ENGR 122 Laboratory Instructions 2018

Lab 3: The Resistor Diode Circuit Meets Newton Raphson Zeros

3.1 Aim

This lab exercise is designed to help you learn to use a high-level programming environment (Python) to solve mathematical problems of the sort design engineers encounter quite regularly.

Include your Python work in your answers.

3.2: Newton-Raphson and the Resistor-Diode Circuit

3.2.1 Newton-Raphson

Recall from lectures and the textbook the Newton-Raphson method for finding zeros of functions. Essentially the idea is that you guess at the zero. Perhaps you have some idea where the zero might be on physical grounds, for example. Let's call your guess x_1. The N-R method then calculates and improved estimate x_2 of the zero using the formula

$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)} \qquad \text{Equation 1}$$

A challenge problem asks you to derive this formula, but for now we will just use it.

A Simple Example

Consider the function

$$f(x) = x^2 - 7\text{x} + 10 \qquad \text{or} \qquad f(x) = (x-2)(x-5)$$

This has zeros $x = 2$ and $x = 5$. But let's try the Newton-Raphson method on this, first by hand, and then in Python. This should give us confidence in the method. Let's guess $x_1 = 4.0$ as a zero.

First $\quad f'(x) = 2x\text{-}7$

So we have

$$x_2 = x_1 - \frac{x_1^2 - 7x_1 + 10}{2x_1 - 7}$$

Core 1: Complete the table below (10 marks)

<><><><><><><><><><><><><><><><><><><><><><><><>

Estimate	Refined estimate
4.0	$4.0 - \frac{-2.0}{1.0} = 6.0$
6.0	$6.0 - \frac{4.0}{5.0} = 5.2$
5.2	$5.2 - \frac{0.64}{3.4} = 5.0118$
5.0118	$5.0118 - \frac{0.0354}{3.0235} = 5.00009$

<><><><><><><><><><><><><><><><><><><><><><><><>

3.2.2 Using Python for Newton-Raphson

Copy and paste the following code into Python. Note: 2 is not the same thing as 2.0. If you type in 2, it will be interpreted as an integer while 2.0 will be interpreted as a floating point number. Also, remember that you have to use ***ctrl-shift-v***, not *ctrl-v* to paste into Python.

```
x1=4.0
x2 = x1-(x1*x1-7.0*x1+10.0)/(2.0*x1-7.0)
print x2
```

You should get 6.0 as above. Now copy and paste

```
x1=6.0
x2 = x1-(x1*x1-7.0*x1+10.0)/(2.0*x1-7.0)
print x2
```

You should get 5.2 as above, and so on.

Well, that's tedious. There must be some way to get a loop going, right? Yes, there is. Try the following code. The indents (tabs) are important and so is the line between x_1=x_2 and print("done"). The tabs tell Python that the line is part of the loop, and the blank line without a tab tells Python the loop has ended.

```
x1=4.0
for I in range(0,8):
	x2 = x1-(x1*x1-7.0*x1+10.0)/(2.0*x1-7.0)
	print "%.4f" % x2
	x1=x2

print("done")
```

Note the code "%.4f" % just formats the printing of the number so that four digits past the decimal are included. You will need these kind of formatting statements frequently so experiment a bit it if you are not familiar with the notation.

Note *N-R* got to the zero with great accuracy very quickly in this case.

Core 2: Copy and paste your results. Annotate the code, explaining what each line does briefly, and also count the number of steps iterations and comment on how that relates to the range(0, 8). (10 marks)

<><><><><><><><><><><><><><><><><><><><><><><>

```
6.0000
5.2000
5.0118
5.0000
5.0000
5.0000
5.0000
5.0000
done
```

```
x1 = 4.0 #original x estimate

for I in range(0,8): # repeat everything inside the for loop 8 times
    x2 = x1-(x1*x1-7.0*x1+10.0)/(2.0*x1-7.0)  # the new estimate for x
    print "%.4f" % x2 # print out the new x estimate
    x1 = x2 # set the old estimate equal to the new one.

print("done")

```

The range(0,8) means that the loop will begin at the integer value 0, and step by 1 each time through till it reaches 8.

<><><><><><><><><><><><><><><><><><><><><><><>

Improving the code

We used a for loop with eight iterations. How many iterations do we need? Of course that will depend on the function we are trying to find the zeros of. We could just set the number of steps in the for loop very high and then exit early if we reach a specified accuracy as in the code below.

```
x1=4.0
DeltaMin=0.02
for I in range(0,100):
    x2 = x1-(x1*x1-7.0*x1+10.0)/(2.0*x1-7.0)
    print "%.4f" % x2
    Delta = abs(x2-x1)
```

```
    if Delta < DeltaMin:
      break
    x1=x2

print("done")
```

Core 3: Copy and paste your results. Annotate the code with comments explaining how it works. Just a few lines will be sufficient. (10 marks)

<><><><><><><><><><><><><><><><><><><><><><><>

```
6.0000
5.2000
5.0118
5.0000
done
```

```
x1 = 4.0 #original x estimate
DeltaMin = 0.02 #minimum difference between the old estimate and the new one

for i in range(0,100): # repeat the loop 100 times
    x2 = x1-(x1*x1-7.0*x1+10.0)/(2.0*x1-7.0) # calculate the new x estimate
    print "%.4f" % x2
    Delta = abs(x2-x1)
    if Delta < DeltaMin:# if the difference between the two estimates is less then delta min, break the loop
      break
    x1 = x2

print("done")
```

<><><><><><><><><><><><><><><><><><><><><><><>

Import exp

We will now need to use the e^x function and make graphs, so import maths and plotting:

```
import numpy as np
import matplotlib.pyplot as plt
```

Note: In Python e^x is given by np.exp(x), and x^2 is given by x**2 (not x^2).

Completion 1: (10 marks)

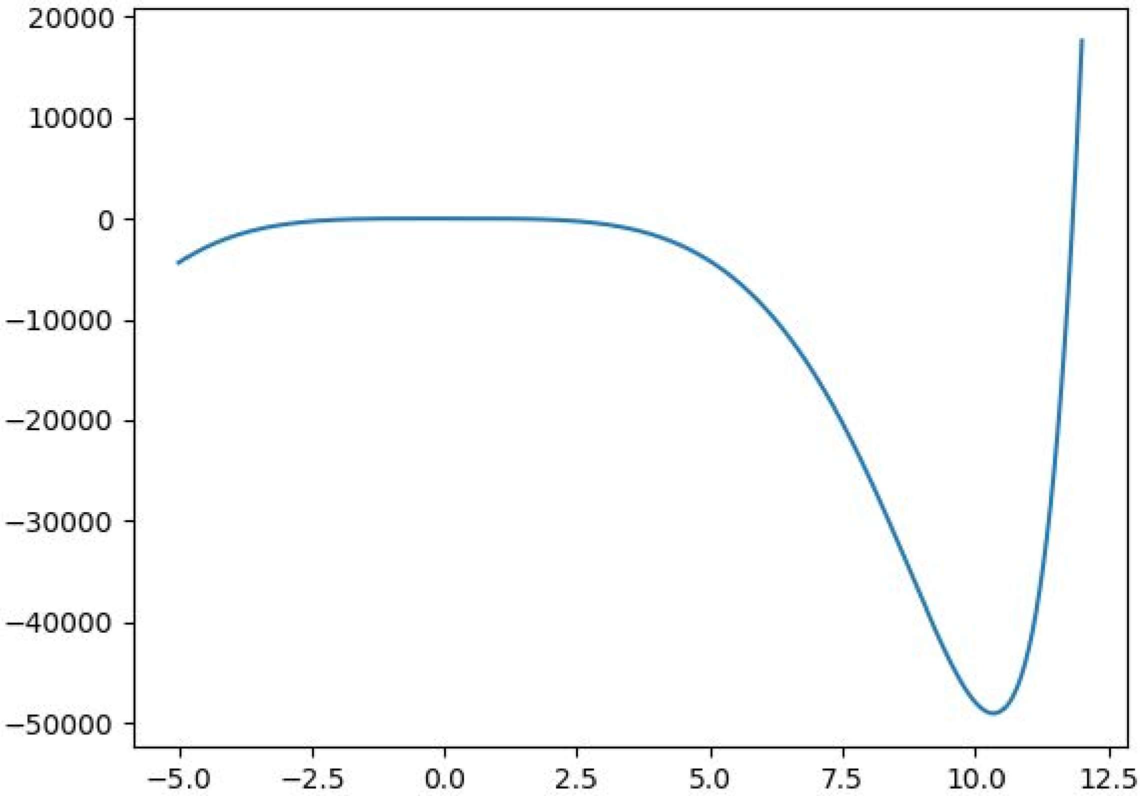
Consider the function $f(x) = e^x - 7x^4$. Use Python to make a rough graph of this function from $x = 0$ to $x = 20$ and find out where the zeros are approximately. You can do this by finding a value x_1 where the function is positive and x_2 where the function is negative, and then taking the average of x_1 and x_2. You should find zeros somewhere between 0 and 1 and somewhere around 12 roughly.

Plotting Hints: to make a plot in python, make an array of x values, e.g., x=np.linspace(0,20,100) makes an array of 100 values of x from 0 to 20.

Next, define y. For example y = 17*x+23 would give you a line.

Then make your plot: plt.plot(x,y) and then plt.show(). Note you have to close the plot before you can enter commands into python via the terminal.

<><><><><><><><><><><><><><><><><><><><><><><>



This curve has roots at around -0.54, 0.73, and 11.7

<><><><><><><><><><><><><><><><><><><><><><><>

Completion 2: (10 marks)

We now know roughly where to find a couple of zeros of $f(x)=e^x-7x^4$ **.** Find one of the zeros **to an accuracy of 0.001.**

<><><><><><><><><><><><><><><><><><><><><><><>

$\frac{dy}{dx} = e^x - 28x^3$

```python
x1 = 10.0
deltaMin = 0.001

for i in range(0,100):
    x2 = x1 - (np.exp(x1)-(7*(x1**4)))/(np.exp(x1)-(28*(x1**3)))
    print("%.4f" % x2)
    if(abs(x2 - x1) < deltaMin):
        print("done")
        break
    x1 = x2
```

```
1.9690
1.4943
1.1521
0.9209
0.7906
0.7450
0.7397
0.7397
done
```

<><><><><><><><><><><><><><><><><><><><><><><>

3.2.3 A Resistor – Diode Circuit…

… But first some background.

Linear Functions

A linear function is one for which

$f(x_1+x_2)=f(x_1)+f(x_2)$ Equation 2

For example the function $f(x)=12x$ is linear. To illustrate this consider $x_1=2$ and $x_3=3$. We have $12*(2+3) = 12*2+12*3$.

Core 4: Are Lines Linear? (10 marks)

The function $f(x)=5x+2$ describes a line, but is it linear? Explain.

<><><><><><><><><><><><><><><><><><><><><><><>

It is not linear, this is because if we were to do $f(x_1)+f(x_2)$, we would get two groups of +2, meaning there would be a total of +4. Because of this any answer taken using $f(x_1)+f(x_2)$ will always be 2, larger than an answer taken with $f(x_1+x_2)$.

<><><><><><><><><><><><><><><><><><><><><><><>

Ohm's Law is Linear in the Voltage

For convenience let's write Ohm's Law as

$$I = \frac{V}{R}.$$

Note then that $I(V_1+V_2)=I(V_1)+I(V_2)$, so Ohm's law is linear in the voltage.

Completion 3: Interpreting the Maths (5 marks)

Interpret the result that Ohm's Law is linear in the voltage physically. Hint: two batteries in series.

<><><><><><><><><><><><><><><><><><><><><><><>

Because Ohms Law is linear, this means that is you were to put two batteries (EMF's) in series, the total voltage would be additive. E.g. putting a 3V and a 9V battery in series would total a 12V EMF. Because of this we know that the currents within a circuit are additive as well.

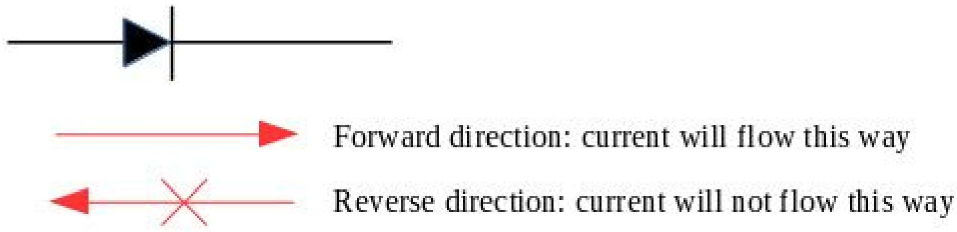
<><><><><><><><><><><><><><><><><><><><><><><>

Non-Ohmic Devices

Not all circuit elements obey Ohm's Law. Devices that do not are called non-Ohmic or non-linear. There are many examples. Even an incandescent light bulb is non-linear. If you double the voltage the current does not double because the filament gets hotter and its resistance increases.

Diodes

Diodes are very far from linear. As you probably know, a diode is said to allow current to flow in one direction but not the other as indicated in the diagram below.



But there is more to it than that.

1) If a voltage is applied to drive a current in the reverse direction (a “reverse bias”), the current is not really zero. But it is small.

2) If a small forward bias is applied (say 0.1 volts), only a very small current will flow. But as the forward bias is increased, the diode “opens up”, often at about 0.6 Volts (for standard silicon diodes) and a large current will flow. As the voltage is increased further the current increases greatly.

Core 5: Diode Data for the 1N4148 Diode (5 marks)
We will use 1N4148 diodes. These are extremely common, “garden variety” diodes found in many devices. Get current versus voltage curves for your diode in the forward and reverse directions from the data sheet and put it here. Does the data agree with statements 1 and 2 above? Explain.

<><><><><><><><><><><><><><><><><><><><><><><>

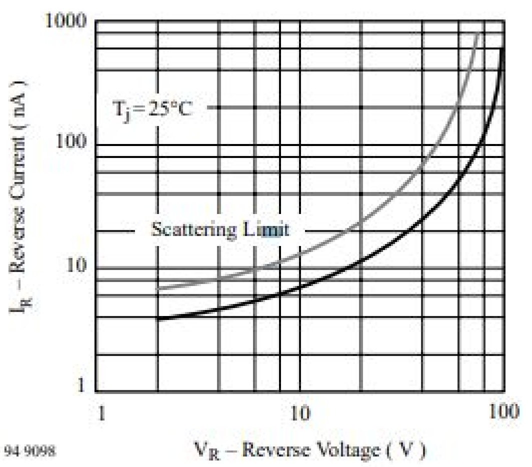


Figure 4. Reverse Current vs. Reverse Voltage

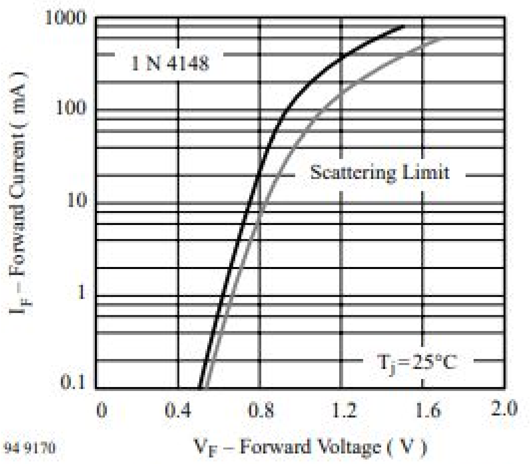


Figure 2. Forward Current vs. Forward Voltage

Yes these curves agree with the above statements. With the reverse voltage curve we can see that as a reverse voltage is applied to the diode there is a very small reverse current (in the nano-Amps).

The forward voltage curve also agrees with statement 2, with very small voltage jump (0.4-0.8 volts) there is an increase in forward current of 10mA. But with an increase in voltage from 0.8-1.2V there is an increase in current of 300mA.

<><><><><><><><><><><><><><><><><><><><><><><><

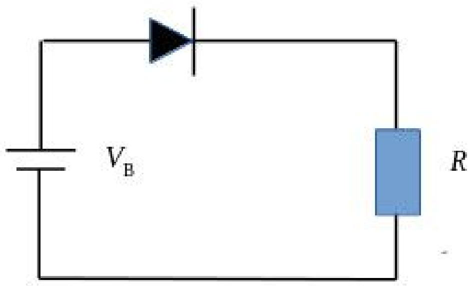
The following material is adapted from your textbook, Engineering Mathematics …, Croft et al, pages 403-404.

For a forward bias the current is given approximately by

$I = I_0(e^{aV} - 1)$ Equation 3

where V is the voltage across the diode, I_0 is a constant called the reverse bias saturation current, and a is another constant for the diode. These constants vary from one diode to another and also vary with temperature. For example, I_0 might be on the order of nanoamps (nA, 10^{-9} A) but can vary substantially. For our diode, I_0 is about 6.1 nA and the constant a for our diode is about 19.5 V^{-1} at a temperature of 300 K

In a circuit with a resistor and a diode in series as shown below, what current do we expect to flow? Let's find out.



Starting at the bottom of the battery and looping we have

$$V_B - V_d - iR = 0$$

where V_B is the battery voltage and V_d is the voltage across the diode. Since the resistor and diode are in series, the resistor current and diode current are the same. So we can plug our expression for the diode current into the loop equation and get

$$V_B - I_0 R(e^{aV_d} - 1) - V_d = 0 \qquad \text{Equation 4}$$

Completion 4: (10 marks)

For our diodes I_0 = 6.2x10^{-9} A and a = 19.5 V^{-1}. and we will use 4700 Ohm resistors and 5 V power supplies. Find the voltage across the diode and the current through the resistor. Use Newton-Raphson on Python. Hint: let the expression above be called $f(V_d)$ and find its zeros.

<><><><><><><><><><><><><><><><><><><><><><><>

```
for i in range(0,10000):
    x2 = x1 - ((5-(IR*(math.exp(19.5*x1)-1))-x1)/(((-IR*19.5)*(math.exp(19.5*x1)))-1))
    print("%.4f" % x2)
    if(abs(x2 - x1) < deltaMin):
        print("done")
        break
    x1 = x2
```

```
0.9487
0.8975
0.8464
0.7956
0.7457
0.6981
0.6562
0.6262
0.6133
0.6114
0.6114
done
```

If the voltage across the diode is 0.6114, then the voltage across the resistor must be 4.3886. Because of this we can calculate that the current through the resistor is 9.34x10^{-4} amps.

<><><><><><><><><><><><><><><><><><><><><><><>

Completion 5**:** (10 marks)
Connect the circuit and measure the current. Is it reasonably close?
<><><><><><><><><><><><><><><><><><><><><><><>
The measured voltage across the diode was 0.614V, and the voltage across the resistor was 4.43V. This is reasonably close the the theoretical voltages of 0.6114 across the diode, and 4.3886 across the resistor.
<><><><><><><><><><><><><><><><><><><><><><><><

Challenge 1: (10 marks)
Derive the Newton-Raphson formula (equation 1). Diagrams and explanations will be needed.

<><><><><><><><><><><><><><><><><><><><><><><>

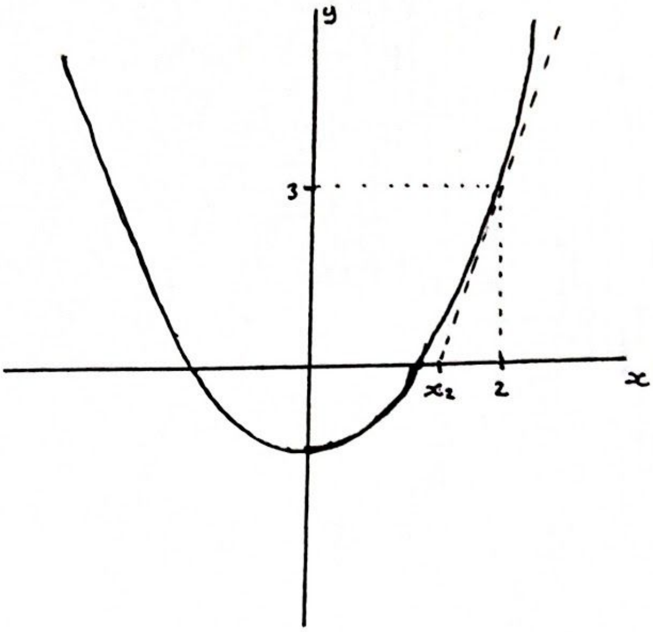
$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)}$$

$$f(x) = x^2 - 1$$

$$x_1 = 2$$

$$f(2) = 3$$

$$f'(2) = 4$$



The equation of the line tangent to f(x) at point 2 is

$$y - y_0 = m(x - x_0)$$

$$y - 3 = 4(x - 2)$$

x_2 is the point along this line where $y = 0$.

$$-3 = 4(x - 2)$$

$$-\frac{3}{4} = x - 2$$

$$2 - \frac{3}{4} = x$$

$$\boxed{x_2 = 2 - \frac{3}{4}}$$

here we see that $x_2 = x_1 - \dfrac{f(x_1)}{f'(x_1)}$