Boundary Value Problems!

Finite Difference Method

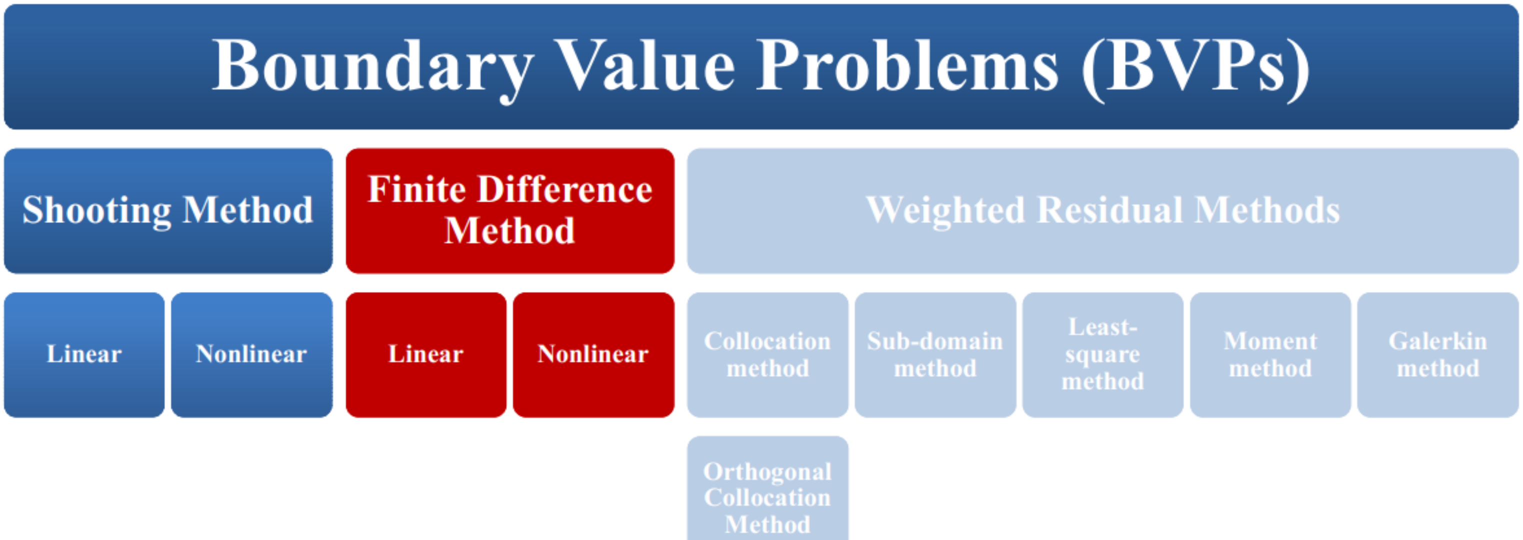
تمرین ۴. معادله‌ی دیفرانسیل توزیع دما در یک پره به صورت تابعی از طول پره، در شرایط پایا، به‌دست آمده است. معادله‌ی دیفرانسیل و شرایط مرزی پس از بی‌بعدسازی و مرتب کردن آن، به‌صورت زیر حاصل شده است:

$$\frac{d^2\theta}{dz^2} - \varphi^2\theta^{\frac{4}{3}} = 0, \quad \varphi^2 = m^2 w^2 (T_s - T_a)^{\frac{1}{3}}$$

$$\begin{cases} z = 0 \ : \ \theta = 1 \\ z = 1 \ : \ \dfrac{d\theta}{dz} = -\psi^2\theta^{\frac{4}{3}}, \quad \psi^2 = \dfrac{w\alpha}{k}(T_s - T_a)^{\frac{1}{3}} \end{cases}$$

الف- این معادله را با برنامه‌نویسی در محیط نرم‌افزار MATLAB به روش پرتابی حل کنید؛

ب- با کمک روش اختلاف‌های محدود در محیط MATLAB، مسئله را حل کرده و نتایج به دست آمده را با نتایج حاصل از بند الف مقایسه کنید.



- ۱ – انتخاب x_i , step size:
- $h=\dfrac{b-a}{n}$ و $x_i = a+ih$
- ۲ – نوشتن مشتقات اول و دوم و ... از طریق تفاضل‌های محدود (به شکل مشتقات مرکزی)
- مشتق دوم:

$$y_i''=\frac{1}{h^2}(y_{i+1}-2y_i+y_{i-1})+o(h^2)$$

- مشتق اول:

$$y_i'=\frac{1}{2h}(y_{i+1}-y_{i-1})+o(h^2)$$

الگوریتم:

- ۱ – حدس y_i^j (j=1 مرحله اول حدس) برای تمام نقاط *i=1 to n-1*
- ۲ – جایگزینی y_i^j در تابع g_i و $\dfrac{\partial g_i}{\partial y_k}$ برای $\begin{cases} k=1\ to\ n-1 \\ i=1\ to\ n-1 \end{cases}$
- ۳ – حل دستگاه (n-1) معادله و (n-1) مجهول به منظور تعیین Δy_i^j :
- شکل ماتریس دستگاه:

$$\begin{bmatrix} \frac{\partial g_1}{\partial y_1} & \frac{\partial g_1}{\partial y_2} & 0 & 0 & 0 & 0 \\ \frac{\partial g_2}{\partial y_1} & \frac{\partial g_2}{\partial y_2} & \frac{\partial g_2}{\partial y_3} & 0 & 0 & 0 \\ 0 & \frac{\partial g_3}{\partial y_2} & \frac{\partial g_3}{\partial y_3} & \frac{\partial g_3}{\partial y_4} & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \frac{\partial g_{n-1}}{\partial y_{n-2}} & \frac{\partial g_{n-1}}{\partial y_{n-1}} \end{bmatrix} \begin{bmatrix} \Delta y_1 \\ \Delta y_2 \\ \Delta y_3 \\ \vdots \\ \Delta y_{n-1} \end{bmatrix} = - \begin{bmatrix} g_1 \\ g_2 \\ g_3 \\ \vdots \\ g_{n-1} \end{bmatrix}$$

- ۴ – محاسبه y_i^{j+1} با $y_i^{j+1}=y_i^j+\Delta y_i^j$
- ۵ – اگر $g_i(y^{j+1})=0$ باشد جواب بدست آمده است در غیر این صورت y_i^{j+1} را به عنوان حدس بعدی در نظر گرفت و به مرحله ۲ بازگشت .
- همچنین می‌توان با روش relaxation نیز حدس جدید را به صورت محتاطانه‌تری اصلاح کرد. $y_i^{j+1}=y_i^j+r\Delta y_i^j$

ODE :

$$\frac{1}{h^2}(y_{i+1} - 2 * y_i + y_{i-1}) = \mathrm{M1} * y_i^{\frac{4}{3}};$$

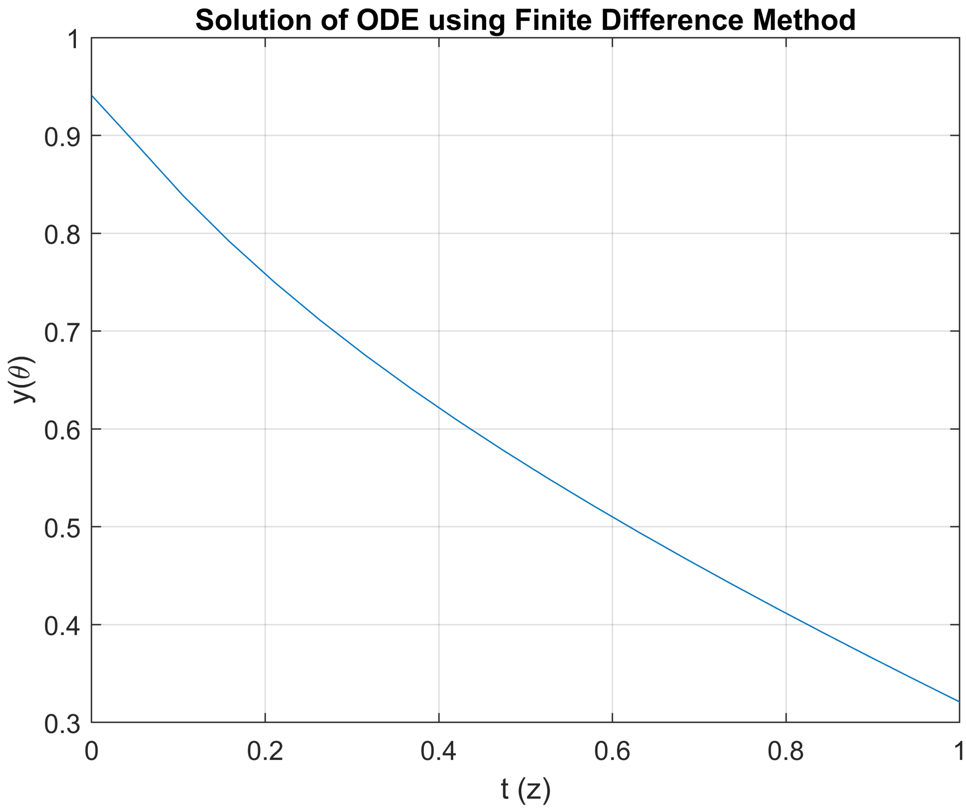
at the first Point : $y_0 = 1;$

at the final point : $y_{i+1} - y_i = h * \left(-\mathrm{M2} * y_i^{\frac{4}{3}}\right)$

```
clear; clc; close all;

% initilizations:
m = 1;   w = 2; Ts  =210; Ta = 180; alpha = 0.2; k = 1;

h = 0.05; % Step Size
b=1; a=0.01;
interval=b-a;
num_of_lines = round(interval/h);


global MULTING  MULTING2

MULTING = m^2 * w^2 * (Ts - Ta)^(1/3);
MULTING2 = w*alpha/k*(Ts - Ta)^(1/3) ;

Y  = zeros(num_of_lines,1);
A = zeros(num_of_lines,num_of_lines); % Coefficient Matrix
g = Y;

y0 = 1;

iter_max = 10;

threshold = 1e-3;




for iter =1 :iter_max
    % Obtain Coefficient Matrix:
    for i=1:length(Y)
        for j=1:length(Y)
            if(i==j)
            A(i,j) = h^2 * MULTING*4/3*Y(i)+2;
            elseif(abs(j-i)==1)
            A(i,j) =-1;
            end
```

```
        end
        if((i>2) && (i<length(Y)))
            g(i,1)  = h^2*MULTING*Y(i,1).^4/3 - Y(i+1,1) + 2*Y(i,1) - Y(i-1,1) ;
        elseif(i==1)
            g(i,1)  = h^2*MULTING*Y(i,1).^4/3 - Y(i+1,1) + 2*Y(i,1) - y0 ;
        elseif(i==length(Y))
            g(i,1)  = h^2*MULTING*Y(i,1).^4/3 - (-2*h*MULTING2*Y(i,1).^4/3 + Y(i,1)) + 2*Y(i,1
        end
    end

    delta_Y =  - g'*pinv(A);
    Y = Y + delta_Y';
    % CHeck Convergence:
    if(norm(g)<threshold)
        return
    end

end
```

```
t = linspace(0,1,length(Y(:,1)));
figure(1)
plot(t,Y(:,1));
grid on
ylabel("y(\theta)")
xlabel("t (z)")
title("Solution of ODE using Finite Difference Method")
```



Solution of ODE using Finite Difference Method
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
y(θ)
0
0.2
0.4
0.6
0.8
1
t (z)