Madhuri Sarda
18CH30030

ASSIGNMENT-4

PROBLEM STATEMENT-1

Consider the following BVP representing a dimensionless form of the diffusion with chemical reaction in a catalyst pore.

$$\frac{d^2C}{dx^2} = 4c$$

At x = 0 (the mouth of the pore), the dimensionless concentration, c = 1.

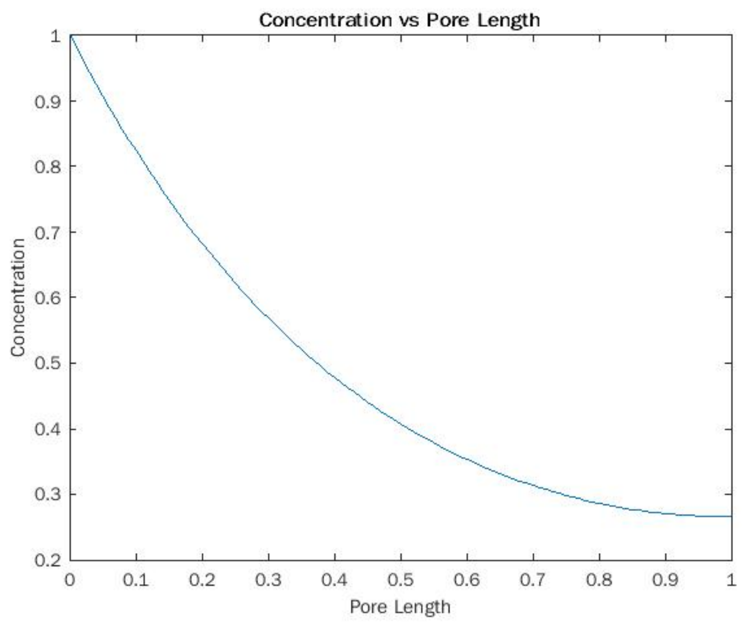
At x = 1(the pore end), the gradient of the concentration, dc/dx = 0.

Solve the BVP using (a) Shooting Method (your code) and (b) MATLAB function bvp4c. Plot the concentration profile along pore length x and compare the results.

SHOOTING METHOD (MATLAB Code)

```
%SHOOTING METHOD
f= @(x,y1,y2) [ y2];      % c=y1, dc/dx = dy1/dx = y2
g = @(x,y1,y2)[4*y1];     % dy2/dx = 4y1
y1(1)=1;                  % x=0, c=1, y1=1
y2(1)=-1.925;             % guess at x=0
h=0.01;      % step height
x=[0:h:1];   % length of pore
for i=1:1/h
    y1(i+1)= y1(i) + h*f(x(i),y1(i),y2(i)); % EULER METHOD TO SOLVE THE SYSTEM OF
EQUATIONS
    y2(i+1)= y2(i) + h*g(x(i),y1(i),y2(i));

end
plot(x,y1);
```

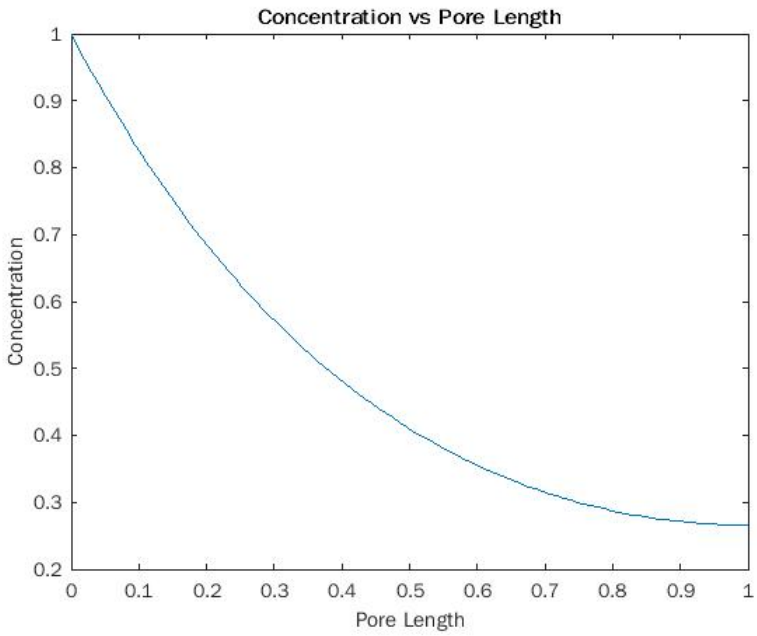
Plot



MATLAB FUNCTION bvp4c

```
f= @(x,y) [ y(2) ; 4*y(1)]; % define function
                            % c=y1, dc/dx = dy1/dx = y2
                            % dy2/dx = 4y1

bc = @(ya, yb) [ya(1)-1; yb(2)]; % boundary condition
                                 % x=0, c=1; x=1, dc/dx=0

xmesh = linspace(0 , 1 , 100);    % steps

solinit = bvpinit(xmesh, [0 1]);

sol=bvp4c(f, bc,solinit);

c = sol.y;

plot( sol.x, c(1,:));
```

Plot



CONCLUSION

PROBLEM STATEMENT-2

Consider the following BVP representing steady state Heat Transfer in a rod of length L = 10 m.

$$\frac{d^2T}{dx^2} + h'(T_\infty - T) + \sigma\left(T_\infty{}^4 - T^4\right) = 0$$

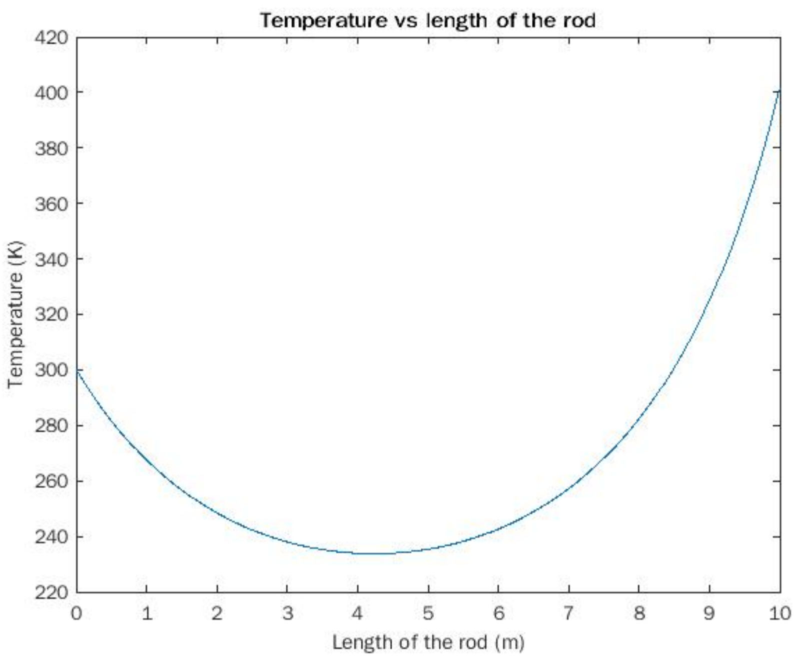
Given: h' = 0.05 m-2, σ = 2.7 × 10-9 K-3 m-2, T_∞= 200 K, T(0) = 300 K, T(10) = 400 K.

Solve the BVP using (a) Shooting Method (your code) and (b) MATLAB function bvp4c. Plot the temperature distribution along the length of the rod and compare the results.

SHOOTING METHOD (MATLAB code)

```
h= 0.05;
T0=300;
TL=400;
Tinf=200;
sigma= 2.7*10^-9;
f= @(x,y1,y2) [ y2];  % T=y1, dy1/dx = y2
g = @(x,y1,y2)[(-h*(Tinf-y1) - sigma*(Tinf^4 - (y1)^4))]; %dy2/dx=(-h*(Tinf-y1)-sigma*(Tinf^4-y1^4))
y1(1)=300;     % x=0, T(0)=300
y2(1)=-41.72;   % guess at x=0
h=0.01;        % step height
x=[0:h:10];  % Length of the rod
for i=1:10/h
    y1(i+1)= y1(i) + h*f(x(i),y1(i),y2(i));   % EULER METHOD FOR SOLVING SYSTEM OF EQUATIONS
     y2(i+1)= y2(i) + h*g(x(i),y1(i),y2(i));

end
plot(x,y1);
```

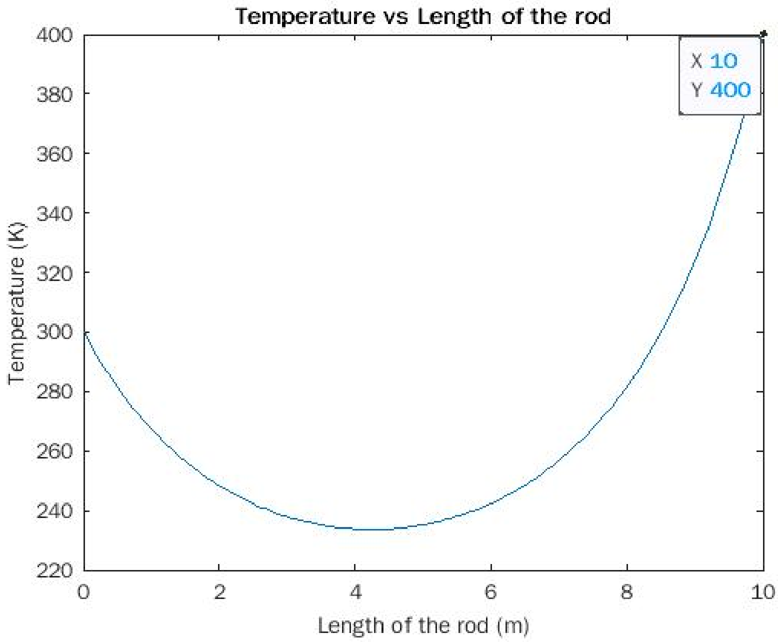
Plot



MATLAB FUNCTION bvp4c

```
h= 0.05;
T0=300;
TL=400;
Tinf=200;
sigma= 2.7*10^-9;
f= @(x,y) [ y(2) ; (-h*(Tinf-y(1)) - sigma*(Tinf^4 - (y(1))^4))]; % T=y1, dy1/dx = y2
                                                             %dy2/dx=(-h*(Tinf-y1)-sigma*(Tinf^4-y1^4))

bc = @(ya, yb) [ya(1)-T0; yb(1) - TL];   % boundary condition
                                          % x=0, T=T0; x=10, T=TL

xmesh = linspace(0 , 10 , 100);    % mesh

solinit = bvpinit(xmesh, [0 10]);

sol=bvp4c(f, bc,solinit);

T = sol.y;

plot( sol.x, T(1,:));
```

Plot



CONCLUSION