MATH 214

NUMERICAL METHODS

FINAL PROJECT

MUHAMMED CEMAL ERYIGIT

1801022024

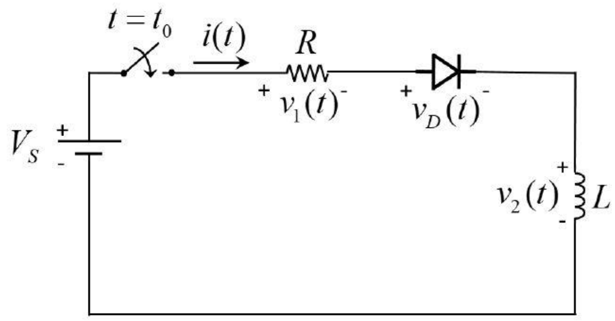
UPLOAD DATE

31/01/2021

PREPARATION DATE

30/01/2021

1- Problem definition and introduction formulas



In this circuit, which is difficult to encounter in daily life, the circuit was formulated as a differential equation.

$$Vs = V_1 + V_D + V_2$$

$$V_2 = L * \frac{di(t)}{dt}$$

$$\frac{di(t)}{dt} = \frac{1}{L} * (V_s - V_1 - V_D)$$

Euler's method, one of the easiest methods, was chosen to solve this equation. But we need a polynomial to apply the Euler method to this equation, because we only have 5 values. For this, a polynomial is fitted on 5 values with the least square method.

Since the diode characteristic is exponential, the least square method was applied differently.

At the end of the least squares method, we take the logarithm of the resulting exponential equation and leave x ,to get vd.

<u>Least squares</u>

The problem is to fit the given values on a polynomial with the least error. Least squares method is used to fit with the least error. The least squares method is the most convenient procedure for determining best linear approximations. Sometimes it is appropriate to assume that the data are exponentially related.

$$y = be^{a\,x}$$

The normal equation associated with this procedure is obtained as

$$E = \sum_{i=1}^{m} [y_i - be^{a\,x}]^2$$

$$0 = \frac{\partial E}{\partial b} = 2\sum_{i=1}^{m}(y_i - be^{a\,x})(-e^{a\,x})$$

and

$$0 = \frac{\partial E}{\partial a} = 2\sum_{i=1}^{m}(y_i - be^{a\,x})(-bxe^{a\,x})$$

The method that is commonly used when the data are suspected to be exponentially related is to consider the logarithm of the approximation equation:

$$ln\, y = ln\, b + ax$$

In either case, a linear problem now appears, and solutions for ln b and a can be obtained. Using this equations we can find a and b;

$$a = \frac{(m * \sum_{i=1}^{m} x_i * ln\, y_i) - (\sum_{i=1}^{m} x_i * ln\, y_i)}{m * (\sum_{i=1}^{m} x_i^2) - (\sum_{i=1}^{m} x_i)^2}$$

$$ln\, b \; = \frac{(\sum_{i=1}^{m} x_i^2 * \sum_{i=1}^{m} ln\, y_i) - (\sum_{i=1}^{m} x_i * ln\, y_i) \; * \sum_{i=1}^{m} x_i}{m * (\sum_{i=1}^{m} x_i^2) - (\sum_{i=1}^{m} x_i)^2}$$

$$b = exp(lnb)$$

<u>Euler Method</u>

Euler method is a first-order numerical method for solving ordinary differential equations (ODEs) with a given initial value. It is the most elementary method for numerical derivation of initial value problems for ordinary differential equations and is the simplest Runge–Kutta method. The object of Euler’s method is to obtain approximations to the well-posed initial-value with this formula;

$$w_{n+1} = w_n + h * f(t_n, w_n)$$

for each n=0,1,2….N

$$f(t, w) = \frac{di(t)}{dt}$$

As a result of these operations, we calculate the next value of the current with the Euler method.

2- Code description and inputs

All result and figures are generated using the code, which is given at the appendix section. The input data given by fprdata.dat are imported and column of the imported data are assigned to [v], [a] respectively. The values given in the problem are Vs = 2V, R=14.2Ω t = 0.6 s, L = 0.98 H. The h1 and h2 values used in the code are time intervals. h1=0.025, h2=0.0025. Since h = (b-a) / n, the intervals are n1 = 24 n2 = 240.

topsa, topv, toplna, topv2 and topvlna these allows us to find coefficients for least square (kata and katb). The current function obtained from least square is defined as id@(ger). The Vd voltage function is created by taking the logarithm of this id function and defined as vd@(aki).

Arrays in which current and voltage values of tvd1, tvd2 and tid1, tid2 are kept.
Arrays in which the times san1, san2 and say1, say2 are kept.

Method solutions are graphed with the "plot" command and these graphics are outputs of this code. There are 5 graph at the end of the code. These are Diode Voltage/Time graph, Main Current/Time graph, Inductance Voltage/Time graph, Resistor Voltage/Time graph, Fitting graph.

3- Project Result and Analysis

Comment and comparisons will be evaluate in this section.

A-Result

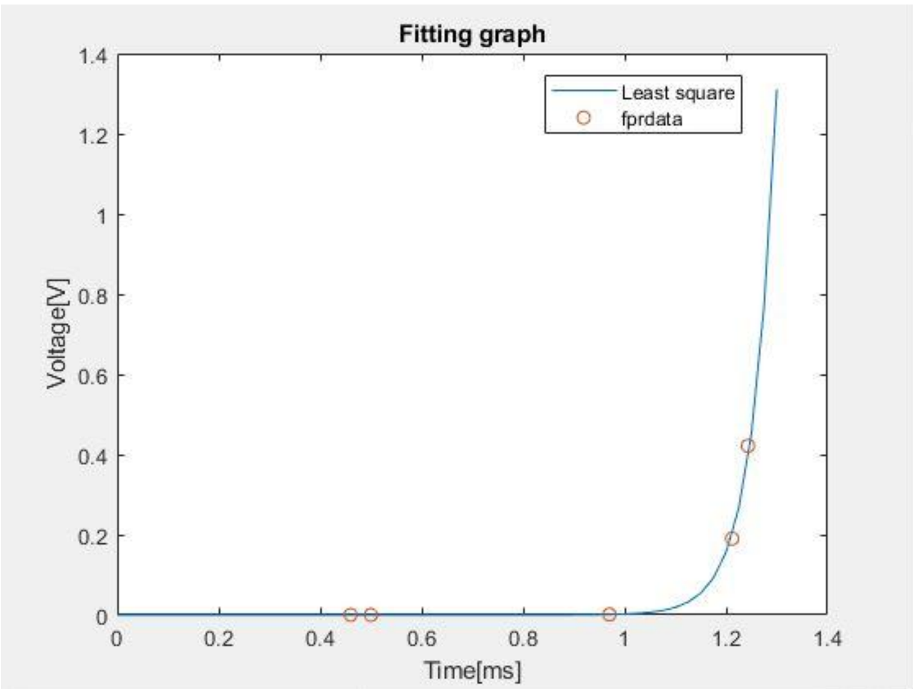


Figure 1. Fitting graph for fprdata.dat

An exponential polynomial is fitted for each point given in the fprdata file. Because the number of points in the fprdata file is low, there was less error in the fit. Errors are indicated in the analysis section.

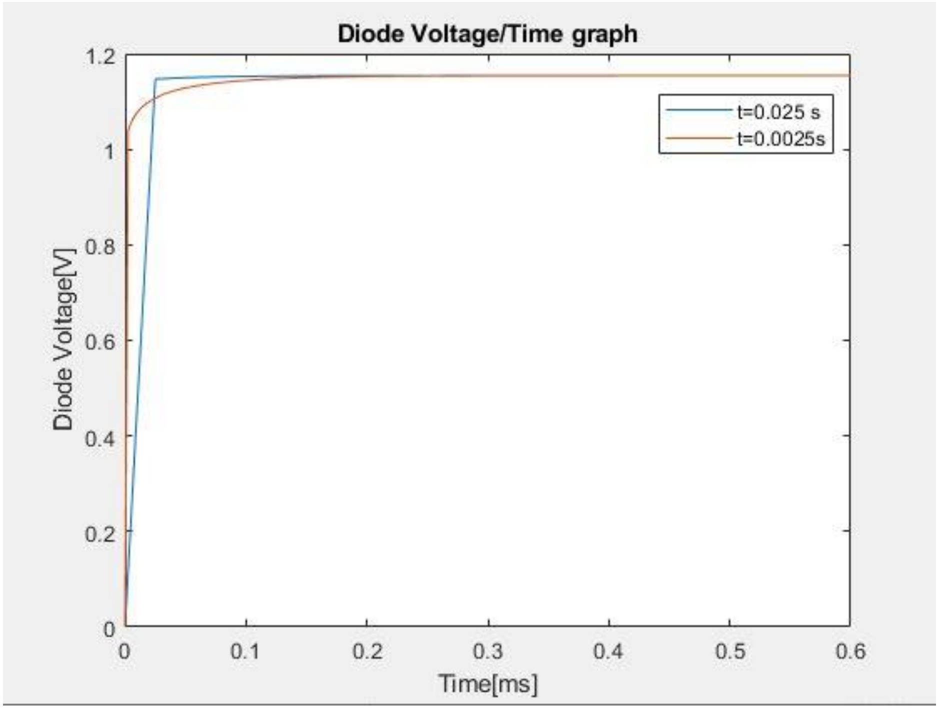


Figure 2 Diode voltage time graph (Vd)

Since the voltage on the diode depends on the current, the change in current has given us a voltage time graph in this way. The graph peaks faster where the current change is faster.

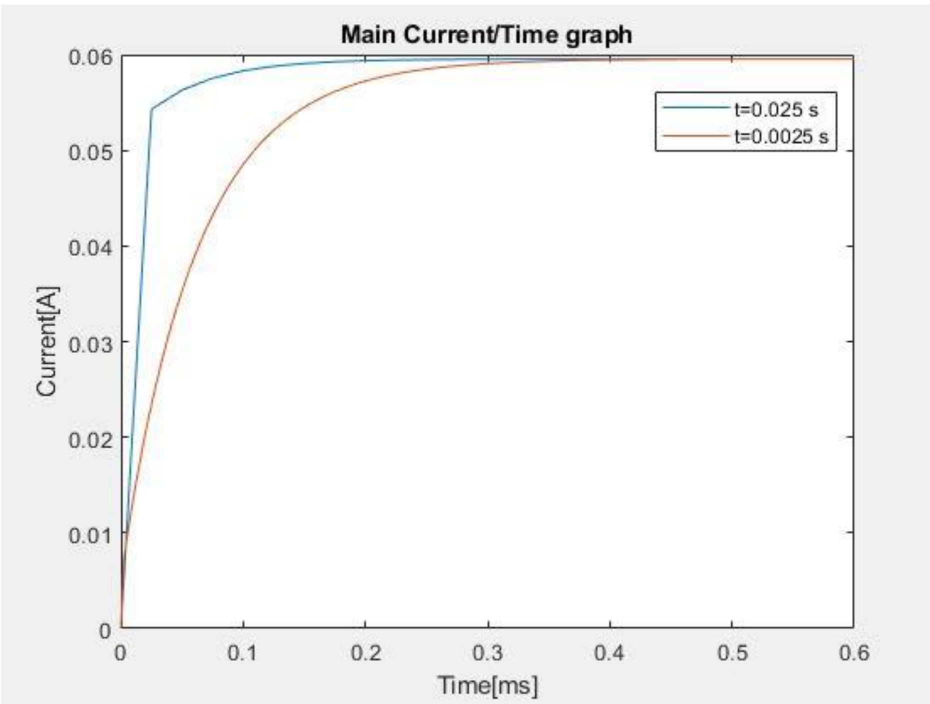


Figure 3 Main Current time graph (I)

The main current is determined by the level of electron passing on diode ,resistance and inductance. Since the current in the inductance does not change rapidly, it takes time for the current in the circuit to flow continuously. This time and change can be observed on the chart.

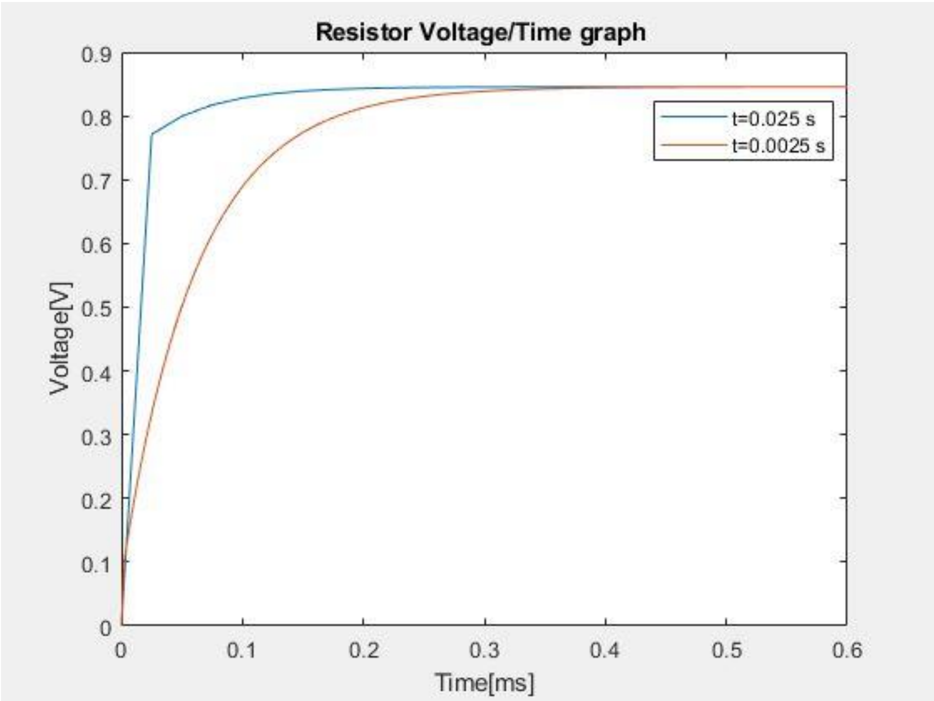


Figure 4 Resistor voltage time graph (V1)

The voltage on the resistor depends on the current, it changes rapidly or slowly according to the changes in the current.

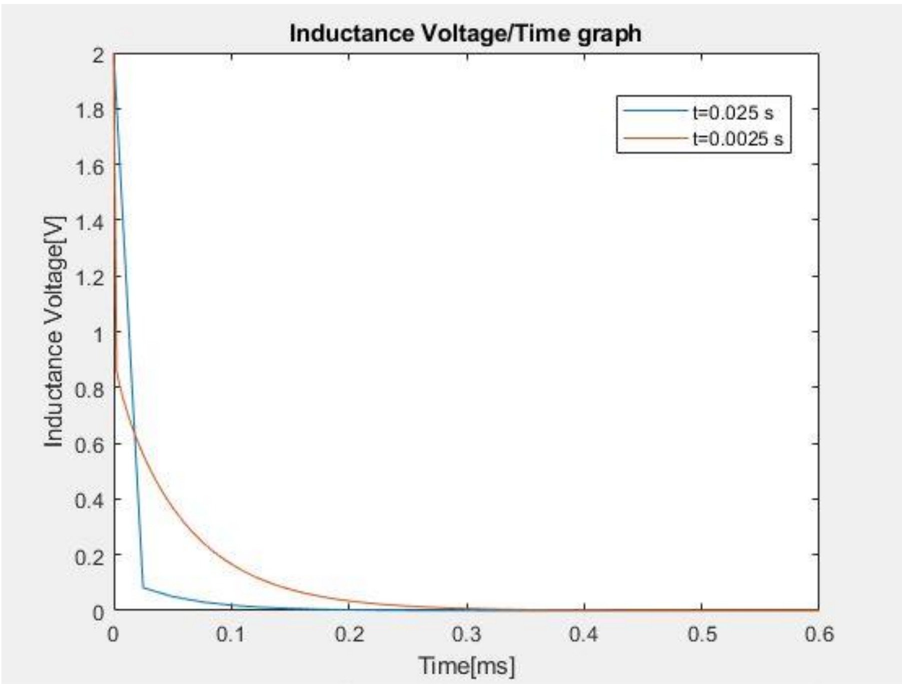


Figure 5 Inductance voltage time graph (V2)

Inductance voltage depends on the current change. Since the inductance current does not change rapidly, these graphs are obtained for these seconds values.

B-Analysis

In Euler method The Initial Value Problem variables for $\Delta t = 0.025$ ms and $\Delta t = 0.0025$ ms are shown in Figures 2,3,4,5

In this project, the Euler method was used to solve the next value of the current with the initial value problem. The first reason for choosing the euler's method is that it is easy and comfortable to apply. When applying the euler method, $w_0 = h.Vs / L$ because the value of w for the first step is 0 at the time of t = 0.

Could another method be used instead of the Euler's method? Yes it could be used. It would be better to use the Runge kutta order n method, but the faster and easier euler method was chosen because it could be difficult and time consuming to apply.

Considering the first two term of the Taylor series (values up to the 1st derivative), the Euler's Method is found. This method converging with the error order of "O(h2)".

The data in the fprdata file is fitted to the exponential function. Since there are 5 values in this fit and there are measurement noises in the values, there were some error deviations. We can see this fittings in figure 1.

In order to see the error deviations, we can observe the error amount by taking the difference from the function value at time t in the measurement value.

Could another method be used instead of the least square method? Yes, it could be used. With Lagrange interpolation method, many polynomials could be fitted and the correct result could be achieved. However, this method was not preferred because it is difficult to apply for an exponential function.

$ln\, y = ln\, b + ax$ is obtained by taking the logarithm of the $y = be^{a\,x}$ function. In this function, leaving the x value alone, the following function is obtained for the diode;

$$x = (lny - lnb)/a$$

4- Appendix Section

Matlab code is used to see the values in the graph. The arrays in which the values are kept are learned from the code description section.

CODE

```
clear
clc
%definition values
load fprdata.dat
v(:,1)=fprdata(:,1); a(:,1)=fprdata(:,2);
lnv=zeros(5,1);
lna=zeros(5,1);
L=0.98 ; Vs=2; R=14.2; n1=600/25; h1=25/1000; n2=600/2.5; h2=2.5/1000;
for i=1 : length(v)
   lnv(i,1)=log(v(i,1));
   lna(i,1)=log(a(i,1));
end
%least square
topa=0; topv=0; toplna=0; topv2=0; topvlna=0;
for i=1 : length(v)
   topa=a(i,1)+topa;
   topv=v(i,1)+topv;
   toplna=lna(i,1)+toplna;
   topv2=v(i,1)^2+topv2;
   topvlna=v(i,1)*lna(i,1)+topvlna;

end
kata=(length(v)*topvlna-topv*toplna)/(length(v)*topv2-topv^2);
katlnb=(topv2*toplna-topvlna*topv)/(length(v)*topv2-topv^2);
katb=exp(katlnb);

%functions
id = @ (ger)  katb*exp(kata*ger);
vd = @ (aki) (log(aki)-log(katb))/kata;

%euler method
w1=zeros(n1+1,1);
w2=zeros(n2+1,1);
tvd1=zeros(n1+1,1); tvd2=zeros(n2+1,1);
tvd1(1,1)=0; tvd2(1,1)=0;
tid1=zeros(n1+1,1); tid2=zeros(n2+1,1);
tid2(1,1)=0; tid2(1,1)=0;
w1(1,1)=h1*Vs/L;
w2(1,1)=h2*Vs/L;
san1=0 : h1 : 0.6;
san2=0 : h2 : 0.6;
for i=1 : n1
  w1(i+1,1)= w1(i,1)+h1*((Vs-vd(w1(i,1))-w1(i,1)*R)/L);
  tvd1(i+1,1)=vd(w1(i,1));
  tid1(i+1,1)=w1(i+1,1);
end
for i=1 : n2
  w2(i+1,1)= w2(i,1)+h2*((Vs-vd(w2(i,1))-w2(i,1)*R)/L);
```

```
  tvd2(i+1,1)=vd(w2(i,1));
  tid2(i+1,1)=w2(i+1,1);
end
say1=0 :h1 : 1.3;
say2=0 :h2 : 1.3;
%graphs
figure
plot(san1,tvd1)
hold on
plot(san2,tvd2)
xlabel('Time[ms]')
ylabel('Diode Voltage[V]')
title('Diode Voltage/Time graph');
legend('t=0.025 s','t=0.0025s')
figure
plot(san1,tid1)
hold on
plot(san2,tid2)
xlabel('Time[ms]')
ylabel('Current[A]')
title('Main Current/Time graph');
legend('t=0.025 s','t=0.0025 s')
figure
plot (san1,Vs-tvd1-tid1*R)
hold on
plot (san2,Vs-tvd2-tid2*R)
xlabel('Time[ms]')
ylabel('Inductance Voltage[V]')
title('Inductance Voltage/Time graph');
legend('t=0.025 s','t=0.0025 s')
figure
plot(san1,tid1*R)
hold on
plot(san2,tid2*R)
xlabel('Time[ms]')
ylabel('Voltage[V]')
title('Resistor Voltage/Time graph ');
legend('t=0.025 s','t=0.0025 s')
figure
plot(say1,id(say1))
xlabel('Time[ms]')
ylabel('Voltage[V]')
title('Fitting graph');
hold on
plot(v,a,'o');
legend('Least square','fprdata')
```