Experiment 4

Determining the Shape of Suspension Bridge Cable

Aim:

To determine the shape of the Suspension Bridge Cable when different loads and tensions act on the cable.

Mathematical Background (Basic Model and Method Applied):

The Suspension Bridge will consist of the main cable, the hangers, and the deck. The weight of the deck and the loads on the deck are transferred to the cable through the hangers. Let the cable be loaded by the external load w(x). Now to derive the equation of the cable, we consider a cable element. Assuming that $d\theta$ tends to zero, we have sin $d\theta$ approx. equals $d\theta$, and cos $d\theta$ approx. equals 1. After applying Newton's Second Law (Equilibrium Condition) and using tanθ = dy/dx, we have the Differential Equation for the shape of the Suspension Cable:

D2y = w(x) / T_H

Now to solve this DE, we will use the method of **Variation of Parameters.** Consider the DE to be of the form:

$$y'' + q(t)\,y' + r(t)\,y = g(t)$$

Let y1(t) and y2(t) be the solution for the below DE (Complementary Function):

$$y'' + q(t)\,y' + r(t)\,y = 0$$

W(y1,y2) is the determinant of the Wonskian Matrix (The case when the functions y1 and y2 are linearly dependent has to be taken care in the code):

$$W(y_1, y_2) = y_1 y_2' - y_2 y_1' \neq 0$$

The Particular Solution is then given by the below formula:

$$Y_P(t) = -y_1 \int \frac{y_2 g(t)}{W(y_1, y_2)}\,dt + y_2 \int \frac{y_1 g(t)}{W(y_1, y_2)}\,dt$$

The Complete Solution is given by: C1y1(t) + C2y2(t) + YP

Code:

```
%Determining the Shape of a Suspension Cable - 16BCE0783
%sym2str - Copyright (c) 2008, Martin Lawson. All rights reserved.
clear vars;close all;clc
syms m x c1 c2
a = input('Input the coefficient of Second Derivative of y(D2): ');
b = input('Input the coefficient of First Derivative of y(D1): ');
c = input('Input the coefficient of y(D0): ');
f = input('Input the Inhomogenous function f(x): ');
i3 = input('Input conditional x0: ');
i1 = input('Input value at condition y(x0): ');
i4 = input('Input value at condition y1(x0): ');
m = solve(a*m^2+b*m+c,m);
if (b^2-4*a*c<0)
    alpha = (m(1) + m(2))/2;
    beta = abs((m(1) - m(2))/2);
    y1 = exp(alpha*x)*cos(beta*x);
    y2 = exp(alpha*x)*sin(beta*x);
elseif (b^2-4*a*c==0)
    y1 = exp(m(1)*x);
    y2 = x*exp(m(2)*x);
else
    y1 = exp(m(1)*x);diff(y,x)
    y2 = exp(m(2)*x);
end
Wm = [y1 y2;diff(y1,x) diff(y2,x)];
Wd = det(Wm);
if (Wd==0)
    y2 = y2*x;
end
Wm = [y1 y2;diff(y1,x) diff(y2,x)];
Wd = det(Wm);
yc = c1*y1 + c2*y2;
u1 = -int((y2*f)/(a*Wd),x);
u2 = int((y1*f)/(a*Wd),x);
yp = u1*y1 + u2*y2;
y = yc + yp;
eq1 = subs(y,x,i3);
eq2 = subs(diff(y,x),x,i3);
[sol_c1,sol_c2] = solve(eq1==i1,eq2==i4,c1,c2);
y_final = subs(y,{c1,c2},[sol_c1,sol_c2]);
figure
ezplot(y_final,[-3 10]);
title([char(y_final),' - 16BCE0783'])
xlabel('Displacement')
ylabel('Height')
cable_length = int((1+diff(y_final,x)^2)^0.5,x,-3,10);
disp([' Equation for the shape of the cable is:',char(10),sym2str(vpa(y_final))]);
disp(['The cable length is:',char(10),sym2str(vpa(cable_length))]);
```

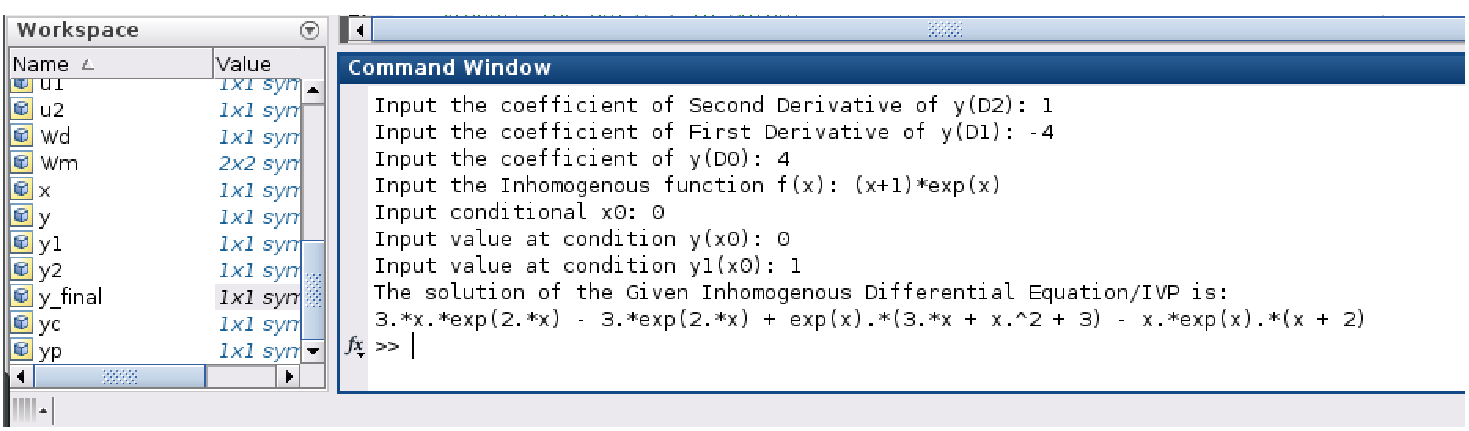
Question 1: Solving an Inhomogenous DE using Variation of Parameters and IVP

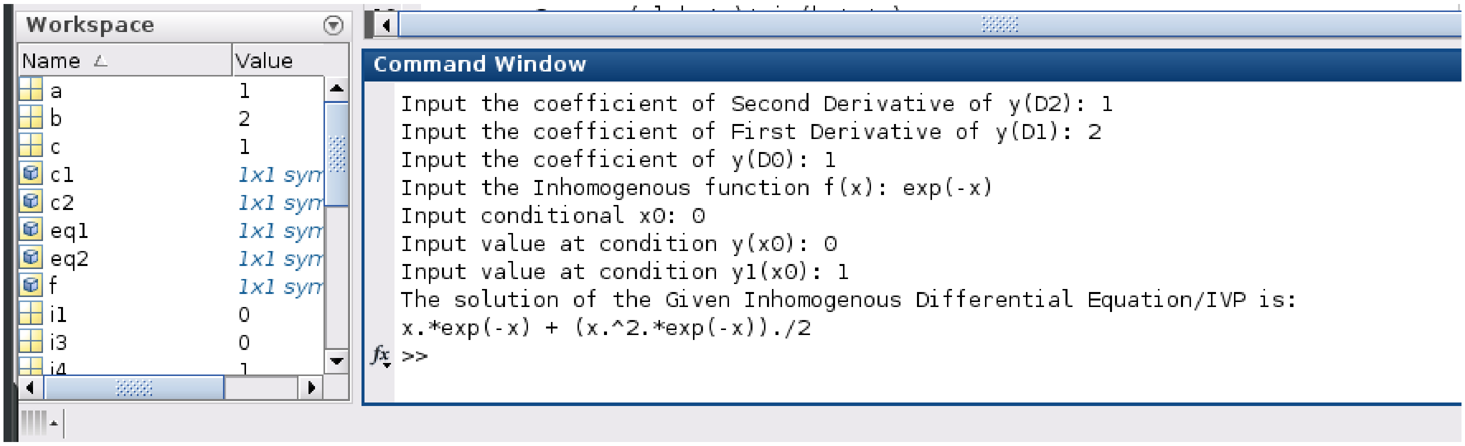
Code:

```
%Code for Solving Inhomogenous DE using Variation of Parameters and IVP -
%Illustrations taken from Theory Classes - Conditions supplied randomly
%sym2str - Copyright (c) 2008, Martin Lawson. All rights reserved.
%Ignore the dots(.) in Output. - 16BCE0783
```

```
clear vars;close all;clc
syms m x c1 c2
a = input('Input the coefficient of Second Derivative of y(D2): ');
b = input('Input the coefficient of First Derivative of y(D1): ');
c = input('Input the coefficient of y(D0): ');
f = input('Input the Inhomogenous function f(x): ');
i3 = input('Input conditional x0: ');
i1 = input('Input value at condition y(x0): ');
i4 = input('Input value at condition y1(x0): ');
m = solve(a*m^2+b*m+c,m);
if (b^2-4*a*c<0)
    alpha = (m(1) + m(2))/2;
    beta = abs((m(1) - m(2))/2);
    y1 = exp(alpha*x)*cos(beta*x);
    y2 = exp(alpha*x)*sin(beta*x);
elseif (b^2-4*a*c==0)
    y1 = exp(m(1)*x);
    y2 = x*exp(m(2)*x);
else
    y1 = exp(m(1)*x);
    y2 = exp(m(2)*x);
end
Wm = [y1 y2;diff(y1,x) diff(y2,x)];
Wd = det(Wm);
if (Wd==0)
    y2 = y2*x;     %Taking care if the Functions are Dependent
end
Wm = [y1 y2;diff(y1,x) diff(y2,x)];
Wd = det(Wm);
yc = c1*y1 + c2*y2;
u1 = -int((y2*f)/(a*Wd),x);
u2 = int((y1*f)/(a*Wd),x);
yp = u1*y1 + u2*y2;
y = yc + yp;
eq1 = subs(y,x,i3);
eq2 = subs(diff(y,x),x,i3);
[sol_c1,sol_c2] = solve(eq1==i1,eq2==i4,c1,c2);
y_final = subs(y,{c1,c2},[sol_c1,sol_c2]);
disp(['The solution of the Given Inhomogenous Differential Equation/IVP is:
',char(10),sym2str(y_final)])
```

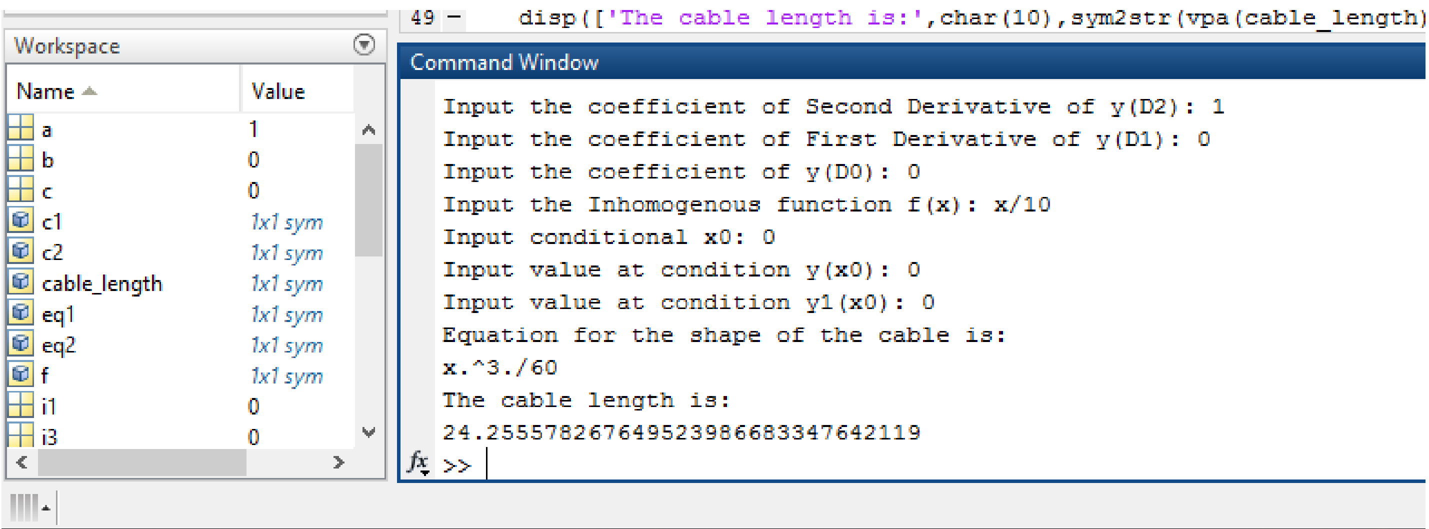
Input and Output:

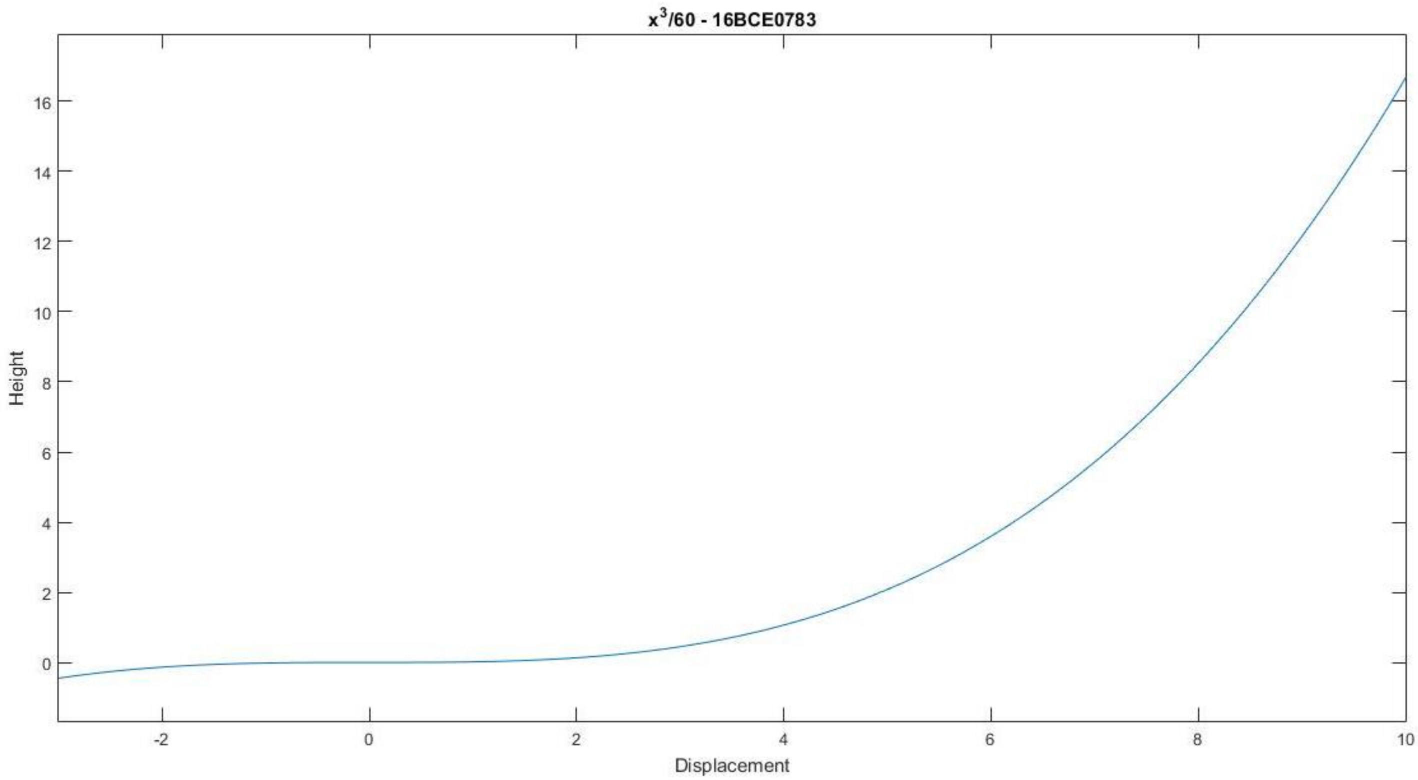




Question 2: Finding the shape of the Bridge Cable having the given Inputs (using above code)

Input and Outputs:





Question 3: Main Assumptions made for deriving the shape of the Suspension Cable

1. The cable is assumed to be inextensible.

2. The weight of the cable is assumed to be null since it is very less as compared to the length of the deck having the load.

3. In real life the weight distributed is discrete, but we have assumed it to be some function of the displacement.

Experiment 5

Functionality of Windshield Wipers

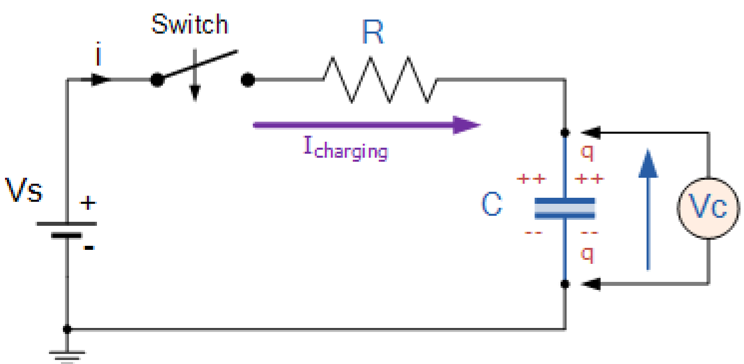
Aim:

1. To understand the use of dsolve command in solving Differential Equations and System of DEs

2. Visualizing and understanding the working mechanism of Windshield Wipers in Vehicles

Mathematical Background:

We all know Windshield Wipers are used at the time of rainfall to provide comfort to the driver while driving and avoid any accidents. The working of such wipers depend on the charging and discharging of capacitor. The wipers are actually part of an RC-circuit. As the voltage across the capacitor increases, the capacitor reaches at a point at which it starts discharging and the wipers start working. The circuit then starts charging again and the cycle continues. The time interval between the individual sweeps of the wipers is determined by the value of the time constant. The basic circuit and circuit and circuit equations are given below:



$$R\frac{dq}{dt} + \frac{1}{C}q = V$$

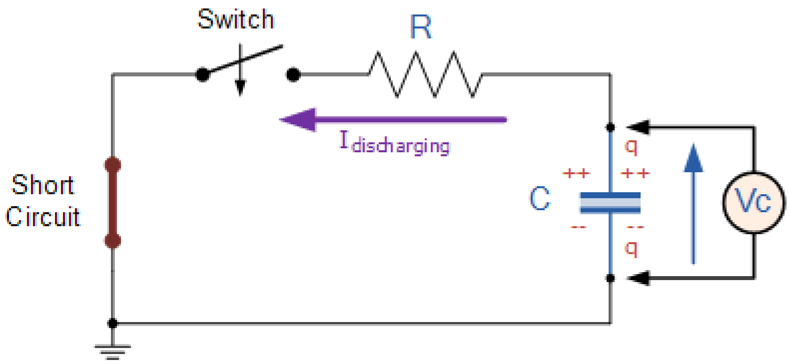
$$Vc = Vs\,(1 - e^{-t/RC})$$

where:

Vc is the voltage across the capacitor and Vs is the supply voltage

t is the elapsed time since the application of the supply voltage

RC is the time constant of the RC charging circuit



$$\frac{Q}{C} + R\frac{dQ}{dt} = 0\,.$$

$$V_C = V_S \times e^{-t/RC}$$

Code: Written and modified as in Question 2

Question 1: Using dsolve in two different ways

About dsolve: dsolve function in MATLAB is used to solve a differential equation or a system of Differential Equations with or without initial conditions. When we solve a system of DEs, the output is not an array but a structure consisting of solutions. Its basic syntax is:

dsolve(eq_1,eq_2…..eq_n,cond_1,cond_2….cond_n)

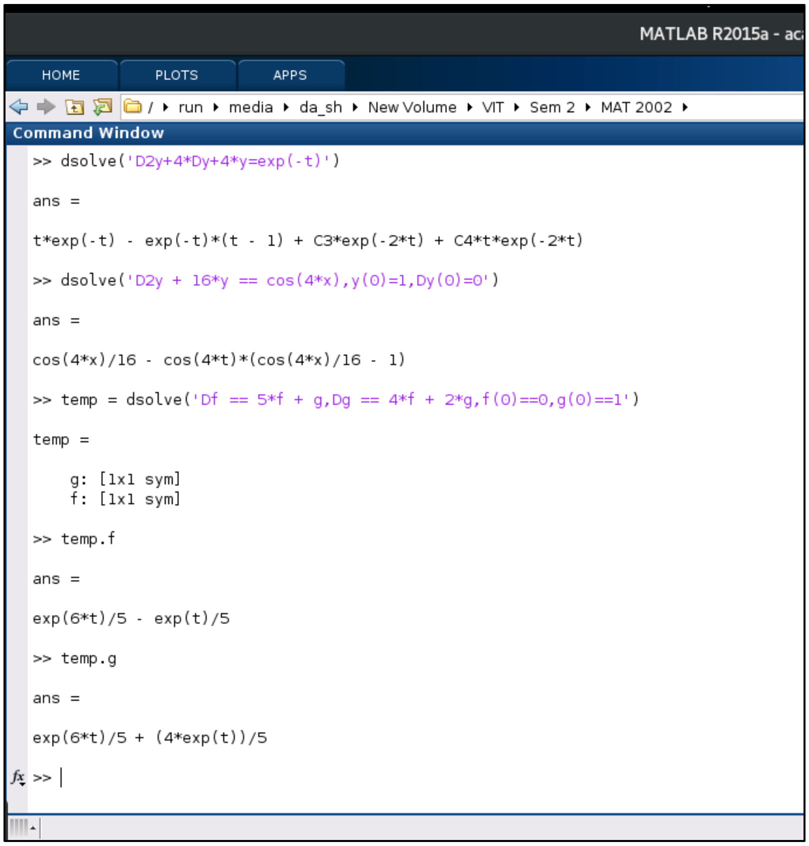
Important point to remember:

If you are using D2y – Put in inverted commas.

If you are using diff(y,n) etc., then do not use inverted commas.

In first command, I have given a DE without conditions and in second, I have provided a DE with conditions. In third command, I have provided a system of DEs with conditions of which the output can be seen is having a structure consisting of the solutions which can be accessed as:

structure_name<dot>function_name



Question 2: Given RC Circuit, R = 8x10^5, C = 5x10^-6, E = 12V. Find maximum Current, Charge and to plot the charge and current across capacitor while Charging and Discharging.

Code:

```
%Solving an RC Circuit
%sym2str - Copyright (c) 2008, Martin Lawson. All rights reserved.
clear vars;close all;clc;
syms q(t) i(t)
r = input('Input the value of Resistor: ');
c = input('Input the value of Capacitor: ');
e = input('Input the external Voltage: ');
Dq = diff(q);
sol_q = dsolve(r*Dq+(q/c) == e,Dq(0) == e/r);
sol_q = vpa(sol_q);
%time constant is the time at which the charge on the capacitor is equal to
%(1-1/e) times final charge(=c*e)(which occurs at infinity) (in charging state)
max_q = limit(sol_q,t,inf);
time_const = solve(sol_q == (1-1/exp(1))*max_q,t);
max_i = limit(diff(sol_q),t,0);
disp(['The time constant for the given circuit
is:',char(10),sym2str(time_const),char(10)]);
disp(['The maximum charge on the capacitor in the given circuit
is:',char(10),sym2str(max_q),char(10)]);
disp(['The maximum current in the given circuit is:',char(10),sym2str(max_i)]);
subplot(2,2,1)
ezplot(sol_q,[0 7]);
title('Capacitor Charge - Charging - 16BCE0783')
subplot(2,2,2)
ezplot(diff(sol_q),[0 7]);
title('Capacitor Current - Charging')
```

```
%Discharging State
sol2_q = dsolve(Dq+q/(r*c)==0,q(0)==max_q);
subplot(2,2,3)
ezplot(sol2_q,[0 7]);
title('Capacitor Charge - Discharging')
subplot(2,2,4)
ezplot(diff(sol2_q),[0 7]);
title('Capacitor Current - Discharging')
```

Input and Outputs:

Input the value of Resistor: 8*10^5

Input the value of Capacitor: 5*10^-6

Input the external Voltage: 12

The time constant for the given circuit is:

3.99999999999999961578319987112 66

The maximum charge on the capacitor in the given circuit is:

0.000059999999999999994743787867790275

The maximum current in the given circuit is:

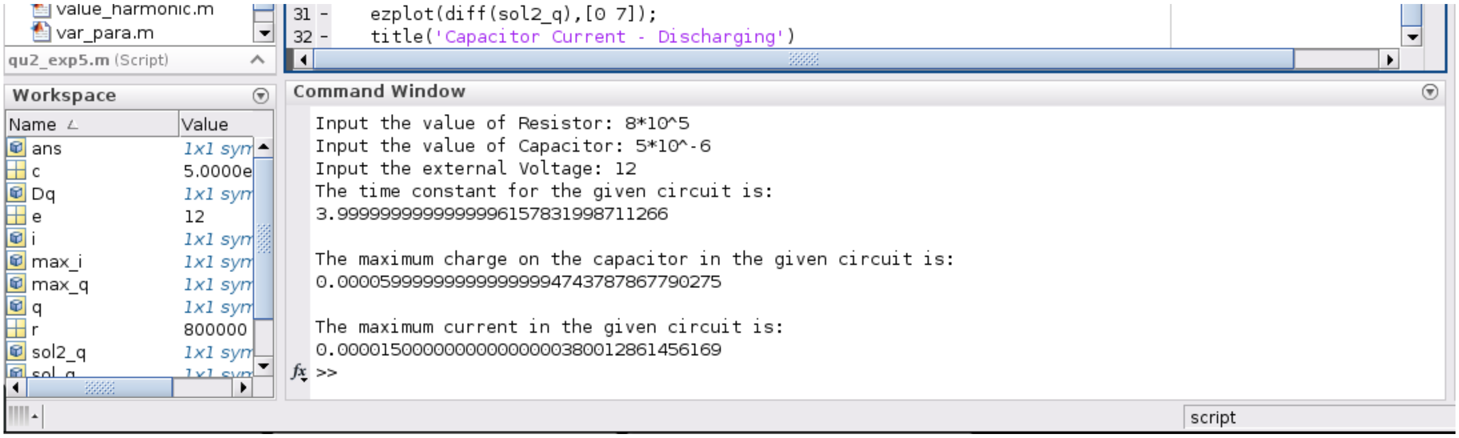
0.000015000000000000000380012861456169

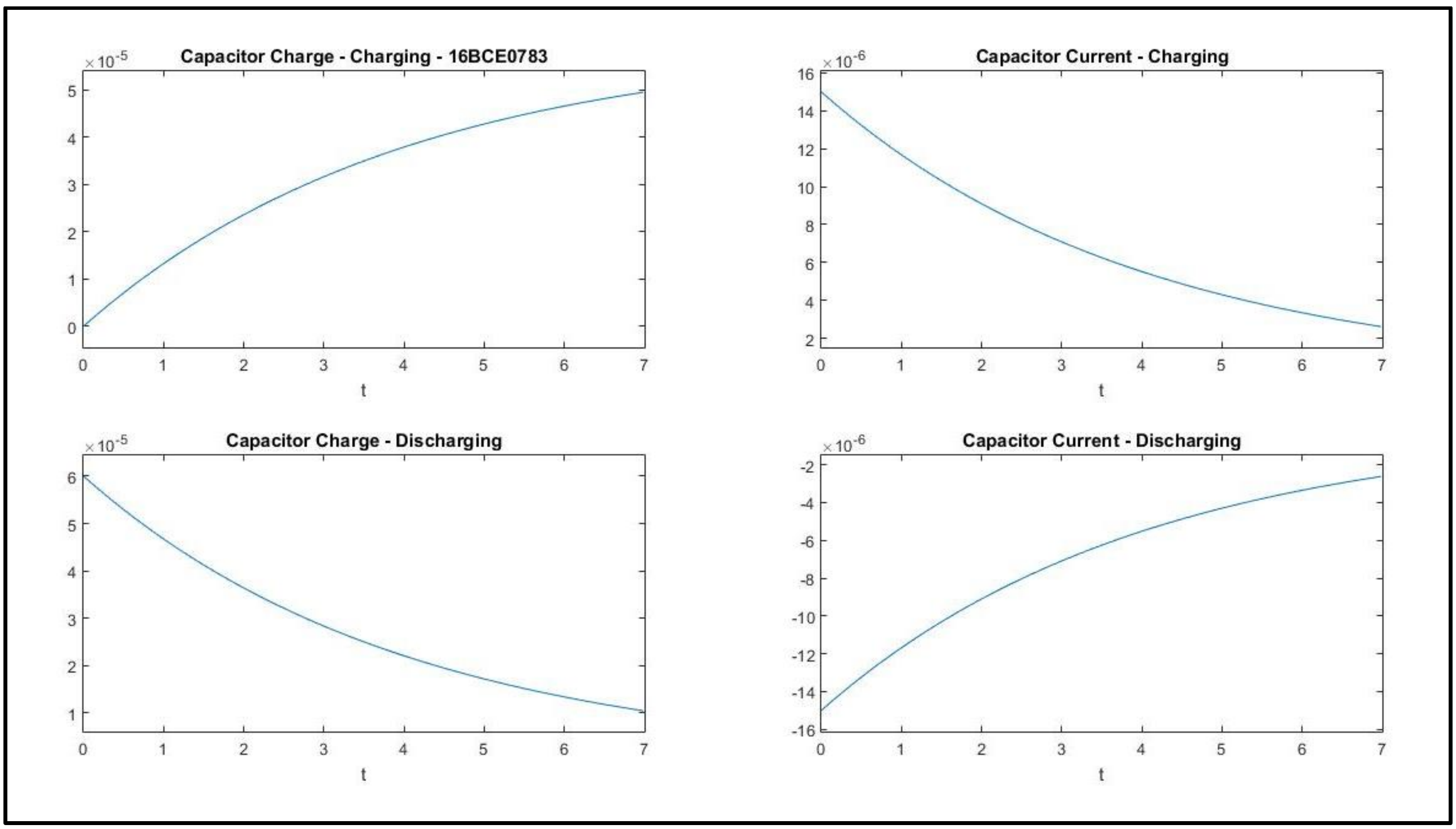
>> pretty(sol_q)

0.000059999999999999994743787867790275 - exp(-0.25000000000000002190088388420719 t)/0.000059999999999999999626383931361 4952

>> pretty(diff(sol_q))

exp(-0.25000000000000002190088388420719 t) / 0.000015000000000000000380012861456169


value_harmonic.m
var_para.m
31 - ezplot(diff(sol2_q),[0 7]);
32 - title('Capacitor Current - Discharging')
qu2_exp5.m (Script)
Workspace
Name
Value
ans 1x1 sym
c 5.0000e
Dq 1x1 sym
e 12
i 1x1 sym
max_i 1x1 sym
max_q 1x1 sym
q 1x1 sym
r 800000
sol2_q 1x1 sym
Command Window
Input the value of Resistor: 8*10^5
Input the value of Capacitor: 5*10^-6
Input the external Voltage: 12
The time constant for the given circuit is:
3.9999999999999996157831998711266
The maximum charge on the capacitor in the given circuit is:
0.000059999999999999994743787867790275
The maximum current in the given circuit is:
0.000015000000000000000380012861456169
>>
script


Capacitor Charge - Charging - 16BCE0783
Capacitor Current - Charging
Capacitor Charge - Discharging
Capacitor Current - Discharging
× 10^-5
× 10^-6
t

Experiment 6

Use of Harmonic Analysis as a tool in solving Engineering Problems

Aim:

To gain a good understanding and practice with the Fourier series for periodic signals. To use MATLAB to plot the approximate signals using the discrete values at different time. (To convert discrete function into continuous using Fourier Transform)

Mathematical Background:

Fourier analysis plays an important role in communication theory. Fourier discovered that any periodic signal could be represented as a sum of sinusoids. A waveform with periodicity T seconds and frequency f0 = 1/T can be represented as a sum of sinusoids with integral multiples of frequency f0. Mathematically, we can write (w = 2*pi/T):

$$f(t) = a_0 + \sum_{n=1}^{\infty} a_n \cos\, n\omega t + \sum_{n=1}^{\infty} b_n \sin\, n\omega t$$

Coefficients are given by:

$$\begin{aligned} a_0 &= \frac{1}{T}\int_0^T f(t)\, dt \\ a_n &= \frac{2}{T}\int_0^T f(t)\cos\, n\omega t\, dt \\ b_n &= \frac{2}{T}\int_0^T f(t)\sin\, n\omega t\, dt \end{aligned}$$

The Fourier Series is an infinite series, but for most periodic signals we may only need a few harmonics to get a good approximation.

Question 1: fourierseries function

Code:

x -> x Values Vector

f -> function value vector

T -> Time Period

N -> Number of Harmonics

```
function snx = fourierseries(x,f,T,N)
    syms t
    n = numel(x);
    if numel(x) ~= numel(f)
        disp('The number of elements in x and function value vector are not equal.')
        return;
    end
    a0 = 2*(sum(f)/n);
    snx = a0/2;
    xx = (2*pi*x)/T;%(converting rango from 0 to 2pi)
    for i=1:N
        aa = 2*(sum(f.*cos(xx*i))/n);
        bb = 2*(sum(f.*sin(xx*i))/n);
        snx = snx + aa*cos(2*i*pi*t/T) + bb*sin(2*i*pi*t/T);
    end
end
```

Question 2: Finding the charge on the capacitor using the fourierseries function for the given inputs

Code:

```
%Converting Discrete to Continous function using Fourier Function 16BCE0783
clear vars;close all;clc;
x = input('Input the time vector: ');
ff = input('Input the Voltage Source vector: ');
T = input('Input Period(2L/T): ');
N = input('Input the Number of Harmonics: ');
c = input('Input the value of the Capacitor: ');
r = input('Input the value of the Resistor: ');
q_val = [];
for k=1:numel(x) q_val(k) = c*ff(k)*(1-exp(-x(k)/r*c)); end
%q_val = c.*ff.*(1-exp(-x./r*c)) will also work instead of loop.
sol = fourierseries(x,q_val,T,N);
disp(['The charge on the capacitor as a function of time is:',char(10)]);
pretty(sol)
ezplot(sol,[0 4*pi]);
ylabel('Charge across Capacitor')
xlabel('time - 16BCE0783')
```

Input and Outputs:

```
Command Window
Input the time vector: [0 1 2 3 4 5]
Input the Voltage Source vector: [9 18 24 28 26 20]
Input Period(2L/T): 6
Input the Number of Harmonics: 2
Input the value of the Capacitor: 5*10^-6
Input the value of the Resistor: 8*10^5
The charge on the capacitor as a function of time is:

                                     / 2 pi t \                          / pi t \
                                  cos| ------ | 336719718545935      sin| ---- | 6312451854085933
        4674461975108281             \   3    /                          \  3   /
------------------------------- - ------------------------------ - -------------------------------
2535301200456458802993406410752   633825300114114700748351602688   5070602400912917605986812821504

          / 2 pi t \                          / pi t \
       sin| ------ | 4757210092934323      cos| ---- | 5334696717355215
          \   3    /                          \  3   /
   - -------------------------------- - -------------------------------
     20282409603651670423947251286016   5070602400912917605986812821504

>> |
```

Input the time vector: [0 1 2 3 4 5]

Input the Voltage Source vector: [9 18 24 28 26 20]

Input Period(2L/T): 6

Input the Number of Harmonics: 2

Input the value of the Capacitor: 5*10^-6

Input the value of the Resistor: 8*10^5

The charge on the capacitor as a function of time is:

```
                    / 2 pi t \              / pi t \
                 cos | ------ | 336719718545935   sin | ---- | 6312451854085933
   4674461975108281        \  3  /                \  3  /
------------------------------ - ---------------------------- - ------------------------------
2535301200456458802993406410752  633825300114114700748351602688
5070602400912917605986812821504


   / 2 pi t \              / pi t \
```

```
 sin| ------ | 4757210092934323    cos| ---- | 5334696717355215
   \   3   /                   \  3  /
- -------------------------------- - ------------------------------
 20282409603651670423947251286016  5070602400912917605986812821504
```