EAST WEST UNIVERSITY

Department of EEE

LAB PROJECT

Section: 01

Course Code: EEE204

Course Title: Numerical Analysis for Electrical Engineering

Course Instructor: M. Ryyan Khan,
Chairperson & Associate Professor, Department of EEE

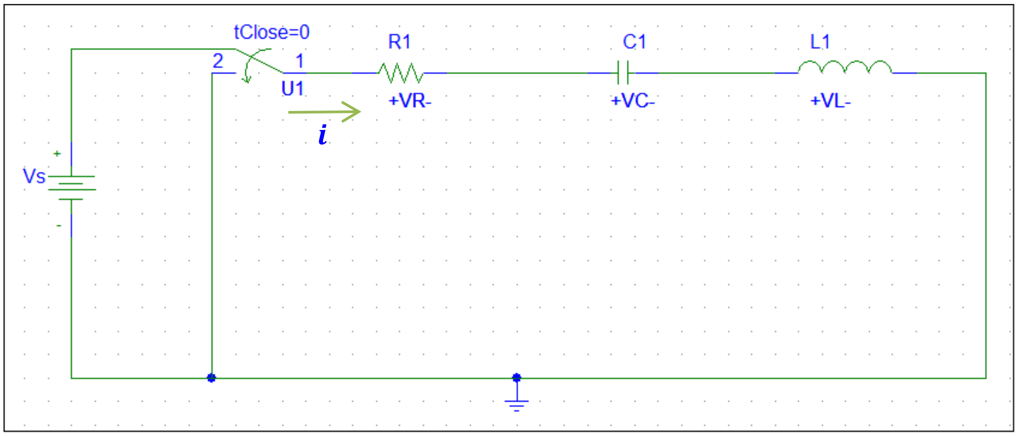
Submission Date: 20/08/2022

Name: Rohit Bhowmick

Id: 2020-1-80-006

01.(a)

Approximately marking the directions (arrows and +/−) of i, V_R, V_L, V_C in the circuit,



01.(b)

The differential equation for the system,

$$L\frac{d^2i}{dx} + R\frac{di}{dx} + \frac{i}{c} = 0$$

Its boundary conditions are,

At $t_o = 0s$ is $i_o = 0$ A
&
At $t_{k+1} = 50$ms is $i_{k+1} = 132.56$mA

01.(c)

From ans 01(b),

$$L\frac{d^2i}{dx}+R\frac{di}{dx}+\frac{i}{c}=0$$

Applying centered difference formula,

$$L\frac{i_{k+1}-2i_k+i_{k-1}}{h^2}+R\frac{i_{k+1}-i_{k-1}}{2h}+\frac{i_k}{C}=0$$

$$\Rightarrow \frac{Li_{k+1}-2Li_k+Li_{k-1}+\frac{R}{2}hi_{k+1}-\frac{R}{2}hi_{k-1}+\frac{i_k}{C}h^2}{h^2}=0$$

$$\Rightarrow \; i_{k-1}\left[L-\frac{R}{2}h\right]+i_k\left[\frac{h^2}{C}-2L\right]+i_{k-1}\left[L+\frac{R}{2}h\right]=0$$

$$\therefore i_{k-1}\left[L-\frac{R}{2}h\right]+i_k\left[\frac{h^2}{C}-2L\right]+i_{k-1}\left[L+\frac{R}{2}h\right]=0$$

From the equation,

$a=L-\frac{R}{2}h$

$b=\frac{h^2}{C}-2L$

$c=L+\frac{R}{2}h$

$d=0$

01.(d)

From ans 01(c),

$a = L - \frac{R}{2}h$

$b = \frac{h^2}{C} - 2L$

$c = L + \frac{R}{2}h$

$d = 0$

$$A = \begin{bmatrix} b & c & 0 & 0 & 0 & \dots & \dots & 0 & 0 & 0 & 0 \\ a & b & c & 0 & 0 & \dots & \dots & 0 & 0 & 0 & 0 \\ 0 & a & b & c & 0 & \dots & \dots & 0 & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & 0 & \dots & \dots & a & b & c & 0 \\ 0 & 0 & 0 & 0 & 0 & \dots & \dots & 0 & a & b & c \\ 0 & 0 & 0 & 0 & 0 & \dots & \dots & 0 & 0 & a & b \end{bmatrix}$$

Its boundary condition is,

At $t_o = 0s$ is $i_o = 0$ A

&

At $t_{k+1} = 50$ms is $i_{k+1} = 132.56$mA

$$B = \begin{bmatrix} d - at_o \\ d \\ d \\ \vdots \\ \vdots \\ d \\ d \\ d - ct_{k+1} \end{bmatrix}$$

$$i = \begin{bmatrix} i_o \\ i_1 \\ i_2 \\ \vdots \\ \vdots \\ i_{k-1} \\ i_k \\ i_{k+1} \end{bmatrix}$$

Now,

We know that,

$A \times i = B$

$$\Rightarrow \begin{bmatrix} b & c & 0 & 0 & 0 & \dots & \dots & 0 & 0 & 0 & 0 \\ a & b & c & 0 & 0 & \dots & \dots & 0 & 0 & 0 & 0 \\ 0 & a & b & c & 0 & \dots & \dots & 0 & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & 0 & \dots & \dots & a & b & c & 0 \\ 0 & 0 & 0 & 0 & 0 & \dots & \dots & 0 & a & b & c \\ 0 & 0 & 0 & 0 & 0 & \dots & \dots & 0 & 0 & a & b \end{bmatrix} \times \begin{bmatrix} i_o \\ i_1 \\ i_2 \\ \vdots \\ \vdots \\ i_{k-1} \\ i_k \\ i_{k+1} \end{bmatrix} = \begin{bmatrix} d - at_o \\ d \\ d \\ \vdots \\ \vdots \\ d \\ d \\ d - ct_{k+1} \end{bmatrix}$$

02.

In MATLAB solving the problem,

```
clc
close all
clear

N=1000;

%Initial values
i_0=0; %A
i_n=132.56*10^(-3) %A

t_0=0; %s
t_n=50*10^(-3); %s

%The parameters are:
Vs=5 %V
R=0.1 %ohm
L=10^(-3) %H
C=10^(-3) %F

h=(t_n-t_0)/(N+1);

%Coefficient Values
a=(L-(R/2)*h);
b=(((h^2)/C)-(2*L));
c=(L+(R/2)*h);
d=0;

A=diag(b*ones(1,N))+diag(c*ones(1,N-1),1)+diag(a*ones(1,N-1),-1);

B_1=repmat(d,N-2,1);

B=[d-a*i_0
   B_1
   d-c*i_n];

T=inv(A)*B;

t=linspace(t_0,t_n,N+2);

i=[i_0
   T
   i_n];
```

03.

The exact solution is,

```
clc
close all
clear

N=1000,

%Initial values
t_0=0; %s
t_n=50*10^(-3); %s

%The parameters are:
Vs=5 %V
R=0.1 %ohm
L=10^(-3) %H
C=10^(-3) %F

t=linspace(t_0,t_n,N+2);

%Exact Solution
alp=R/(2*L);
w=1/(sqrt(L*C));
s_1=-alp+sqrt((alp^2)-(w^2));
s_2=-alp-sqrt((alp^2)-(w^2));
A=-Vs/(L*(s_1-s_2));
i_exact=A*exp(s_1*t)-A*exp(s_2*t);
```

04.(a)

The voltage across Inductor $v_L(t)$,

```
clc
close all
clear

N=1000;
w=1000; %From ans 03
%Initial values
i_0=0; %A
i_n=132.56*10^(-3) %A
t_0=0; %s
t_n=50*10^(-3); %s

%The parameters are:
Vs=5 %V
R=0.1 %ohm
L=10^(-3) %H
C=10^(-3) %F

h=(t_n-t_0)/(N+1);

%Coefficient Values
a=(L-(R/2)*h);
b=(((h^2)/C)-(2*L));
c=(L+(R/2)*h);
d=0;

A=diag(b*ones(1,N))+diag(c*ones(1,N-1),1)+diag(a*ones(1,N-1),-1);

B_1=repmat(d,N-2,1);

B=[d-a*i_0
   B_1
   d-c*i_n];

T=inv(A)*B;

t=linspace(t_0,t_n,N+2);

i=[i_0
   T
   i_n];

x_L=w*L;
V_L=i.*x_L

figure(1)
plot(t,V_L);
xlabel('t');
ylabel('V_L');
grid on
```

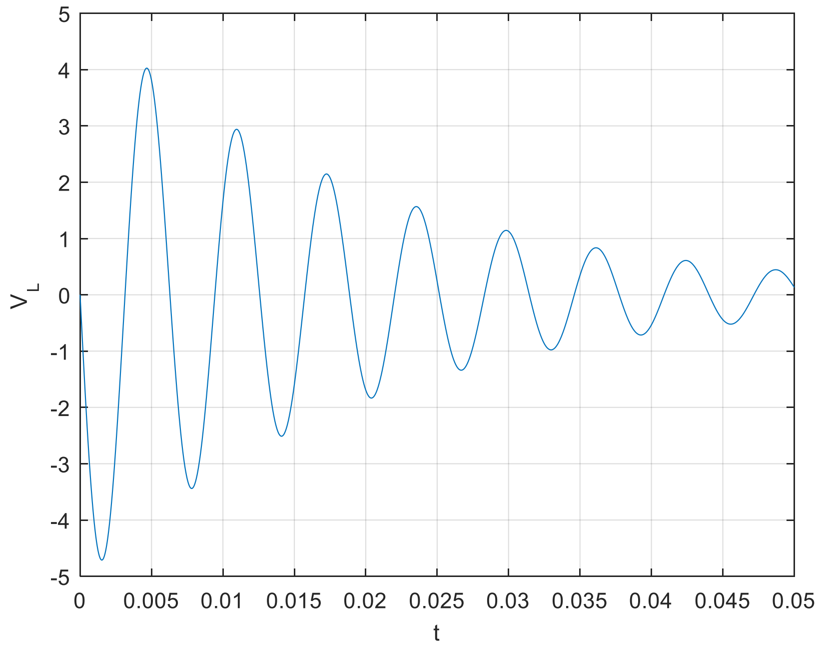


Figure: Voltage across Inductor v_L

04.(b)

The instantaneous power P_R ,

```
clc
close all
clear

N=1000;

%Initial values
i_0=0; %A
i_n=132.56*10^(-3) %A
t_0=0; %s
t_n=50*10^(-3); %s

%The parameters are:
Vs=5 %V
R=0.1 %ohm
L=10^(-3) %H
C=10^(-3) %F

h=(t_n-t_0)/(N+1);

%Coefficient Values
a=(L-(R/2)*h);
b=(((h^2)/C)-(2*L));
c=(L+(R/2)*h);
d=0;

A=diag(b*ones(1,N))+diag(c*ones(1,N-1),1)+diag(a*ones(1,N-1),-1);

B_1=repmat(d,N-2,1);

B=[d-a*i_0
   B_1
   d-c*i_n];

T=inv(A)*B;

t=linspace(t_0,t_n,N+2);

i=[i_0
   T
   i_n];

PR=(i.^2)*R;

figure(1)
plot(t,PR);
xlabel('t');
ylabel('PR');
grid on
```

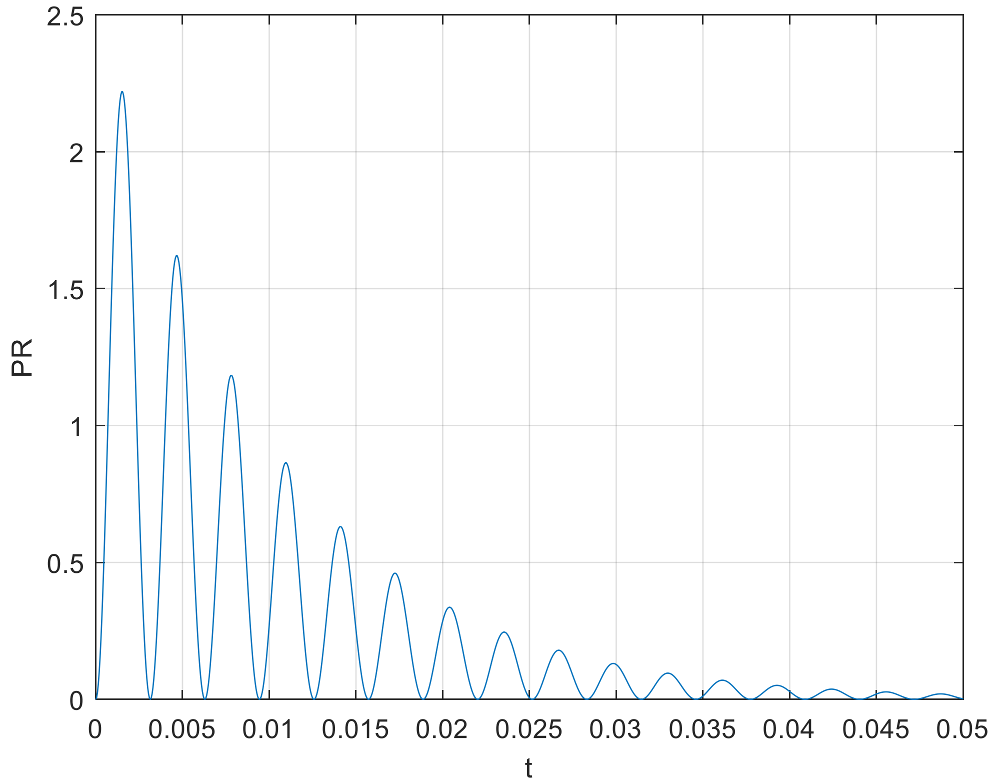


Figure: Instantaneous power, P_R

04.(c)

Instantaneous power absorbed by the inductor P_L

```
clc
close all
clear

N=1000;

x_L=1 %From ans 04(a)

%Initial values
i_0=0; %A
i_n=132.56*10^(-3) %A
t_0=0; %s
t_n=50*10^(-3); %s

%The parameters are:
Vs=5 %V
R=0.1 %ohm
L=10^(-3) %H
C=10^(-3) %F

h=(t_n-t_0)/(N+1);

%Coefficient Values
a=(L-(R/2)*h);
b=(((h^2)/C)-(2*L));
c=(L+(R/2)*h);
d=0;

A=diag(b*ones(1,N))+diag(c*ones(1,N-1),1)+diag(a*ones(1,N-1),-1);

B_1=repmat(d,N-2,1);

B=[d-a*i_0
   B_1
   d-c*i_n];

T=inv(A)*B;

t=linspace(t_0,t_n,N+2);

i=[i_0
   T
   i_n];

PL=(i.^2)*x_L;

figure(1)
plot(t,PL);
```

```
xlabel('t');
ylabel('PL');
grid on
```

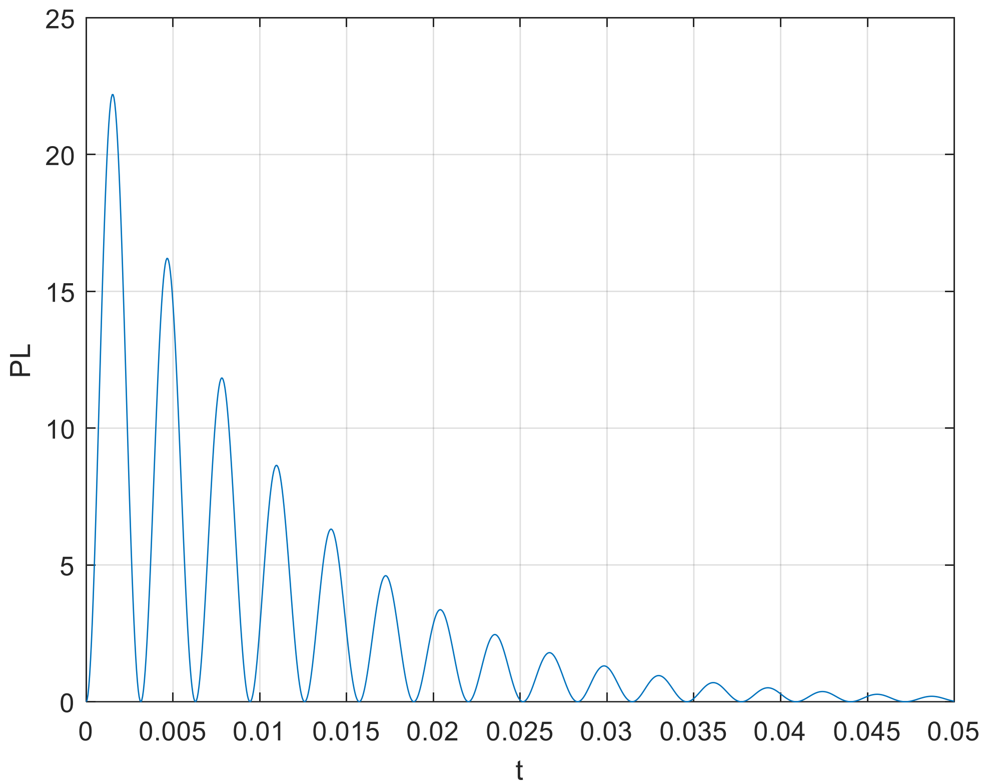


Figure: Instantaneous power absorbed by the inductor P_L

05.(a)

Plotting i vs t when n=100,

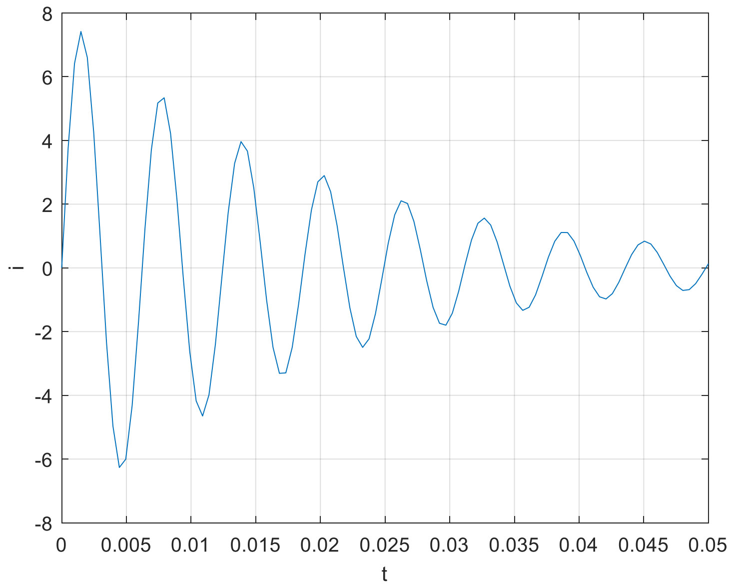


Figure: Plot i vs t

Plotting i_exact & i vs t in same plot

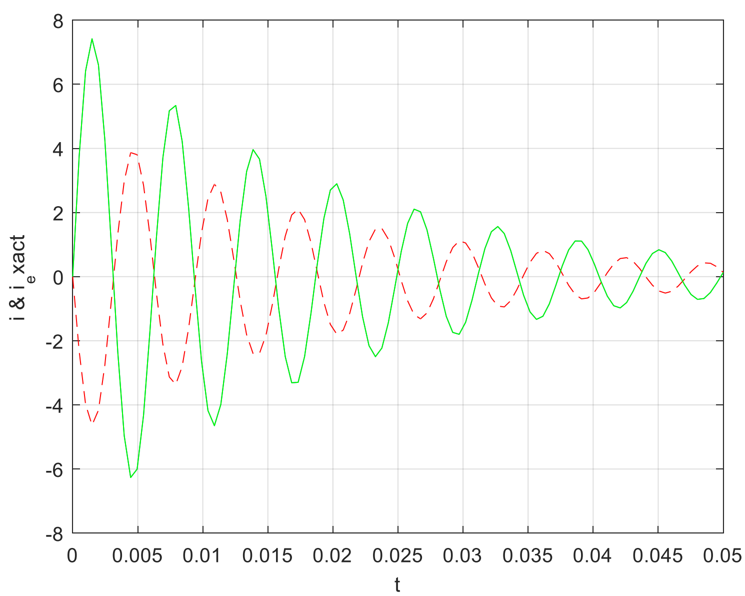


Figure: i_exact & i vs t in same plot

Finding error

```
clc
close all
clear

N=100;

%Initial values
i_0=0; %A
i_n=132.56*10^(-3) %A
t_0=0; %s
t_n=50*10^(-3); %s

%The parameters are:
Vs=5 %V
R=0.1 %ohm
L=10^(-3) %H
C=10^(-3) %F

h=(t_n-t_0)/(N+1);

%Coefficient Values
a=(L-(R/2)*h);
b=(((h^2)/C)-(2*L));
c=(L+(R/2)*h);
d=0;

A=diag(b*ones(1,N))+diag(c*ones(1,N-1),1)+diag(a*ones(1,N-1),-1);

B_1=repmat(d,N-2,1);

B=[d-a*i_0
   B_1
   d-c*i_n];

T=inv(A)*B;

t=linspace(t_0,t_n,N+2)

i=[i_0
   T
   i_n];

%Exact Solution
alp=R/(2*L);
w=1/(sqrt(L*C));
s_1=-alp+sqrt((alp^2)-(w^2));
s_2=-alp-sqrt((alp^2)-(w^2));
A=-Vs/(L*(s_1-s_2));
i_exact=A*exp(s_1*t)-A*exp(s_2*t);
```

```
ei=i_exact'-i;
En=sqrt (sum(ei.^2)/(N+2));
```

The amount of error En is 4.0680.

05.(b)

Plotting v_L vs t when n=100,

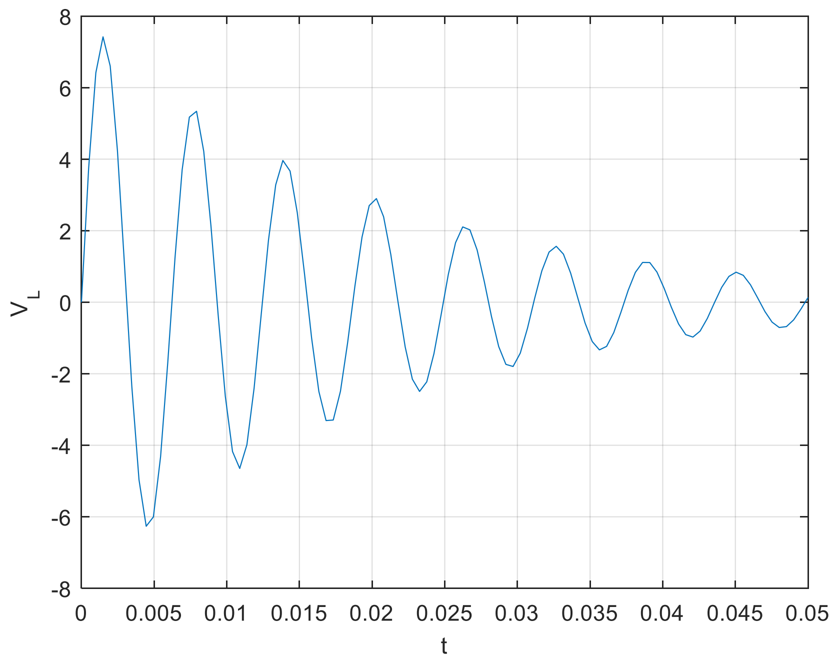


Figure: v_L vs t

05.(c)

Plotting P_R vs t when n=100,

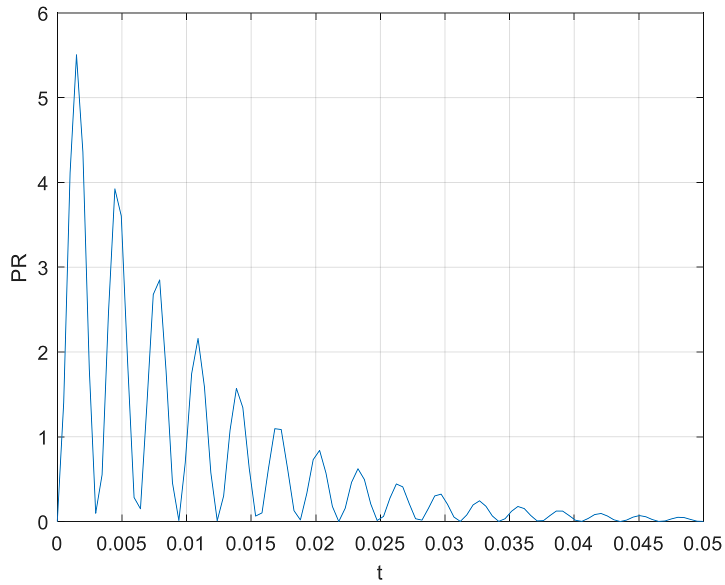


Figure: P_R vs t

Plotting P_R & vs $P_R exact$ vs t in same plot,

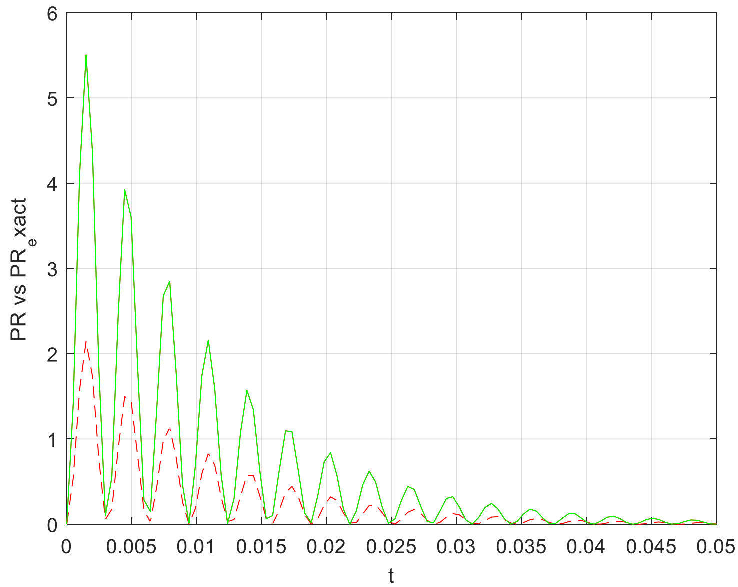


Figure: P_R & $P_R exact$ vs t in same plot

05.(d)

Plotting i vs t when n=200,

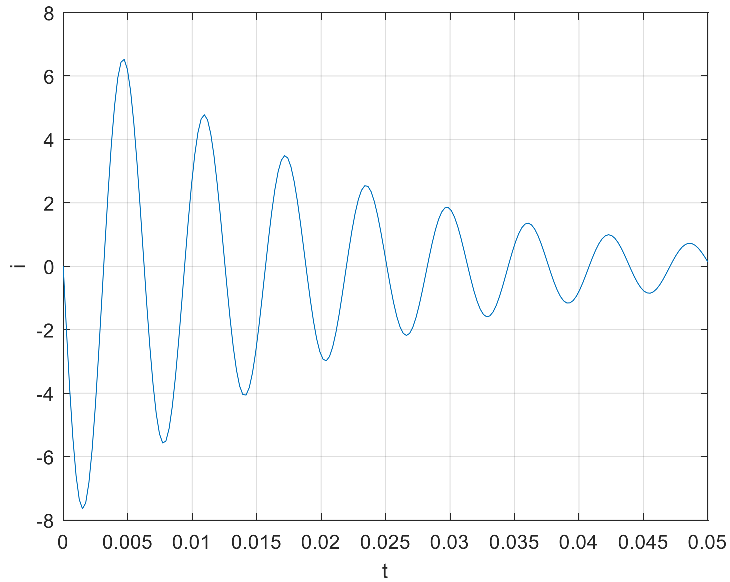


Figure: i vs t

Plotting i_exact & i vs t in same plot

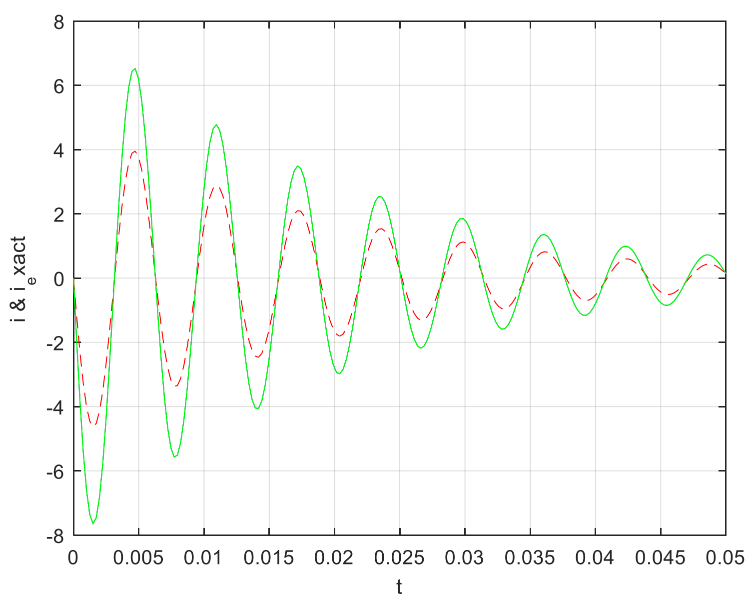


Figure: i_exact and i in same plot

Plotting v_L vs t when n=200,

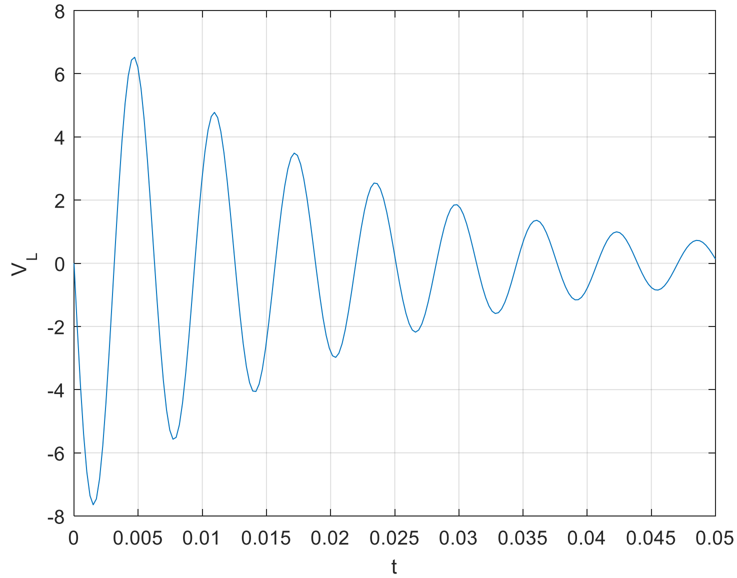


Figure: v_L vs t

Plotting P_R vs t when n=200,

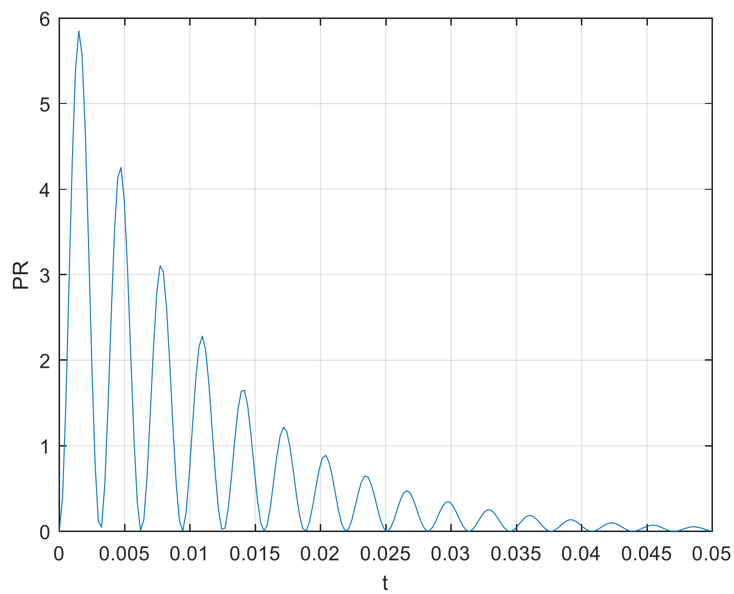


Figure: P_R vs t

Plotting P_R & vs $P_R exact$ vs t in same plot,

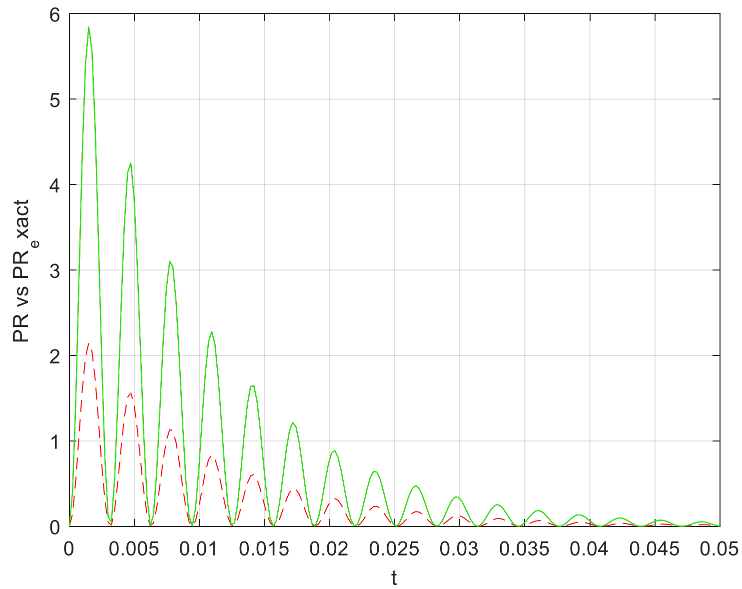


Figure: P_R & $P_R exact$ vs t in same plot

Plotting i vs t when n=500,

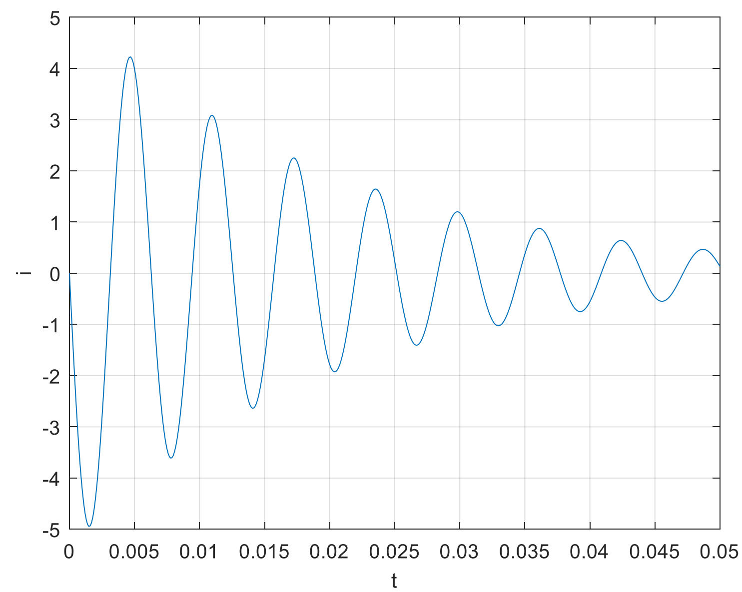


Figure: i vs t

Plotting i_exact & i vs t in same plot

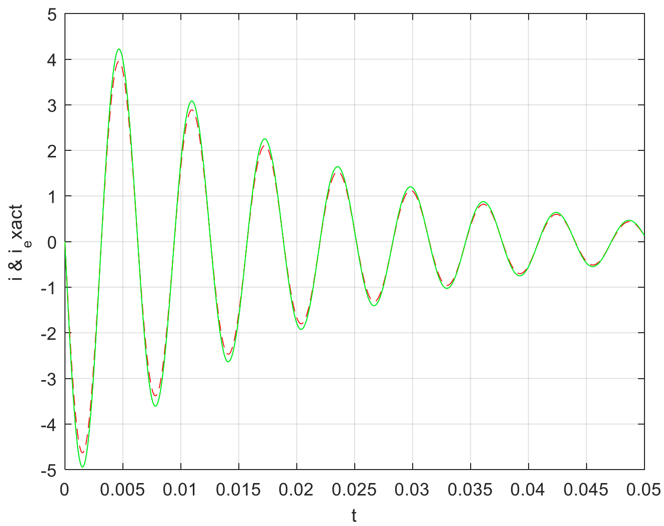


Figure: i_exact and i in same plot

Plotting v_L vs t when n=500,

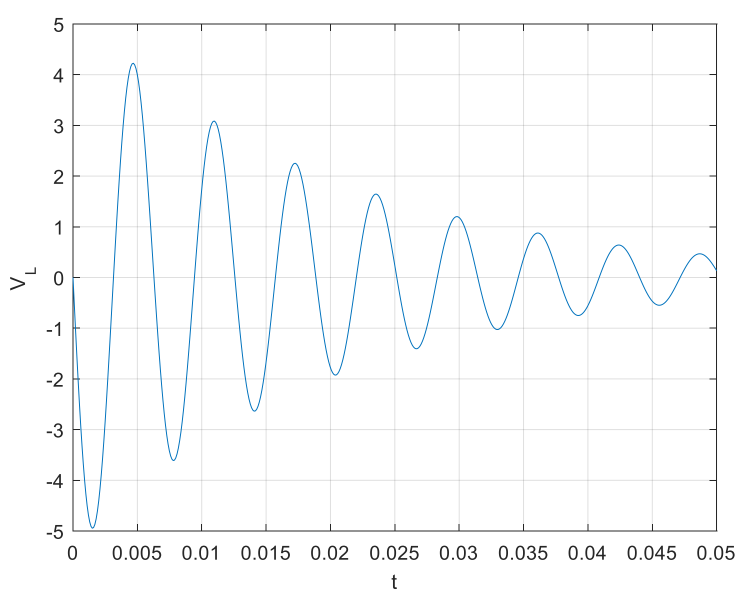


Figure: v_L vs t

Plotting P_R vs t when n=500,

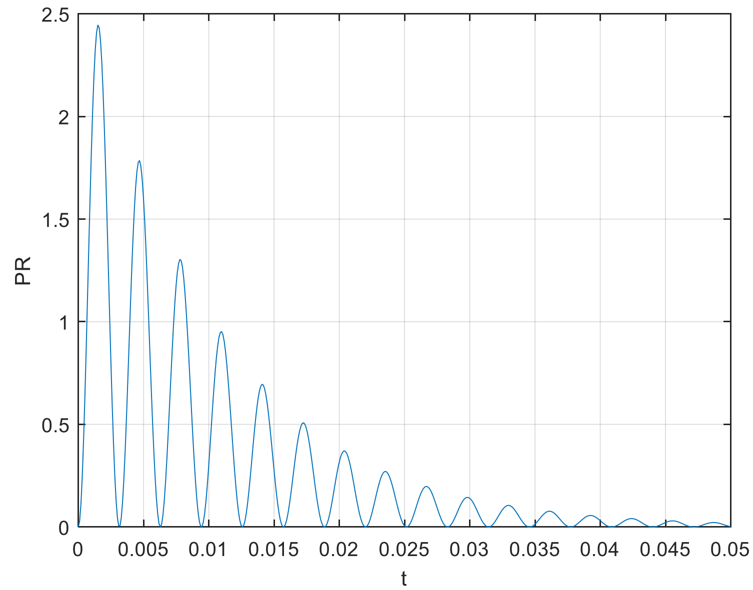


Figure: P_R vs t

Plotting P_R & vs $P_R exact$ vs t in same plot,

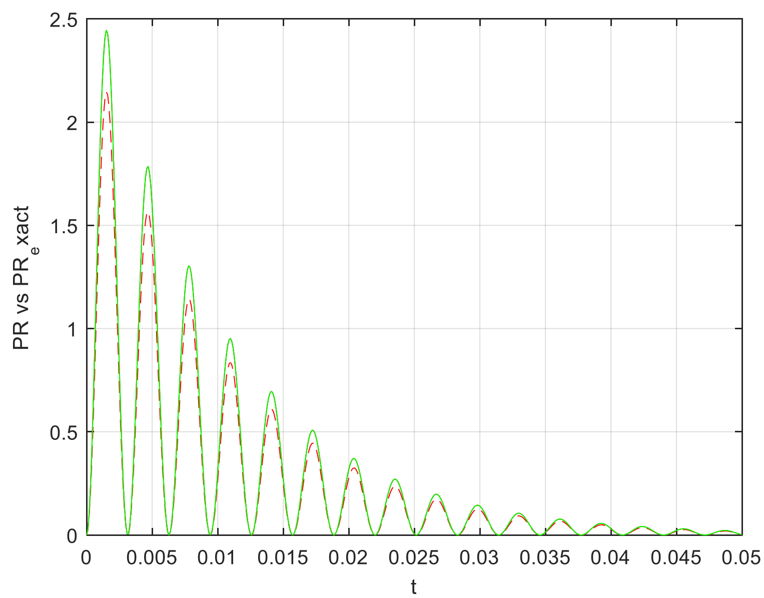


Figure: P_R & $P_R exact$ vs t in same plot

Plotting i vs t when n=1000,

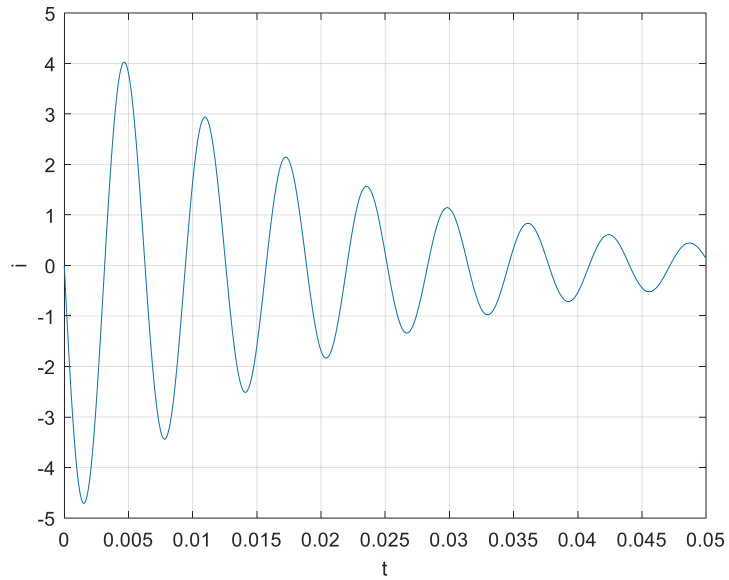


Figure: i vs t

Plotting i_exact & i vs t in same plot

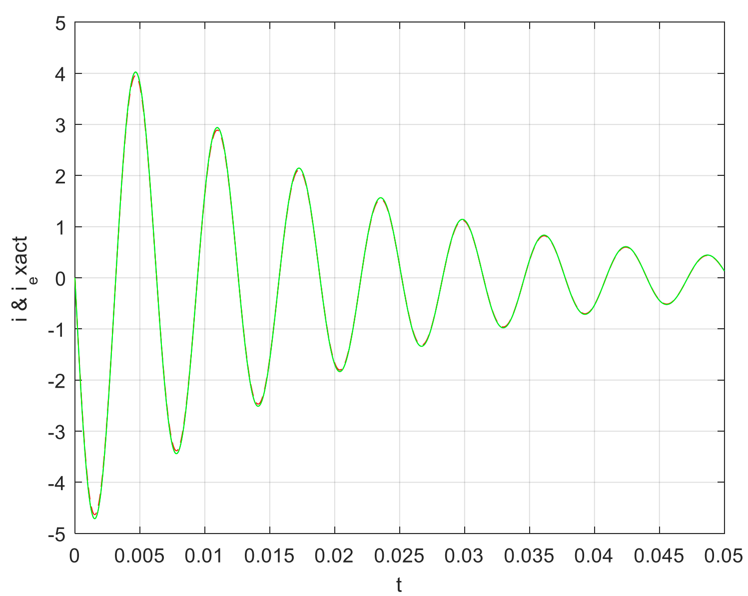


Figure: i_exact and i in same plot

Plotting v_L vs t when n=1000,

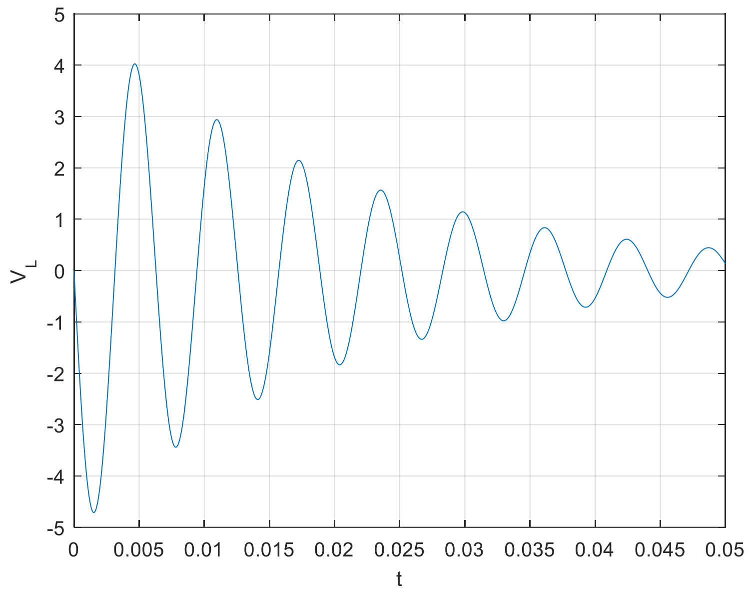


Figure: v_L vs t

Plotting P_R vs t when n=1000,

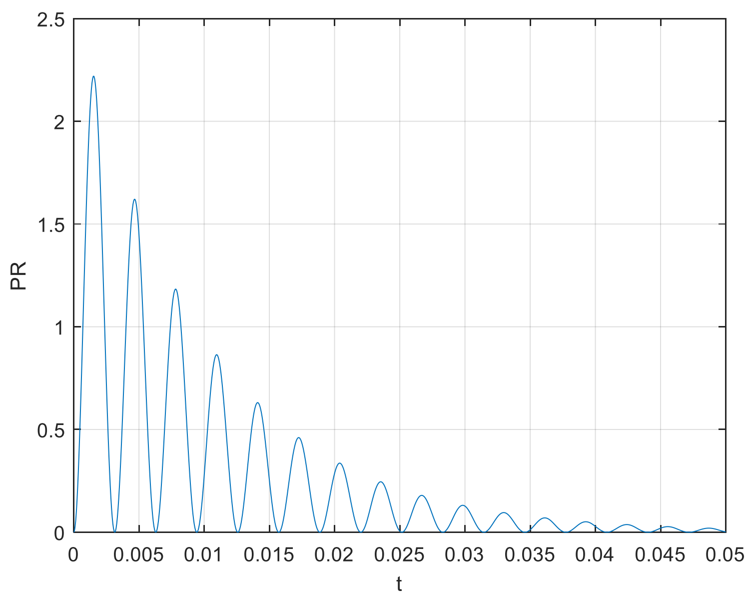


Figure: P_R vs t

Plotting P_R & vs $P_R exact$ vs t in same plot,

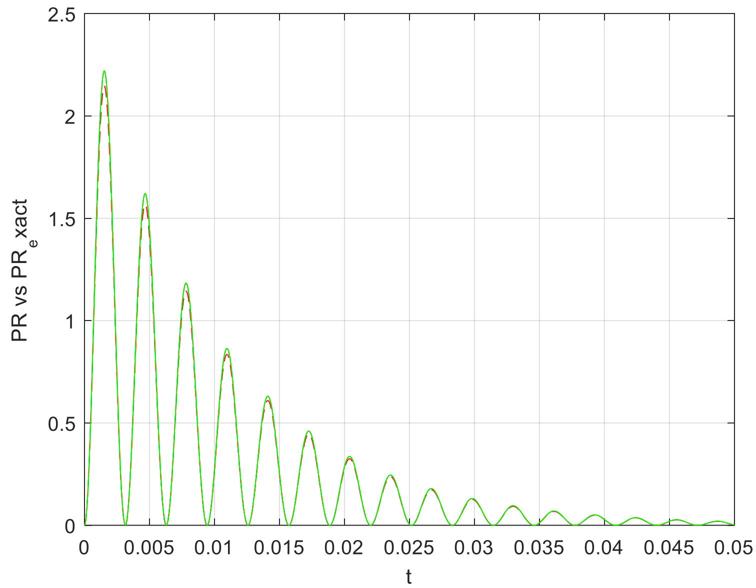


Figure: P_R & $P_R exact$ vs t in same plot

Plotting i vs t when n=5000,

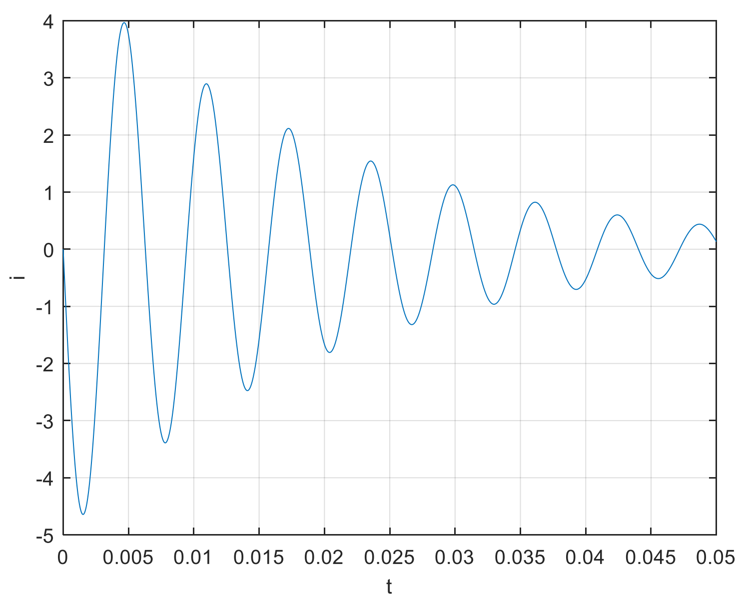


Figure: i vs t

Plotting i_exact & i vs t in same plot

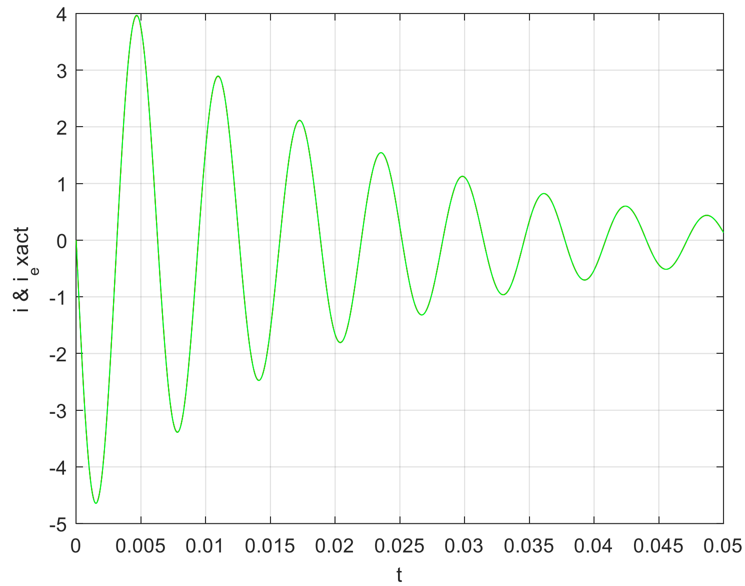


Figure: i_exact and i in same plot

Plotting v_L vs t when n=5000,

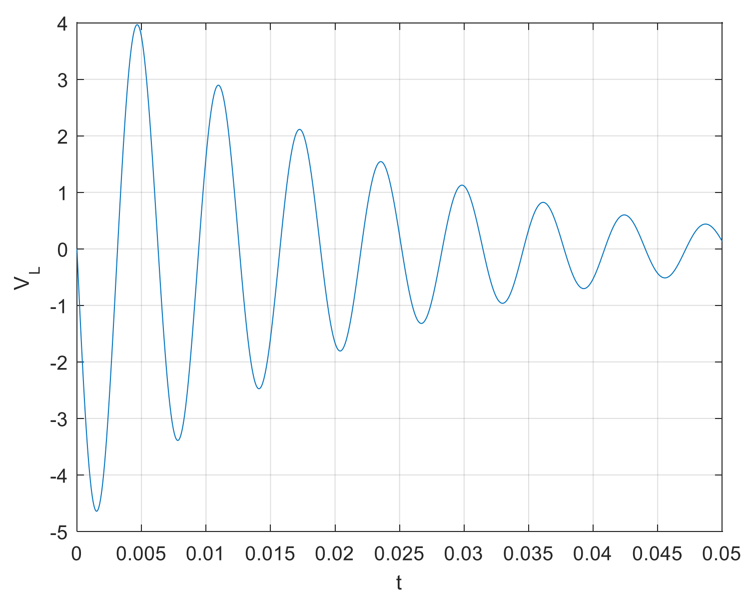


Figure: v_L vs t

Plotting P_R vs t when n=5000,

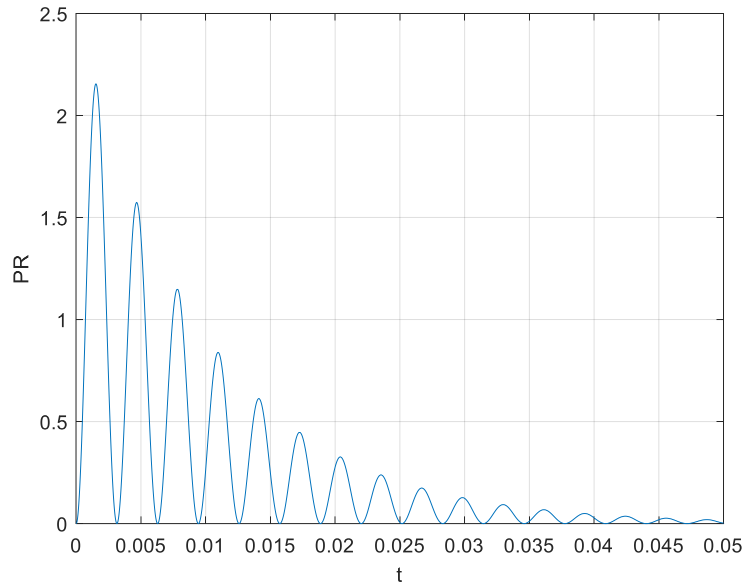


Figure: P_R vs t

Plotting P_R & vs $P_R exact$ vs t in same plot,

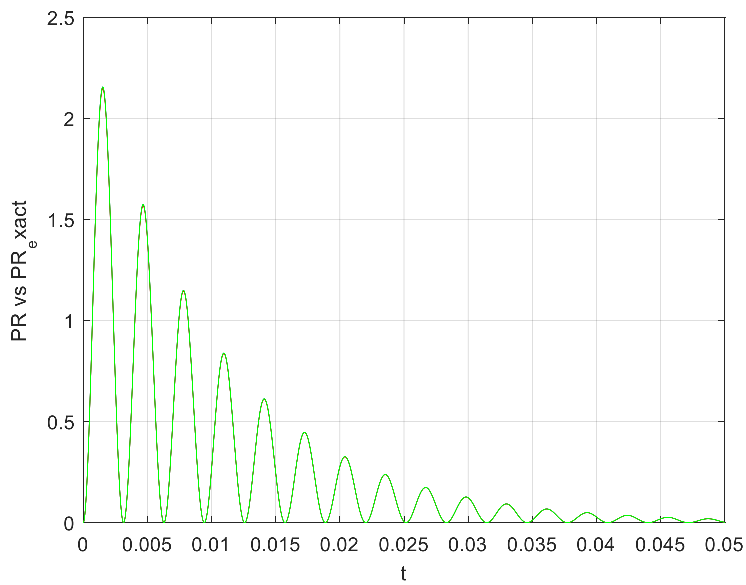


Figure: P_R & $P_R exact$ vs t in same plot

05.(e)

Yes, the accuracy improves with increasing the value of n. Because we know that n is proportional to error, En. So when the value of n is increased, the amount of error is increased. We collect some error of this circuit, when the value of n is 100 the error is 4.068 but when we increase the value of n we see that the error is decreased to 0.003.

06.

Find the En for n=[75:25:1000 2000 3500 5000]

The code is,

```
clc
close all
clear

n=[75:25:1000 2000 3500 5000];

for j=1:length(n)
N=n(j)

%Initial values
i_0=0; %A
i_n=132.56*10^(-3); %A
t_0=0; %s
t_n=50*10^(-3); %s

%The parameters are:
Vs=5; %V
R=0.1; %ohm
L=10^(-3); %H
C=10^(-3); %F

h=(t_n-t_0)/(N+1);

%Coefficient Values
a=(L-(R/2)*h);
b=(((h^2)/C)-(2*L));
c=(L+(R/2)*h);
d=0;

A=diag(b*ones(1,N))+diag(c*ones(1,N-1),1)+diag(a*ones(1,N-1),-1);

B_1=repmat(d,N-2,1);

B=[d-a*i_0
   B_1
   d-c*i_n];

T=inv(A)*B;

t=linspace(t_0,t_n,N+2);

i=[i_0
   T
   i_n];
```

```
%Exact Solution (03)
alp=R/(2*L);
w=1/(sqrt(L*C));
s_1=-alp+sqrt((alp^2)-(w^2));
s_2=-alp-sqrt((alp^2)-(w^2));
A=-Vs/(L*(s_1-s_2));
i_exact=A*exp(s_1*t)-A*exp(s_2*t);

ei=i_exact'-i;
En(j)=sqrt (sum(ei.^2)/(N+2));

end

figure(1)
plot(n,En,'o-');
grid on;
```

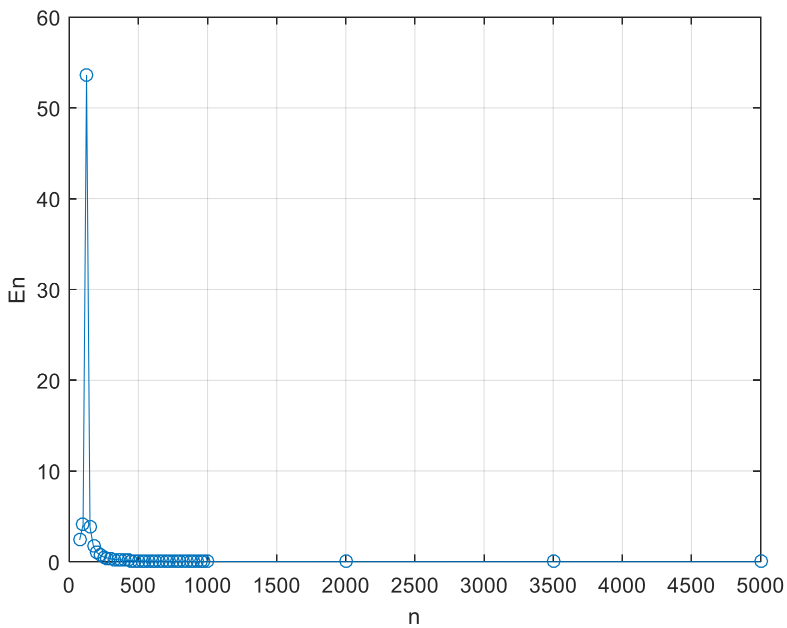


Figure: E_n vs n

We know that as the value of n increases, the value of En decreases because n is proportional to En. We can see that here as the value of n increases the value of En decreases. But here unexpectedly, when the value of n goes from 100 to 125, En increases, but when it goes from 125 to 150 and increases to 25 gap then En decreases again.

APPENDIX

The MATLAB code for step: (02-05),

```
clc
close all
clear

N=1000;

%Initial values
i_0=0; %A
i_n=132.56*10^(-3) %A
t_0=0; %s
t_n=50*10^(-3); %s

%The parameters are:
Vs=5 %V
R=0.1 %ohm
L=10^(-3) %H
C=10^(-3) %F

h=(t_n-t_0)/(N+1);

%Coefficient Values
a=(L-(R/2)*h);
b=(((h^2)/C)-(2*L));
c=(L+(R/2)*h);
d=0;

A=diag(b*ones(1,N))+diag(c*ones(1,N-1),1)+diag(a*ones(1,N-1),-1);

B_1=repmat(d,N-2,1);

B=[d-a*i_0
   B_1
   d-c*i_n];

T=inv(A)*B;

t=linspace(t_0,t_n,N+2)

i=[i_0
   T
   i_n];

%Exact Solution (03)
alp=R/(2*L);
w=1/(sqrt(L*C));
s_1=-alp+sqrt((alp^2)-(w^2));
s_2=-alp-sqrt((alp^2)-(w^2));
A=-Vs/(L*(s_1-s_2));
```

```
i_exact=A*exp(s_1*t)-A*exp(s_2*t);

%Find and Plot (04)
%04(a)
x_L=w*L;
V_L=i.*x_L

figure(1)
plot(t,V_L);
xlabel('t');
ylabel('V_L');
grid on

%04(b)
PR=(i.^2)*R;

figure(2)
plot(t,PR);
xlabel('t');
ylabel('PR');
grid on

%04(c)
PL=(i.^2)*x_L;

figure(3)
plot(t,PL);
xlabel('t');
ylabel('PL');
grid on

%Visualizations and plots

%05(a)

figure(4)
plot(t,i);
xlabel('t');
ylabel('i');
grid on;

figure(5)
plot(t,i); hold on

figure(5)
plot(t,i_exact,'r--');
plot(t,i,'g');
xlabel('t');
ylabel('i & i_exact');
grid on

ei=i_exact'-i;
```

```
En=sqrt (sum(ei.^2)/(N+2));

%05(b)

%N=100
figure(6)
plot(t,V_L);
xlabel('t');
ylabel('V_L');
grid on

%05(c)

%N=100
figure(7)
plot(t,PR);
xlabel('t');
ylabel('PR');
grid on

figure(8)
plot(t,PR,'r'); hold on

PR_exact=(i_exact.^2)*R;

figure(8)
plot(t,PR_exact,'r--');
plot(t,PR,'g');
xlabel('t');
ylabel('PR & PR_exact');
grid on
```

The MATLAB code for step:06,

```
clc
close all
clear

n=[75:25:1000 2000 3500 5000];

for j=1:length(n)
N=n(j)

%Initial values
i_0=0; %A
i_n=132.56*10^(-3); %A
t_0=0; %s
t_n=50*10^(-3); %s

%The parameters are:
Vs=5; %V
R=0.1; %ohm
L=10^(-3); %H
C=10^(-3); %F

h=(t_n-t_0)/(N+1);

%Coefficient Values
a=(L-(R/2)*h);
b=(((h^2)/C)-(2*L));
c=(L+(R/2)*h);
d=0;

A=diag(b*ones(1,N))+diag(c*ones(1,N-1),1)+diag(a*ones(1,N-1),-1);

B_1=repmat(d,N-2,1);

B=[d-a*i_0
   B_1
   d-c*i_n];

T=inv(A)*B;

t=linspace(t_0,t_n,N+2);

i=[i_0
   T
   i_n];

%Exact Solution (03)
alp=R/(2*L);
w=1/(sqrt(L*C));
s_1=-alp+sqrt((alp^2)-(w^2));
s_2=-alp-sqrt((alp^2)-(w^2));
```

```
A=-Vs/(L*(s_1-s_2));
i_exact=A*exp(s_1*t)-A*exp(s_2*t);

ei=i_exact'-i;
En(j)=sqrt (sum(ei.^2)/(N+2));

end

figure(1)
plot(n,En,'o-');
xlabel('n');
ylabel('En');
grid on;
```

CONCLUSION

In this project, we try to solve a RLC series circuit in MATLAB. Here we have to use NxN and 1xN matrix in MATLAB, I took the help of internet for this N x N & 1xN and use an array. We have been able to extract all the plots successfully. In the last point, the error we asked to get out was unexpectedly not what we expected but we managed to make the whole project successful.