

$$\frac{d\bar{u}}{dt} = \underline{f}(u_1, u_2, \dots, u_m) = \underline{f}(\bar{u})$$

노트 제목

2019-10-23

TR :  $\underline{u}_{n+1} - \underline{u}_n = \frac{h}{2} [\underline{f}(\underline{u}_{n+1}) + \underline{f}(\underline{u}_n)] + \Theta(h^3)$

### ① Linearization

$$\begin{aligned} \underline{f}_i(\underline{u}_{n+1}) &= \underline{f}_i(\underline{u}_n) + (\underline{u}_{1,n+1} - \underline{u}_{1,n}) \frac{\partial \underline{f}_i}{\partial \underline{u}_1} \Big|_n + (\underline{u}_{2,n+1} - \underline{u}_{2,n}) \frac{\partial \underline{f}_i}{\partial \underline{u}_2} \Big|_n \\ &\quad + \dots + (\underline{u}_{m,n+1} - \underline{u}_{m,n}) \frac{\partial \underline{f}_i}{\partial \underline{u}_m} \Big|_n + \Theta(h^2), \quad i=1, 2, \dots, m \end{aligned}$$

$$\underline{f}(\underline{u}_{n+1}) = \underline{f}(\underline{u}_n) + A(\underline{u}_{n+1} - \underline{u}_n) + \Theta(h^2) \quad \text{← neglect}$$

$$\Rightarrow \underline{u}_{n+1} - \underline{u}_n = \frac{h}{2} [\underline{f}(\underline{u}_n) + A(\underline{u}_{n+1} - \underline{u}_n) + \underline{f}(\underline{u}_n)] + \Theta(h^3)$$

$$A = \begin{pmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} & \cdots & \frac{\partial f_1}{\partial u_m} \\ \frac{\partial f_2}{\partial u_1} & \frac{\partial f_2}{\partial u_2} & \cdots & \frac{\partial f_2}{\partial u_m} \\ \vdots & & & \\ \frac{\partial f_m}{\partial u_1} & \frac{\partial f_m}{\partial u_2} & \cdots & \frac{\partial f_m}{\partial u_m} \end{pmatrix} \quad t = t_n$$

Jacobian matrix

↑  
full matrix

$$\Rightarrow \boxed{(I - \frac{h}{2} A) \underline{u}_{n+1} = (I - \frac{h}{2} A) \underline{y}_n + h \underline{f}(\underline{y}_n)}$$

linearized  
TR method

full matrix  $\rightarrow$  inverting  $(I - \frac{h}{2} A)$  requires  $\Theta(m^3)$  operations

$\Rightarrow$  may require iterative methods to solve this eq at each time step.  
linearization may have errors for strongly nonlinear prob.

② Iterative method w/o linearization

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

Newton - iterative method  $F(z) = 0$

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

$k$  : iteration index

$$\frac{dF}{dx}(x^{k+1} - x^k) = F^{k+1} - F^k = -F^k$$

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

$$\rightarrow \left. \frac{\partial F_i}{\partial u_j} \right|_{(u_j^{k+1} - u_j^k)} = -F_i^k$$

$$\left( \frac{\partial F_i}{\partial u_j}^k = \frac{\partial u_i}{\partial u_j} - o - \frac{h}{2} \left[ \frac{\partial f_i}{\partial u_j}^k + o \right] = \delta_{ij} - \frac{h}{2} \frac{\partial f_i}{\partial u_j}^k \right)$$

$$\rightarrow \boxed{(I - \frac{h}{2} A^k) (\underline{u}^{k+1} - \underline{u}^k) = -\underline{F}^k} = -\underline{u}^k + \underline{u}_n + \frac{h}{2} [f(\underline{u}^k) + f(\underline{u}^n)]$$

$k$ : iteration index

$$k=0 : \underline{u}^0 = \underline{u}_n$$

Solve this system of eqs. iteratively at each time step

→ 3~4 iterations per time step.

⑥

## Inherent instability

Consider  $y'' - k^2 y = 0$ ,  $y(0) = y_0$ ,  $y'(0) = -ky_0$  ( $k > 0$ )

exact sol.  $y = y_0 e^{-kx}$  well-behaved

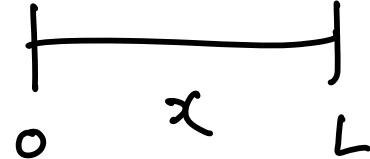
numerical sol.  $y = c_1 e^{-kx} + c_2 e^{kx}$

truncation and round-off errors will be  
exponentially amplified  
→ sol. diverges.

⇒ Inherently unstable

None of the standard numerical method will provide the correct sol.

## 4.11 Boundary value problems

$$\begin{cases} y'' = f(y, y', x) \\ y(0) = y_0, \quad y(L) = y_L \end{cases}$$


① shooting method

② direct method

① Shooting method : boundary value prob.  $\rightarrow$  initial value prob.

$y'' = f \rightarrow$  convert to two 1st-order ODEs.

$$u = y$$

$$v = u' = y' \rightarrow v' = y'' = f$$

$$\Rightarrow \begin{cases} u' = v & u(0) = y_0 \\ v' = f(u, v, x) & u(L) = y_L \end{cases}$$

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

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

$\rightarrow$  check if  $u(L) = y_L$ .

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

Consider a linear eq.

$$y''(x) + A(x)y'(x) + B(x)y(x) = f(x), \quad y(0) = y_0, \quad y(L) = y_L.$$

two guesses for  $y'(0)$  ( $= v(0)$ )  $\rightarrow y_1(x) \& y_2(x)$

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



iteration

$$y(x) = c_1 q_1(x) + c_2 q_2(x)$$

$$@ \quad x=0, \quad y(0) = c_1 y_1(0) + c_2 y_2(0) \rightarrow c_1 + c_2 = 1$$

$\frac{y_1}{y_1} \qquad \qquad \frac{y_2}{y_2}$

$$\textcircled{2} \quad x=L, \quad g(L) = c_1 \underbrace{g_1(L)}_{g_L} + c_2 \underbrace{g_2(L)}_{\text{numerical sols.}}$$

$$\Rightarrow c_1 = \frac{q_L - q_2(L)}{q_r(L) - q_2(L)}, \quad c_2 = 1 - c_1 \quad \Rightarrow \quad q(x) = c_1 q_1(x) + c_2 q_2(x)$$

↑  
numerical sol.

Satisfying  $q(0) = q_0$   
&  $q(L) = q_L$

Consider a nonlinear eq.

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

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

$$y'(0) = \text{④}$$

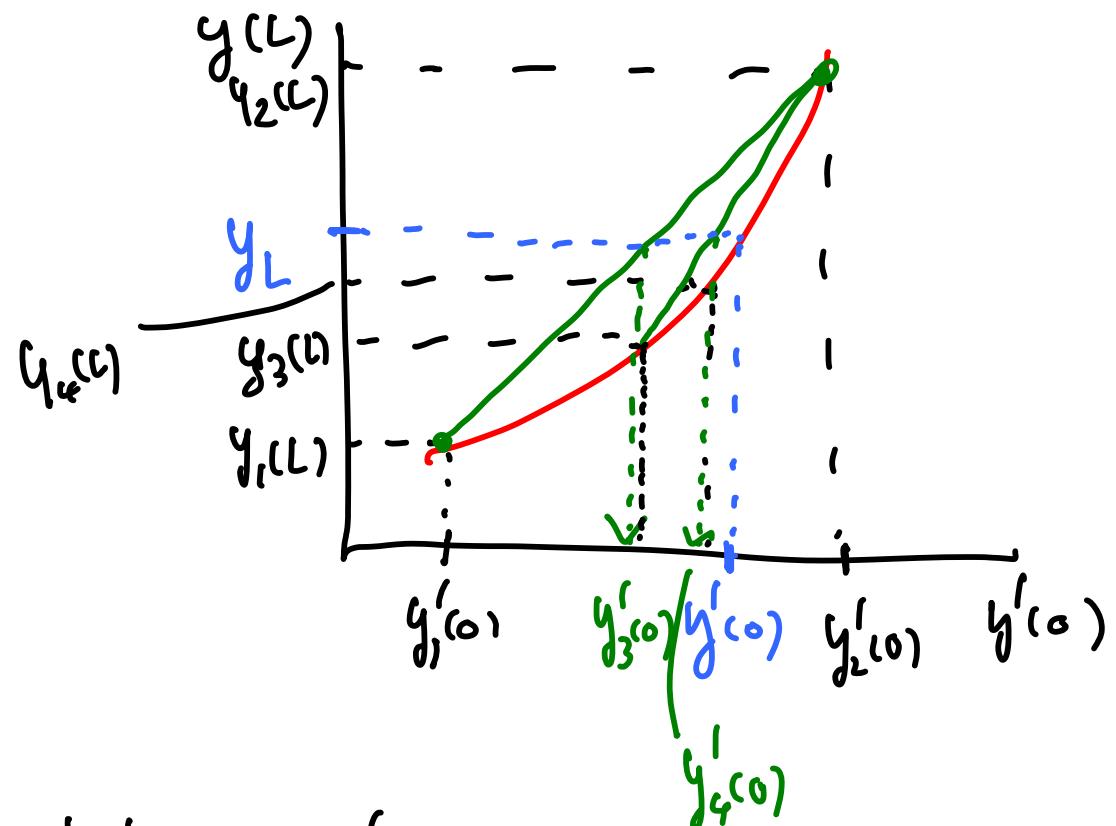
Secant method

$$y'_1(0) \rightarrow y_1(L) \neq y_L$$

$$y'_2(0) \rightarrow y_2(L)$$

→ form a straight line between  $(y'_1(0), y_1(L))$  &  $(y'_2(0), y_2(L))$ .

$$\text{slope } m = \frac{y - y_2(L)}{y' - y'_2(0)} = \frac{y_1(L) - y_2(L)}{y'_1(0) - y'_2(0)}$$



$$\rightarrow y' = q_2'(0) + \frac{1}{m} (y - q_2(L))$$

$$\text{next guess: } q_3'(0) = q_2'(0) + \frac{1}{m} (y_L - q_2(L))$$

$k$ : iteration index.

$$y_{k+1}'(0) = q_k'(0) + \frac{y_k'(0) - y_{k-1}'(0)}{y_k(L) - y_{k-1}(L)} (y_L - q_k(L))$$

ex) Blasius eq  $f''' + ff'' = 0$   $f(0), f(L), f'(L)$  given

$\hookrightarrow$  3 1st-order ODEs  $f(0)$

$$f'(0) = \epsilon_1 \rightarrow f(L)$$

$$f''(0) = \epsilon_2 \rightarrow f'(L)$$

② Direct method

Approximate the derivatives in the diff'l eq. w/ finite difference.  
Also, incorporate the b.c's as required.

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

$$\begin{matrix} 0(2) & \dots & j-1 & j+ & \dots & N-1 & N \\ | & & | & & | & & | \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{matrix}$$

$$x=0 \qquad \qquad x \qquad \qquad x=L$$

$$\rightarrow \frac{y_{j+1} - 2y_j + y_{j-1}}{h^2} + A_j \frac{y_{j+1} - y_{j-1}}{h} + B_j y_j = C_j \quad (\text{Eq 2})$$

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

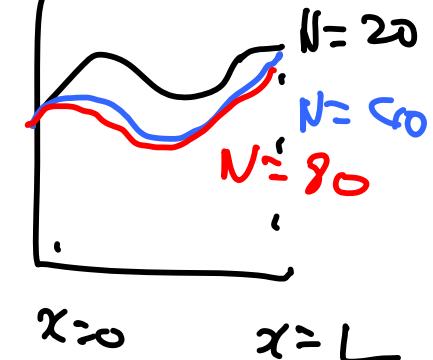
$$@ \quad j=1 : \alpha_1 y_2 + \beta_1 y_1 + \gamma_1 y_0 = c_1 \rightarrow \alpha_1 y_2 + \beta_1 y_1 = c_1 - \gamma_1 y_0$$

$$@ \quad j=N-1 : \alpha_{N-1} y_N + \beta_{N-1} y_{N-1} + \gamma_{N-1} y_{N-2} = c_{N-1}$$

$$\rightarrow \beta_{N-1} y_{N-1} + \gamma_{N-1} y_{N-2} = c_{N-1} - \alpha_{N-1} y_N$$

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

TDMA



$$y'' + y^2 y' = 1 \xrightarrow{\text{CD2}} \frac{y_{j+1} - 2y_j + y_{j-1}}{h^2} + y_j^2 \frac{y_{j+1} - y_{j-1}}{2h} = 1$$

nonlinear algebraic eq.

←  
requires iterative approach.

In some cases, shooting method may be better.