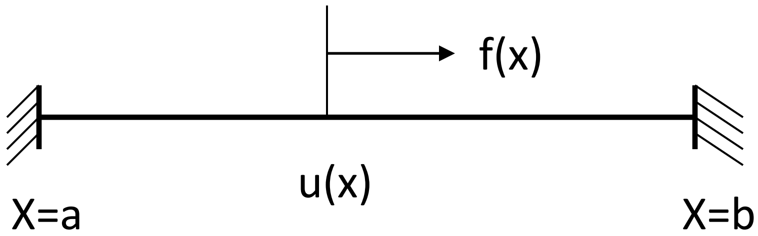
Finite Difference Methods in MATLAB

Padmanabhan Seshaiyer

Sept 5, 2013

PEER Program

Displacement of a Linear Elastic Bar



u(x): Displacement
f(x): Tangential Load
Bar Fixed at both ends

$$\sigma \ \alpha \ \frac{du}{dx} \quad \Rightarrow \quad \sigma = K \frac{du}{dx}$$

$$\frac{d\sigma}{dx} = -f(x)$$

$$-\frac{d}{dx}\left(K \frac{du}{dx}\right) = f(x)$$

$$u(a) = 0$$

$$u(b) = 0$$

Difference Formulas

$$-\frac{d^2u}{dx^2} = f(x)$$

$$u(a) = ua$$

$$u(b) = ub$$

$$\frac{du}{dx} \approx \frac{u_{i+1} - u_i}{h}$$

$$\frac{d^2u}{dx^2} \approx \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2}$$

Difference Formulas

$$i = 2 \qquad -\left(u_3 - 2u_2 + u_1\right) = h^2 f(x_2)$$

$$i = 3 \qquad -\left(u_4 - 2u_3 + u_2\right) = h^2 f(x_3)$$

.....

$$i = N-1 \qquad -\left(u_N - 2u_{N-1} + u_{N-2}\right) = h^2 f(x_{N-1})$$

$$-\left(u_3 - 2u_2 + u_1\right) = h^2 f(x_2)$$

$$-\left(u_4 - 2u_3 + u_2\right) = h^2 f(x_3)$$

.....

$$-\left(u_N - 2u_{N-1} + u_{N-2}\right) = h^2 f(x_{N-1})$$

Difference Formulas

$$u_3 - 2u_2 \qquad = -h^2 f(x_2) - u_1$$

$$u_4 - 2u_3 + u_2 \qquad = -h^2 f(x_3)$$

$$.....$$

$$-2u_{N-1} + u_{N-2} \qquad = -h^2 f(x_{N-1}) - u_N$$

$$\begin{bmatrix} -2 & 1 & 0 & ..... & 0 \\ 1 & -2 & 1 & ...... & 0 \\ .... & & & & \\ .... & & 1 & -2 & 1 \\ 0 & 0 .... & & 1 & -2 \end{bmatrix} \begin{bmatrix} u_2 \\ u_3 \\ . \\ . \\ u_{N-1} \end{bmatrix} = \begin{bmatrix} -h^2 f(x_2) - u_1 \\ -h^2 f(x_3) \\ ..... \\ .... \\ -h^2 f(x_{N-1}) - u_N \end{bmatrix}$$

Solve the BVP

$$\frac{d^2u}{dx^2} = \left(6 + 4x^2\right) x\, e^{x^2}$$

$$u(0) = 0$$

$$u(1) = e$$

$$u(x) = x\ e^{x^2}$$

MATLAB program
Finite Difference Method

```
% myfd.m
% This is a finite difference code
% u_xx = (6 + 4x^2)*x*e^(x^2), u(0)=0, u(1)=e
% Input: a, b, N
% OUTPUT: Plot exact vs approximate

% Initializing Values
a=0
b=1;
N=4;
ua=0;
ub=exp(1);
h = (b - a)/N;                          % Mesh step size
```

MATLAB program
Finite Difference Method

```
% Setting up Finite Difference Discretization

x = a+h:h:b-h;                            % Creates the x-grid
v = h^2*(6 + 4*x.^2).*x.*exp(x.^2);       % Right Vector
v(1) = v(1) - ua;                         % Left Boundary condition
v(N-1) = v(N-1) - ub;                     % Right Boundary condition
```

MATLAB program
Finite Difference Method

```
%% Create the matrix A
A = diag(-2*ones(N-1,1),0) + diag(ones(N-2,1),1);
A = A + diag(ones(N-2,1),-1);
y = A\v';                       % Solving the system
yfd = [0 y' exp(1)]             % Finite Difference solution

%% Plotting the exact vs approximate
x=a:h:b;                        % Grid for plotting
plot(x,yfd,'go');               % Plotting Finite Difference solution
hold on
yexact = x.*exp(x.^2);          % Exact solution
plot(x,yexact,'r');              %Plotting Exact solution
```