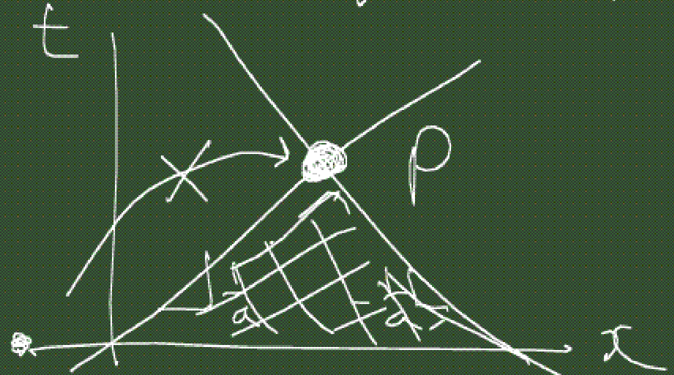
Elliptic PDE if $b^2 - 4ac < 0$

no real characteristics

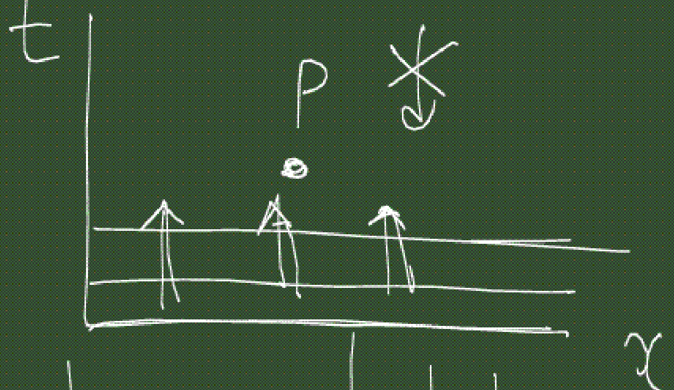
Ex.

Ex. $\frac{\partial^2 u}{\partial t^2} - a^2 \frac{\partial^2 u}{\partial x^2} = f$ : hyperbolic

two char. $\frac{dt}{dx} = \pm \frac{1}{a}$

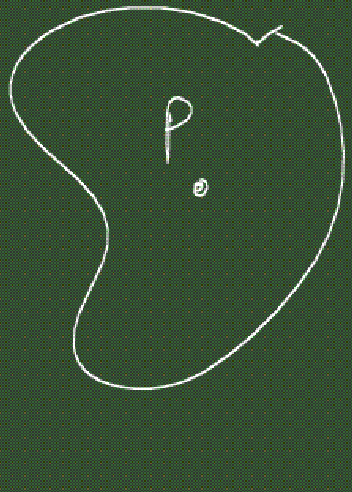


$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2} \quad \text{parabolic}$$




p knows what has happened previously along the entire x-axis.

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = 0 \quad \text{elliptic}$$




at any point P, the sol. is influenced by all other pts.

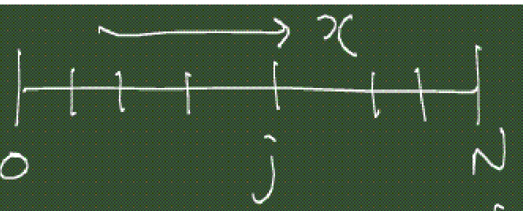
노트 제목 No class on Wednesday 2009-11-02

5.1 Semi-discretization:

PDE to a sys. of ODEs

- $\dfrac{\partial \phi}{\partial t} = \alpha \dfrac{\partial^2 \phi}{\partial x^2}$ : diffusion eq.

$$\begin{cases} \phi(0,t) = \phi(L,t) = 0 \\ \phi(x,0) = f \end{cases}$$




@ j, $\dfrac{\partial \phi_j}{\partial t} = \alpha \dfrac{\phi_{j+1} - 2\phi_j + \phi_{j-1}}{\Delta x^2}$

$j = 1, 2, \cdots, N-1$

$$\frac{\partial \phi}{\partial t} = \underset{\sim}{A} \phi \qquad \phi = \begin{pmatrix} \phi_1 \\ \vdots \\ \phi_{N-1} \end{pmatrix}$$

$$\frac{d}{dt}\begin{bmatrix}\phi_1\\ \phi_2\\ \vdots\\ \phi_{N-1}\end{bmatrix} = \frac{\alpha}{\Delta x^2}\begin{bmatrix}-2 & 1 & & & 0\\ 1 & -2 & 1 & & \\ & 1 & -2 & 1 & \\ & & \ddots & \ddots & \ddots\\ 0 & & & 1 & -2\end{bmatrix}\begin{bmatrix}\phi_1\\ \phi_2\\ \vdots\\ \phi_{N-1}\end{bmatrix}$$

Semi-discretization converts PDE to a system of ODEs.

$$A = \frac{\alpha}{\Delta x^2} \underbrace{B[1,-2,1]}_{\text{banded}}$$

$$\boxed{\begin{aligned} &A = B[a,b,c] \quad m \times m \\ &\lambda_j = b + 2\sqrt{ac}\cos\alpha_j, \quad j = 1,2,\dots,m \\ &\alpha_j = \frac{j\pi}{m+1} \end{aligned}}$$

$$\therefore \lambda_j = \frac{\alpha}{\Delta x^2}\left(-2+2\cos\frac{j\pi}{N}\right), \quad j=1,2,\cdots,N-1$$

$$\lambda_1 = \frac{\alpha}{\Delta x^2}\left(-2+2\cos\frac{\pi}{N}\right)$$

$$N \gg 1 \rightarrow \cos\frac{\pi}{N} = 1-\frac{1}{2!}\left(\frac{\pi}{N}\right)^2+\cdots$$

$$\lambda_1 \sim -\left(\frac{\pi}{N}\right)^2\frac{\alpha}{\Delta x^2}$$

$$\lambda_{N-1} \sim -4\frac{\alpha}{\Delta x^2}$$

$$\frac{\lambda_{N-1}}{\lambda_1} \sim \frac{4N^2}{\pi^2} \Rightarrow \text{large for large } N!$$

$\Rightarrow$ stiff

Eigenvalues are real & negative.

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0 \; ; \text{ wave eq.}$$

semi-discretization

$$\frac{\partial u_j}{\partial t} + c\,\frac{u_{j+1} - u_{j-1}}{2\Delta x} = 0$$

$$\frac{\partial u}{\partial t} = -\frac{c}{2\Delta x} B[-1, 0, 1]\, u$$

$$\lambda_j = -\frac{c}{2\Delta x} \cdot 2i \cdot \cos\frac{j\pi}{N}$$

$$= -i\frac{c}{\Delta x}\cos\frac{j\pi}{N}$$

purely imaginary.

↳ wave-like behavior

Matrix stability analysis

$$\frac{\partial \phi}{\partial t} = \alpha \frac{\partial^2 \phi}{\partial x^2}$$

$$\text{SD} \rightarrow \frac{\partial \phi}{\partial t} = \frac{\alpha}{\Delta x^2} B[1,-2,1]\,\phi$$

$$\text{EE}: \frac{\phi^{n+1}-\phi^n}{\Delta t} = \frac{\alpha}{\Delta x^2} B[1,-2,1]\,\phi^n$$

$$\rightarrow \phi^{n+1} = \left(I + \Delta t \frac{\alpha}{\Delta x^2} B\right)\phi^n$$

$$\rightarrow \phi^n = \left(I + \Delta t \frac{\alpha}{\Delta x^2} B\right)^n \phi^0$$

For stability,

$$\left| 1 + \Delta t \frac{\alpha}{\Delta x^2} \lambda_j \right| \leq 1$$

$$\rightarrow \Delta t \leq \frac{2}{\alpha/\Delta x^2 |\lambda_j|}$$

worst case $\lambda_{max} = -4$

$$\therefore \boxed{\Delta t \leq \frac{\Delta x^2}{2\alpha}}$$

very restrictive for small $\Delta x$.

More accuracy in $x \to$ smaller $\Delta t$

$\Delta t \sim \Delta x^2$ !

## 5.2 Von Neumann stability analysis

$$\frac{\partial \phi}{\partial t} = \alpha \frac{\partial^2 \phi}{\partial x^2}$$

$$SD \longrightarrow \frac{\partial \phi_j}{\partial t} = \alpha \frac{\phi_{j+1} - 2\phi_j + \phi_{j-1}}{\Delta x^2}$$

EE:

$$\frac{\phi_j^{n+1} - \phi_j^n}{\Delta t} = \alpha \frac{\phi_{j+1}^n - 2\phi_j^n + \phi_{j-1}^n}{\Delta x^2}$$

Assume sol. of the form

$$\boxed{\phi_j^n = \sigma^n e^{ikx_j}}$$

assume spatial periodicity.

Then, $\sigma^{n+1} e^{ikx_j} - \sigma^n e^{ikx_j}$

$$= \frac{\alpha \Delta t}{\Delta x^2}\left(\sigma^n e^{ikx_{j+1}} - 2\sigma^n e^{ikx_j} + \sigma^n e^{ikx_{j-1}}\right)$$

$$x_{j+1} = x_j + \Delta x, \quad x_{j-1} = x_j - \Delta x$$

$$\text{RHS} = \frac{\alpha \Delta t}{\Delta x^2} \sigma^n e^{ikx_j}\left(2\cos k\Delta x - 2\right)$$

$$\text{LHS} = \sigma^n e^{ikx_j}(\sigma - 1)$$

$$\therefore \quad \sigma = 1 + \frac{\alpha \Delta t}{\Delta x^2}\left(2\cos k\Delta x - 2\right)$$

Stability $|\sigma| \le 1$

$$\Delta t \le \frac{2}{\alpha/\Delta x^2 \left(2 - 2\cos k\Delta x\right)}$$

worst case : $\cos k\Delta x = -1$

$\Rightarrow \boxed{\Delta t \le \frac{\Delta x^2}{2\alpha}}$ same result as mat. stab. anal.

Von Neumann stability analysis

- works for const. coeff diff'l eq.

$$\frac{\partial u}{\partial t} = \alpha(t)\frac{\partial^2 u}{\partial x^2} \qquad \Delta t \le \frac{\Delta x^2}{2\alpha(t)}$$

- assumes that the bdry conds. are periodic.

In virtually all the cases, numerical stability problems arise solely from (full) discretization of the PDE and not from the bdry conds.

5.3 Modified wavenumber analysis

$$\frac{\partial \phi}{\partial t} = \alpha \frac{\partial^2 \phi}{\partial x^2}$$

Assume $\phi(x,t) = \psi(t)\, e^{ikx}$

$$\frac{d\psi}{dt} \cancel{e^{ikx}} = \alpha(-k^2)\psi \cancel{e^{ikx}}$$

$$\rightarrow \frac{d\psi}{dt} = -\alpha k^2 \psi$$

(Model eq. $\frac{dy}{dt} = \lambda y$)

CD2 (SD) $\frac{\partial \phi_j}{\partial t} = \alpha \frac{\phi_{j+1} - 2\phi_j + \phi_{j-1}}{\Delta x^2}$

Assume $\phi_j = \psi(t)\, e^{ikx_j}$

$$\frac{d\psi}{dt} e^{ikx_j} = \frac{\alpha}{\Delta x^2}\left[\psi e^{ikx_{j+1}} - 2\psi e^{ikx_j} + \psi e^{ikx_{j-1}}\right)$$

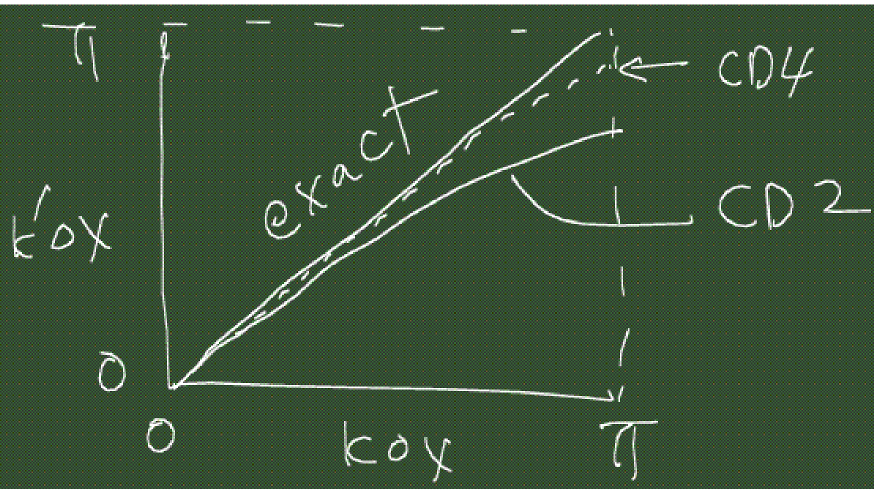
$$\rightarrow \frac{d\psi}{dt} = \frac{2\alpha}{\Delta x^2}\left(\cos k\Delta x - 1\right)\psi$$

$$= -\alpha \frac{2(1-\cos k\Delta x)}{\Delta x^2}\psi$$

$$k'^2 = \frac{2}{\Delta x^2}\left(1-\cos k\Delta x\right)$$

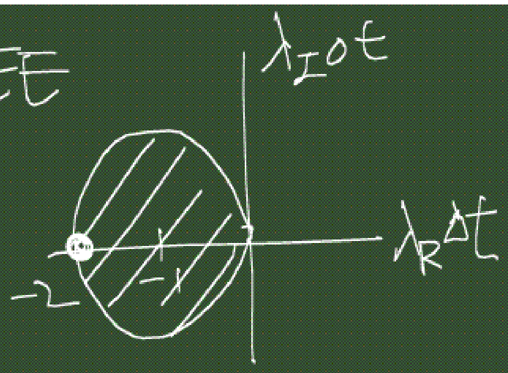
$k'$: modified wavenumber

$$\rightarrow \frac{d\psi}{dt} = -\alpha k'^2 \psi$$




$$\frac{d\psi}{dt} = -\alpha k'^2\psi = \lambda\psi$$

EE




$$\Delta t \le \frac{2}{|\lambda|} = \frac{2}{\alpha \frac{2}{\Delta x^2}(1-\cos k\Delta x)}$$

worst case : $\cos k\Delta x = -1$

$$\Delta t \le \frac{\Delta x^2}{2\alpha} \qquad \text{RK4: } \Delta t \le \frac{2.79\,\Delta x^2}{4\alpha}$$

Modified wavenumber analysis

- calculate the modified wavenumber for spatial derivatives
- use results from ODE with $\lambda$ replaced w/ the worst case for $k'$.

노트 제목 2009-11-09

$[LT^{-1}]$

- $\dfrac{\partial u}{\partial t} + c\underline{\underline{\dfrac{\partial u}{\partial x}}} = 0$ : wave eq.

$u(x,t) = \psi(t)\, e^{ikx}$

$\dfrac{d\psi}{dt}\cancel{e^{ikx}} + c\, ik\, \psi\, \cancel{e^{ikx}} = 0$

$\dfrac{d\psi}{dt} = \underline{-ikc}\,\psi$ — 1

For FD in x (CD2)

$\dfrac{\partial u_j}{\partial t} + c\,\dfrac{u_{j+1} - u_{j-1}}{2\Delta x} = 0$

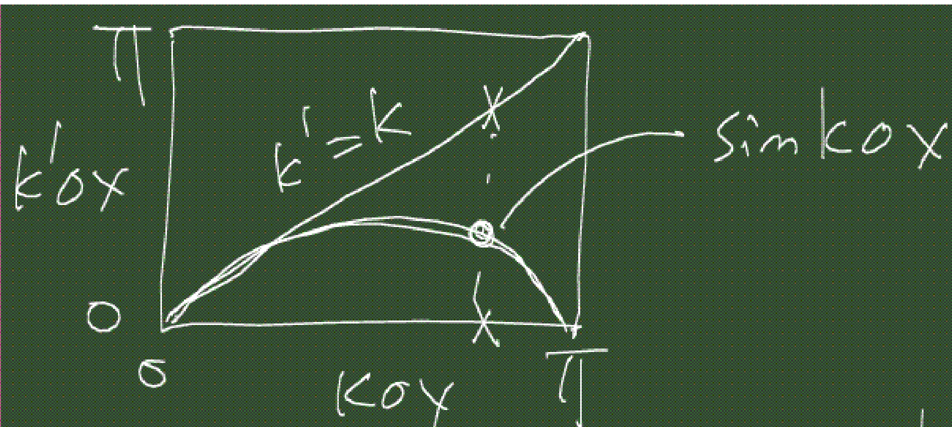
$u_j = \psi(t)\, e^{ikx_j}$

$\dfrac{d\psi}{dt} e^{ikx_j} + \dfrac{c}{2\Delta x}\left(\psi e^{ikx_{j+1}} - \psi e^{ikx_{j-1}}\right) = 0$

$$\rightarrow \frac{d\psi}{dt} + \frac{ic}{\Delta x} \sin k\Delta x \cdot \psi = 0$$

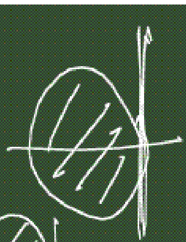
$$\rightarrow \frac{d\psi}{dt} = -i \frac{\sin k\Delta x}{\Delta x} c \psi \quad \text{—} (2)$$

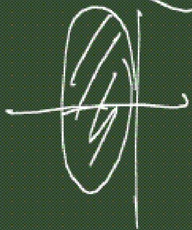
$\therefore k' = \frac{\sin k\Delta x}{\Delta x}$ ; modified wavenumber

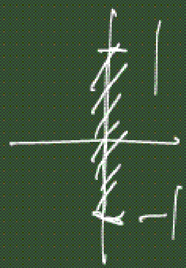



$\frac{d\psi}{dt} = \lambda\psi$ , $\lambda = -i \frac{\sin k\Delta x}{\Delta x} c$

purely imaginary

EE  unstable

RK2  unstable

Leapfrog method 



$|\lambda \Delta t| \le 1$

$$\Delta t \le \frac{1}{|\lambda|} = \frac{1}{\left|\frac{\sin k\Delta x}{\Delta x} c\right|}$$

worst case : $\sin k\Delta x = 1$

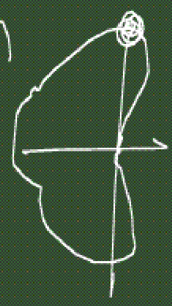
$$\rightarrow \Delta t_{max} = \frac{\Delta x}{c}$$

$$\rightarrow \boxed{\frac{c\Delta t}{\Delta x}} \le 1$$

non-dimensional parameter

CFL number

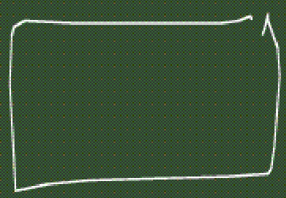
CFL: Courant, Friedrich and Lewy



RK4 → CFL ≤ 2.83

$\delta t_{max} = \frac{\delta x}{c}$ : not bad as compared to the diff. eq.

$\delta x \to \frac{\delta x}{2} \Rightarrow \delta t \to \frac{\delta t}{2}$



$$\begin{array}{ccc} \delta x_1 & \longrightarrow & \delta t_1 \\ \downarrow & & \downarrow \\ \delta x_2 = \frac{\delta x_1}{2} & & \delta t_2 = \frac{\delta t_1}{2} \end{array}$$

CPU time = $2 \times 2 = 4$ times

## 5.4 Implicit time advancement

- **Crank – Nicolson method**

$$\frac{\partial \phi}{\partial t} = \alpha \frac{\partial^2 \phi}{\partial x^2}$$

apply the trapezoidal method

$$\frac{\phi_j^{n+1} - \phi_j^n}{\Delta t} = \frac{\alpha}{2}\left[\frac{\partial^2 \phi^{n+1}}{\partial x^2} + \frac{\partial^2 \phi^n}{\partial x^2}\right]_j$$

$$= \frac{\alpha}{2}\left[\frac{\phi_{j+1}^{n+1} - 2\phi_j^{n+1} + \phi_{j-1}^{n+1}}{\Delta x^2} + \frac{\phi_{j+1}^{n} - 2\phi_j^{n} + \phi_{j-1}^{n}}{\Delta x^2}\right]$$

$$\beta \equiv \frac{\alpha \Delta t}{2 \Delta x^2}$$

$$\rightarrow -\beta \phi_{j+1}^{n+1} + (1+2\beta)\phi_j^{n+1} - \beta \phi_{j-1}^{n+1}$$

$$= \beta \phi_{j+1}^{n} + (1-2\beta)\phi_j^{n} + \beta \phi_{j-1}^{n}$$

$$j = 1, 2, \cdots, N-1$$

tri-diagonal matrix system

Solve for $\phi^{n+1}$ → $O(N)$ operation

- Application of implicit methods to PDE always requires solving a system of algebraic eqs.

## Stability

for trapezoidal method,

$$y' = \lambda y, \quad \sigma = \frac{1+\lambda \Delta t/2}{1-\lambda \Delta t/2} \quad (y^n = \sigma^n y_0)$$

$$\frac{\partial \phi}{\partial t} = \alpha \frac{\partial^2 \phi}{\partial x^2} \xrightarrow[\text{CD2}]{} k'^2 = \frac{2}{\Delta x^2}(1-\cos k\Delta x)$$

$$\hookrightarrow \left( \frac{d\psi}{dt} = -\alpha k'^2 \psi \right)$$

$$\lambda = -\alpha k'^2$$

$$\sigma = \frac{1-\alpha \frac{\Delta t}{\Delta x^2}(1-\cos k\Delta x)}{1+\alpha \frac{\Delta t}{\Delta x^2}(1-\cos k\Delta x)}$$

$|\sigma| \leq 1$ ∴ unconditionally stable

for large $\Delta t$, $\sigma \to$ '$-1$'

## 5.5 Accuracy via modified eq.

Since the numerical sol. is an approx. of the exact sol., it does not exactly satisfy the continuous PDE at hand, but it satisfies a modified PDE.

Let $\tilde{\phi}$ be the exact sol., and $\phi$ be the numerical sol.

(e.g. from EE + CD2)

$$\frac{\partial \tilde{\phi}}{\partial t} = \alpha \frac{\partial^2 \tilde{\phi}}{\partial x^2}$$

$$\frac{\phi_j^{n+1} - \phi_j^n}{\Delta t} = \alpha \frac{\phi_{j+1}^n - 2\phi_j^n + \phi_{j-1}^n}{\Delta x^2}$$

$$L(\phi_j^n) \equiv \frac{\phi_j^{n+1} - \phi_j^n}{\Delta t} - \alpha \frac{\phi_{j+1}^n - 2\phi_j^n + \phi_{j-1}^n}{\Delta x^2}$$

difference operator

Taylor series expansion

$$\phi_j^{n+1} = \phi_j^n + \Delta t \frac{\partial \phi_j^n}{\partial t} + \frac{1}{2}\Delta t^2 \frac{\partial^2 \phi_j^n}{\partial t^2} + \cdots$$

$$\rightarrow \frac{\phi_j^{n+1} - \phi_j^n}{\Delta t} = \frac{\partial \phi_j^n}{\partial t} + \frac{1}{2}\Delta t \frac{\partial^2 \phi_j^n}{\partial t^2} + \cdots$$

Similarly,

$$\frac{\phi_{j+1}^n - 2\phi_j^n + \phi_{j-1}^n}{\Delta x^2} = \left.\frac{\partial^2 \phi^n}{\partial x^2}\right|_j + \frac{\Delta x^2}{12} \left.\frac{\partial^4 \phi^n}{\partial x^4}\right|_j + \cdots$$

Then,

$$L(\phi_j^n) = \frac{\partial \phi_j^n}{\partial t} + \frac{1}{2}\Delta t \frac{\partial^2 \phi_j^n}{\partial t^2}$$

$$- \alpha \left.\frac{\partial^2 \phi^n}{\partial x^2}\right|_j - \alpha \frac{\Delta x^2}{12} \left.\frac{\partial^4 \phi^n}{\partial x^4}\right|_j + \cdots$$

$$L(\phi) - \left(\frac{\partial \phi}{\delta t} - \alpha \frac{\partial^2 \phi}{\delta x^2}\right)$$

$$= -\alpha \frac{\delta x^2}{12} \frac{\partial^4 \phi}{\delta x^4} + \frac{\delta t}{2} \frac{\partial^2 \phi}{\delta t^2} + \cdots .$$

$L(\phi) = 0 \quad \because \phi$ is a num. sol.

Then, the numerical sol. actually Satisfies the following modified PDE.

$$\boxed{\frac{\partial \phi}{\delta t} - \alpha \frac{\partial^2 \phi}{\delta x^2} = \alpha \frac{\delta x^2}{12} \frac{\partial^4 \phi}{\delta x^4} - \frac{\delta t}{2} \frac{\partial^2 \phi}{\delta t^2}}$$

As $\Delta x$ & $\delta t \to 0$, $\dfrac{\partial \tilde{\phi}}{\delta t} = \alpha \dfrac{\partial^2 \tilde{\phi}}{\delta x^2}$

Error

$$\varepsilon = L(\tilde{\phi}) = -\alpha \frac{\delta x^2}{12} \frac{\partial^4 \tilde{\phi}}{\delta x^4} + \frac{\delta t}{2} \frac{\partial^2 \tilde{\phi}}{\delta t^2} + \cdots$$

here $\frac{\partial^2 \hat{\phi}}{\partial t^2} = \frac{\partial}{\partial t}\left(\frac{\partial \hat{\phi}}{\partial t}\right) = \frac{\partial}{\partial t}\left(\alpha \frac{\partial^2 \hat{\phi}}{\partial x^2}\right)$

$$= \alpha^2 \frac{\partial^4 \hat{\phi}}{\partial x^4}$$

$\rightarrow \varepsilon = \left(-\alpha \frac{\Delta x^2}{12} + \alpha^2 \frac{\Delta t}{2}\right) \frac{\partial^4 \hat{\phi}}{\partial x^4} + \cdots$

If we set $\alpha \frac{\Delta x^2}{12} = \alpha^2 \frac{\Delta t}{2}$,

we can increase the accuracy.

$$\Delta t = \frac{1}{6} \frac{\Delta x^2}{\alpha} \quad \left(\text{cf.} \quad \Delta t \leq \frac{\Delta x^2}{2\alpha}\right)$$

this is within the stability limit but is rather restrictive.

5.6 Dufort - Frankel Method:

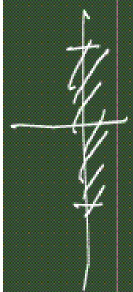
an inconsistent numerical method

$$\frac{\partial \phi}{\partial t} = \alpha \frac{\partial^2 \phi}{\partial x^2}$$

2nd-order in time & space ✓

Leapfrog method

$$\frac{\phi_j^{n+1} - \phi_j^{n-1}}{2\Delta t} = \alpha \frac{\phi_{j+1}^n - 2\phi_j^n + \phi_{j-1}^n}{\Delta x^2}$$



Leapfrog method is unconditionally unstable for $\lambda$ real & negative.

Dufort - Frankel scheme

$$\underline{\phi_j^n = \frac{1}{2}\left(\phi_j^{n+1} + \phi_j^{n-1}\right) + O(\Delta t^2)}$$

$$\rightarrow \phi_j^{n+1} - \phi_j^{n-1} = \frac{2\alpha \Delta t}{\Delta x^2}\left(\phi_{j+1}^n - \phi_j^{n+1} - \phi_j^{n-1} + \phi_{j-1}^n\right)$$

$(\beta \equiv 2\alpha\Delta t/\Delta x^2)$

$$\rightarrow \boxed{(1+2\beta)\,\phi_j^{n+1} = (1-2\beta)\,\phi_j^{n-1} + 2\beta\,\phi_{j+1}^{n} + 2\beta\,\phi_{j-1}^{n}}$$

It turns out that this method is <u>unconditionally stable</u>

No matrix inversion is required which is expected from implicit methods! → too good to be true.

what is the modified eq. for the Du Fort – Frankel scheme?

$$\begin{cases} \phi_j^{n+1} = \phi_j^n + \Delta t\, \phi_j'^{\,n} + \dfrac{\Delta t^2}{2} \phi_j''^{\,n} + \cdots \\ \phi_j^{n-1} = \cdots \\ \phi_{j+1}^n = \phi_j^n + \Delta x \dfrac{\partial \phi}{\partial x}\Big|_j^n + \dfrac{\Delta x^2}{2} \dfrac{\partial^2 \phi}{\partial x^2}\Big|_j^n + \cdots \\ \phi_{j-1}^n = \cdots \end{cases}$$

$$\Rightarrow \frac{\partial \phi}{\partial t} - \alpha \frac{\partial^2 \phi}{\partial x^2} = -\frac{\Delta t^2}{6} \frac{\partial^3 \phi}{\partial t^3} + \alpha \frac{\Delta x^2}{12} \frac{\partial^4 \phi}{\partial x^4}$$

$$- \alpha \boxed{\frac{\Delta t^2}{\Delta x^2}} \frac{\partial^2 \phi}{\partial t^2} - \alpha \frac{\Delta t^4}{12 \Delta x^2} \frac{\partial^4 \phi}{\partial t^4} + \cdots$$

difficulty!

For a given $\Delta t$, the error increases if we refine the mesh.

One cannot increase the accuracy of num. sol. by arbitrarily letting $\Delta x \to 0$ and $\Delta t \to 0$.

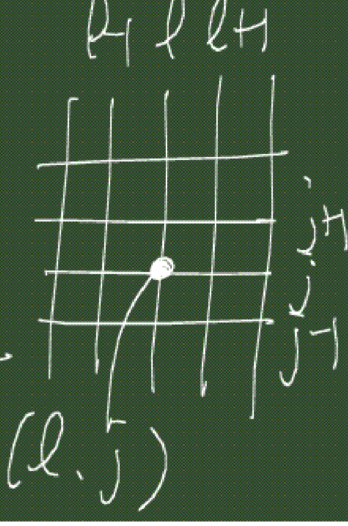
The third term approaches zero only if $\Delta t \to 0$ faster than $\Delta x$.

$\rightarrow$ Example of inconsistent num. meth.

# 5.7 Higher dimensions

2-D diffusion eq.

$$\frac{\partial \phi}{\partial t} = \alpha\left(\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2}\right)$$




CD2

$$\frac{\partial \phi_{l,j}}{\partial t} = \alpha\left[\frac{\phi_{l+1,j} - 2\phi_{l,j} + \phi_{l-1,j}}{\Delta x^2} + \frac{\phi_{l,j+1} - 2\phi_{l,j} + \phi_{l,j-1}}{\Delta y^2}\right]$$

EE:

$$\phi_{l,j}^{n+1} - \phi_{l,j}^{n} = \frac{\alpha \Delta t}{\Delta x^2}\left(\phi_{l+1,j}^{n} - 2\phi_{l,j}^{n} + \phi_{l-1,j}^{n}\right)$$

$$+ \frac{\alpha \Delta t}{\Delta y^2}\left(\phi_{l,j+1}^{n} - 2\phi_{l,j}^{n} + \phi_{l,j-1}^{n}\right)$$

start from initial cond. $\phi_{l,j}^{(0)}$

march in time w/ b.c for $j = 1, \dots, N-1$, $l = 1, \dots, M-1$

· stability

Modified wavenumber analysis

$$\frac{\partial \phi}{\partial t} = \alpha\left(\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2}\right)$$

$$\phi(x, y, t) = \underline{\psi(t)}\, e^{ik_1 x} e^{ik_2 y}$$

$$\frac{d\psi}{dt} = \alpha\left(-k_1^2 - k_2^2\right)\psi$$

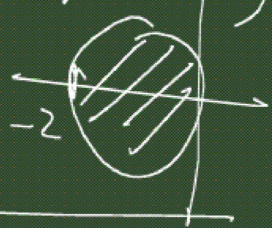
CD2: $\frac{d\psi}{dt} = \boxed{\alpha\left(-k_1'^2 - k_2'^2\right)}\psi$

$$k_1'^2 = \frac{2(1-\cos k_1 \Delta x)}{\Delta x^2}$$

$$k_2'^2 = \frac{2(1-\cos k_2 \Delta y)}{\Delta y^2}$$

$\frac{dy}{dt} = \lambda y$ for real and negative $\lambda$,

$$\Delta t \le \frac{2}{|\lambda|}$$




$$\Delta t \le \frac{2}{\alpha\left[\frac{2(1-\cos k_1 \Delta x)}{\Delta x^2} + \frac{2(1-\cos k_2 \Delta y)}{\Delta y^2}\right]}$$

worst case: $\cos k_1 \Delta x = \cos k_2 \Delta y = -1$

$$\rightarrow \Delta t \le \frac{1}{2\alpha\left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2}\right)}$$

If $\Delta x = \Delta y$, $\quad \Delta t \le \dfrac{\Delta x^2}{4\alpha}$ (2D)

too restrictive. $\quad \dfrac{\Delta x^2}{2\alpha}$ (1D)

$\dfrac{\Delta x^2}{6\alpha}$ (3D)

## 5.8 Implicit methods in higher dimension.

Trapezoidal method (C–N method)

$$\frac{\phi^{n+1} - \phi^n}{\Delta t} = \frac{\alpha}{2}\left(\frac{\partial^2 \phi^{n+1}}{\partial x^2} + \frac{\partial^2 \phi^n}{\partial x^2}\right) + \frac{\alpha}{2}\left(\frac{\partial^2 \phi^{n+1}}{\partial y^2} + \frac{\partial^2 \phi^n}{\partial y^2}\right)$$

$\Delta x = \Delta y = h \quad C\theta 2$

$$\phi^{n+1}_{l,j} - \phi^{n}_{l,j} = \underbrace{\frac{\alpha \Delta t}{2h^2}}_{\beta}\left(\phi^{n+1}_{l+1,j} - 2\phi^{n+1}_{l,j} + \phi^{n+1}_{l-1,j}\right)$$

$$+ \ \text{''} \ \left(\phi^{n+1}_{l,j+1} - 2\phi^{n+1}_{l,j} + \phi^{n+1}_{l,j-1}\right)$$

$$+ \ \text{''} \ \left( \ {}^{n} \qquad {}^{n} \qquad {}^{n} \ \right)$$

$$+ \ \text{''} \ \left( \ {}^{n} \qquad {}^{n} \qquad {}^{n} \ \right)$$

$$-\beta \phi^{n+1}_{l+1,j} + (1+4\beta)\phi^{n+1}_{l,j} - \beta \phi^{n+1}_{l-1,j}$$

$$-\beta \phi^{n+1}_{l,j+1} - \beta \phi^{n+1}_{l,j-1} = F^{n}_{l,j}$$

$$l = 1, 2, \cdots, M-1$$

$$j = 1, 2, \cdots, N-1$$

System of eqs. $[(M-1)(N-1) \times (M-1)(N-1)]$

$$\begin{array}{c} \xrightarrow{\quad (M-1)\times(N-1) \quad} \\ \begin{bmatrix} B & C & & 0 \\ A & B & C & \\ & \ddots & \ddots & \ddots \\ 0 & & & \ddots \end{bmatrix} \begin{bmatrix} \phi_{11} \\ \phi_{21} \\ \vdots \\ \vdots \\ \phi_{M-1\,N-1} \end{bmatrix} = F \end{array}$$

Block-tridiagonal matrix

If $M=N=100$, # of elements in the matrix is $10^8$.

too difficult to get $\phi$.

5.9 Alternating direction implicit method and approximate (ADI) factorization

Trapezoidal method

$$\frac{\phi^{n+1}-\phi^{n}}{\Delta t} = \frac{\alpha}{2}\left[A_x\phi^{n+1} + A_y\phi^{n+1}\right] + \frac{\alpha}{2}\left[A_x\phi^{n} + A_y\phi^{n}\right] + O(\Delta t^2) + O(\Delta x^2)$$

$A_x, A_y$: difference operators representing derivatives in $x$ & $y$ dimensions, resp.

$$\underbrace{\left[I - \frac{\Delta t\alpha}{2}A_x - \frac{\Delta t\alpha}{2}A_y\right]}\phi^{n+1} = \underline{\left[I + \frac{\Delta t\alpha}{2}A_x + \frac{\Delta t\alpha}{2}A_y\right]}\phi^{n} + \Delta t\left[O(\Delta t^2) + O(\Delta x^2)\right]$$

$\checkmark$ $I - \frac{\alpha\Delta t}{2}A_x - \frac{\alpha\Delta t}{2}A_y$

$$= \left(I - \frac{\alpha \Delta t}{2} A_x\right)\left(I - \frac{\alpha \Delta t}{2} A_y\right) - \frac{\alpha^2 \Delta t^2}{4} A_x A_y$$

$$\checkmark \quad I + \frac{\alpha \Delta t}{2} A_x + \frac{\alpha \Delta t}{2} A_y$$

$$= \left(I + \frac{\alpha \Delta t}{2} A_x\right)\left(I + \frac{\alpha \Delta t}{2} A_y\right) - \frac{\alpha^2 \Delta t^2}{4} A_x A_y$$

$$\therefore \left(I - \frac{\alpha \Delta t}{2} A_x\right)\left(I - \frac{\alpha \Delta t}{2} A_y\right) \phi^{n+1}$$

$$= \left(I + \frac{\alpha \Delta t}{2} A_x\right)\left(I + \frac{\alpha \Delta t}{2} A_y\right) \phi^{n}$$

$$+ \frac{\alpha^2 \Delta t^2}{4} A_x A_y \left(\underline{\phi^{n+1} - \phi^{n}}\right) \Leftarrow$$

$$+ \underline{\Delta t} \left( \underline{O(\Delta t^2)} + O(\Delta x^2) \right)$$

$O(\Delta t^3)$

$O(\Delta t)$

$$\Rightarrow \boxed{\left(I - \frac{\alpha\Delta t}{2}A_x\right)\left(I - \frac{\alpha\Delta t}{2}A_y\right)\phi^{n+1} = \left(I + \frac{\alpha\Delta t}{2}A_x\right)\left(I + \frac{\alpha\Delta t}{2}A_y\right)\phi^{n}} = F$$

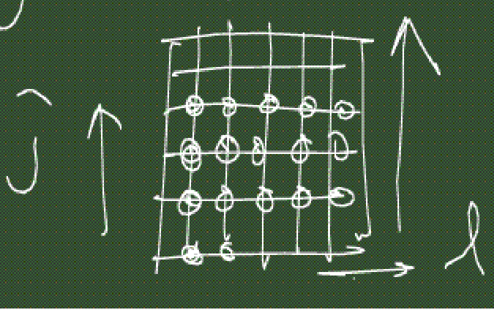
Let $\left(I - \frac{\alpha\Delta t}{2}A_y\right)\phi^{n+1} = Z \Leftarrow$

Then, $\left(I - \frac{\alpha\Delta t}{2}A_x\right)Z = F$

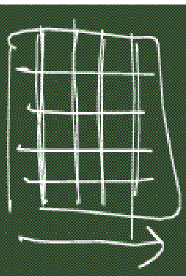
✓ $Z_{l,j} - \frac{\alpha\Delta t}{2}\,\frac{Z_{l+1,j} - 2Z_{l,j} + Z_{l-1,j}}{\Delta x^2} = F_{l,j}$

(*) For each $j$, solve a tri-diagonal matrix for $Z_{l,j}$, $l = 1, \cdots, M-1$

$O(N \cdot M)$



Having solved for $z$,



$$\left(I - \frac{\alpha \Delta t}{2} A_y\right)\phi^{n+1} = z$$

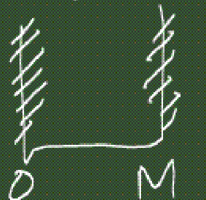
(**)
$$\phi^{n+1}_{l,j} - \frac{\alpha \Delta t}{2}\,\frac{\phi^{n+1}_{l,j+1} - 2\phi^{n+1}_{l,j} + \phi^{n+1}_{l,j-1}}{\Delta y^2} = z_{l,j}$$

For each $l$, solve a tri-diagonal matrix for $\phi^{n+1}_{l,j}$. $j = 1, 2, \cdots, N-1$

$\rightarrow O(MN)$

In total, $O(2MN)$ is required.

(*) needs $z_{0,j}$ and $z_{M,j}$.




(**)

$$z_{0,j} = \phi^{n+1}_{0,j} - \frac{\alpha \Delta t}{2}\,\frac{\phi^{n+1}_{0,j+1} - 2\phi^{n+1}_{0,j} + \phi^{n+1}_{0,j-1}}{\Delta y^2}$$

from b.c. of $\phi$

$z_{M,j} = \cdots$

Alternating directional implicit (ADI) method.

5.9.1 Stability

Replace $A_x$ and $A_y$ with $-k_1'^2$ and $-k_2'^2$, respectively.

$$k_1'^2 = 2\,\frac{(1-\cos k_1\Delta x)}{\Delta x^2}$$

$$k_2'^2 = 2\,\frac{(1-\cos k_2\Delta y)}{\Delta y^2}$$

unconditionally stable

↓

$$\sigma = \frac{\left[1+\frac{\alpha\Delta t}{\Delta y^2}(\cos k_2\Delta y-1)\right]\left[1+\frac{\alpha\Delta t}{\Delta x^2}(\cos k_1\Delta x-1)\right]}{\left[1-\frac{\alpha\Delta t}{\Delta y^2}(\cos k_2\Delta y-1)\right]\left[1-\frac{\alpha\Delta t}{\Delta x^2}(\cos k_1\Delta x-1)\right]}$$

## 5.9.2 Alternating Direction Implicit method

Peaceman & Rachford (1955)

① $$\frac{\phi^{n+\frac{1}{2}} - \phi^n}{\Delta t/2} = \alpha\left(\frac{\partial^2 \phi^{n+\frac{1}{2}}}{\partial x^2} + \frac{\partial^2 \phi^n}{\partial y^2}\right)$$

② $$\frac{\phi^{n+1} - \phi^{n+\frac{1}{2}}}{\Delta t/2} = \alpha\left(\frac{\partial^2 \phi^{n+\frac{1}{2}}}{\partial x^2} + \frac{\partial^2 \phi^{n+1}}{\partial y^2}\right)$$

① → $$\left(I - \frac{\alpha \Delta t}{2} A_x\right)\phi^{n+\frac{1}{2}} = \left(I + \frac{\alpha \Delta t}{2} A_y\right)\phi^n$$

② → $$\left(I - \frac{\alpha \Delta t}{2} A_y\right)\phi^{n+1} = \left(I + \frac{\alpha \Delta t}{2} A_x\right)\phi^{n+\frac{1}{2}}$$

$$\phi^{n+\frac{1}{2}} = \left(I - \frac{\alpha \Delta t}{2} A_x\right)^{-1}\left(I + \frac{\alpha \Delta t}{2} A_y\right)\phi^n$$

$$\left(I - \frac{\alpha \Delta t}{2} A_y\right)\phi^{n+1} = \left(I + \frac{\alpha \Delta t}{2} A_x\right)\left(I - \frac{\alpha \Delta t}{2} A_x\right)^{-1} \cdot \left(I + \frac{\alpha \Delta t}{2} A_y\right)\phi^n$$

$$\left(I-\frac{\alpha\Delta t}{2}A_x\right)\left(I-\frac{\alpha\Delta t}{2}A_y\right)\phi^{n+1}$$

$$=\left(I+\frac{\alpha\Delta t}{2}A_x\right)\left(I+\frac{\alpha\Delta t}{2}A_y\right)\phi^{n}$$

(Note that $I-\frac{\alpha\Delta t}{2}A_x$ and $I+\frac{\alpha\Delta t}{2}A_x$ commute)

same as before

B.C for $\phi^{n+\frac{1}{2}}$

① $\left(I-\alpha\frac{\Delta t}{2}A_x\right)\phi_B^{n+\frac{1}{2}}=\left(I+\frac{\alpha\Delta t}{2}A_y\right)\phi_B^{n}$

② $\left(I-\alpha\frac{\Delta t}{2}A_x\right)\phi_B^{n+1}=\left(I+\frac{\alpha\Delta t}{2}A_x\right)\phi_B^{n+\frac{1}{2}}$

---

$$\phi_B^{n+\frac{1}{2}}=\frac{1}{2}\left(I+\alpha\frac{\Delta t}{2}A_y\right)\phi_B^{n}+\frac{1}{2}\left(I-\alpha\frac{\Delta t}{2}A_y\right)\phi_B^{n+1}$$

$$\frac{\phi^{n+1}-\phi^{n}}{\Delta t} = \frac{\alpha}{2}\left(A_x\phi^{n+1}+A_y\phi^{n+1}\right) + \frac{\alpha}{2}\left(A_x\phi^{n}+A_y\phi^{n}\right)$$

$\boxed{\delta\text{-form}}$

$$\delta\phi \equiv \phi^{n+1}-\phi^{n}$$

$$\frac{\delta\phi}{\Delta t} = \frac{\alpha}{2}\left(A_x(\delta\phi+\phi^{n})+A_y(\delta\phi+\phi^{n})\right)$$

$$+\frac{\alpha}{2}\left(A_x\phi^{n}+A_y\phi^{n}\right)$$

$$\delta\phi = \frac{\alpha\Delta t}{2}\left(A_x\delta\phi+A_y\delta\phi + A_x\phi^{n}+A_y\phi^{n}\right) + \frac{\alpha\Delta t}{2}\left(A_x\phi^{n}+A_y\phi^{n}\right)$$

$$+A_y\phi^{n})$$

$$\rightarrow \left(I-\frac{\alpha\Delta t}{2}A_x-\frac{\alpha\Delta t}{2}A_y\right)\delta\phi = \alpha\Delta t(A_x\phi^{n}$$

$$\left(I - \frac{\alpha \Delta t}{2} A_x\right)\left(I - \frac{\alpha \Delta t}{2} A_y\right)\delta\phi$$

$$= \alpha \Delta t\,(A_x \phi^n + A_y \phi^n)$$

$$+ \frac{\alpha^2 \Delta t^2}{4} A_x A_y \delta\phi$$

$O(\Delta t)$

$O(\Delta t^3)$ neglect!

$$\left(I - \frac{\alpha \Delta t}{2} A_x\right)\left(I - \frac{\alpha \Delta t}{2} A_y\right)\delta\phi$$

$$= \alpha \Delta t\,(A_x + A_y)\,\phi^n$$

less issue in b.c.

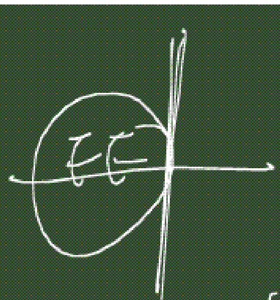
# §9.3 Mixed time advancement

노트 제목

2009-11-16

✓ $$\frac{\partial u}{\partial t} = \underbrace{-c\frac{\partial u}{\partial x}}_{EE} + \underbrace{\nu\frac{\partial^2 u}{\partial x^2}}_{\cancel{TR}\ EE}$$

$$\frac{u^{n+1}-u^{n}}{\Delta t} = -c\frac{\partial u^{n}}{\partial x} + \frac{\nu}{2}\left(\frac{\partial^2 u^{n+1}}{\partial x^2} + \frac{\partial^2 u^{n}}{\partial x^2}\right)$$




$$\frac{\partial u}{\partial t} = -c\frac{\partial u}{\partial x}$$

$$u = \psi(t)\, e^{ikx}$$

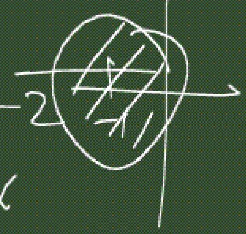
$$\frac{d\psi}{dt}\cancel{e^{ikx}} = -cik\psi\cancel{e^{ikx}} - \nu k^2\psi\cancel{e^{ikx}}$$

$$\rightarrow \frac{d\psi}{dt} = (-\nu k^2 - ick)\,\psi$$

complex eigenvalue prob.

CD2: $\dfrac{d\psi}{dt} = \left(-\nu k'^2 - i c k''\right)\psi$

$$k'' = \frac{\sin k\Delta x}{\Delta x}$$




$$k'^2 = 2\,\frac{1-\cos k\Delta x}{\Delta x^2}$$

$$\lambda = -2\nu\,\frac{1-\cos k\Delta x}{\Delta x^2} - ic\,\frac{\sin k\Delta x}{\Delta x}$$

$$\Rightarrow u_j^n = \sigma^n e^{ikx_j}$$

$$\frac{\sigma^{n+1}-\sigma^n}{\Delta t}\, e^{ikx_j} = -c\,\frac{\sigma^n e^{ikx_{j+1}} - \sigma^n e^{ikx_{j-1}}}{2\Delta x}$$

$$+\frac{\nu}{2}\left[\ \cdot\ -\ -\ \ -\ \right]$$

$$|\sigma| \le 1 \longrightarrow \Delta t$$

AB2 + CN

$$\frac{u^{n+1}-u^n}{\Delta t} = -\frac{1}{2}\left(3\frac{\partial u^n}{\partial x} - \frac{\partial u^{n-1}}{\partial x}\right) + \frac{\nu}{2}\left(\frac{\partial^2 u^{n+1}}{\partial x^2} + \frac{\partial^2 u^n}{\partial x^2}\right)$$

stability

$$\Rightarrow \frac{\Delta t\, u}{\Delta x} \le 1$$

## 5.10 Elliptic PDE

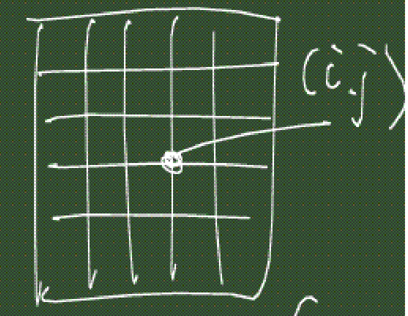
Ex. $\nabla^2\phi = 0$ Laplace eq.

$\nabla^2\phi = g$ Poisson eq.

$\nabla^2\phi + k^2\phi = g$ Helmholtz eq.

$$\nabla^2 \phi = f$$

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = f$$




CD2:

$$\frac{\phi_{i+1,j} - 2\phi_{i,j} + \phi_{i-1,j}}{\Delta x^2} + \frac{\phi_{i,j+1} - 2\phi_{i,j} + \phi_{i,j-1}}{\Delta y^2} = f_{i,j}$$

$i = 1, 2, \cdots, M-1$; $j = 1, 2, \cdots, N-1$

Block tri-diagonal matrix.

→ Matrix becomes too large.

- Treat the elliptic PDE as steady-state version of parabolic PDE.

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

Integrate in time to reach steady state ($\partial\phi/\partial t = 0$).

EE ($\Delta x = \Delta y = h$)

$$\frac{\phi_{i,j}^{n+1} - \phi_{i,j}^{n}}{\Delta t} = \frac{1}{h^2}\left(\phi_{i+1,j}^{n} - 2\phi_{i,j}^{n} + \phi_{i-1,j}^{n} + \phi_{i,j+1}^{n} - 2\phi_{i,j}^{n} + \phi_{i,j-1}^{n}\right) - f_{i,j}^{n}$$

Don't care about the time accuracy

Use largest time step.

Stability $\rightarrow \Delta t \leq \frac{h^2}{4} \rightarrow \Delta t = \frac{h^2}{4}$

Then,

$$\phi_{i,j}^{n+1} = \frac{1}{4}\left(\phi_{i-1,j}^{n} + \phi_{i+1,j}^{n} + \phi_{i,j-1}^{n} + \phi_{i,j+1}^{n}\right) - \frac{h^2}{4} f_{i,j}^{n}.$$

This is the same as the Jacobi iteration.

Or IE.

$$\frac{\phi_{i,j}^{n+1} - \phi_{i,j}^{n}}{\Delta t} = \frac{1}{h^2}\left( \phi_{i+1,j}^{n+1} - 2\phi_{i,j}^{n+1} + \phi_{i-1,j}^{n+1} + \phi_{i,j+1}^{n+1} - 2\phi_{i,j}^{n+1} + \phi_{i,j-1}^{n+1} \right) - f_{i,j}^{n+1}$$

+ ADI : TDMA to get $\phi^{n+1}$ w/ large $\Delta t$.

5.10.1 Iterative solution methods

Discretization (e.g. CD2)

$\rightarrow A\phi = b$

$A = A_1 - A_2$

① $A_1 \phi = A_2 \phi + b$

Iterative scheme

② $A_1 \phi^{k+1} = A_2 \phi^k + b$ k; iteration index

$\rightarrow$ Obtain $\phi^{k+1}$

↳ $A_1$ should be easy to convert.

Error $\varepsilon^k = x - x^k$

① - ② ; $A_1 (\phi - \phi^{k+1}) = A_2 (\phi - \phi^k)$

$\rightarrow A_1 \varepsilon^{k+1} = A_2 \varepsilon^k$

$\rightarrow \varepsilon^{k+1} = A_1^{-1} A_2 \varepsilon^k$

$\rightarrow \boxed{\varepsilon^k = (A_1^{-1} A_2)^k \varepsilon^0}$

For convergence ($\varepsilon^k \rightarrow 0$)

eigenvalue of $A_1^{-1} A_2$ should have modulus less than 1.

$$|\lambda_{max}| < 1$$

For $\nabla^2 \phi = b$ and CD2,

$$A = \left[\begin{array}{ccc|ccc|c} -4 & 1 & & 1 & & & \\ 1 & -4 & 1 & & 1 & & 0 \\ & 1 & -4 & & & 1 & \\ \hline 1 & & & -4 & 1 & & \\ & 1 & & 1 & -4 & 1 & \\ & & 1 & & 1 & -4 & \\ & & & & & & \ddots \end{array}\right]$$

5.10.2 Point Jacobi method

Simplest choice for $A_1$ is the diagonal matrix $D$.

$$A_1 = D = \begin{bmatrix} -4 & & & 0 \\ & -4 & & \\ & & -4 & \\ 0 & & & \ddots & -4 \end{bmatrix}, \quad D^{-1} = A_1^{-1} = \begin{bmatrix} -\frac{1}{4} & & 0 \\ & -\frac{1}{4} & \\ 0 & & \ddots & -\frac{1}{4} \end{bmatrix}$$

$$A_2 = \left[\begin{array}{ccc|ccc|c} 0 & -1 & & -1 & & & \\ -1 & 0 & -1 & & -1 & & \\ & -1 & 0 & & & -1 & \\ \hline & & & & & & \end{array}\right]$$

$$A_1^{-1}A_2 = D^{-1}A_2 = -\frac{1}{4}A_2$$

$$\phi^{k+1} = -\frac{1}{4}A_2\phi^k - \frac{1}{4}b$$

$$\phi_{i,j}^{k+1} = \frac{1}{4}\left[\phi_{i-1,j}^k + \phi_{i+1,j}^k + \phi_{i,j-1}^k + \phi_{i,j+1}^k\right] - \frac{1}{4}b_{i,j}$$

; point Jacobi method

$\lambda$'s for $D^{-1}A_2$ are

$$\lambda_{i,j} = \frac{1}{2}\left(\cos\frac{i\pi}{M} + \cos\frac{j\pi}{M}\right), \quad i=1,\cdots,M-1,\ j=1,\cdots,N-1$$

$|\lambda|<1 \rightarrow$ It converges

$$\left(\cos x = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \cdots\right)$$

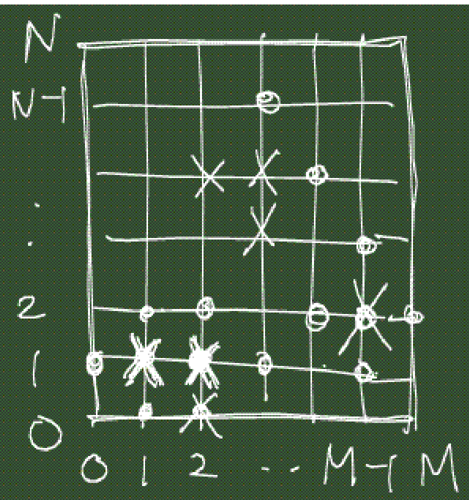
$$\lambda_{max} = \frac{1}{2}\left(\cos\frac{\pi}{M} + \cos\frac{\pi}{N}\right)$$

$$= 1 - \frac{1}{4}\left(\frac{\pi^2}{M^2} + \frac{\pi^2}{N^2}\right) + \cdots$$

For large M and N, $\lambda_{max}$ is close to 1. $\rightarrow$ slow convergence.

- Gauss-Seidel method.

$$\phi_{i,j}^{k+1} = \frac{1}{4}\left(\phi_{\underline{i-1},j}^{k+1} + \phi_{\underline{i+1},j}^{k} + \phi_{i,\underline{j-1}}^{k+1} + \phi_{i,\underline{j+1}}^{k}\right) - \frac{1}{4} b_{i,j}$$



$$A\phi = b$$

$$A = \underset{\underset{D-L}{\parallel}}{A_1} - \underset{\underset{U}{\parallel}}{A_2}$$

$\underline{\underline{(D-L)}}\,\phi^{k+1} = U\phi^{k} + b$ : G-S iteration

lower triangular matrix
$\rightarrow$ easy to convert

$\lambda$'s for $A_1^{-1}A_2$ are

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

$$\lambda_{GS} = \lambda_J^2 \qquad \varepsilon_{GS}^k = \lambda_{GS}^k \varepsilon^0$$

$$= (\lambda_J^2)^k \varepsilon^0$$

$$\varepsilon_J^{k'} = \lambda_J^{k'} \varepsilon^0$$

$$= \lambda_J^{2k} \varepsilon^0$$

$\rightarrow k' = 2k$

$\rightarrow$ GS is 2 times faster than J.

## 5.10.4 Successive Over Relaxation (SOR) scheme

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

$$\phi^{k+1} = \phi^k + d \cdot \omega$$

$\omega$: relaxation factor

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

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

$$\Rightarrow \phi^{k+1} = \phi^{k} + \omega\left[\ \text{''}\ - \phi^{k}\right]$$

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

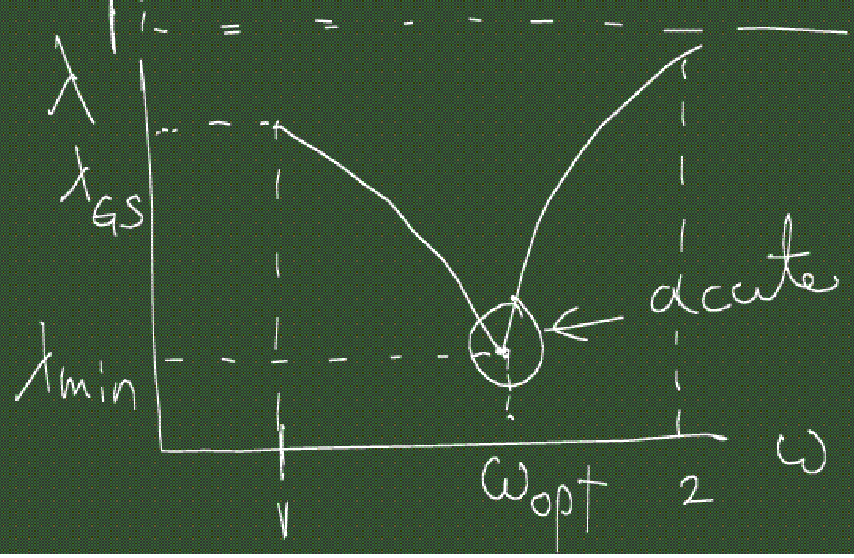
$$\phi^{k+1} = \underbrace{(I-\omega D^{-1}L)^{-1}\left[(1-\omega)I + \omega D^{-1}U\right]}_{G_{SOR}}\phi^{k} + (I-\omega D^{-1}L)^{-1}\omega D^{-1}b$$

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

where $\mu$ is an eigenvalue of the point Jacobi matrix.

Optimum $\omega$? (to minimize $\lambda$)

$\Rightarrow \frac{d\lambda}{d\omega} = 0$ has no solution.



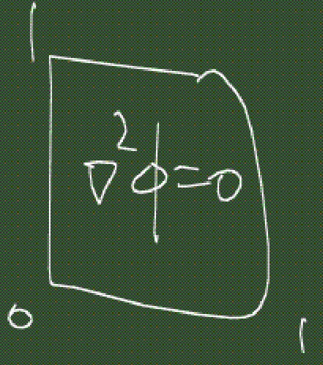

$$\frac{d\lambda}{d\omega} \to \infty \Rightarrow \omega_{opt} = \frac{2}{1+\sqrt{1-\mu_{max}^2}}$$

For probs. w/ irregular geometry and non-uniform mesh, $\omega_{opt}$ cannot be obtained analytically, and one must find it by trial and error.

$\omega_{opt} = 1.7 \sim 1.9$

As $M$ & $N \uparrow$, $\mu_{max} \to 1$, $\omega_{opt} \to 2$.

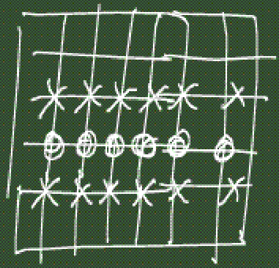
Ex




$M = N = 100$

| Method | $k=100$ |
|---|---|
| J | 0.887 |
| GS | 0.786 |
| SOR ($\omega = 1.75$) | $5 \times 10^{-7}$ |

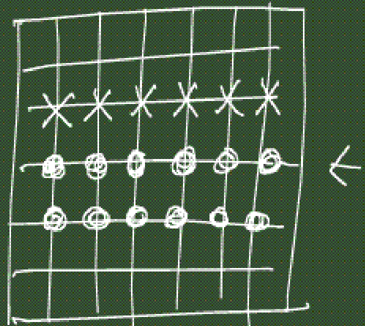
- Line Jacobi method



$$\phi_{i,j}^{k+1} = \frac{1}{4}\left(\phi_{i-1,j}^{k+1} + \phi_{i+1,j}^{k+1} + \phi_{i,j-1}^{k} + \phi_{i,j+1}^{k}\right) - \frac{1}{4} b_{i,j}$$

TDMA

- Line Gauss-Seidel method



$$\phi_{i,j}^{k+1} = \frac{1}{4}\left(\phi_{i-1,j}^{k+1} + \phi_{i+1,j}^{k+1} + \phi_{i,j-1}^{k+1} + \phi_{i,j+1}^{k}\right) - \frac{1}{4} b_{i,j}$$

- ADI method using artificial time derivative term

- Strongly Implicit Procedure (SIP)

Stone (1968) SIAM J. Num. Anal. <u>5</u>, 530

$$A\phi = b \rightarrow \underline{L}U\phi = b$$

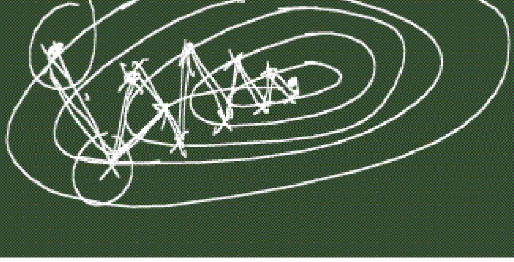
$$(LU)\phi^{k+1} = (LU - A)\phi^{k} + b$$

find L & U such that $LU \simeq A$.

incomplete LU decomposition (ILU)

- conjugate gradient solver (CGS)

Kershaw (1978) JCP <u>26</u> 43.



single gradient method

- Pre-conditioning

$$A\phi = b \rightarrow \underline{BA}\phi = Bb$$
$$\rightarrow C\phi = b'$$
$$\rightarrow C_1\phi^{k+1} = C_2\phi^{k} + b'$$

5.10.5 Multigrid acceleration (method)

Brandt (1977) Math. Comput. $\frac{31}{333}$.

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

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

$A\phi = b$ A: matrix obtained from a FD.

$\psi = \phi^k$ is an approx. to the sol. $\phi$ after $k$ iterations.

$r$: residual vector

$$\boxed{r = b - A\psi}$$

error $\varepsilon = \phi - \psi$

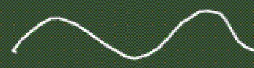
$$A\varepsilon = A\phi - A\psi = b - A\psi = r$$

$\boxed{A\varepsilon = r}$ ; residual eq.

example. $$\frac{d^2 y}{dx^2} = \sin k\pi x \qquad 0 \le x \le 1$$

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

k ; wavenumber high k 

low k 

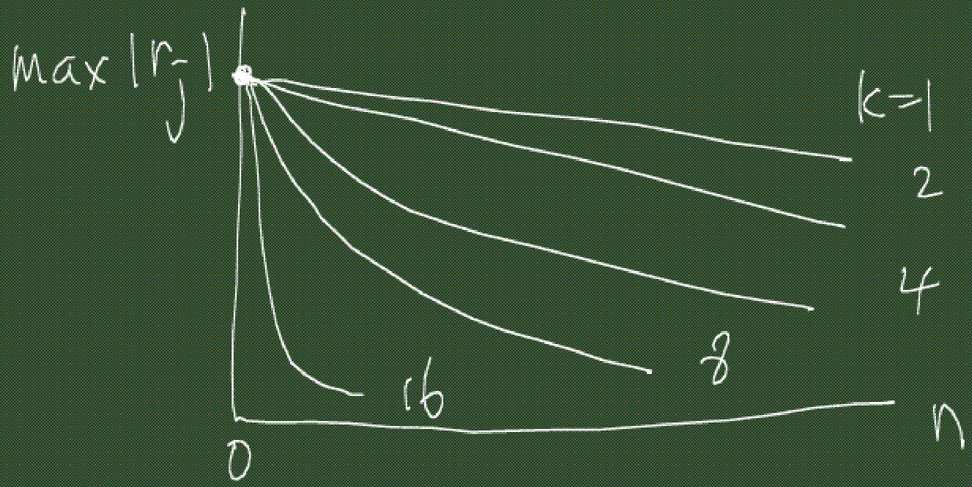
CD2 :
$$\begin{cases} \dfrac{u_{j+1} - 2u_j + u_{j-1}}{\Delta x^2} = \sin k\pi x_j & j = 1, 2, \cdots, N-1 \\ u_0 = u_N = 0 \end{cases}$$

initial guess $u^{(0)} = 0$ for all $j$'s.

$\rightarrow$ residual $r_j^0 = \sin k\pi j \Delta x$

use GS.

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




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

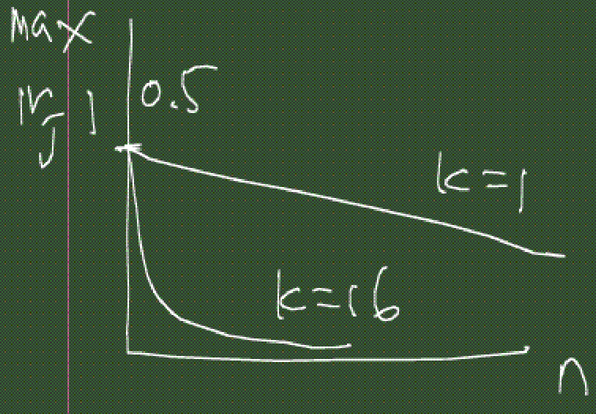
convergence is faster for higher k.

another example

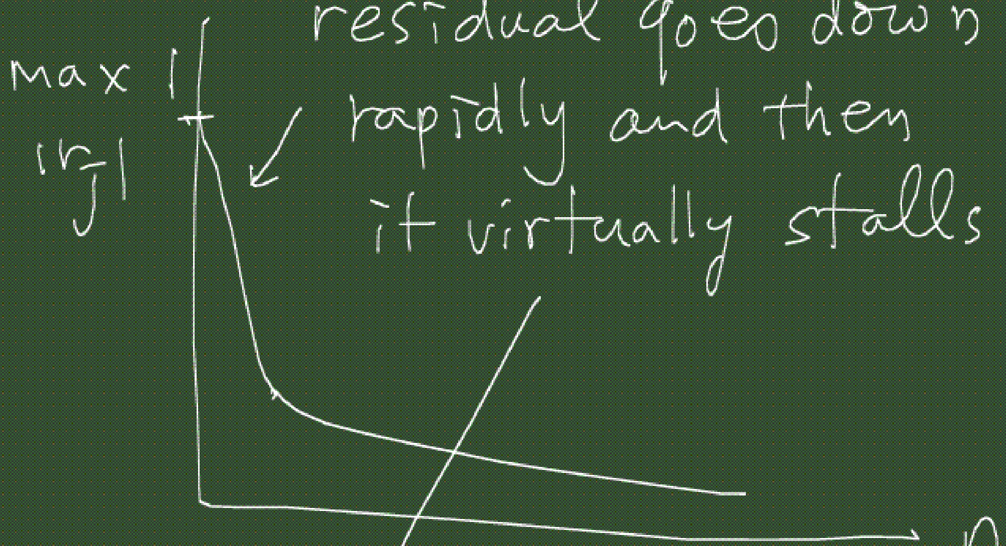
k=1 k=16

$$\frac{d^2u}{dx^2} = \frac{1}{2}\left(\sin \pi x + \sin 16\pi x\right)$$

$$u_0 = u_N = 0$$

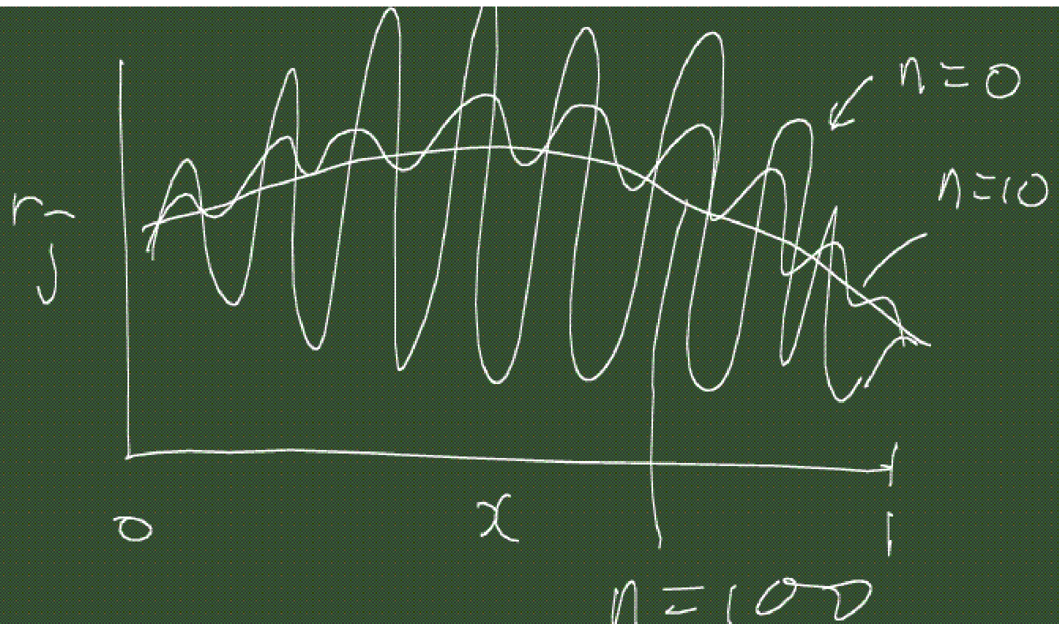



residual goes down rapidly and then it virtually stalls.




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 grid pts. are required for high k's. 
  but the convergence is fast for high k and low for low k.
- As N↑, $\lambda \to 1$ ⟵
  reduce N to N/2 for low k.

then $|\lambda|$ gets smaller.

→ faster convergence.

- $A\phi = b$

$A = A_1 - A_2$

$A_1 \phi^{n+1} = A_2 \phi^n + b$

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

$A_1 (\phi^{n+1} - \phi^n) = (A_2 - A_1)\phi^n + b.$

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

$A_1 \delta\phi^{n+1} = r^n$ $\quad \delta\phi^n$: convergence error

① compute $r^n = b - A\phi^n$

② solve $A_1 \delta\phi^{n+1} = r^n$ for $\delta\phi^{n+1}$

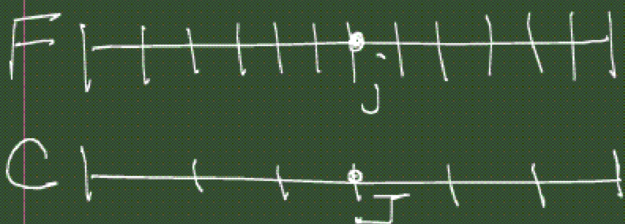
③ update $\phi^{n+1} = \phi^n + \delta\phi^{n+1}$

(single grid computation)

- Multigrid algorithm

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

② Restrict (smooth) residual to coarser grid




$r_J = r_j$

or

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

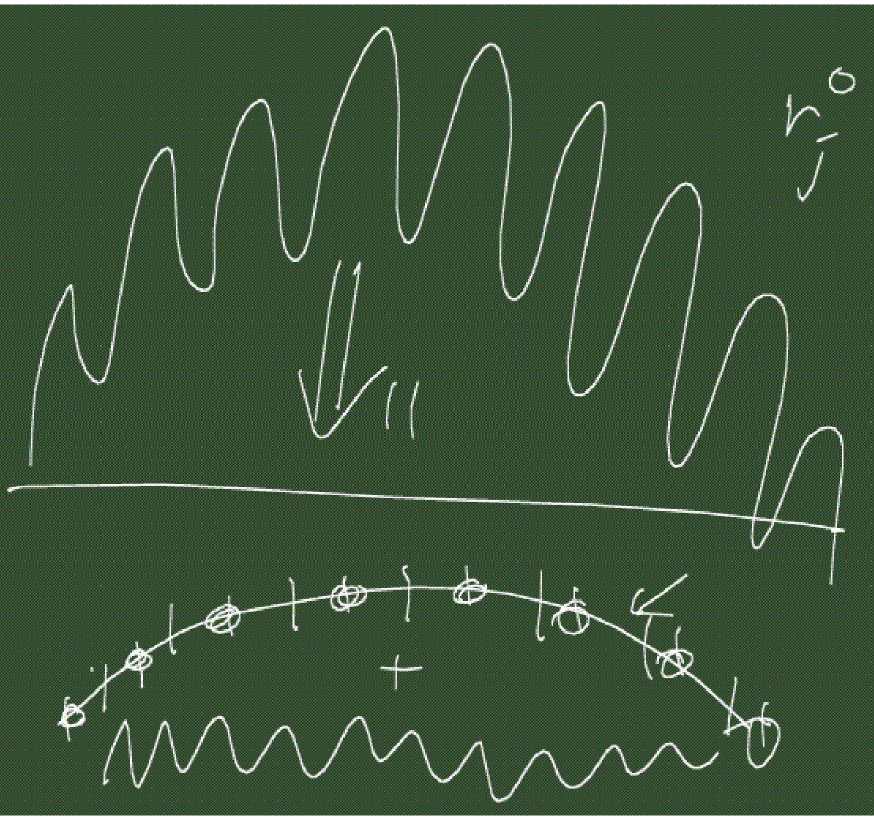
③ Iterate $A_1 \delta\phi^{n+1} = r^n$ on coarse grid

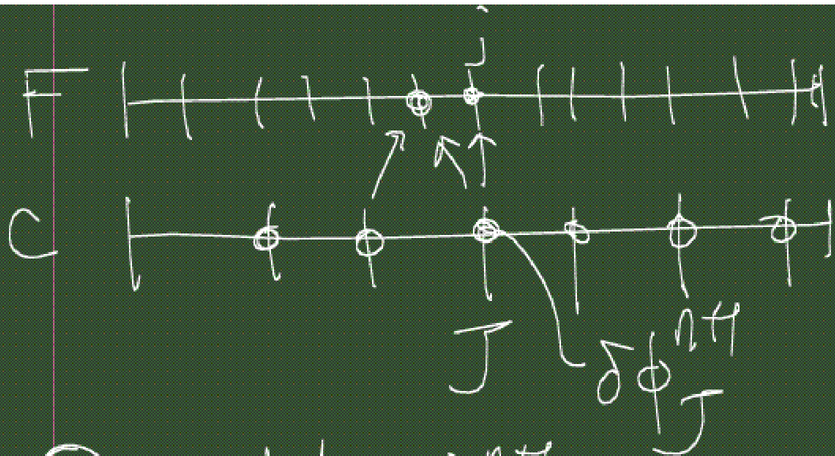
④ obtain $\delta\phi^{n+1}$

⑤ Prolong (interpolate) $\delta\phi^{n+1}$ to fine grid

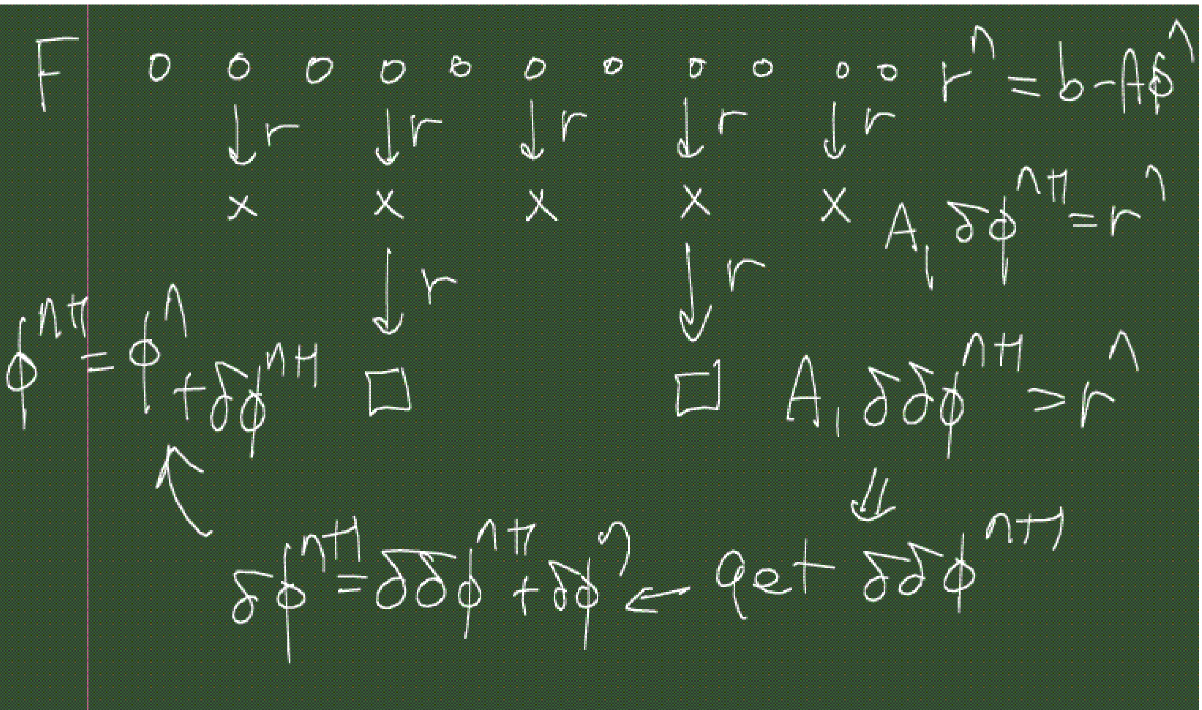
$j$ even, $\delta\phi_j^{n+1} = \delta\phi_J^{n+1}$

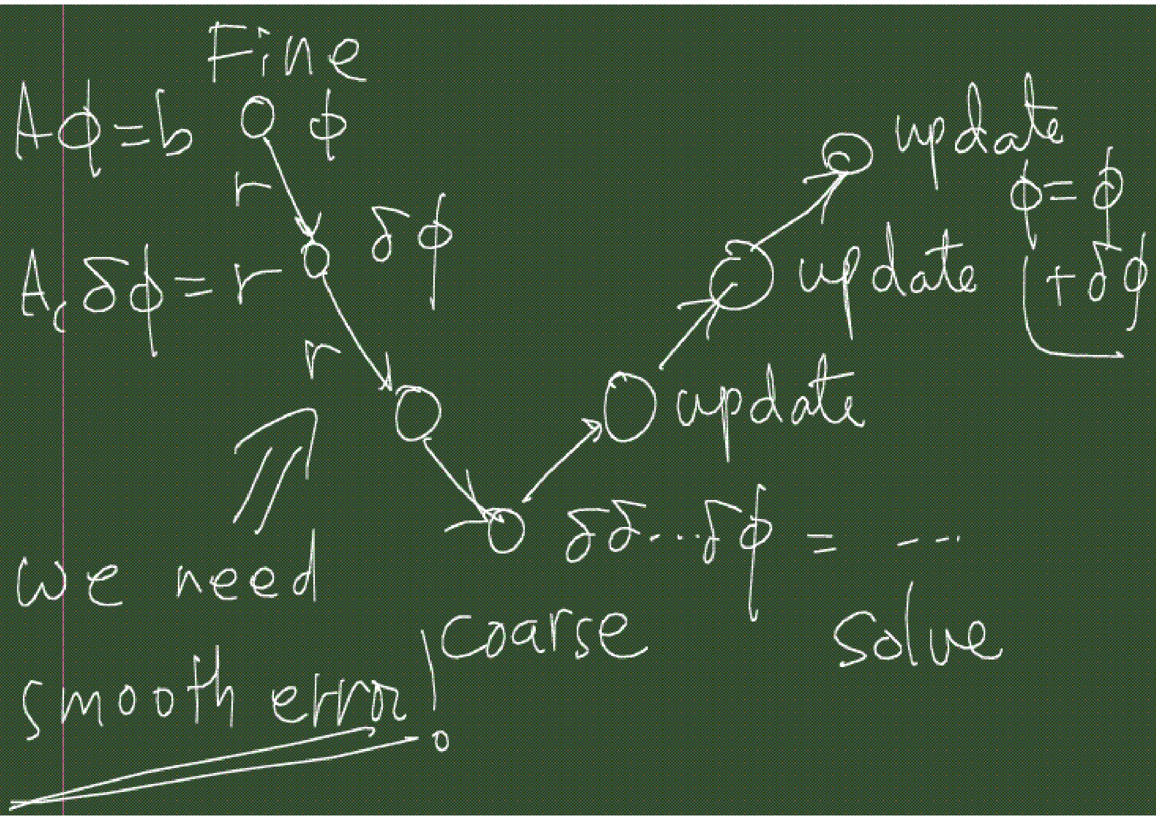
$$\delta\phi_{j+1}^{n+1} = \frac{1}{2}\left(\delta\phi_J^{n+1} + \delta\phi_{J+1}^{n+1}\right)$$






⑥ Update $\phi_j^{n+1} = \phi_j^n + \delta\phi_j^{n+1}$

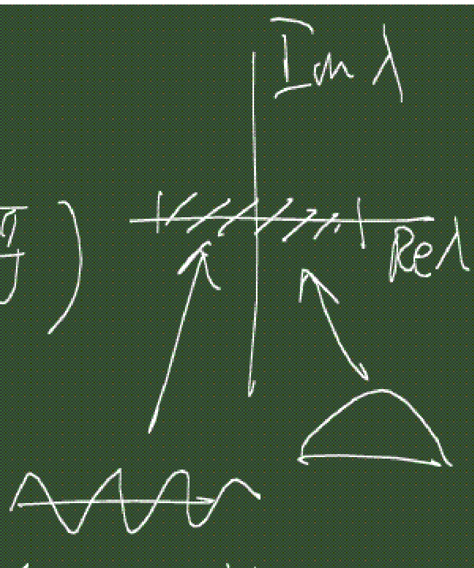





error from Jacobi:

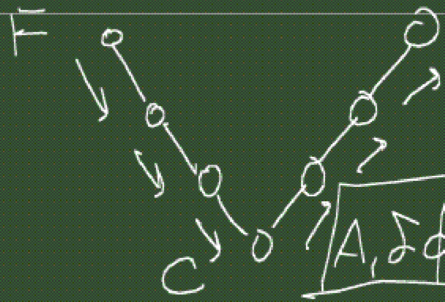
$$\lambda = \frac{1}{2}\left(\cos\frac{i\pi}{M} + \cos\frac{j\pi}{N}\right)$$

$$\varepsilon^n = \lambda^n \varepsilon^0$$




Jacobi is bad for multigrid due to this error behavior.

# multigrid method

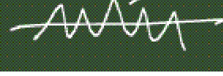


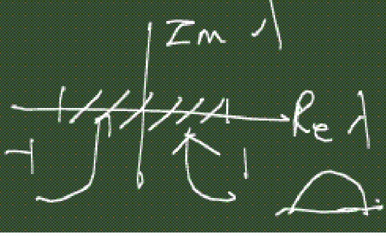

$$\boxed{A_1 \delta\phi = r}$$

need smooth error in space

error from Jacobi $\underline{\varepsilon^k} = \underline{\lambda}^k \underline{\varepsilon^0}$

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



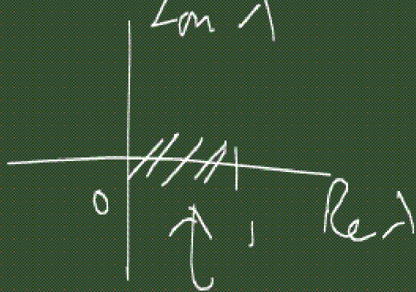


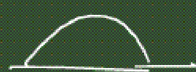

Jacobi is bad for multigrid due to this error behavior.

- from Gauss-Seidel

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

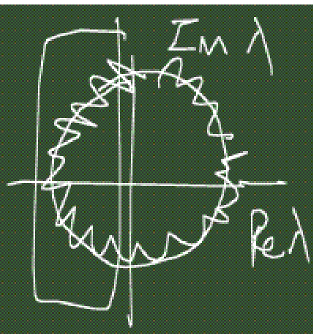
good.



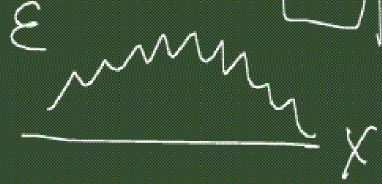



From SOR

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

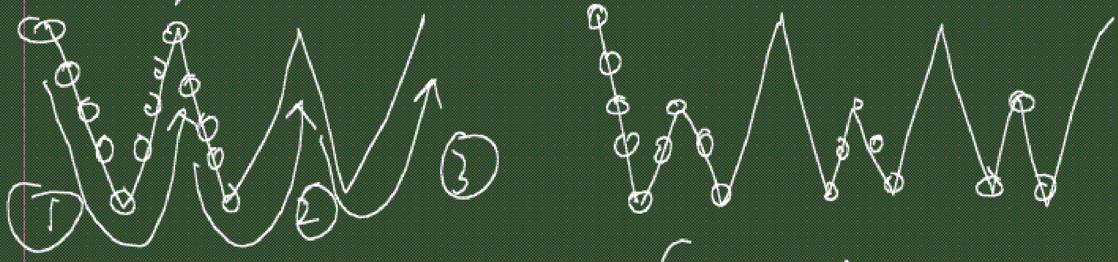



bad.



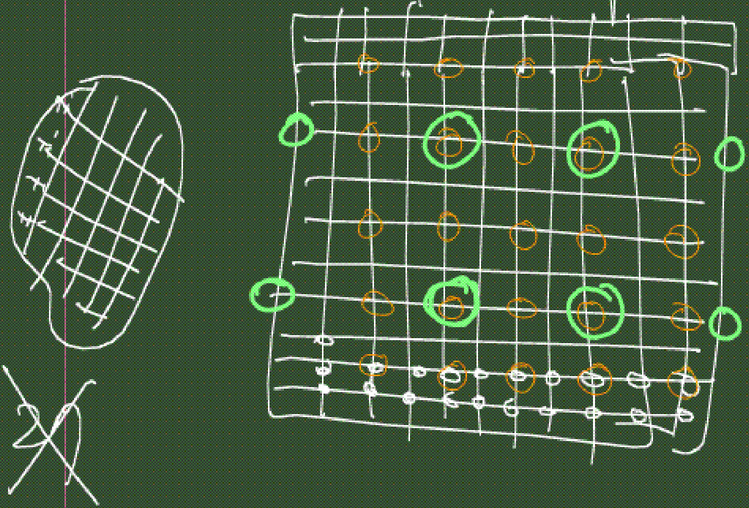

| | Error | Solver | Multigrid method |
|---|---|---|---|
| Jacobi | Rough | Bad | X |
| $\boxed{\text{G-S}}$ | Smooth | Bad | O |
| SOR | Rough | Good | X |
| SIP | Smooth | Good | O |
| ADI | Rough | Good | X |

- V cycle    W cycle



- Single grid transfers the information to an adjacent grid per iteration.

Multigrid transfer the information to all grids per iteration.

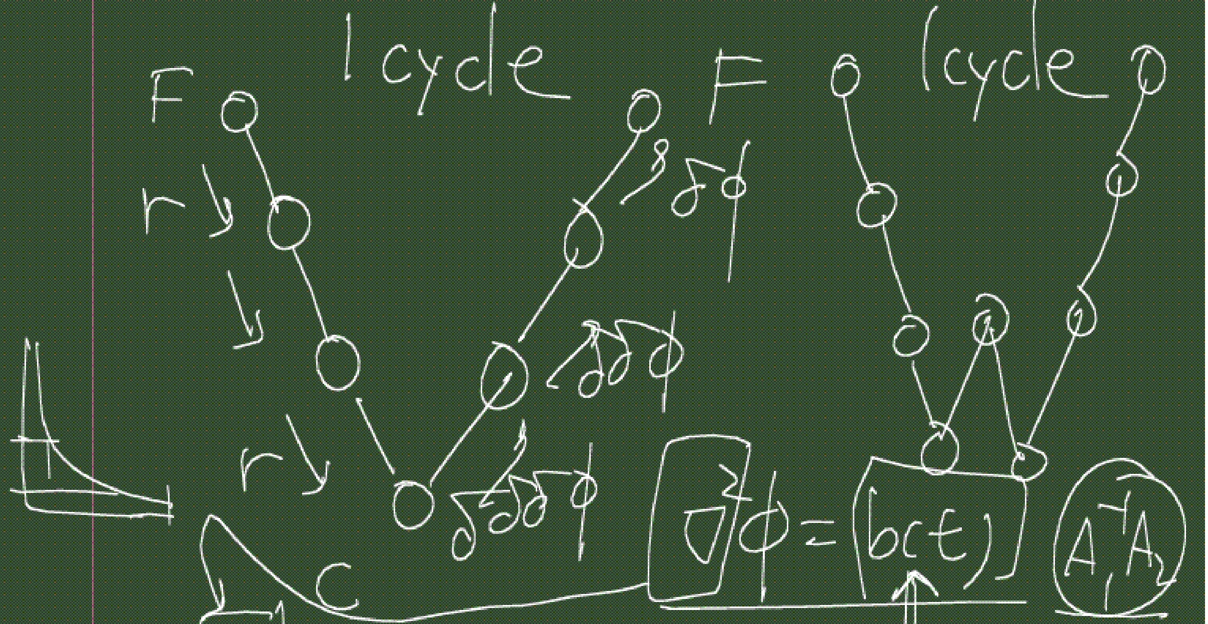


$2^M \times M + 1$

- Convergence rate depends on smoother and cycle
- Number of cycles for convergence is independent of N.

$\boxed{A}\phi = b \qquad r = b - A\phi \Rightarrow \boxed{10^{-5}}$ 10 $\quad 10^{-6}$

$|\phi^{n+1} - \phi^{n}| \le 10^{-7}$ 100




$\boxed{\nabla^2}\phi = \boxed{b(t)} \qquad A_1^{-1}A_2$

$\boxed{A}\phi = b(t) \Rightarrow A_1\underline{\phi} = A_2\phi + \underline{\underline{b}}$

(*) Hyperbolic PDE $(b^2 - ac > 0)$

$$a\frac{\partial^2 u}{\partial x^2} + 2b\frac{\partial^2 u}{\partial x \partial y} + c\frac{\partial^2 u}{\partial y^2} = F$$

Equation for the characteristics

$$a\left(\frac{\partial \phi}{\partial x}\right)^2 + 2b\left(\frac{\partial \phi}{\partial x}\right)\left(\frac{\partial \phi}{\partial y}\right) + c\left(\frac{\partial \phi}{\partial y}\right)^2 = 0$$

$$\rightarrow a\left(\frac{\partial \phi}{\partial x}\right)^2 \Big/ \left(\frac{\partial \phi}{\partial y}\right)^2 + 2b\left(\frac{\partial \phi}{\partial x}\right) \Big/ \left(\frac{\partial \phi}{\partial y}\right) + c = 0$$

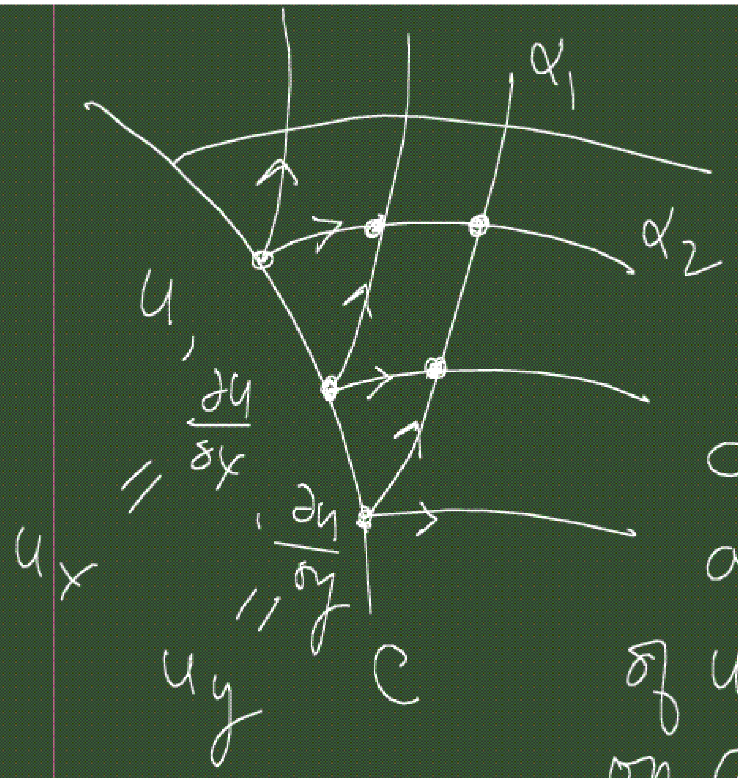
characteristic lines: $\phi(x, y) = \text{const.}$

$$d\phi = \frac{\partial \phi}{\partial x}dx + \frac{\partial \phi}{\partial y}dy = 0$$

$$\rightarrow \frac{dy}{dx} = -\frac{\partial \phi}{\partial x} \Big/ \frac{\partial \phi}{\partial y}$$

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

Method of Characteristics (MOC) on char. lines




We need to know the locations of char. lines and variation of $u_x$ and $u_y$ on char. lines

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

$$\left.\begin{aligned} d\left(\frac{\partial u}{\partial x}\right) &= \frac{\partial^2 u}{\partial x^2}dx + \frac{\partial^2 u}{\partial x \partial y}dy \\ d\left(\frac{\partial u}{\partial y}\right) &= \frac{\partial^2 u}{\partial x \partial y}dx + \frac{\partial^2 u}{\partial y^2}dy \end{aligned}\right]$$

We have set of eqs.

$$\begin{cases} a\, u_{xx} + 2b\, u_{xy} + c\, u_{yy} = F \\ dx\, u_{xx} + dy\, u_{xy} = du_x \\ dx\, u_{xy} + dy\, u_{yy} = du_y \end{cases}$$

$$\begin{vmatrix} a & 2b & c \\ dx & dy & 0 \\ 0 & dx & dy \end{vmatrix} = 0 \rightarrow \text{eq. of char.}$$

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

$$\begin{vmatrix} a & F & c \\ dx & du_x & 0 \\ 0 & du_y & dy \end{vmatrix} = 0.$$

$$\rightarrow dy\, a\, du_x - dy\, F\, dx + du_y \cdot c\, dx = 0$$

$$\rightarrow \frac{dy}{dx} a\, du_x - F dy + c\, du_y = 0$$

For $dy/dx = \alpha_1$, $\quad \alpha_1 a\,du_x + c\,du_y - F\,dy = 0$

$\qquad\quad = \alpha_2$, $\quad \alpha_2 a\,du_x + c\,du_y - F\,dy = 0$

From chain rule, $\quad du = u_x dx + u_y dy$

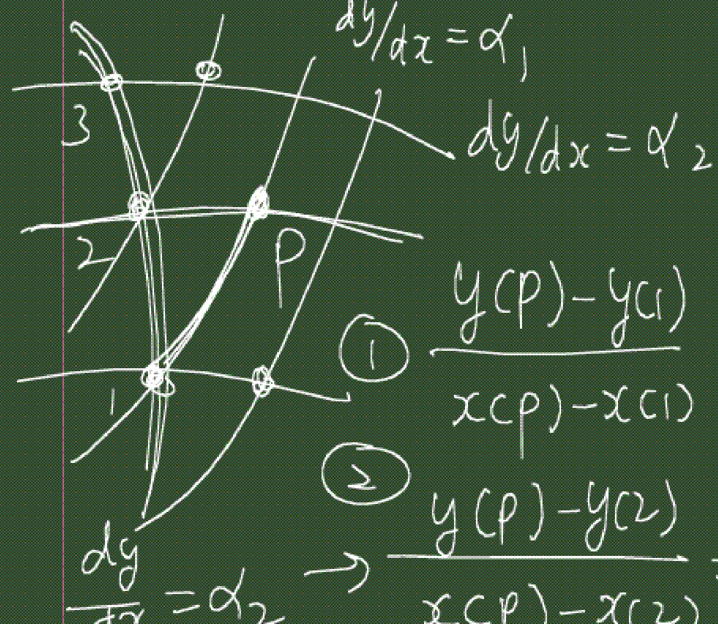
$dy/dx = \alpha_1$

$dy/dx = \alpha_2$

5 eqs

unknowns ; $x$, $y$, $u$, $u_x$, $u_y$

PDE $\xrightarrow{\text{MOC}}$ ODEs




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

① $$\frac{y(P) - y(1)}{x(P) - x(1)} = \frac{1}{2}\left[\alpha_1(P) + \alpha_1(1)\right]$$

② $$\frac{dy}{dx} = \alpha_2 \rightarrow \frac{y(P) - y(2)}{x(P) - x(2)} = \frac{1}{2}\left[\alpha_2(P) + \alpha_2(2)\right]$$

$$\alpha_1 a \, du_x + c \, du_y - F dy = 0$$

$$\rightarrow \cancel{\tfrac{1}{2}}\left[\alpha_1(p) a(p) + \alpha_1(1) a(1)\right] \cdot \left[u_x(p) - u_x(1)\right]$$

$$+ \cancel{\tfrac{1}{2}}\left[c(p) + c(1)\right]\left[u_y(p) - u_y(1)\right]$$

$$- \cancel{\tfrac{1}{2}}\left[F(p) + F(1)\right]\left[y(p) - y(1)\right] = 0$$

3

$$\alpha_2 a \, du_x + c \, du_y - F dy = 0$$

$$\left[\alpha_2(p) a(p) + \alpha_2(2) a(2)\right]\left[u_x(p) - u_x(2)\right]$$

$$+ \left[c(p) + c(2)\right]\left[u_y(p) - u_y(2)\right]$$

$$- \left[F(p) + F(2)\right]\left[y(p) - y(2)\right] = 0$$

4

$$du = u_x\,dx + u_y\,dy$$

$$u(p) - u(1) = \frac{1}{2}\left[u_x(p) + u_x(1)\right]\left[x(p) - x(1)\right] + \frac{1}{2}\left[u_y(p) + u_y(1)\right]\left[y(p) - y(1)\right] \quad (5)$$

5 eqs for $x(p)$, $y(p)$, $u(p)$, $u_x(p)$ & $u_y(p)$

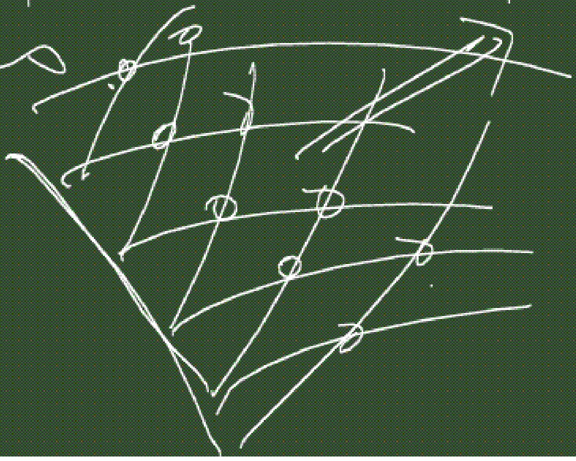
In general, they are nonlinear.
$\rightarrow$ iterative method.

Advantages of MOC:

① important properties of the exact sol. are preserved.

② method is easily adapted to the computation of probs. that contains discontinuities.

③ ability to compute the sol. over a long span of the indep. variables



Disadvantages of MOC:

① difficulty of keeping track of the locations of the characteristics and the values of the variables in 3D.

② mixed eq. type. | Hy | Ell: |

① Explicit methods for hyperbolic equation.

✓ convection eq. $\boxed{\frac{\partial \phi}{\partial t} + c\frac{\partial \phi}{\partial x} = 0}$

$x - c t = \text{const}$

✓ wave eq

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

$$\rightarrow \begin{cases} \dfrac{\partial \phi}{\partial t} = c\dfrac{\partial w}{\partial x} \\ \dfrac{\partial w}{\partial t} = c\dfrac{\partial \phi}{\partial x} \end{cases} \qquad \rightarrow \frac{\partial^2 \phi}{\partial t^2} = c\frac{\partial^2 w}{\partial x \partial t} = c\frac{\partial}{\partial x}\left(c\frac{\partial \phi}{\partial x}\right)$$

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

EE + CD2 : unstable

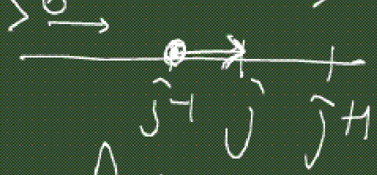
Leapfrog + 〃 : stable

no amplitude error for $\lambda = i\omega$

$|\omega h| < 1$

spurious sol.

upwind scheme ($\because$ one char.)




+ EE :

$$\frac{\phi_j^{n+1} - \phi_j^n}{\Delta t} = -\frac{c}{\Delta x}\left(\phi_j^n - \phi_{j-1}^n\right)$$

Von Neumann stability

$$\rightarrow \frac{c\Delta t}{\Delta x} < 1 \quad \text{CFL} < 1$$

For wave eq., two chars.

$\longrightarrow$ $\longleftarrow$ no upwind scheme

central diff. scheme

EE $\to$ unstable

Leap frog $\to$ stable for CFL < 1

- Lax - Wendroff method

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

$$\phi(x, t+\Delta t) = \phi(x) + \Delta t \boxed{\frac{\partial \phi}{\partial t}}(x,t) + \frac{1}{2}\Delta t^2 \boxed{\frac{\partial^2 \phi}{\partial t^2}}(x,t) + \cdots$$

$$\frac{\partial \phi}{\partial t} = -c\frac{\partial \phi}{\partial x}$$

$$\frac{\partial^2 \phi}{\partial t^2} = \frac{\partial}{\partial t}\left(\frac{\partial \phi}{\partial t}\right) = \frac{\partial}{\partial t}\left(-c\frac{\partial \phi}{\partial x}\right) = c^2\frac{\partial^2 \phi}{\partial x^2}$$

$$\rightarrow \phi_j^{n+1} = \phi_j^n - c\Delta t \frac{\phi_{j+1}^n - \phi_{j-1}^n}{2\Delta x} + \frac{1}{2}\Delta t^2 c^2 \frac{\phi_{j+1}^n - 2\phi_j^n + \phi_{j-1}^n}{\Delta x^2} + \cdots$$

2nd-order accurate in space & time

explicit

stable when CFL<1

dispersive error

problem :

when c is not const, nonlinear eq, high dimension $\rightarrow$ difficulty

This method can be applied to

$$\frac{\partial^2\phi}{\partial t^2} = c^2 \frac{\partial^2\phi}{\partial x^2} \quad \text{by} \quad \begin{cases} \dfrac{\partial u}{\partial t} = c\dfrac{\partial w}{\partial x} \\ \dfrac{\partial w}{\partial t} = c\dfrac{\partial u}{\partial x} \end{cases}$$

If there is a discontinuity


↓

ENO, TVD, - - -

↓

essentially non-oscillating scheme

total variation diminished scheme

---

∘ Implicit methods for hyperbolic eqs.

Crank-Nicolson method

$$\frac{\partial\phi}{\partial t}+c\frac{\partial\phi}{\partial x}=0 \qquad \frac{\phi^{n+1}_{j+1}-\phi^{n+1}_{j-1}}{2\Delta x} \qquad \frac{\phi^{n}_{j+1}-\phi^{n}_{j-1}}{2\Delta x}$$

$$\parallel \qquad \parallel$$

CD2; $$\frac{\phi_j^{n+1}-\phi_j^{n}}{\Delta t}+c\,\frac{1}{2}\left[\frac{\partial\phi^{n+1}}{\partial x}+\frac{\partial\phi^{n}}{\partial x}\right]=0$$

no amplitude error

dispersive error, stable

$$\frac{\partial^2\phi}{\partial t^2} = c^2\frac{\partial^2\phi}{\partial x^2}$$

→ split into two eqs. $\phi$ & $w$.

and apply C-N to two eqs,

or

→ apply C-N directly to wave eq.

$$\frac{\phi_j^{n+1} - 2\phi_j^n + \phi_j^{n-1}}{\Delta t^2} = \frac{c^2}{2\Delta x^2}\left[(\phi_{j+1}^{n+1} - 2\phi_j^{n+1} + \phi_{j-1}^{n+1}) + (\;''^{\,n-1} - \;^{n-1} + \;^{n-1})\right]$$

uncond. stable

a little more accurate than the method applied to factored equation.

C-N to nonlinear hyperbolic eq may have instability.

multi-dimension → ADI

implicit method for nonlinear eq → difficulty

⇒ Predictor - Corrector method

MacCormack - very popular

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

$$\rightarrow \quad \phi_j^{*} = \phi_j^{n} - \frac{c\Delta t}{\Delta x}\left(\phi_{j+1}^{n} - \phi_j^{n}\right)$$

$$\phi_j^{n+1} = \frac{1}{2}\left(\phi_j^{n} + \phi_j^{*}\right) - \frac{c\Delta t}{2\Delta x}\left(\phi_j^{*} - \phi_{j-1}^{*}\right)$$

equivalent to the Lax-Wendroff method for linear prob.

CFL < 1

accurate and easy to program

desirable nonlinear properties

readily extended to 2 & 3 D.

# Ch 6. Discrete Transform Methods - Spectral method.

## 6.1 Fourier series

$$f(x) = \sum_{k=-\infty}^{\infty} \hat{f}_k e^{ikx}$$

continuous ft. $\quad \hat{f}_k$ : Fourier coefficient

$$f'(x) = \sum_{k=-\infty}^{\infty} \underbrace{ik\hat{f}_k}_{\text{Four. coeff. of } f'} e^{ikx}$$

$$f(x) \xrightarrow{FT} \hat{f}_k \rightarrow ik\hat{f}_k \xrightarrow{IFT} f'$$

## 6.1.1 Discrete Fourier series

$f$ : periodic ft.

$N$ grid pts. $\rightarrow$ even number

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 1 | 2 | $\cdots$ $j$ | | $N-2$ | $N-1$ |
| $x_0$ | $x_1$ | $x_2$ | $x_j$ | | $x_{N-2}$ | $x_{N-1}$ |
| $f_0$ | $f_1$ | $f_2$ | $f_j$ | | $f_{N-2}$ | $f_{N-1}$ |

Let us take $N$ to be even number

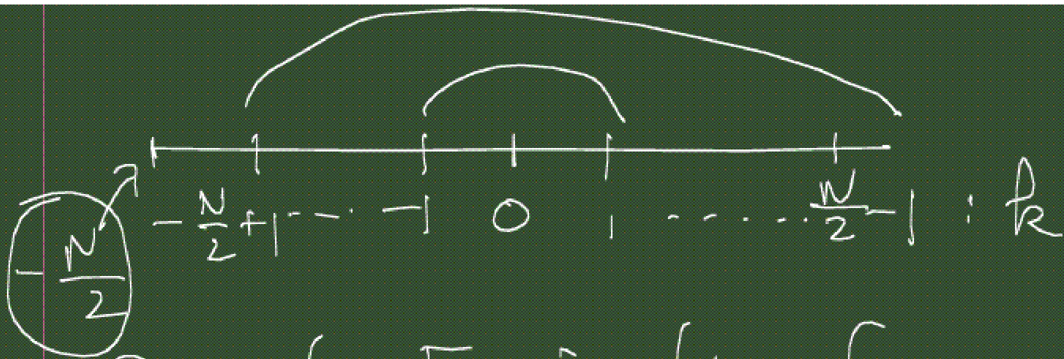
& $\boxed{\text{period of } f \text{ to be } 2\pi}$.

$e^{ikx}$ : $\Delta k \cdot (2\pi) = 2\pi \rightarrow \Delta k = 1$

grid spacing $h = 2\pi / N$

$x_j = j h$

$f$ ; periodic ft. $\Rightarrow f_N = f_0$




Discrete Fourier transform

$$f_j = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{f}_k e^{ikx_j}, \quad j = 0, 1, 2, \cdots N-1$$

↳ discrete Fourier coeff. of $f$

(If period is L,

$e^{ikx}$ : $\Delta k \cdot L = 2\pi \rightarrow \Delta k = \frac{2\pi}{L}$

and $h = L/N$ )

$$\boxed{f_j = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{f}_k e^{ikx_j}} \quad - \ (*)$$

$$\begin{pmatrix} \text{———} \\ \text{———} \\ \text{———} \\ \text{———} \\ \text{———} \\ \text{———} \\ \text{—} \end{pmatrix} \begin{pmatrix} \hat{f}_{-\frac{N}{2}} \\ \hat{f}_{-\frac{N}{2}+1} \\ \vdots \\ \hat{f}_0 \\ \vdots \\ \hat{f}_{\frac{N}{2}-1} \end{pmatrix} = \begin{pmatrix} f_0 \\ f_1 \\ \vdots \\ \vdots \\ f_{N-1} \end{pmatrix}$$

Gauss elimination to get $\hat{f}_k$

$\rightarrow$ expensive!

Use orthogonality!

$$I = \sum_{j=0}^{N-1} e^{ikx_j} e^{-ik'x_j}$$

$$= \sum_{j=0}^{N-1} e^{i\underset{\substack{\parallel \\ 2\pi/N}}{h}(k-k')j} = \begin{cases} N & \text{if } k = k' + mN \\ 0 & \text{otherwise} \end{cases}$$

$(m = 0, \pm 1, \pm 2, \cdots)$

$$\sum_{j=0}^{N-1} (*)\, e^{-ik'x_j}$$

$$\sum_{j=0}^{N-1} f_j e^{-ik'x_j} = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \sum_{j=0}^{N-1} \hat{f}_k e^{i(k-k')x_j}$$

(using the orthogonality)

$$\rightarrow \boxed{\hat{f}_k = \frac{1}{N} \sum_{j=0}^{N-1} f_j e^{-ikx_j}} \quad (**)$$

## 6.1.2 Fast Fourier Transform (FFT)

Operation count for (*) or (**)

is about $4N^2$.

FFT $\longrightarrow O(N \log_2 N)$

↳ See Numerical Recipes for the routine.

## 6.1.3 FT of a real function

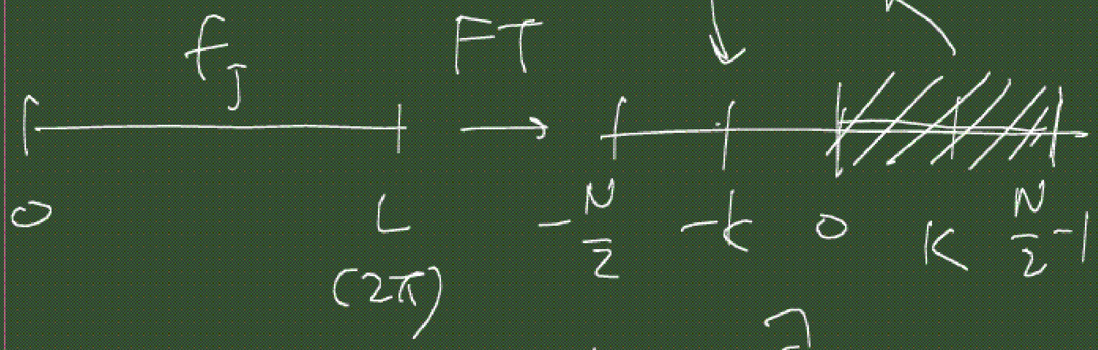
$f$: real $\longrightarrow \hat{f}$: complex

$$\hat{f}_k = \frac{1}{N} \sum_0^{N-1} f_j e^{-ikx_j}$$

$$\left( \hat{f}_{-k} = \quad " \quad e^{ikx_j} \right)^* \leftarrow \text{conjugate}$$

$$\rightarrow \hat{f}_{-k}^* = \frac{1}{N} \sum f_j e^{-ikx_j} = \hat{f}_k$$

$$\therefore\ \hat{f}_k = \hat{f}_{-k}^* \quad \text{or} \quad \boxed{\hat{f}_{-k}} = \hat{f}_k^*$$




no need to store $\hat{f}_{-k}$
when f is real

12/16 2:30 — 4:30

~~Mid~~ Final

# 6.1.4 Discrete Fourier Series in Higher Dimensions

$f(x,y)$ : doubly periodic in $x$ & $y$ directions

$N_1$ in $x$ & $N_2$ in $y$

$$f(x_m, y_l) = \sum_{k_1=-\frac{N_1}{2}}^{\frac{N_1}{2}-1} \sum_{k_2=-\frac{N_2}{2}}^{\frac{N_2}{2}-1} \hat{f}_{k_1,k_2} e^{ik_1x_m} e^{ik_2y_l}$$

$m = 0, 1, \cdots, N_1 - 1$

$l = 0, 1, \cdots, N_2 - 1$

Using the orthogonality,

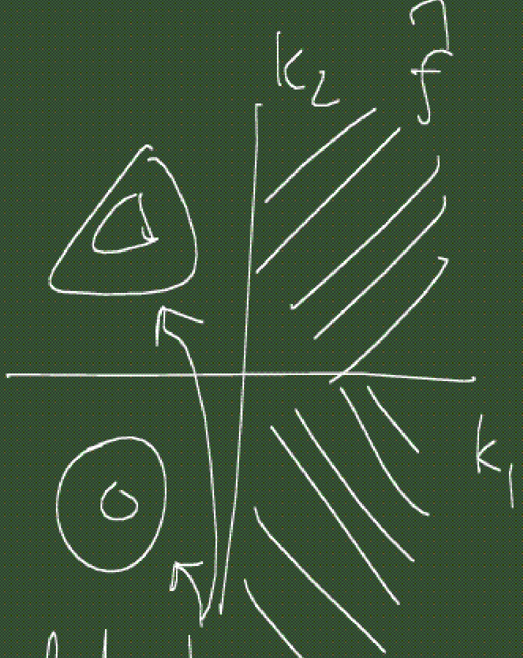
$$\hat{f}_{k_1,k_2} = \frac{1}{N_1}\frac{1}{N_2} \sum_{m=0}^{N_1-1} \sum_{l=0}^{N_2-1} f_{m,l} e^{-ik_1x_m} e^{-ik_2y_l}$$

If $f$ is real,

$$\hat{f}^*_{-k_1,-k_2} = \hat{f}_{k_1,k_2}$$

or

$$\hat{f}_{-k_1,-k_2} = \hat{f}^*_{k_1,k_2}$$




no need to store data here

## 6.1.5 DFT of a product of two fts.

Let $H(x) = f(x)g(x)$

$$\hat{H}_m = \widehat{fg}_m = \frac{1}{N}\sum_{j=0}^{N-1} f_j g_j e^{-imx_j}$$

$$= \frac{1}{N}\sum_{j=0}^{N-1} \sum_k \hat{f}_k e^{ikx_j} \sum_{k'} \hat{g}_{k'} e^{ik'x_j} e^{-imx_j}$$

$$= \frac{1}{N}\sum_j \sum_k \sum_{k'} \hat{f}_k \hat{g}_{k'} e^{i(k+k'-m)x_j}$$

$$= \sum_k \sum_{k'} \hat{f}_k \hat{g}_{k'} \underbrace{\frac{1}{N} \sum_j e^{i(k+k'-m)x_j}}$$

$$x_j = h j = \frac{2\pi}{N} j$$

$$\begin{cases} 1 & \text{if } k+k'-m = \pm \alpha N \\ 0 & \text{otherwise} \end{cases}$$

Sum over $j$ is non-zero only if

$$k+k' = m \quad \text{or} \quad k+k' = m \pm N \quad \| \cancel{\pm 2N}$$

The part of the summation corresponding to $k+k' = m \pm N$ is known as the aliasing error and should be discarded because the Fourier exponentials corresponding to these wavenumbers cannot be resolved on the grid of size $N$.

$\therefore \hat{H}_m = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{f}_k \hat{g}_{m-k}$ : convolution sum of $\hat{f}$ & $\hat{g}$

~~xxx~~

$\hookrightarrow O(N^2)$ operation $\rightarrow$ expensive!

$f \rightarrow \hat{f}$, $g \rightarrow \hat{g}$ $\Big\} \rightarrow \hat{H}$

$O(N) - f\,g \xrightarrow[O(N\log_2 N)]{FFT} \widehat{fg} = \hat{H}$ : much less cost

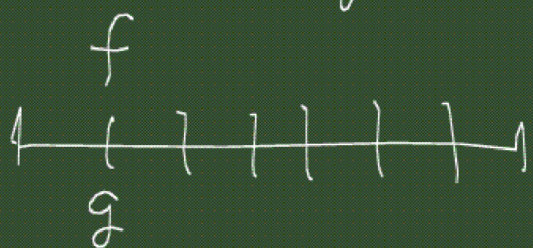
If we simply multiply f and g at each grid pt., the resulting discrete ft. will be contaminated by the aliasing errors and not be equal to the inverse FT of $\hat{H}_m$.

Sufficient grid pts. → small aliasing error.

**ex** $f(x) = \sin 2x$

$g(x) = \sin 3x \qquad 0 \le x \le 2\pi$

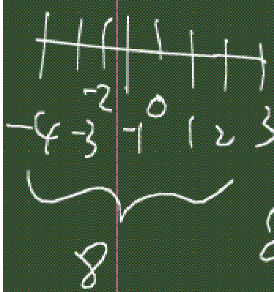
$$H(x) = f(x)\,g(x) = \frac{1}{2}(\cos x - \cos 5x)$$




$f \cdot g$

$$x_j = \frac{2\pi}{N} j, \quad j = 0, 1, \cdots, N-1$$

For $N \ge 8$, DFTs are




$$\hat{f}_k = \begin{cases} \mp i\frac{1}{2} & \text{for } k = \pm 2 \\ 0 & \text{otherwise} \end{cases}$$

$$\hat{g}_k = \begin{cases} \mp i\frac{1}{2} & \text{for } k = \pm 3 \\ 0 & \text{otherwise} \end{cases}$$

With $N=16$ ($k=-8,\dots,7$),

can resolve up to $\cos 7x$

using or multiplying $f$ & $g$ at each grid pt.,

$$\rightarrow \hat{H}_k = \begin{cases} \frac{1}{4} & \text{for } k=\pm 1 \\ -\frac{1}{4} & \text{for } k=\pm 5 \\ 0 & \text{otherwise} \end{cases}$$

---

$N=16$. $\hat{H}_k = \sum\limits_{m=-8}^{7} \hat{f}_m \hat{g}_{k-m}$

$k=5$: $\hat{H}_5 = \sum \hat{f}_m \hat{g}_{5-m}$

$$= \hat{f}_2 \hat{g}_{5-2} + \hat{f}_{-2} \cancelto{0}{\hat{g}_{5+2}}$$

$$= -\frac{1}{4}$$

$k=1$: $\hat{H}_1 = \dots = \frac{1}{4}$

$\vdots$

With $N=8$, $(k=-4, \cdots, 3)$

Using ~~[illegible]~~

$$\hat{H}_k = \begin{cases} \frac{1}{4} & \text{for } k=\pm 1 \\ 0 & \text{otherwise} \end{cases} \rightarrow$$

accurate for $k=1$ but lose $\cos 5x$!

$$\boxed{\frac{1}{2}\cos x}$$

But it is ok within the grid pt.!

Multiplying $f$ & $g$ at each grid pt. and then FT

$$\hat{H}_k = \begin{cases} \frac{1}{4} & \text{for } k=\pm 1 \\ -\frac{1}{4} & k=\pm 3 \\ 0 & \text{otherwise} \end{cases}$$

aliasing error

$$\Downarrow$$

$$\frac{1}{2}(\cos x - \cos 3x)$$

$$\hat{H}_k = \frac{1}{N}\sum_{j=0}^{N-1} H_j e^{-ikx_j} \qquad e^{i\boxed{5}x_j}$$

orthogonality

$$\sum_{j=0}^{N-1} e^{ih(k-k')j} = \begin{cases} N & \text{if } k-k'=mN \\ 0 & \text{otherwise} \end{cases}$$

if $5 = k' + mN$, non-zero value

<u>$N=8$</u>: $k' = 5 - mN = 5 - 8m = \boxed{-3}$ $(m=1)$

$(k' = -4 \ldots 3)$ $\quad k' = -3$ : $\hat{H}_k = -\frac{1}{4}$ <u>bad</u>

<u>$N=12$</u>; $k' = 5 - 12m = 5$ $(m=0)$

$\cancel{-7 \ (m=1)}$

$k' = 5$ ; $\hat{H}_5 = -\frac{1}{4}$ <u>good</u>

$N=16$ ; $k' = 5 - 16m = 5$ $(m=0)$

$N_1 = 8$ was sufficient for $f$ or $g$.

$N_2$ for $fg$? $\quad N_2 \geq N_1 \times \frac{3}{2}$

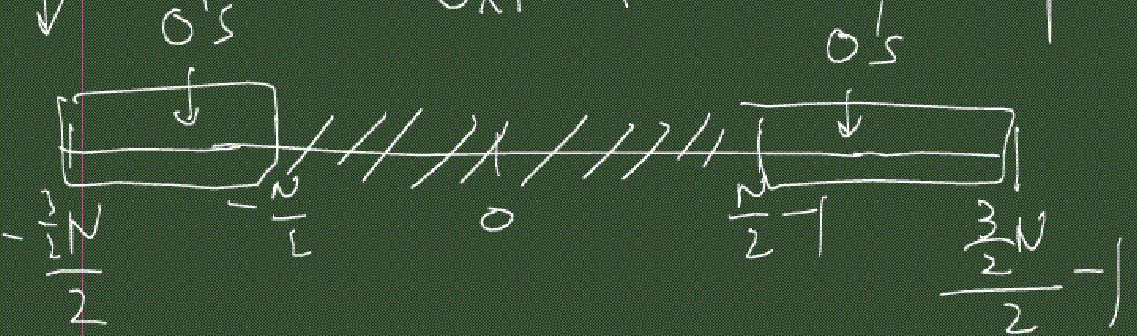
aliasing control.

$f,\ g$ w/ $N$

$\downarrow$

$\hat{f}$ $\hat{g}$ w/ $N$

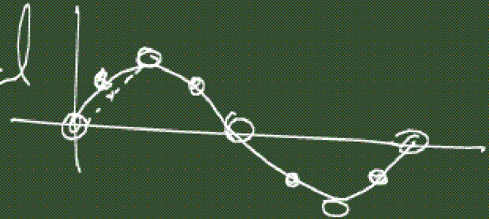
$\downarrow$ extend wavenumber space



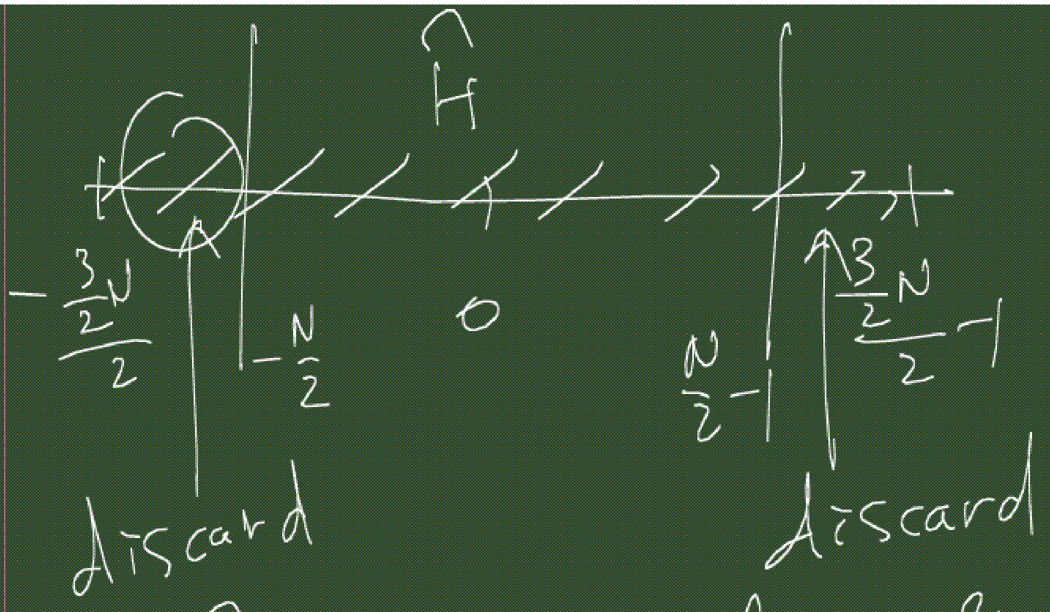

2FT $\downarrow$ (spectral interpolation)

$f$ & $g$ w/ $\frac{3}{2}N$

$\downarrow$ $fg$ in physical space w/ $\frac{3}{2}N$



FT $\downarrow$ $\hat{H} = \widehat{fg}$ w/ $\frac{3}{2}N$




$\hat{H}$ w/ N aliasing control

## 6.1.6 Discrete sine and cosine transfs.

f is not perodic $\longrightarrow$ ~~FT~~

f is even ft. i.e. $f(x) = f(-x)$

$\longrightarrow$ cosine transform

f is odd ft. i.e. $f(x) = -f(-x)$

$\longrightarrow$ sine transform

$f$ $N+1$ pts. on $0 \le x \le \pi$.

cosine transform

$$\begin{cases} f_j = \sum_{k=0}^{N} a_k \cos k x_j \quad , \quad j = 0, 1, \cdots, N \\ a_k = \dfrac{2}{c_k N} \sum_{j=0}^{N} \dfrac{1}{c_j} f_j \cos k x_j \end{cases}$$

$k = 0, 1, \cdots, N$

where $c_\ell = \begin{cases} 2 & \text{if } \ell = 0, N \\ 1 & \text{otherwise} \end{cases}$

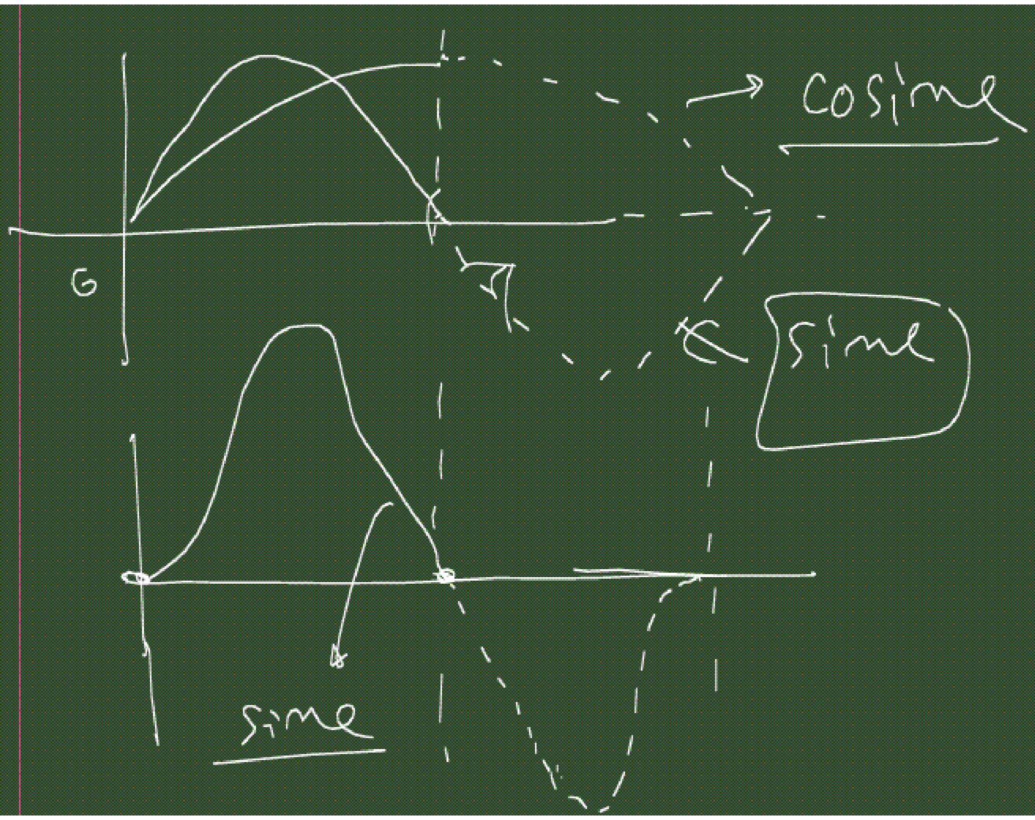
$x_j = jh$, $h = \pi/N$.

orthogonality

$$\sum_{j=0}^{N} \frac{1}{c_j} \cos k x_j \cos k' x_j = \begin{cases} 0 & \text{for } k \neq k' \\ \frac{1}{2} c_k N & k = k' \end{cases}$$

sine transform

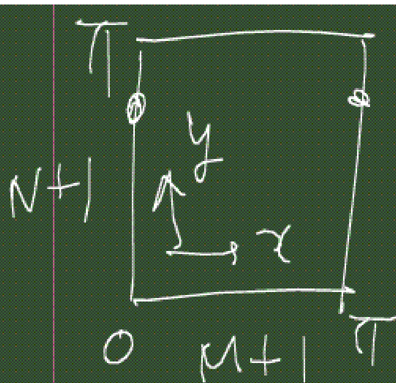
$$\begin{cases} f_j = \sum_{k=0}^{N} b_k \sin k x_j & j = 0, 1, \dots, N \\ b_k = \frac{2}{N} \sum_{j=0}^{N} f_j \sin k x_j & k = 0, 1, \dots, N \end{cases}$$



## 6.2 Applications of discrete Fourier series

### 6.2.1 Direct sol. of finite differenced elliptic eqs.

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = Q(x, y)$$ with $\phi = 0$ on boundary




$\Delta x = \pi / M$

$\Delta y = \pi / N$

CD2: $\frac{1}{\Delta x^2}\left(\phi_{i+1,j} - 2\phi_{i,j} + \phi_{i-1,j}\right)$

$(*)$ $+ \frac{1}{\Delta y^2}\left(\phi_{i,j+1} - 2\phi_{i,j} + \phi_{i,j-1}\right) = Q_{i,j}$

→ System of algebraic eqs for $(M-1)(N-1)$ unknowns

→ expensive to solve.

Use Fourier sine series in $x$

Assume
$$\phi_{i,j} = \sum_{k=1}^{M-1} \hat{\phi}_{k,j} \sin k x_i$$
$$Q_{i,j} = \text{"} \ \hat{Q}_{k,j} \ \text{"}$$

---

$$(*) \longrightarrow \sum_{k=1}^{M-1} \hat{\phi}_{k,j} \left\{ \sin\frac{\pi k}{M}(i+1) - 2\sin\frac{\pi k}{M} i + \sin\frac{\pi k}{M}(i-1) \right\}$$

$$\left( x_i = \frac{\pi}{M} i \right)$$

$$+ \frac{\Delta x^2}{\Delta y^2} \sum_{k=1}^{M-1} \left( \hat{\phi}_{k,j+1} - 2\hat{\phi}_{k,j} + \hat{\phi}_{k,j-1} \right) \sin\frac{\pi k}{M} i$$

$$= \Delta x^2 \sum_{k=1}^{M-1} \hat{Q}_{k,j} \sin\frac{\pi k}{M} i$$

$$\sin\frac{\pi k}{M} i \left( 2\cos\frac{\pi k}{M} - 2 \right)$$

Equating the coeffs. of $\sin\frac{\pi k}{M} i$ gives

$$\hat{\phi}_{k,j+1} + \left[\frac{\Delta y^2}{\Delta x^2}\left(2\cos\frac{\pi k}{M} - 2\right) - 2\right]\hat{\phi}_{k,j} + \hat{\phi}_{k,j-1} = \Delta y^2\, \hat{Q}_{k,j} \,!$$

For each k, tri-diagonal sys. of eqs.

Procedure

$$Q \xrightarrow[\Uparrow\; N\times O(M\log_2 M)]{ST} \hat{Q} \longrightarrow \begin{array}{c}\text{Solve}\\ \text{TDMA}\\ \text{for each } k\\ M\times O(N)\end{array} \longrightarrow \hat{\phi}_{k,j} \xrightarrow[N\times O(M\log_2 M)]{IST} \phi_{i,j}$$

Direct and low-cost method!

constraints :

- uniform grid in one direction
- coeff. of PDE should not be a fct. of transform direction.

- For probs. w/ Neumann b.c.'s

→ use cosine series.

Final exam: Dec. 16 (Wed) 2:30-  #118

노트 제목  2009-12-0?

6.2.2 Differentiation of a periodic ft. using Fourier spectral method

a periodic ft. $f(x)$

N equally spaced grid pts.

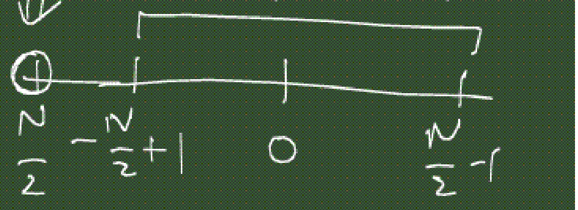
$$x_j = j\Delta x, \quad j = 0, 1, 2, \cdots, N-1$$

• Get the spectral derivative of $f$

① $\hat{f}_k = \frac{1}{N}\sum_{j=0}^{N-1} f_j e^{-ikx_j} \qquad k = \frac{2\pi}{N}\cdot n$

$n = -\frac{N}{2}, \cdots, 0, \cdots, \frac{N}{2}-1$

② $\widehat{Df}_k = ik\hat{f}_k$

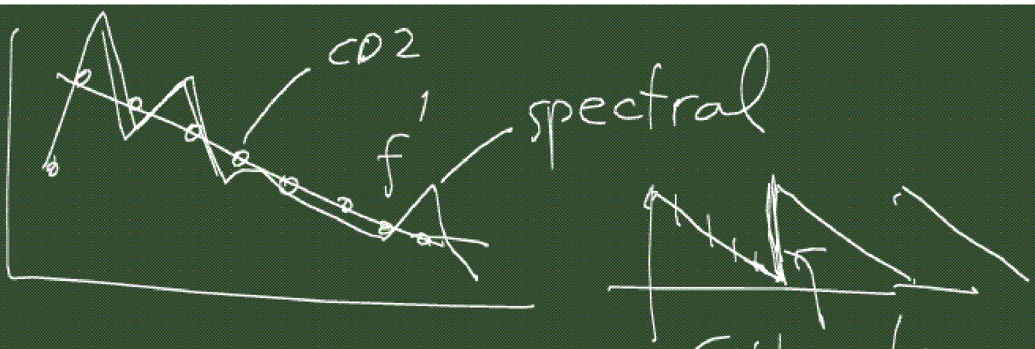



$\widehat{Df}_{k=-\frac{N}{2}} \equiv 0$ odd ball

③ $\left.\frac{\partial f}{\partial x}\right|_j = \sum_{k=-N/2}^{N/2-1} \widehat{Df}_k e^{ikx_j}$

provides exact derivative of the harmonic ft. $f(x) = e^{ikx}$ at the grid pts if $|k| \le \frac{N}{2} - 1$.

- The spectral derivative is more accurate than any FD scheme for periodic ft.
- Cost $\longrightarrow$ FFT

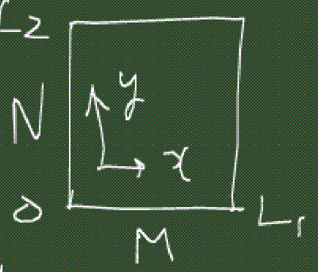



Gibbs phenomena

6.2.3 Numerical sol. of linear, constant coeff. diff'l eq. w/ periodic b.c's

Poisson eq

$$\Rightarrow \frac{\partial^2 \rho}{\partial x^2} + \frac{\partial^2 \rho}{\partial y^2} = Q(x, y)$$




$$\xrightarrow{FT} \left( \rho = \Sigma \hat{\rho} e^{ik_1 x} e^{ik_2 y} \right)$$

$$-k_1^2 \hat{\rho}_{k_1, k_2} - k_2^2 \hat{\rho}_{k_1, k_2} = \hat{Q}_{k_1, k_2}$$

$$\left( k_1 = \frac{2\pi}{L_1} n_1 , \quad k_2 = \frac{2\pi}{L_2} \cdot n_2 \right)$$

$$\rightarrow \hat{\rho}_{k_1, k_2} = -\frac{1}{k_1^2 + k_2^2} \hat{Q}_{k_1, k_2} \quad \text{for } k_1 = k_2 \neq 0$$

$$\hat{\rho}_{k_1, k_2} = \frac{1}{M} \frac{1}{N} \sum_{l=0}^{M-1} \sum_{j=0}^{N-1} \rho_{l,j} e^{-ik_1 x_l} e^{-ik_2 y_j}$$

$$\hat{\rho}_{0,0} = \frac{1}{M} \frac{1}{N} \sum_l \sum_j \rho_{l,j} \ : \ \text{average of } \rho \text{ over the domain.}$$

Solution of Poisson eq. w/ periodic b.c. is indeterminant ~~to~~ within an arbitrary constant.

Thus, set $\hat{P}_{0,0} = 0$

$\longrightarrow$ IFT to get $P(x_l, y_j)$.

- $$\iint \left( \frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2} \right) dx\, dy = \iint Q\, dx\, dy$$

$$\underset{\text{due to periodicity}}{\underbrace{\iint \left( \frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2} \right) dx\, dy}_{=0}} \qquad \iint Q\, dx\, dy = 0$$

$\rightarrow \sum_l \sum_j Q_{l,j} = 0$ ; required for the well posedness.

$\downarrow$

$\hat{Q}_{0,0} = 0$

ex

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2} + f(x,t)$$

$u$ : periodic

$$\xrightarrow{FT} \frac{d\hat{u}_k}{dt} + ik\hat{u}_k = -\nu k^2 \hat{u}_k + \hat{f}_k$$

$$\rightarrow \frac{d\hat{u}_k}{dt} = -(ik + \nu k^2)\hat{u}_k + \hat{f}_k$$

apply numerical method
for time integration

$\rightarrow$ do IFT to get $u(x,t)$.

## 6.3 Matrix operator for Fourier spectral numerical differentiation

Spectral Fourier derivative in physical space?

$$(Du)_l = \sum_{j=0}^{N-1} d_{lj} u_j \qquad l = 0, 1, \cdots, N-1$$

full matrix $\Rightarrow$ $d_{lj} = \begin{cases} \frac{1}{2}(-1)^{l-j}\cot\frac{\pi(l-j)}{N} & \text{for } l \neq j \\ 0 & \text{otherwise} \end{cases}$

→ Fourier spectral differentiation is matrix multiplication in physical space.

→ requires $O(N^2)$ operations

but $O(N\log_2 N)$ 〃 for FFT

## 6.4 Discrete Chebyshev transform and applications.

non-periodic ft?

or non-uniform mesh?

- Chebyshev transform

$u(x)$ defined in $-1 \leq x \leq 1$.

$$u(x) = \sum_{n=0}^{N} a_n T_n(x),$$

where $T_n(x)$ is chebyshev polynomials

that are sols. of diff'l eq.

$$\frac{d}{dx}\left(\sqrt{1-x^2}\,\frac{dT_n}{dx}\right) + \frac{\lambda_n}{\sqrt{1-x^2}} T_n = 0, \quad \lambda_n = n^2$$

$T_0 = 1,\ T_1 = x,\ T_2 = 2x^2-1,\ T_3 = 4x^3-3x, \dots$

$$-1 \le x \le 1 \longrightarrow 0 \le \theta \le \pi$$

$$x = \cos\theta$$

$$T_n(\cos\theta) = \cos n\theta \ !$$

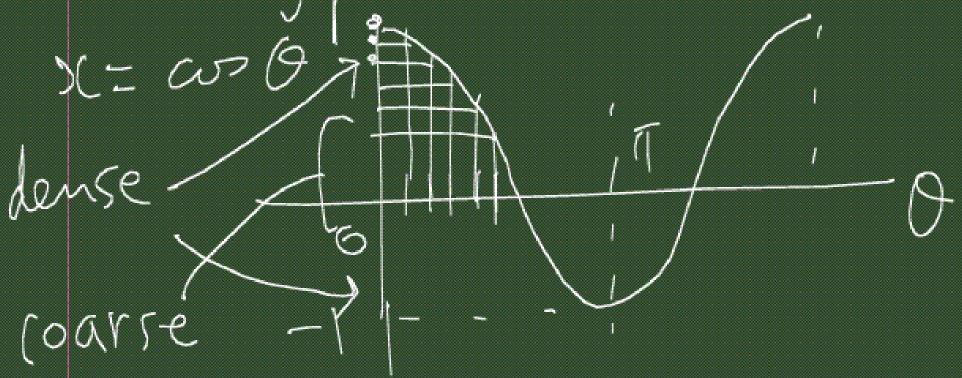
recursive relation:

$$T_{n+1}(x) + T_{n-1}(x) = 2x T_n(x) \qquad n \ge 1$$

To use chebyshev polynomials for numerical analysis, the domain

$-1 \le x \le 1$ is discretized using the 'cosine' mesh:

$$x_j = \cos \frac{\pi}{N} j \, , \quad j = N, N-1, \cdots, 1, 0$$




Orthogonality

$$\sum_{n=0}^{N} \frac{1}{c_n} T_m(x_n) T_p(x_n) = \begin{cases} N & \text{if } m=p=0, N \\ N/2 & \text{if } m=p \neq 0, N \\ 0 & m \neq p \end{cases}$$

$$c_n = \begin{cases} 2 & \text{if } n=0, N \\ 1 & \text{otherwise} \end{cases}$$

Discrete Chebyshev transform

$$\begin{cases} u_j = \sum_{n=0}^{N} a_n T_n(x_j) = \sum_{n=0}^{N} a_n \cos \frac{n\pi}{N} j \\ \qquad\qquad\qquad j = 0, 1, \cdots, N \\ a_n = \frac{2}{c_n N} \sum_{j=0}^{N} \frac{1}{c_j} u_j T_n(x_j) \end{cases}$$

$$= \frac{2}{c_n N} \sum_{j=0}^{N} \frac{1}{c_j} u_j \cos \frac{n\pi}{N} j \qquad n = 0, 1, \cdots, N$$

The Chebyshev coefficients for any ft. $u$ in $-1 \leq x \leq 1$ are exactly the coeffs. of the cosine transform obtained using the values of $u$ at the cosine mesh, i.e,

$$u_j = u\left(\cos \frac{\pi j}{N}\right).$$

## 6.4.1 Numerical differentiation using Chebyshev polynomials

$T_n(x) = \cos n\theta \quad , \quad x = \cos\theta$

$$\rightarrow \frac{dT_n}{dx} = \frac{d\cos n\theta}{d\theta}\frac{d\theta}{dx} = \frac{n\sin n\theta}{\sin\theta}$$

With $2\sin\theta\cos n\theta = \sin(n+1)\theta$
$\qquad - \sin(n-1)\theta$

$$\rightarrow 2T_n(x) = \frac{1}{n+1}T_{n+1}' - \frac{1}{n-1}T_{n-1}' \quad (n>1)$$

Now, $u(x) = \sum_{n=0}^{N} a_n T_n$

$$u'(x) = \sum_{n=0}^{N-1} b_n T_n = u' = \sum_{n=0}^{N} a_n T_n'$$

$$b_0T_0 + b_1T_1 + \sum_{n=2}^{N-1} b_n \frac{1}{2}\left(\frac{T_{n+1}'}{n+1} - \frac{T_{n-1}'}{n-1}\right)$$

$$= \sum_{n=0}^{N} a_n T_n'$$

Equating the coeffs. of $T_n'$

$\rightarrow \dfrac{b_{n-1}}{2n} - \dfrac{b_{n+1}}{2n} = a_n$

$\rightarrow b_{n-1} - b_{n+1} = 2n a_n \quad (n = 2, 3, \cdots, N-1)$

$$\begin{cases} b_N = 0 \\ b_{N-1} = 2N a_N \\ b_0 - \frac{1}{2} b_2 = a_1 \quad \text{for } n = 1 \end{cases}$$

$\left( \because T_1' = T_0, \; T_2' = 4T_1 \right)$

$\Rightarrow c_{n-1} b_{n-1} - b_{n+1} = 2n a_n, \quad n = 1, 2, \cdots, N$

with $b_N = b_{N+1} = 0$

$$\Rightarrow \boxed{b_m = \frac{2}{c_m} \sum_{\substack{p=m+1 \\ p+m \text{ odd}}}^{N} p\, a_p}$$

procedure for $u'$

$$u \xrightarrow{CT} a_n \longrightarrow b_n \xrightarrow{ICT} u'$$

## 6.4.2 Quadrature using Chebyshev polynomials

$$2T_n(x) = \frac{1}{n+1}T_{n+1}' - \frac{1}{n-1}T_{n-1}' \quad (n>1)$$

Integrate both sides

$$\to \int T_n(x)\,dx = \begin{cases} T_1 + \alpha_0 & n=0 \\ \frac{1}{4}(T_0+T_2)+\alpha_1 & n=1 \\ \frac{1}{2}\left[\frac{1}{n+1}T_{n+1} - \frac{1}{n-1}T_{n-1}\right]+\alpha_n & \text{otherwise} \end{cases}$$

$$g(x) = \int_{-1}^{x} u(\xi)\,d\xi = \sum_{n=0}^{N+1} d_n T_n$$

$$\sum_{n=0}^{N} a_n \int T_n(x)\,dx$$

$\cdots$

$$\begin{cases} d_n = \frac{1}{2n}(c_{n-1}a_{n-1} - a_{n+1}) & n=1,2,\cdots,N+1 \\ \text{with } a_{N+1} = a_{N+2} = 0 & \\ d_0 = d_1 - d_2 + d_3 \cdots + (-1)^{N+2} d_{N+1} & \end{cases}$$

$$u \xrightarrow{CT} a_n \longrightarrow d_n \xrightarrow{ICT} \int u\,dx$$

6.4.3 Chebyshev diff'l in physical space?

$$\underline{f}' = D\underline{f} \qquad D = \{d_{jk}\}$$

$$d_{jk} = \begin{cases} c_j(-1)^{j+k} / c_k(x_j - x_k) & j \neq k \\ -x_j / 2(1-x_j^2) & j = k,\ j \neq 0, N \\ (2N^2+1)/6 & j = k = 0 \\ -(2N^2+1)/6 & j = k = N \end{cases}$$

$$c_j = \begin{cases} 2 & j = 0, N \\ 1 & \text{otherwise} \end{cases}$$