

## TOPICS COVERED

- More examples of discretisation of non-linear BVPs by Newton's method & quasilinear method.
- Intro to Spline interpolation

**EXAMPLE** (Recap of linearisation techniques)

$$y'' = 2 + y^2$$

$$y(0) = y(1) = 0$$

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

Discretise by Newton's linearisation technique:

$$\underbrace{\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} - y_i^2 - 2}_F(y_{i+1}, y_i, y_{i-1}) = 0$$

$$\text{At any iteration, } y_i^{(k+1)} = y_i^{(k)} + \Delta y_i$$

$$\frac{1}{h^2} \left[ (y_{i+1}^{(k)} + \Delta y_{i+1}) - 2(y_i + \Delta y_i) + (y_{i-1} + \Delta y_{i-1}) \right]$$

$$- (y_i^{(k)} + \Delta y_i^{(k)})^2 - 2 = 0, \quad i=1, 2, \dots, N$$

Unknowns are  $\Delta y_i$ 's, expand and retain the terms only up to linear order of  $\Delta y_i$ .

$$a_i \Delta y_{i-1} + b_i \Delta y_i + c_i \Delta y_{i+1} = d_i, \quad i=1, 2, \dots, N-1$$

Note:  $y_i^{(k)}$  are known

$$\begin{aligned} \frac{1}{h^2} \Delta y_{i-1} - \left( -\frac{2}{h^2} + 2y_i^{(k)} \right) \Delta y_i + \frac{1}{h^2} \Delta y_{i+1} \\ = 2 - \underbrace{\frac{y_{i+1}^{(k)} - 2y_i^{(k)} + y_{i-1}^{(k)}}{h^2} + (y_i^{(k)})^2}_{d_i} \end{aligned}$$

\*we do not need to store all  $y_j^{(k)}$   $\forall k$ , just last 2 are needed.

EXERCISE

$$f'' + f f' + \beta(1 - f'^2) = 0$$

$$f(0) = f'(0) = 0, \quad f'(10) = 1$$

$\beta$  is a real number.

This eqn is used to model flow of fluids.

$$\text{Ans. } F = f'$$

$$F'' + f F' + 1 - F^2 = 0$$

Step 1: Discretise

$$\frac{h}{2} (F_i + F_{i-1}) = f_i - f_{i-1}$$

$$\frac{F_{i+1} - 2F_i + F_{i-1}}{h^2} + f_i \frac{F_{i+1} - F_{i-1}}{2h} + \beta(1 - F_i^2) = 0$$

(1)

$$(F_i + \Delta F_i) - (F_{i-1} + \Delta F_{i-1}) - \frac{h}{2} ((F_i + \Delta F_i) + (F_{i-1} + \Delta F_{i-1})) = 0$$

(2)

$$\frac{1}{h^2} [(F_{i+1} + \Delta F_{i+1}) - 2(F_i + \Delta F_i) + (F_{i-1} + \Delta F_{i-1})]$$

$$+ \frac{(F_i + \Delta F_i)}{2h} [(F_{i+1} + \Delta F_{i+1}) - (F_{i-1} + \Delta F_{i-1})]$$

$$+ \beta (1 - F_i^2 - 2\Delta F_i F_i - \Delta F_i^2) = 0$$

$$\begin{pmatrix} -1 & -h/2 \\ 0 & \frac{1}{h^2} - \frac{f_i^{(k)}}{2h} \end{pmatrix} = A_i \quad \begin{pmatrix} 1 & -h/2 \\ \frac{f_{i+1}^{(k)} - f_{i-1}^{(k)}}{2h} & -\frac{2}{h^2} - 2\beta F_i \end{pmatrix} = B_i \quad \begin{pmatrix} 0 & 0 \\ 0 & \frac{1}{h^2} + \frac{f_i^{(k)}}{2h} \end{pmatrix} = C_i$$

$$\begin{pmatrix} -f_i^{(k)} + f_{i-1}^{(k)} + \frac{h}{2} (F_i^{(k)} + F_{i-1}^{(k)}) \\ -\beta + \beta (f_i^{(k)})^2 - \frac{F_{i+1}^{(k)} - 2F_i^{(k)} + F_{i-1}^{(k)}}{h^2} \end{pmatrix} = D_i$$

$$\max \{ |\Delta f_i|, |\Delta F_i| \}$$

Possible initial guess -

$$f = \frac{x^2}{20}$$

**EXERCISE**

$$f'' + ff'' + \beta(1 - f'^2) = 0$$

$$f(0) = f'(0) = 0, \quad f'(10) = 1$$

$\beta$  is a real number.

At every iteration obtain the quasi-linear form.

Ans.

$$F(f'', f', f, f) = 0$$

Let  $f^{(k)}, f'^{(k)}, f''^{(k)}, f'''^{(k)}$  be the approximate form of  $f, f', f'', f'''$  respectively at any iteration  $k$ .

$$\begin{aligned} F(f^{(k)}, f'^{(k)}, f''^{(k)}, f'''^{(k)}) &+ (f - f^{(k)}) \left. \frac{\partial F}{\partial f} \right|^{(k)} \\ &+ (f' - f'^{(k)}) \left. \frac{\partial F}{\partial f'} \right|^{(k)} + (f'' - f''^{(k)}) \left. \frac{\partial F}{\partial f''} \right|^{(k)} \\ &+ (f''' - f'''^{(k)}) \left. \frac{\partial F}{\partial f'''} \right|^{(k)} + \dots \end{aligned}$$

Neglecting the square and higher order terms, we get the modified form

$$\begin{aligned} (f - f^{(k)}) \left. \frac{\partial F}{\partial f} \right|^{(k)} + (f' - f'^{(k)}) \left. \frac{\partial F}{\partial f'} \right|^{(k)} + (f'' - f''^{(k)}) \left. \frac{\partial F}{\partial f''} \right|^{(k)} \\ + (f''' - f'''^{(k)}) \left. \frac{\partial F}{\partial f'''} \right|^{(k)} = -F(f^{(k)}, f'^{(k)}, f''^{(k)}, f'''^{(k)}) \end{aligned}$$

$$In \quad F = f''' + ff'' + \beta(1 - f'^2)$$

$$\frac{\partial F}{\partial f} = f''^{(k)}$$

$$\frac{\partial F}{\partial f'} = -2\beta f'^{(k)}$$

$$\frac{\partial F}{\partial f''} = f^{(k)}$$

$$\frac{\partial F}{\partial f'''} = 1$$

$$(f - f^{(k)}) (f''^{(k)}) + (f' - f'^{(k)}) (-2\beta f'^{(k)})$$

$$+ (f'' - f''^{(k)}) (f^{(k)}) + (f''' - f'''^{(k)})$$

$$= -F(f^{(k)}, f'^{(k)}, f''^{(k)}, f'''^{(k)})$$

$$\Rightarrow f'''^{(k+1)} + f^{(k)} f''^{(k+1)} + f''^{(k)} f^{(k+1)} - 2\beta f'^{(k)} f^{(k+1)} \\ = f^{(k)} f''^{(k)} - \beta (f'^{(k)})^2 - \beta$$

... REDUCED LINEAR BVP.

Poly Writing

Now use central Block tri-diagonal system.

$$\text{Let } g^{(k+1)} = f^{(k+1)} \dots (i)$$

$\therefore$  The equation becomes

$$g''^{(k+1)} + f^{(k)} g'^{(k+1)} - 2\beta g^{(k)} g^{(k+1)} + g'^{(k)} f^{(k+1)} \\ = f^{(k)} g'^{(k)} - \beta(g^{(k)})^2 - \beta \dots (ii)$$

where  $f^{(k+1)}(0) = g^{(k+1)}(0) = 0, g^{(k+1)}(10) = 1$

Eq. (i) using trapezoidal rule & simplifying gives us:

$$-\frac{h}{2}(g_i^{(k+1)} + g_{i-1}^{(k+1)}) + (f_i^{(k+1)} - f_{i-1}^{(k+1)}) = 0 \dots (i)a$$

Eq. (ii) Using central difference method , gives us:

$$\frac{1}{h^2}(g_{i+1}^{(k+1)} - 2g_i^{(k+1)} + g_{i-1}^{(k+1)}) + \frac{f_i^{(k)}}{2h}(g_{i+1}^{(k+1)} - g_{i-1}^{(k+1)})$$

$$-2\beta g_i^{(k)} g_i^{(k+1)} + g_i^{(k)} f_i^{(k+1)} = f_i^{(k)} g_i^{(k)} - \beta(g_i^{(k)})^2 - \beta$$

$$\Rightarrow g_{i-1}^{(k+1)} \left[ \frac{1}{h^2} - \frac{f_i^{(k)}}{2h} \right] + g_i^{(k+1)} \left[ -\frac{2}{h^2} - 2\beta g_i^{(k)} \right] \\ + g_{i+1}^{(k+1)} \left[ \frac{1}{h^2} + \frac{f_i^{(k)}}{2h} \right] + g_i^{(k)} f_i^{(k+1)}$$

$$A_i = \begin{pmatrix} -1 & -h/2 \\ 0 & \frac{1}{h^2} - \frac{f_i^{(k)}}{2h} \end{pmatrix}$$

$$C_i = \begin{pmatrix} 0 & 0 \\ 0 & \frac{1}{h^2} + \frac{f_i^{(k)}}{2h} \end{pmatrix}$$

$$B_i = \begin{pmatrix} 1 & -h/2 \\ g_i^{(k)} & -\frac{2}{h^2} - 2\beta g_i^{(k)} \end{pmatrix}$$

$$D_i = \begin{pmatrix} 0 \\ \vdots \end{pmatrix}$$

## NEW TOPIC:

### Spline Interpolation

Drawbacks of current system:

- Require problem to form a tri-diagonal system

Pros:

- 2nd order accuracy

↳ But maybe we can do better, & that too w/o much hassle.

"compact difference scheme" → HIGHER ORDERS

↳  $f''$  & its derivative are  
both considered to be unknown,  
→ Hermite interpolation technique.

$x$	$y$
$x_0$	$y_0$
$x_1$	$y_1$
:	:
$x_n$	$y_n$

$$P_n(x) = y \quad \text{s.t.} \quad y_i = P_n(x_i)$$

$$i = 0, 1, \dots, n .$$

$x_i$  → node/grid points  
usually associated  
w/ (P) Diff. eqs.

- Consider piecewise-interpol.

↳ instead of taking whole interval  $(x_0, x_n)$ ,  
we take  $P^{(k)}(x)$  as a polynomial of  
any degree (not necessarily  $k$ ), which  
interpolates  $y(x)$  in  $[x_k, x_{k+1}]$

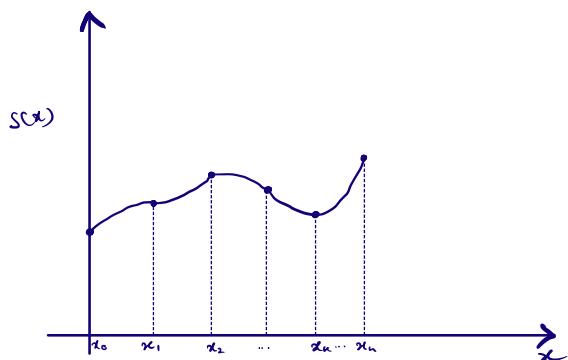
$$\text{for } k=0, 1, \dots, n-1 .$$

→  $S(x)$  is a piecewise interpolation polynomial.

$$\Rightarrow S(x) = \{ p^{(k)}(x) \mid x_k \leq x \leq x_{k+1}, k=0,1,\dots,n-1 \}$$

Recall:  $p^{(k)}(x)$  is a cubic spline in  $[x_k, x_{k+1}]$  if  $p^{(k)}(x)$  is a cubic polynomial and has the same slope and curvature at each knot point  $x_1, x_2, \dots, x_{n-1}$  where  $p^{(k)}(x)$  are joined.

- ( knot point  $x_1, x_2, \dots, x_{n-1}$  where  $p^{(k)}(x)$  are joined )



- $p^{(k)}(x)$  is a cubic polynomial

- $p^{(k)}(x_k) = y_k, p^{(k)}(x_{k+1}) = y_{k+1} \quad \forall k \in \{0, 1, \dots, n\}$

Now, how do we impose the co

- $p_k(x)$  in  $[x_k, x_{k+1}]$

- $p_{k-1}(x)$  in  $[x_{k-1}, x_k]$

$$S(x) = \{ p_k(x) \mid x_k \leq x \leq x_{k+1}, k=0,1,\dots,n-1 \}$$

$$p^{(k)}(x_k) = p^{(k-1)}(x_k) = y_k$$

$$p^{(k)}'(x_k) = p^{(k-1)'}(x_k)$$

$$p^{(k)''}(x_k) = p^{(k-1)''}(x_k)$$

Now,  $s(x)$  is piecewise cubic in  $[x_0, x_n]$ , so

$s''(x)$  is piecewise linear in  $[x_0, x_n]$ .

$$s''(x) = p^{(k)}(x) \text{ in } [x_k, x_{k+1}]$$

is a linear polynomial.

$$\text{Let } M_k = s''(x_k), M_{k+1} = s''(x_{k+1})$$

Then

$$p^{(k)}(x) = M_k \frac{(x - x_{k+1})}{(x_k - x_{k+1})} + M_{k+1} \frac{(x - x_k)}{(x_{k+1} - x_k)}$$

Note:  $M_k, M_{k+1}$  are given.

let  $h = x_{k+1} - x_k$  be constant for equi-spaced points  $x_0, x_1, \dots, x_n$ .

$$\text{so } p^{(k)}(x) = \frac{M_k}{h} (x_{k+1} - x) + \frac{M_{k+1}}{h} (x - x_k)$$

$$\Rightarrow p^{(k)}(x) = \frac{M_k}{h} \frac{(x_{k+1} - x)^3}{6} + \frac{M_{k+1}}{h} \frac{(x - x_k)^3}{6} \\ C_k (x - x_k) + D_k (x_{k+1} - x)$$

$$C_k = \frac{1}{h} \left( y_{k+1} - M_{k+1} \frac{h^2}{6} \right)$$

$$D_k = \frac{1}{h} \left( y_k - M_k \frac{h^2}{6} \right)$$

$$\begin{aligned}
 p_k(x) = & \frac{M_k}{6} \left[ \frac{(x_{k+1}-x)^3}{h} - h(x_{k+1}-x) \right] \\
 & + \frac{y_k}{h} (x_{k+1}-x) + \frac{M_{k+1}}{6} \left[ \frac{(x-x_k)^3}{h} - h(x-x_k) \right] \\
 & + \frac{y_{k+1}}{h} (x-x_k) \quad , \quad k=0, 1, \dots, n-1
 \end{aligned}$$

We have to determine  $M_k$ 's, i.e.  $M_0, M_1, \dots, M_n \rightarrow (n+1)$   
Unknowns

We impose the condition

$$p'_k(x_k) = p'_{k-1}(x_k) \text{ at } k=1, 2, \dots, n-1.$$

$$\begin{aligned}
 p'_k(x) = & \frac{M_k}{6} \left[ \frac{-3(x_{k+1}-x)^2}{h} + h \right] - \frac{y_k}{h} \\
 & + \frac{M_{k+1}}{6} \left[ \frac{3(x-x_k)^2}{h} - h \right] + \frac{y_{k+1}}{h} \\
 \Rightarrow p'_k(x_k) = & \frac{M_k}{6} [-2h] + \frac{M_{k+1}}{6} [-h] + \frac{\Delta y_k}{h} \\
 = & -\frac{h}{3} M_k - \frac{h}{6} M_{k+1} + \frac{\Delta y_{k+1}}{h}
 \end{aligned}$$

8 Solve for

$p'_k(x_{k+1})$ , derive the eqn to get it in terms of  
 $M_k$  &  $\Delta y_k$  (derive again, from start).  
You will obtain a tri-diagonal system.