Table of Contents


```
function Assignment_4()
```

```
% Tianyu Gao
clc
clear all
```

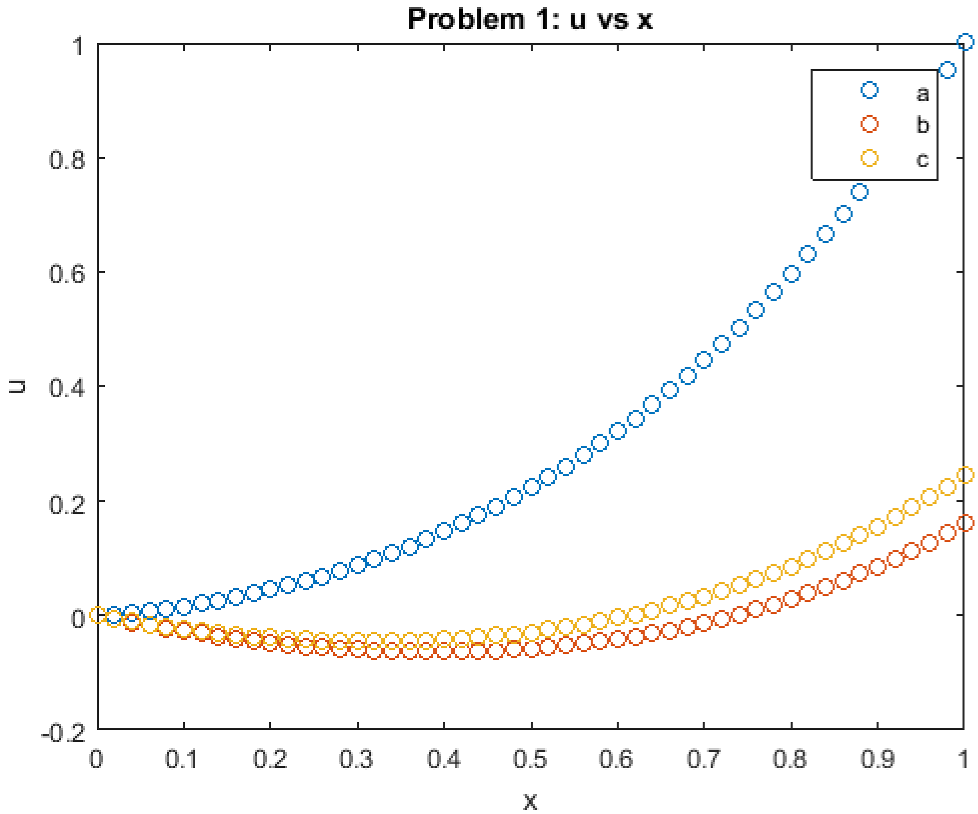
Problem 1 BVP: Finite Difference

$u'' - u = 1$ compared to $y'' = p(x)y' + q(x) + r(x)$

p = 0, q = 1, r = 1 h = 1/N, d = 2 + h^2, u = -1 + h/2, l = -1 - h/2.

```
% a) u(0) = 0, u(1) = 1:
N = 50;
h = 1/N;
d = 2 + h^2;
u = -1 + h/2;
l = -1 - h/2;
r = 1;
% Assign vector B
b = -h^2*ones(N+1,1);
% Dirchlet Boundary Condition
b(1) = 0; %u(0)
b(end) = 1; % u(1)
% Assign matrix A
A = diag(d*ones(1,N+1)) + diag(u*ones(1,N),1) + diag(l*ones(1,N),-1);
A(1,1) = 1;
A(N+1,N+1) = 1;
A(1,2) = 0;
A(N+1,N) = 0;
w1 = A\b;
% Assign xspan for plot
xspan = linspace(0,1,N+1);

% b) u(0) = 0, u(1) + u'(1) = 1:
% Robin Boundary Condition
beta1 = 1;
beta2 = 1;
```

```
beta3 = 1;
A(N+1,N+1) = d-2*h*u*beta1/beta2;
A(N+1,N) = -2;
b(N+1) = -h^2*r-2*h*u*beta3/beta2;

w2 = A\b;

% c) u(0) = 0, u'(1) = 1:
% Neumann Boundary Condtion
beta = 1;
A(N+1,N+1) = d;
A(N+1,N) = -2;
b(N+1) = -h^2*r - 2*h*u*beta;
w3 = A\b;
figure;
plot(xspan, w1,'o',xspan,w2,'o',xspan,w3,'o');
legend('a','b','c');
xlabel('x');
ylabel('u');
title('Problem 1: u vs x');
```



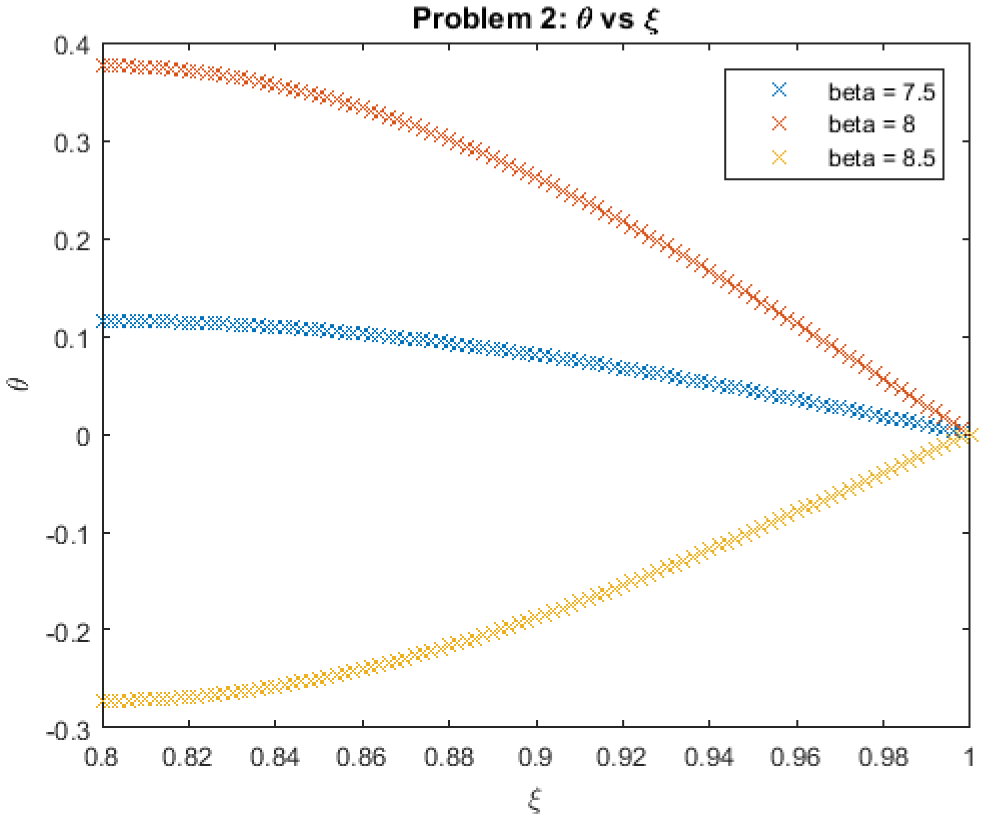
Problem 2: BVP, Finite Difference

$$\frac{d^2\theta}{d\xi^2} + \frac{1}{\xi}\frac{d\theta}{d\xi} + \beta^2\theta = -1$$

$$\frac{d^2\theta}{d\xi^2} = -\frac{1}{\xi}\frac{d\theta}{d\xi} - \beta^2\theta = -1$$

$$p(\xi) = -1/\xi \quad q(\xi) = -\beta^2 \quad r = -1$$

```
N = 100;
xspan = linspace(0.8,1,N+1);
w = zeros(N+1,3);
h = (1-0.8)/N;
beta =[7.5,8,8.5];
r = -1;
b = -h^2*r*ones(N+1,1);

for i = 1:3
    betai =beta(i);
    q = -betai^2;
    r = -1;
    d = 2 + h^2*q;
    A = diag(d*ones(N+1,1));
    for m = 1:N-1
        p = -1/xspan(m);
        u = -1 + h/2*p;
        l = -1 - h/2*p;
        A(m+1,m+2) = u;
        A(m+1,m) = l;
    end

    % Dirchlet Boundary Condition
    A(N+1,N+1) = 1;
    A(N+1,N) = 0;
    b(N+1) = 0;
    % Neumann Boundary Condition
    A(1,1) = d;
    A(1,2) = -2;
    b(1) = -h^2*r;
    w(:,i)=A\b;
end
plot(xspan,w,'x');
legend('beta = 7.5','beta = 8','beta = 8.5');
xlabel('\xi');
ylabel('$\theta$','interpreter','latex');
title('Problem 2: \theta vs \xi');
```



Problem 3 BVP

$$\frac{d^2C}{dr^2} + \frac{2}{r}\frac{dC}{dr} - \frac{k}{D}C = 0$$

define $\xi = r/R \quad \theta = C/C_R$

$$\frac{C_R}{R^2}\frac{d^2\theta}{d\xi^2} + \frac{2C_R}{\xi R^2}\frac{d\theta}{d\xi} - \frac{kC_R}{D}\theta = 0$$

$$\frac{d^2\theta}{d\xi^2} + \frac{2}{\xi}\frac{d\theta}{d\xi} - \frac{kR^2}{D}\theta = 0$$

$$\frac{d^2\theta}{d\xi^2} + \frac{2}{\xi}\frac{d\theta}{d\xi} - \phi^2\theta = 0$$

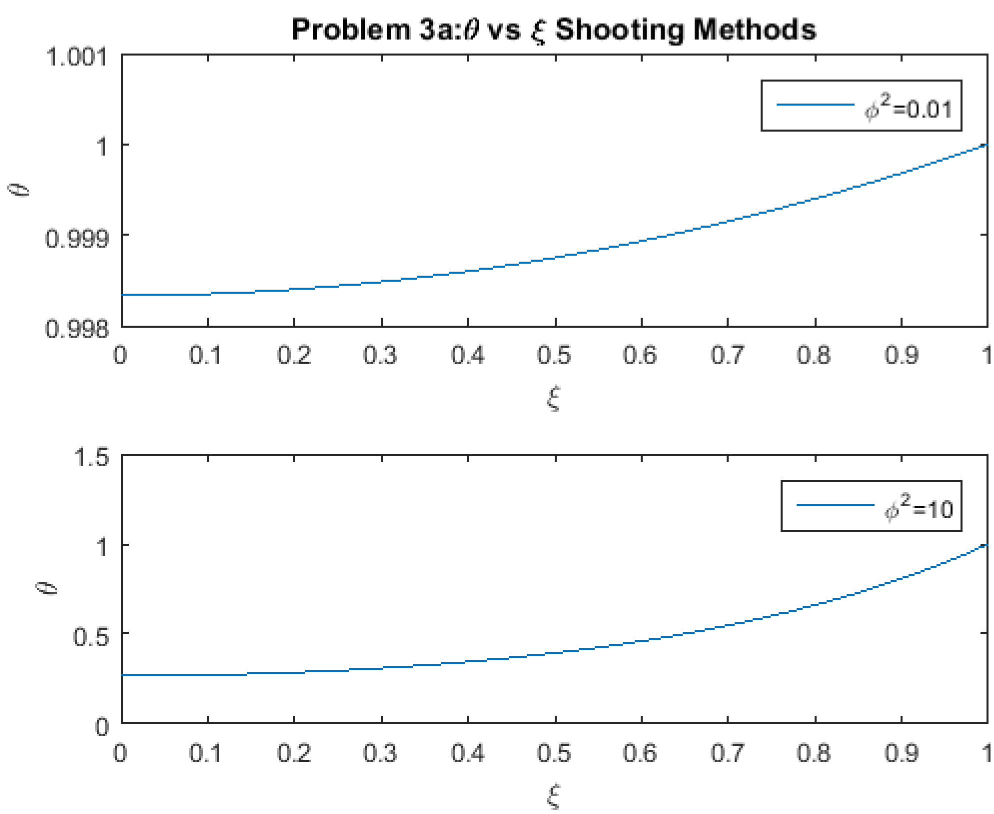
```
phi = [0.01,10];
tspan = [0,1];
% a) BVP Shooting Methods

% phi square equals to 0.01
guess1 = 0.90; % First shooting, goal: y(1) = 1
y0 = [guess1;0];
[x,y]=ode45(@(x,y) prob3(x,y,phi(1)),tspan,y0);
y1 = y(end,1);
guess2 = 1.0; % Second shooting is large because y1 is smaller than 1.
```

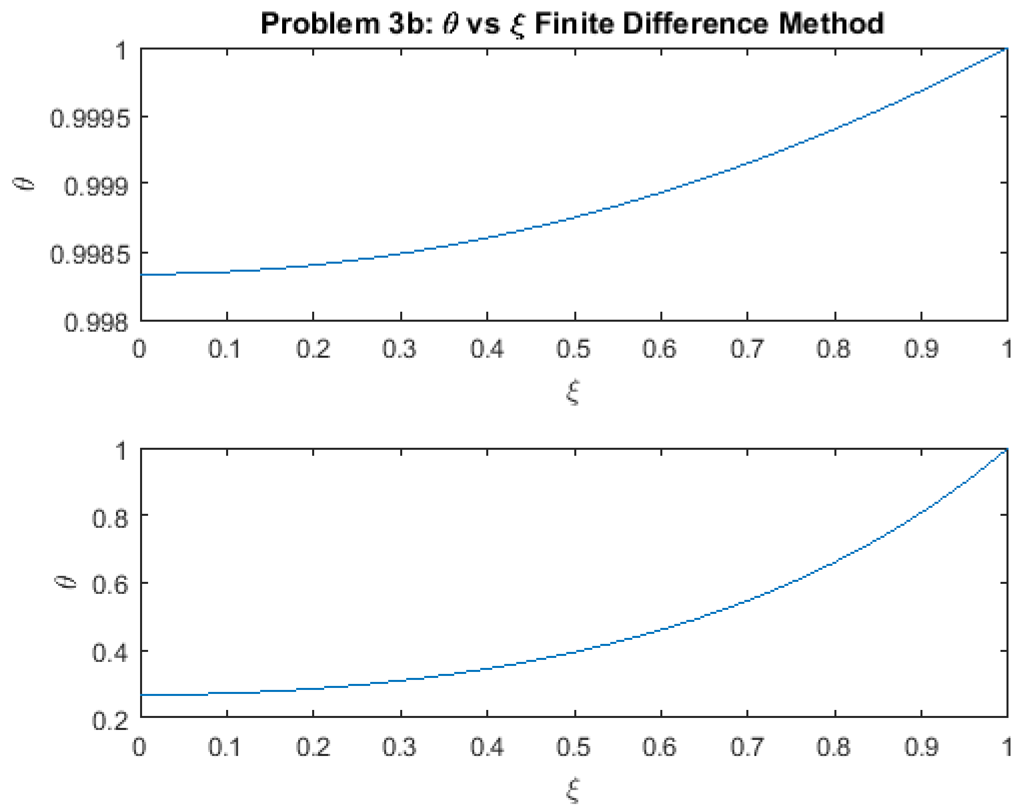
```
y0 = [guess2;0];
[x,y]=ode45(@(x,y) prob3(x,y,phi(1)),tspan,y0);
y2 = y(end,1);
t = fzero(@(t) t*y1+(1-t)*y2-1,0.5); % solve for the 'weighting'
ans1 = t*guess1 + (1-t)*guess2;
sprintf('Initial condition is theta(0) = %f when phi square equals to
 0.01',ans1)
y0 = [ans1, 0];
[xa,ya] = ode45(@(x,y) prob3(x,y,phi(1)),tspan,y0);
grad1 = ya(end,2);
figure
subplot(2,1,1);
plot(xa,ya(:,1))
legend('\phi^2=0.01');
xlabel('\xi');
ylabel('\theta');
title('Problem 3a:\theta vs \xi Shooting Methods');
% phi square equals to 10
guess1 = 0.20; % First shooting, goal: y(1) = 1
y0 = [guess1;0];
[x,y]=ode45(@(x,y) prob3(x,y,phi(2)),tspan,y0);
y1 = y(end,1);
guess2 = 0.3; % Second shooting is large because y1 is smaller than 1.
y0 = [guess2;0];
[x,y]=ode45(@(x,y) prob3(x,y,phi(2)),tspan,y0);
y2 = y(end,1);
t = fzero(@(t) t*y1+(1-t)*y2-1,0.5); % solve for the 'weighting'
ans2 = t*guess1 + (1-t)*guess2;
sprintf('Initial condition is theta(0) = %f when phi square equals to
 10',ans2)
y0 = [ans2, 0];
[xb,yb] = ode45(@(x,y) prob3(x,y,phi(2)),tspan,y0);
grad2 = yb(end,2);
subplot(2,1,2);
plot(xb,yb(:,1));
legend('\phi^2=10','intepreter','latex');
xlabel('\xi');
ylabel('\theta');

    function dydx = prob3(x,y,p)
        dydx = zeros(2,1);
        dydx(1) = y(2);
        if x == 0
            dydx(2) = p*y(1);
        else
            dydx(2) = -2/x*y(2)+p*y(1);
        end
    end
% b) BVP Finite Difference Method
N = 50;
h = 1/N;
xspan = linspace(0,1,N+1);
r = 0;
A = zeros(N+1,N+1);
```

```
b = zeros(N+1,1);
% phi square = 0.01
q = 0.01;
d = 2+h^2*q;
% Assign A,b:
for i = 2:N
    x = xspan(i);
    p = -2/x;
    u = -1+h/2*p;
    l = -1-h/2*p;
    A(i,i) = d;
    A(i,i+1) = u;
    A(i,i-1) = l;
end
% Neumann B.C.
A(1,1) = d;
A(1,2) = -2;
b(1) = 0;
% Dirichlet B.C.
A(N+1,N+1) = 1;
A(N+1,N) = 0;
b(N+1) = 1;
w = A\b;
figure
subplot(2,1,1)
plot(xspan,w)
xlabel('\xi');
ylabel('\theta');
title('Problem 3b: \theta vs \xi Finite Difference Method');
% phi square = 10
A = zeros(N+1,N+1);
b = zeros(N+1,1);
q = 10;
d = 2+h^2*q;
% Assign A,b:
for i = 2:N
    x = xspan(i);
    p = -2/x;
    u = -1+h/2*p;
    l = -1-h/2*p;
    A(i,i) = d;
    A(i,i+1) = u;
    A(i,i-1) = l;
end
% Neumann B.C.
A(1,1) = d;
A(1,2) = -2;
b(1) = 0;
% Dirichlet B.C.
A(N+1,N+1) = 1;
A(N+1,N) = 0;
b(N+1) = 1;
w = A\b;
subplot(2,1,2);
```

```
plot(xspan,w);
xlabel('\xi');
ylabel('\theta');

% c)
sprintf('The gradiend of concentration at the outer edge are %f, %f
 respectively',grad1,grad2)
```

```
ans =

Initial condition is theta(0) = 0.998335 when phi square equals to
 0.01


ans =

Initial condition is theta(0) = 0.268194 when phi square equals to 10

Warning: Ignoring extra legend entries.

ans =

The gradiend of concentration at the outer edge are 0.003331, 2.173630
 respectively
```





Problem 4 BVP Finite Difference

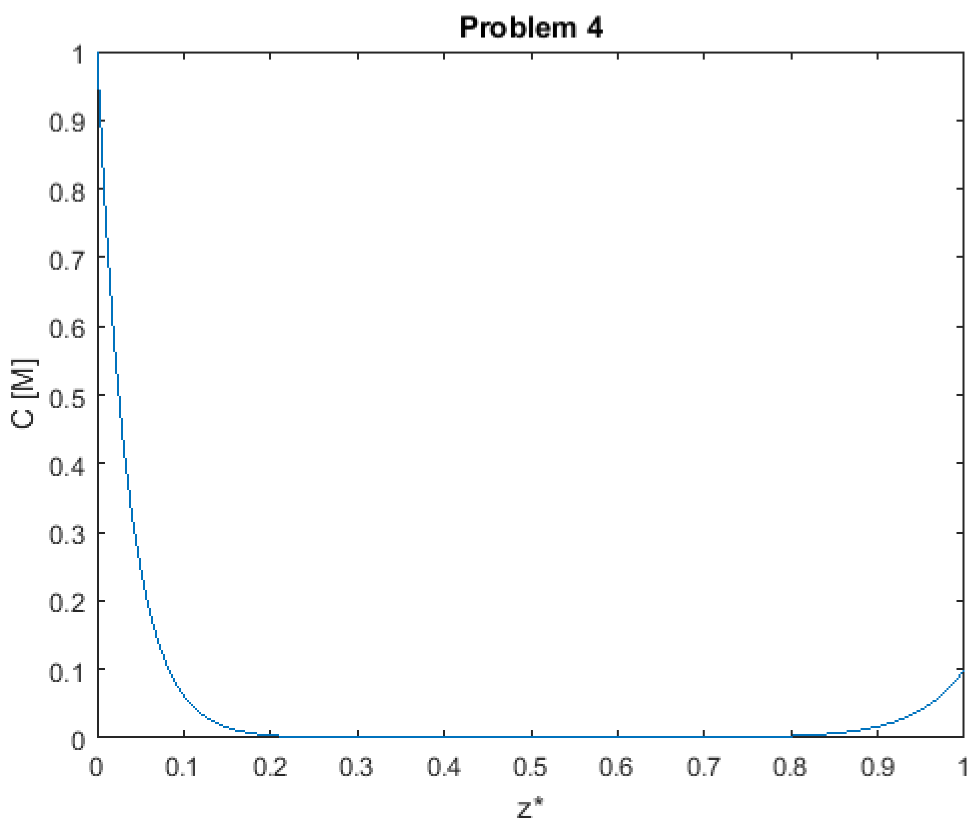
$$D\frac{d^2C}{dz^2} + v\frac{dC}{dz} - kC = 0$$

$$\frac{D}{L^2}\frac{d^2C}{dz*^2} + \frac{v}{L}\frac{dC}{dz*} - kC = 0$$

$$\frac{d^2C}{dz*^2} = -\frac{vL}{D}\frac{dC}{dz*} + \frac{kL^2}{D}C$$

```
D = 10; % um^2/sec
L = 1000; % um
v = 0.1; % um/sec
k = 5e-3; % 1/sec
p = -v*L/D;
q = k*L^2/D;
r = 0;
N = 100;
h = 1/N;

d = 2+h^2*q;
u = -1+h/2*p;
l = -1-h/2*p;
xspan = linspace(0,1,N+1);
```

```
A = zeros(N+1,N+1);
b = -h^2*r*zeros(N+1,1);
% Assign matrix A
for i = 2:N
    A(i,i) = d;
    A(i,i-1) = l;
    A(i,i+1) = u;
end
%Dirichlet BC
A(1,1) = 1;
A(1,2) = 0;
b(1) = 1;
A(N+1,N+1) = 1;
A(N+1,N) = 0;
b(N+1) = 0.1;
w = A\b;
figure;
plot(xspan,w);
xlabel('z*');
ylabel('C [M]');
title('Problem 4');
```



Problem 5: BVP, Non-linear Finite Differences Method

$$\frac{D}{L^2}\frac{d^2C}{dz*^2} + \frac{v}{L}\frac{dC}{dz*} - kC^2 = 0$$

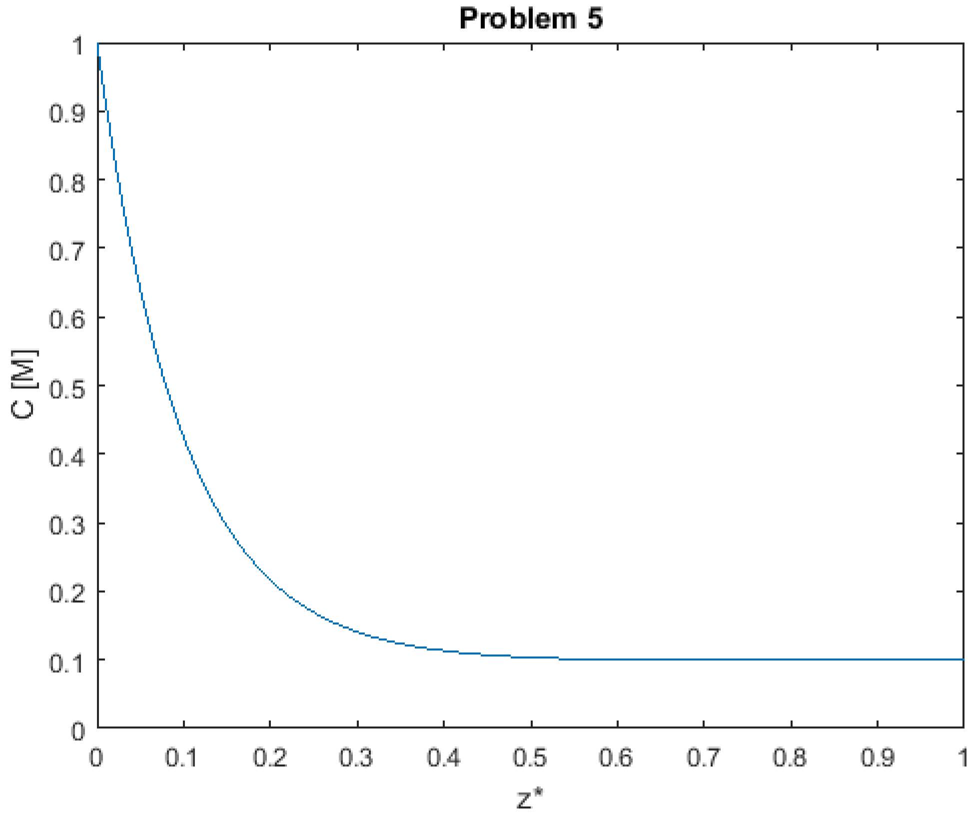
$$\frac{d^2C}{dz*^2} + \frac{vL}{D}\frac{dC}{dz*} - \frac{kL^2}{D}C^2 = 0$$

Discretize

$$\frac{\omega_{i+1} - 2\omega_i + \omega_{i-1}}{h^2} + \frac{vL}{D}\frac{\omega_{i+1} - \omega_{i-1}}{2h} - \frac{kL^2}{D}\omega_i^2 = 0$$

```
D = 10; % um^2/sec
L = 1000; % um
v = 0.1; % um/sec
k = 5e-5; % 1/sec
p = v*L/D;
q = -k*L^2/D;

N = 100;
h = 1/N;
guess = ones(N,1);
w = fsolve(@prob5,guess);
xspan = linspace(0,1,N);
plot(xspan,w);
xlabel('z*');
ylabel('C [M]');
title('Problem 5');
    function y=prob5(x)
        y = zeros(N,1);
        for i = 2:N-1
            y(i)= (x(i+1)-2*x(i)+x(i-1))/h^2+p*(x(i+1)-x(i-1))/2/h
+q*x(i)^2;
        end
        y(1) = x(1)-1; % B.C: C(0) = 1
        y(N) = x(N)-0.1; % B.C: C(1) = 0.1
    end
```

```
Equation solved.

fsolve completed because the vector of function values is near zero
as measured by the default value of the function tolerance, and
the problem appears regular as measured by the gradient.
```

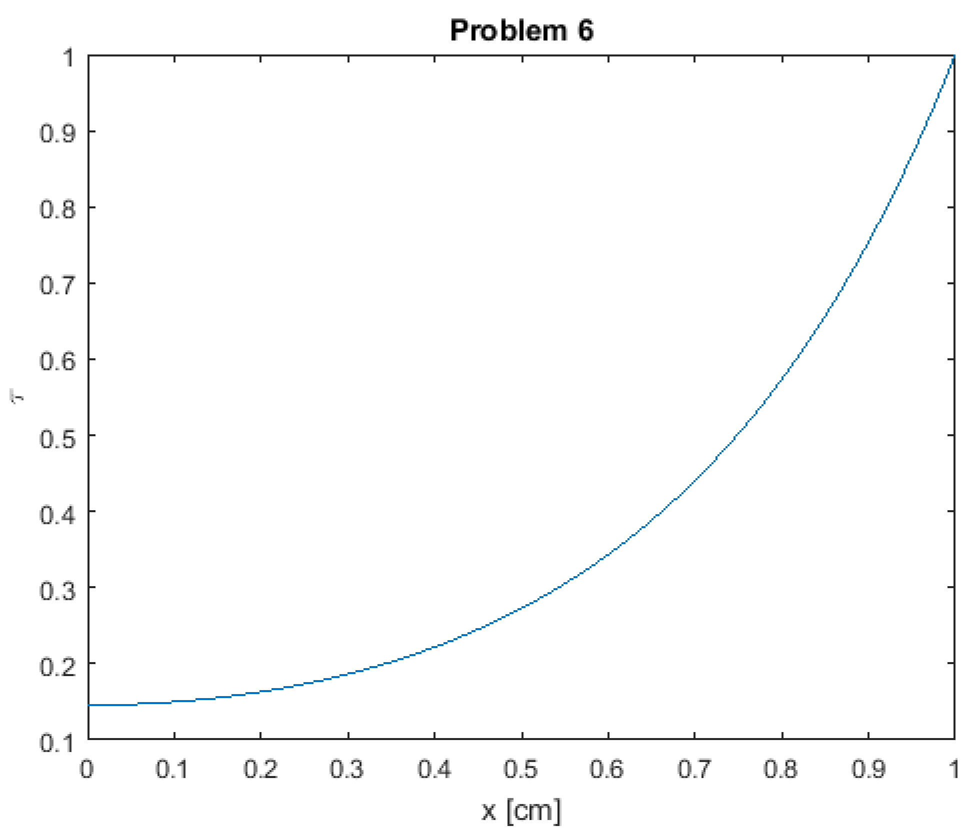


Problem 6 BVP, Finite Differences

$$\frac{d^2\tau}{dx^2} = -\frac{1}{x}\frac{d\tau}{dx} + \beta^2\tau$$

```
R = 1.3; % cm
h = 0.001; % cal/cm^2 s C
e = 0.36;
A = 15; % cm-1
k = 0.0034; % cal/cm s C
B = R^2*h*A/k/(1-e);

r = 0;
q = B;
h = 0.0025;
N = 1/h;
d = 2 + h^2*q;
xspan = linspace(0,1,N+1);
A = zeros(N+1,N+1);
b = -h^2*r*zeros(N+1,1);
% Assign matrix A
for i = 2:N
    x = xspan(i);
    p = -1/x;
    u = -1 + h/2*p;
    l = -1 - h/2*p;
```

```
    A(i,i) = d;
    A(i,i-1) = l;
    A(i,i+1) = u;
end
% Neumann B.C.
A(1,1) = d;
A(1,2) = -2;
b(1) = 0;
% Dirichlet B.C.
A(N+1,N+1) = 1;
A(N+1,N) = 0;
b(N+1) = 1;

w=A\b;
figure
plot(xspan,w);
xlabel('x [cm]');
ylabel('\tau');
title('Problem 6');
```



Problem 7: BVP, Non-linear, Finite differences

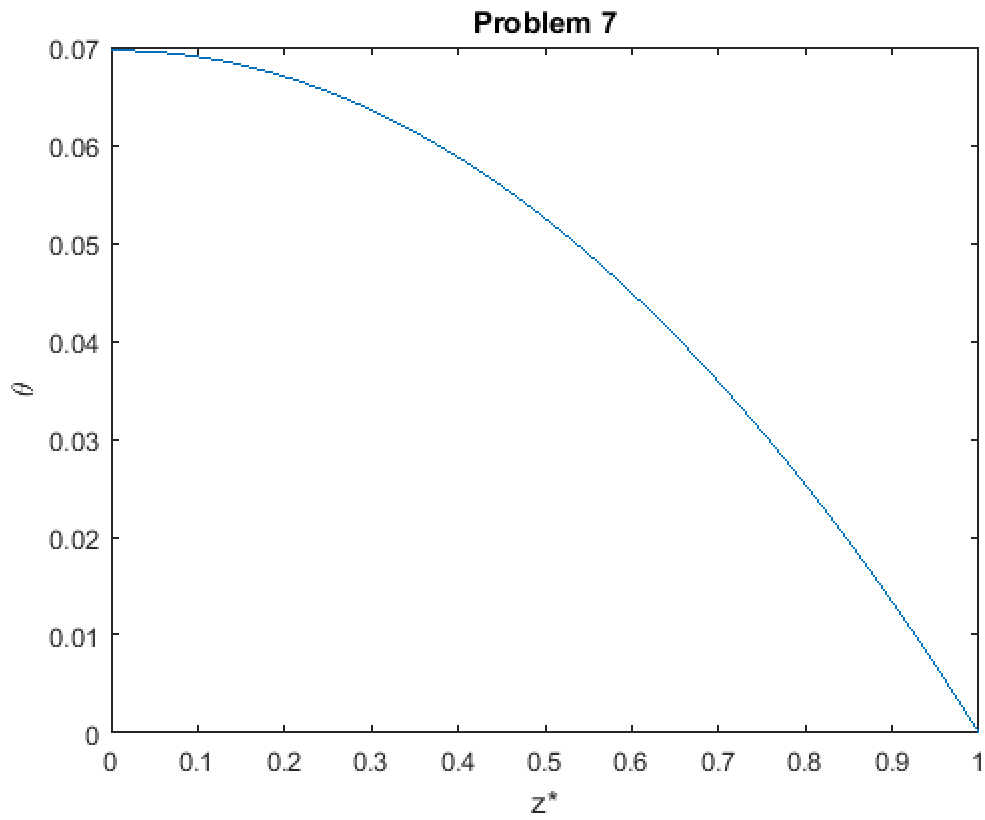
$$\frac{d^2\theta}{dz^2} + B\phi^2(1 - \frac{\theta}{B})exp(\frac{\gamma\theta}{\gamma+\theta}) = 0$$

$$\frac{\omega_{i-1}-2\omega_i+\omega_{i+1}}{h^2}+B\phi^2(1-\frac{\omega_i}{B})exp(\frac{\gamma\omega_i}{\gamma+\omega_i})=0$$

```
B = 0.6;
phi2 = 0.25;
gamma = 30.0;
N = 100;
h = 1/N;
guess = ones(N,1);
xspan = linspace(0,1,N);
w = fsolve(@prob7,guess);
figure
plot(xspan,w);
xlabel('z*');
ylabel('\theta');
title('Problem 7');

    function y=prob7(x)
        y = ones(N,1);
        for i = 2:N-1
            y(i) = (x(i+1)-2*x(i)+x(i-1))/h^2+B*phi2*(1-x(i)/
B)*exp(gamma*x(i)/(gamma+x(i)));
        end
        y(1)= (x(1)-x(2))/h-0; % Neumann Boundary Condition
        y(N)= x(N)-0;          % Dirichlet Boundary Condition
    end
```

Equation solved.

fsolve completed because the vector of function values is near zero as measured by the default value of the function tolerance, and the problem appears regular as measured by the gradient.



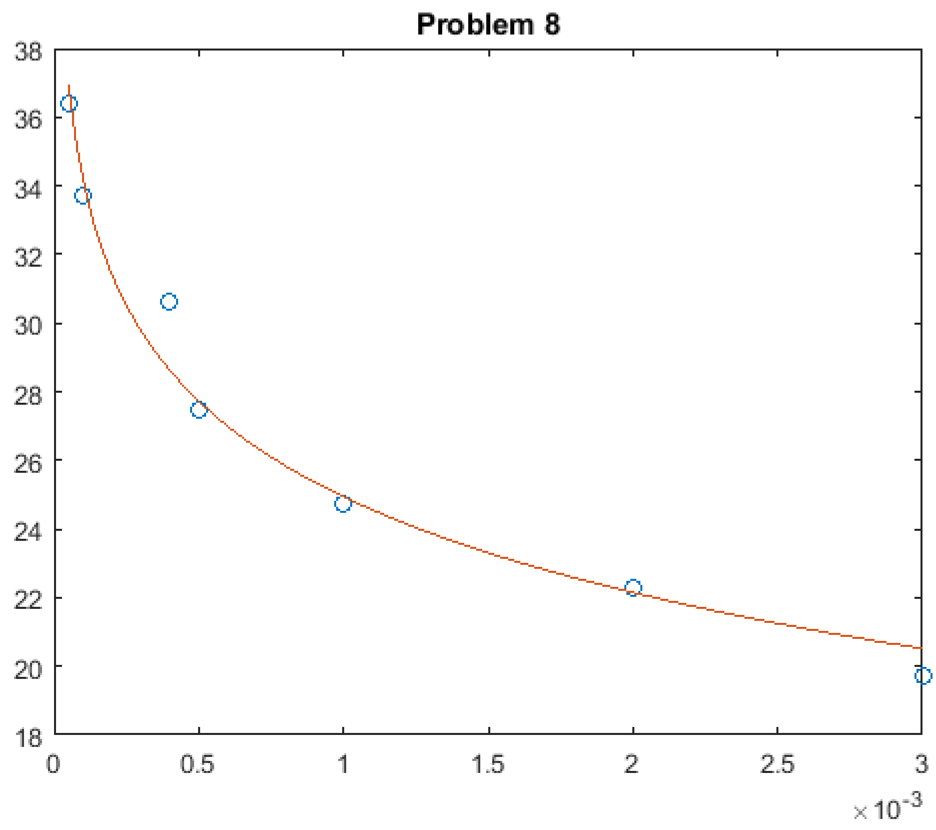
Problem 8: Nonlinear Optimization

```
Cbulk = [5e-5 1e-4 4e-4 5e-4 1e-3 0.002 0.003];
gammaeq = [36.42 33.72 30.63 27.45 24.76 22.30 19.71];
gamma0 = 52.2;
M = 627;
R = 8.314;
T = 298.15; % Room Temperature
x0 = [1e-3,2e-3];  % This guess is very important
x = fminunc(@prob8,x0)

C = linspace(min(Cbulk),max(Cbulk))
gfit=f(C);
figure

plot(Cbulk,gammaeq,'o',C,gfit);

title('Problem 8')
    function s=prob8(x)
        gammaini = x(1);
        a = x(2);
        f=@(t) gamma0 + R*T*gammaini*log(a./(t + a));
        gamfit = f(Cbulk);
        e = gammaeq-gamfit;
        s = 0.5*dot(e,e);
```

```
    end
```

```
Warning: Gradient must be provided for trust-region algorithm; using
quasi-newton algorithm instead.

Local minimum possible.

fminunc stopped because it cannot decrease the objective function
along the current search direction.



x =

    0.0016    0.0000


C =

  Columns 1 through 7

    0.0001    0.0001    0.0001    0.0001    0.0002    0.0002    0.0002

  Columns 8 through 14

    0.0003    0.0003    0.0003    0.0003    0.0004    0.0004    0.0004

  Columns 15 through 21

    0.0005    0.0005    0.0005    0.0006    0.0006    0.0006    0.0006

  Columns 22 through 28

    0.0007    0.0007    0.0007    0.0008    0.0008    0.0008    0.0009

  Columns 29 through 35

    0.0009    0.0009    0.0009    0.0010    0.0010    0.0010    0.0011

  Columns 36 through 42

    0.0011    0.0011    0.0012    0.0012    0.0012    0.0012    0.0013

  Columns 43 through 49

    0.0013    0.0013    0.0014    0.0014    0.0014    0.0015    0.0015

  Columns 50 through 56

    0.0015    0.0015    0.0016    0.0016    0.0016    0.0017    0.0017

  Columns 57 through 63
```

```
    0.0017    0.0017    0.0018    0.0018    0.0018    0.0019    0.0019

  Columns 64 through 70

    0.0019    0.0020    0.0020    0.0020    0.0020    0.0021    0.0021

  Columns 71 through 77

    0.0021    0.0022    0.0022    0.0022    0.0023    0.0023    0.0023

  Columns 78 through 84

    0.0023    0.0024    0.0024    0.0024    0.0025    0.0025    0.0025

  Columns 85 through 91

    0.0026    0.0026    0.0026    0.0026    0.0027    0.0027    0.0027

  Columns 92 through 98

    0.0028    0.0028    0.0028    0.0029    0.0029    0.0029    0.0029

  Columns 99 through 100

    0.0030    0.0030
```



Problem 9 Optimization

```
f=@(x) 10/sin(x) + 10/cos(x);
guess = pi/4;
optx = fminunc(f,guess)
sprintf('The shortest landder theta is %f ft', f(optx))
```

```
Warning: Gradient must be provided for trust-region algorithm; using
quasi-newton algorithm instead.

Initial point is a local minimum.

Optimization completed because the size of the gradient at the initial
 point
is less than the default value of the function tolerance.




optx =

    0.7854


ans =

The shortest landder theta is 28.284271 ft
```

```
end
```

Published with MATLAB® R2015a