Boğaziçi University
The Faculty of Arts and Sciences
Physics Department

+5 LaTeX, +5 Python, -5x2 missing refs. to images, -5 missing access dates.
TOTAL: 95

Frank-Hertz Experiment

experiment

Student:
Gökhan Kebir

Group Members:
Gökhan Kebir
Sevim Açıksöz

Instructor:
Veysi Erkcan Özcan

Contents

1 Abstract

In this experiment the first excitation potential of gaseous mercury atoms was examined by the Franck-Hertz experiment.The values of the potential that we obtained were $\Delta V = 5.05 \pm 0.06$ and accepted value for mercury is 4.98V. We obtained this property from plots which we put our current and voltage values.Our error has been occurred because of equipment which used during experiment and some miscalculation finding peak points.

2 Introduction

The Frank-Hertz experiment was done by James Franck and Gustav Hertz in 1913.The Franck-Hertz experiment, performed in 1914, is an experiment for testing the Bohr Model of that atom. As a result of this experiment,atoms can only absorb a discrete amount of energy.Also, in their second paper about experiment ,they showed that the light emitted by excited mercury atoms had the same as the energy they gained from collisions with electrons. They were awarded the Nobel Prize for this work in 1925.

3 Theory

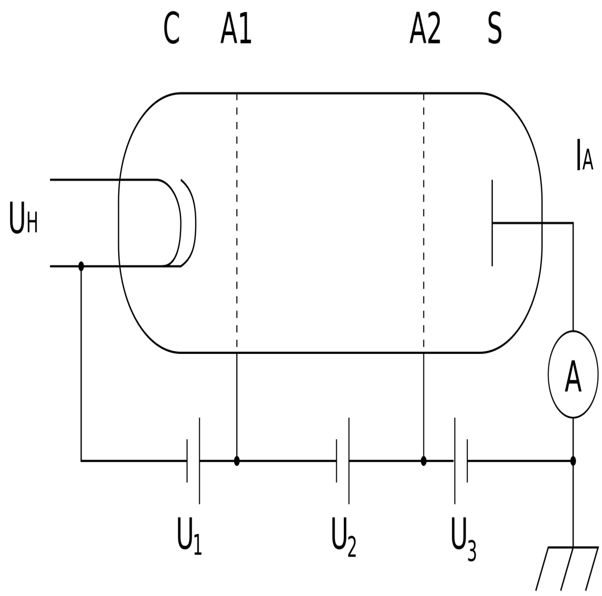


Figure 1: Frank-Hertz Tube

The most important part of the experiment is the Franck-Hertz tube which include small amount of Mercury. When tube is heated, liquid mercury turns into gas.The electrons are shot by an electron gun and U_1 potential helps to separate the electrons from gun.For this reason, We expect higher current in higher U_1 condition. Separated electrons are accelerated by U_2 potential which applied between A1 and A2.U_3 is added to prevent electrons without sufficient energy from reaching to anode.In higher U_3 condition current can be too low.

According to classical model , current will rise when U_2 potential is increased.However, Frank-Hertz obtained different result from this experiment.According to result of Frank-Hertz experiment current dropped when U_2 potential is multiples of $4.9eV$. This is the energy need to excite the electrons of the mercury atoms.Collision between mercury atoms and electrons is elastic when electrons energy smaller than multiples of $4.9eV$.The reason of elastic collision between mercury atoms and separated electrons is huge mass difference. The mass of electrons approx. 400.000 times smaller mercury atoms.thus the energy of electrons do not change because of these collisions but as soon as these electrons energies reach to 4.9 eV, then energy transfer occurs in these collisions.When electrons lose their energy in these collisions, they can not reach to the anode.

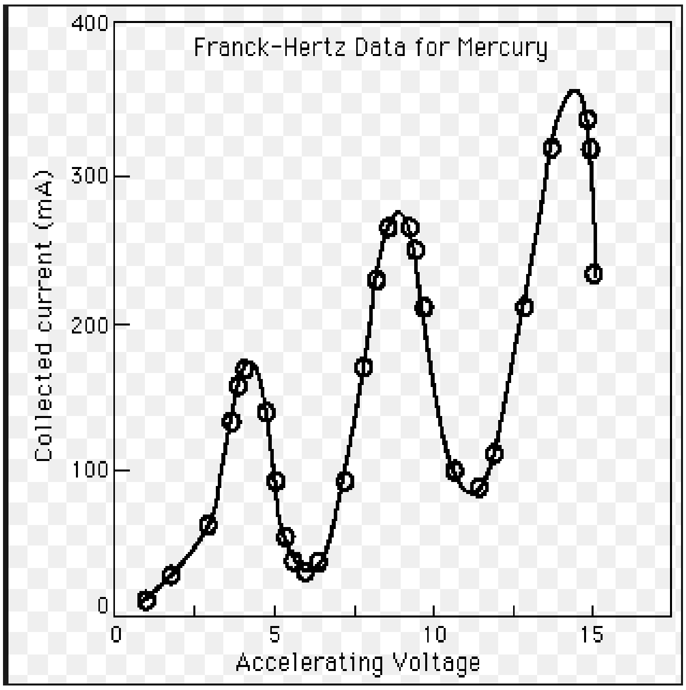


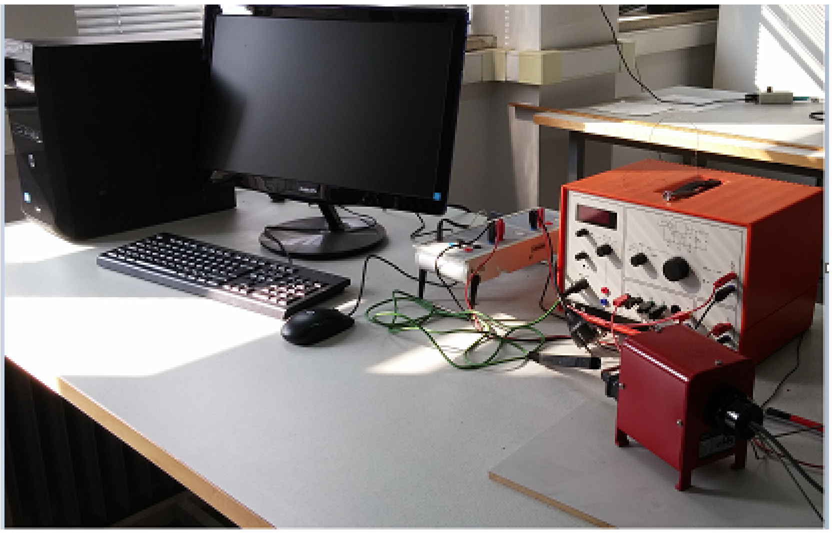
Figure 2: Frank-Hertz Data for Mercury

4 Experiment

4.1 Apparatus

- Franck-Hertz Tube
- Thermometer
- Electric Oven
- Digital Multimeter
- Computer
- Franck-Hertz Supply Unit

4.2 Setup



4.3 Procedure

- Set U_1 and U_3 voltage
- Start from 169 Celsius and increase temperature in steps of 5 C up to 210 Celsius.
- We start to take data by the computer and then we turn the knob to ramp up. We stop data taking from computer and turn the knob to reset back.

5 Data

5.1 Raw Data

Temp (°C)	$U_1(V)$	$U_3(V)$
169	1.95	4.12
175	1.95	3.04
181	1.95	3.05
185	1.95	3.05
189	1.95	3.05
194	1.95	3.05
200	1.95	3.05
205	1.95	3.05
210	1.95	2.55

Table 1: Temperature (°C) U_1 and U_3 values

5.2 Raw Data Plots

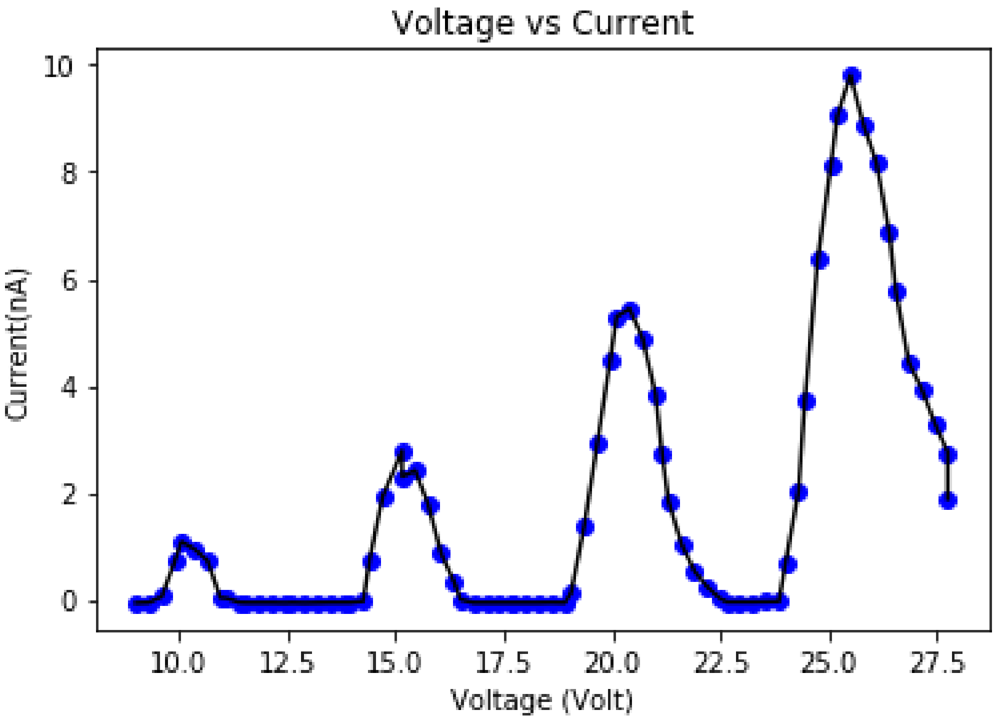


Figure 3: Accelerating potential vs Current plot for 169 °C

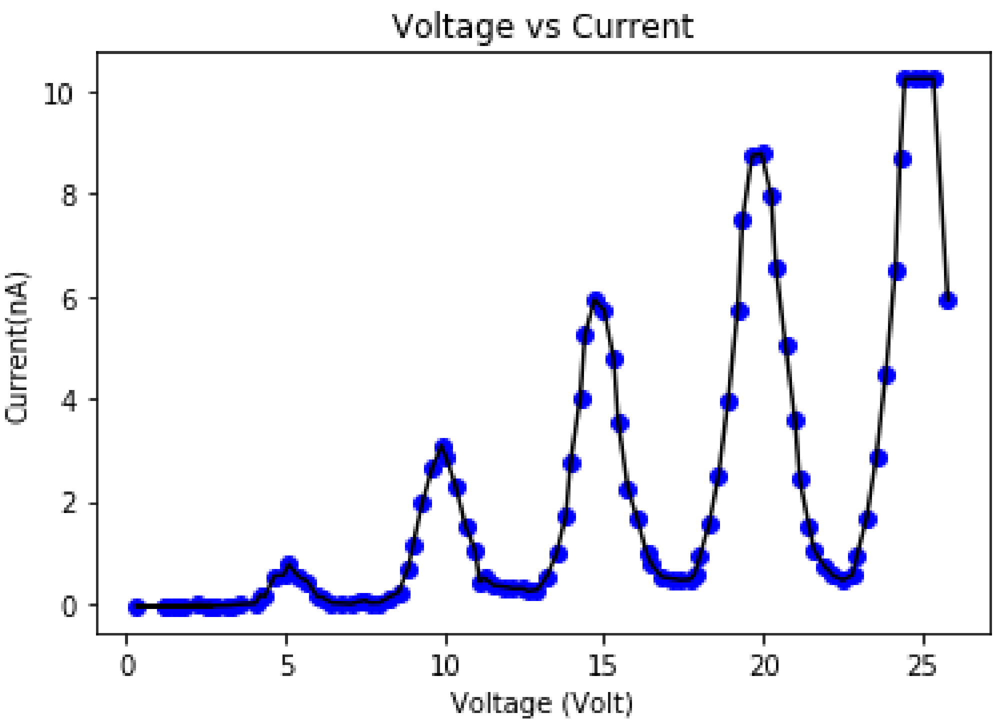


Figure 4: Accelerating potential vs Current plot for 175 °C

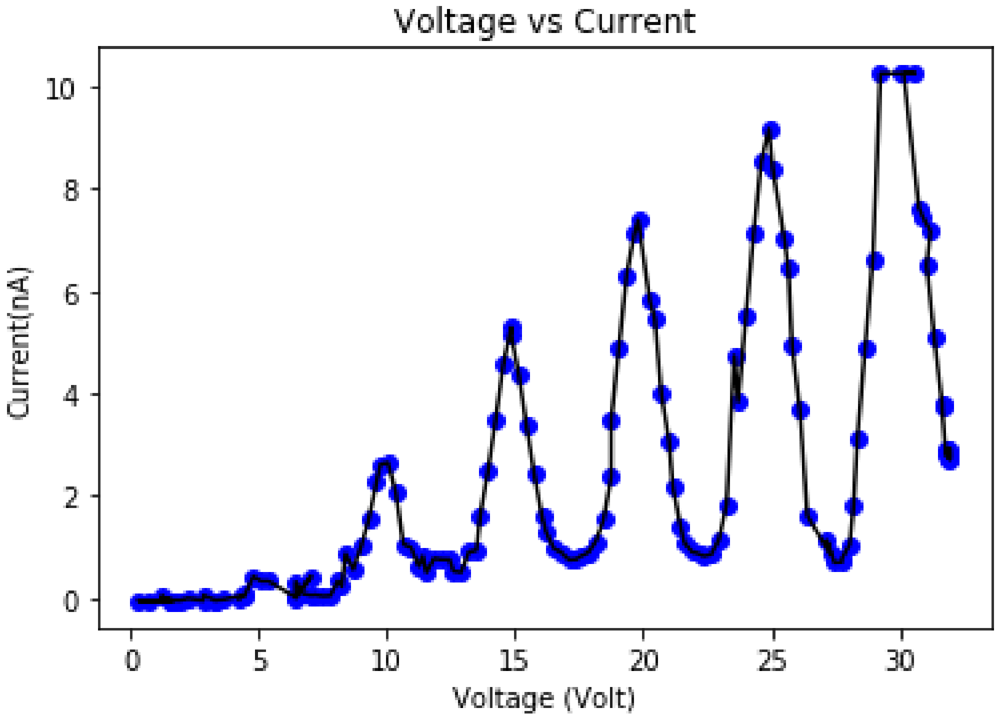


Figure 5: Accelerating potential vs Current plot for 181 °C

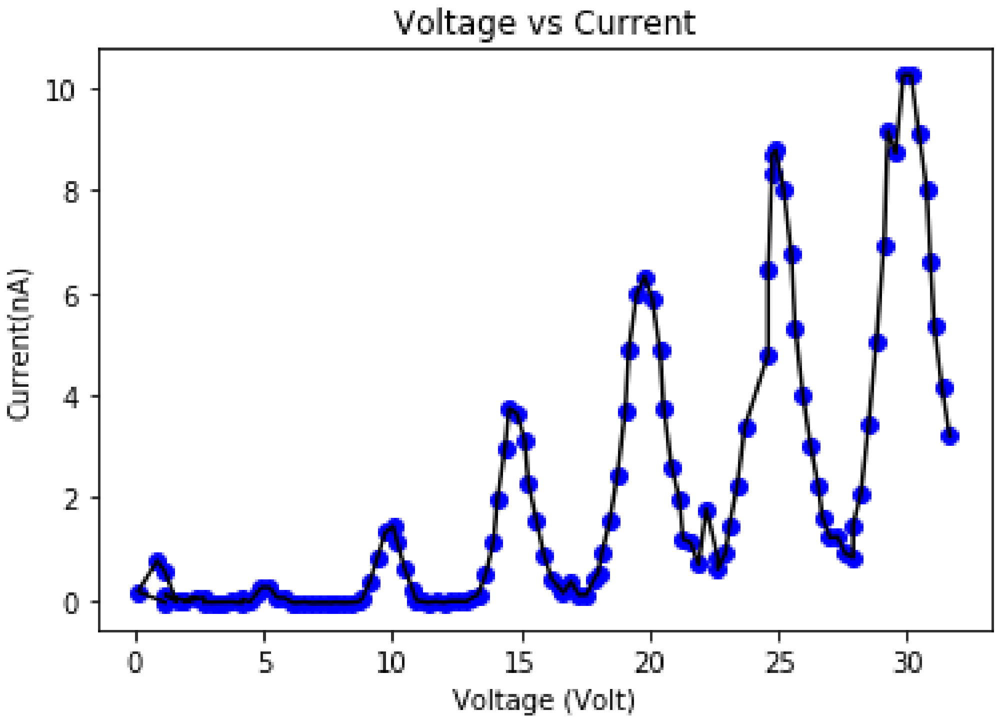


Figure 6: Accelerating potential vs Current plot for 184 °C

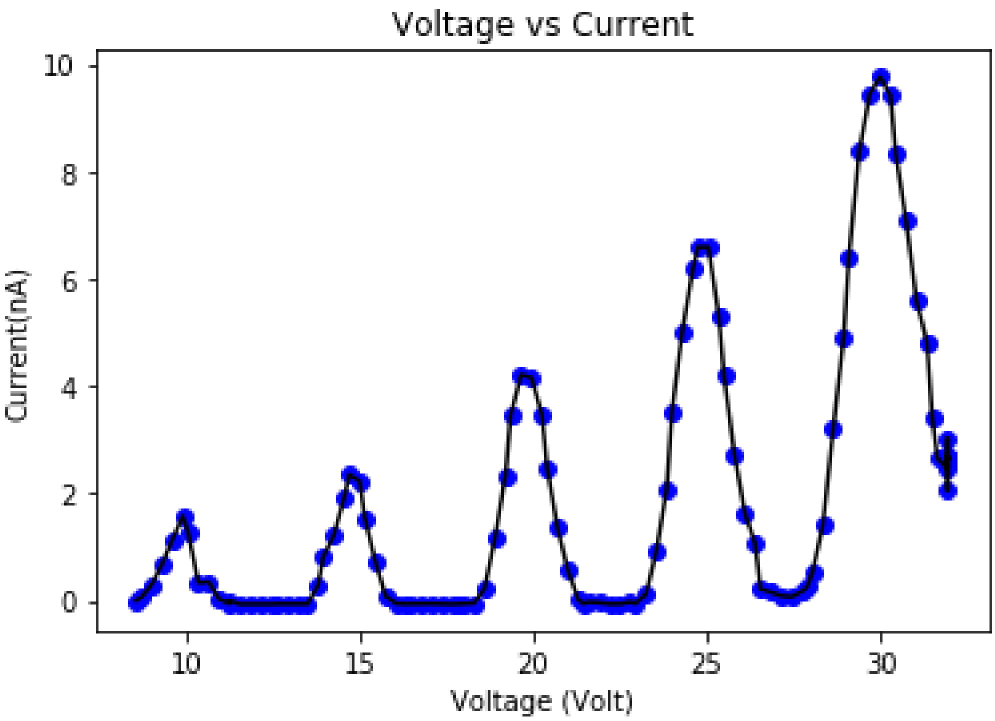


Figure 7: Accelerating potential vs Current plot for 189 °C

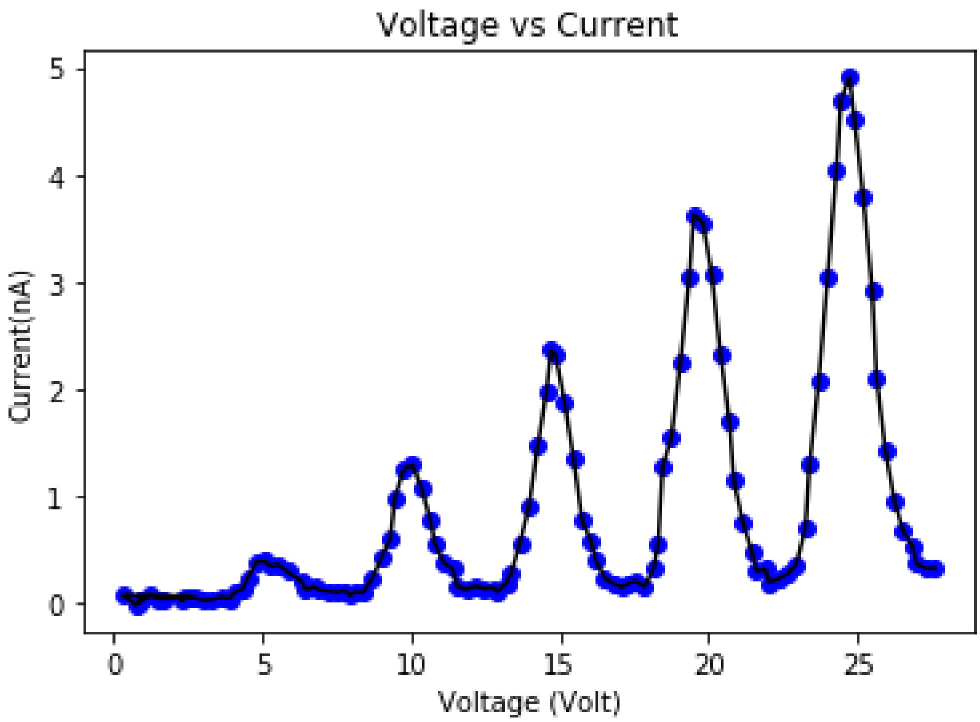


Figure 8: Accelerating potential vs Current plot for 194 °C

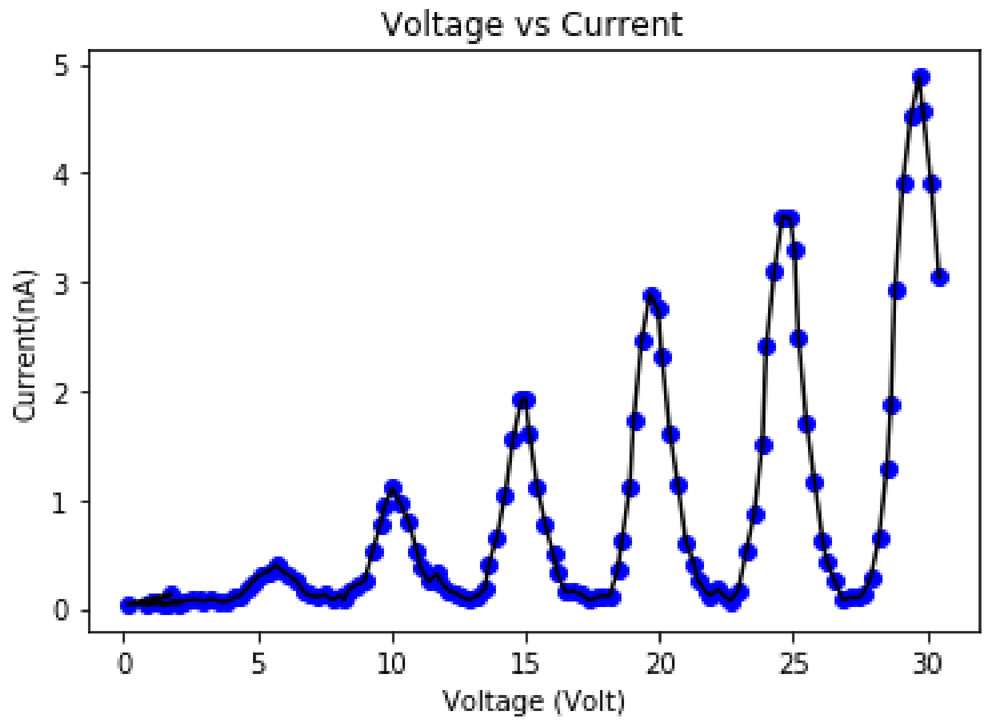


Figure 9: Accelerating potential vs Current plot for 200 °C

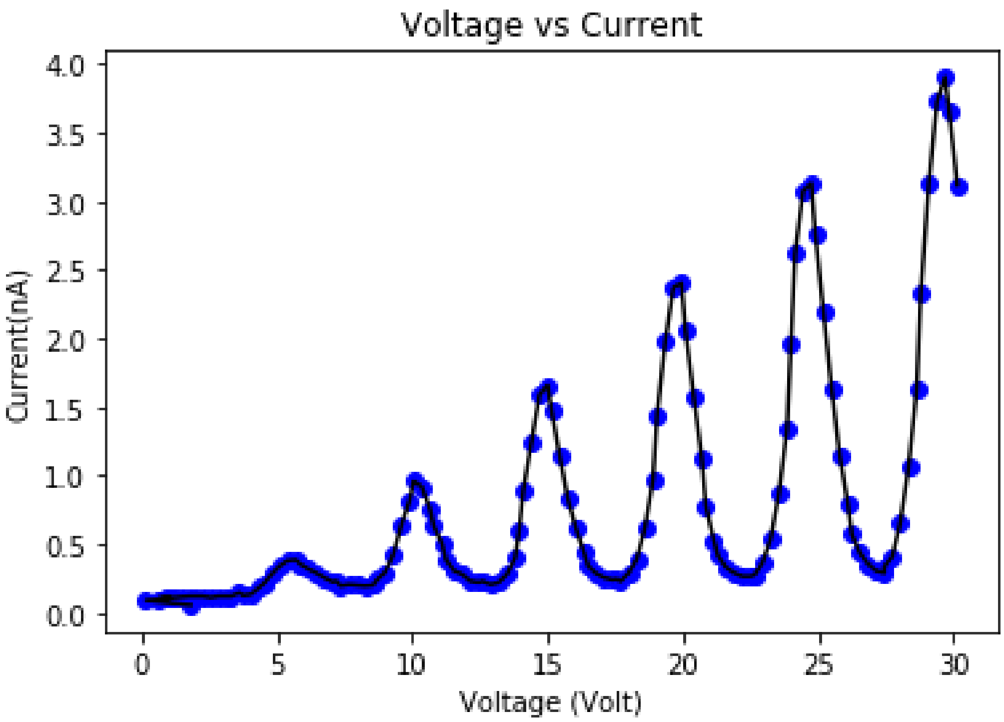


Figure 10: Accelerating potential vs Current plot for 205 °C

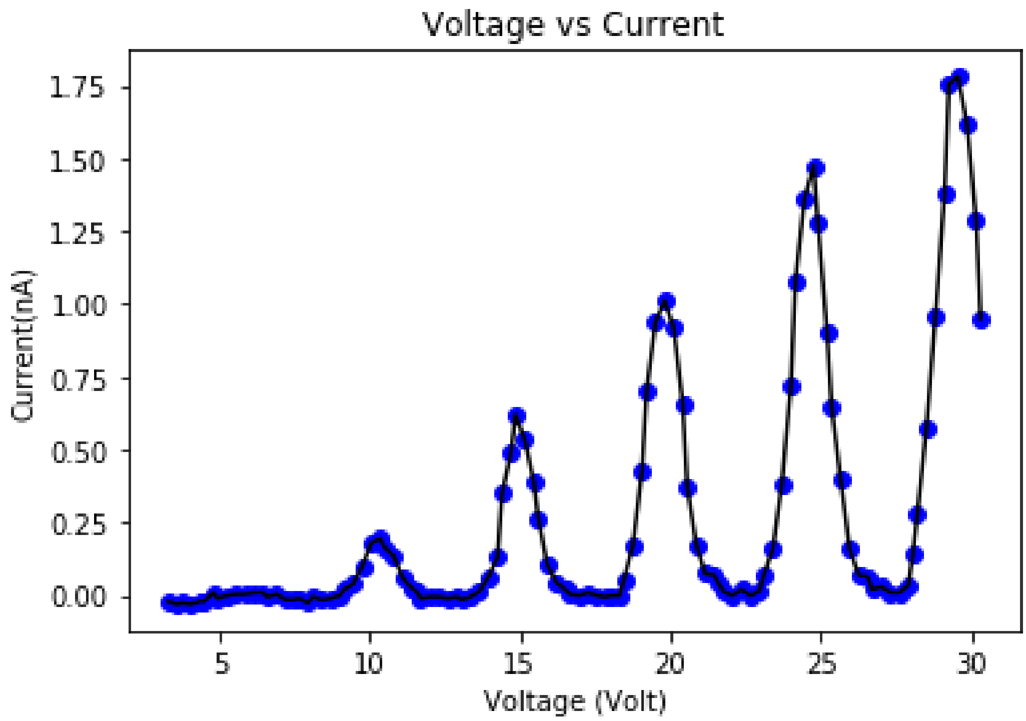


Figure 11: Accelerating potential vs Current plot for 210 °C

5.3 Data Analysis

The peaks arrived to the maximum energy electrons can have before inelastic collision begin to occur and just after this electrons almost have no any kinetic energy. After the inelastic collision , electrons were accelerated to the same kinetic energy as in the previous peak thanks to U_2 value. For this reason, to calculate the firs excitation energy of the mercury, we need to find the difference between peaks. By Using these voltage differences, we calculate the average V_{avg} and its standard deviation from the following formula;

$$\Delta V_{avg} = \frac{1}{N}\sum_{i=1}^{n} \Delta V_i \tag{1}$$

$$\sigma = \sqrt{\frac{1}{N-1}\sum_{i=1}^{n}(\Delta V_i - \Delta V_{avg})^2} \tag{2}$$

Temp (°C)	U_1	V_{avg}	σ^2
169	1.95	5.15	0.00
175	1.95	5.03	0.02
181	1.95	5.00	0.00
185	1.95	4.98	0.08
189	1.95	4.99	0.07
194	1.95	4.66	0.15
200	1.95	4.93	0.02
205	1.95	4.92	0.01
210	1.95	4.80	0.04

Table 2: Temperature (°C) U_1, V_{avg} and σ^2 values

After calculating these values that figured above, the last step will be calculating the weighted average. Weighted average is a must when the multiple instances of each voltage occurs because the process is taking the average of a data set which is already an average of another set. The formula used to calculated the weighted average is ;

$$\Delta V_{weighted} = \sum \frac{\Delta V_i}{\sigma_i^2} / \sum \frac{1}{\sigma_i^2} \tag{3}$$

$$\sigma_{weighted} = \sqrt{1/\sum \sigma_i^{-2}} \tag{4}$$

$$\Delta V_{weighted} = 5.05 Volt \tag{5}$$

$$\sigma_{weighted} = 0.06 \tag{6}$$

For this experiment , our true value is;

$$\Delta V_{true} = 4.98 Volt \tag{7}$$

6 Conclusion

Such experiment is prepared to observe that atoms can only absorb a discrete amount of energy . During this experiment, calculated the first excitation level of the mercury atom by Python is $\Delta V = 5.05 \pm 0.06$ The formula for comparison between the experimental value found during the experiment and the accepted value is below.

$$\Delta\sigma_V = \frac{|5.05 - 4.98|}{0.06} \tag{8}$$

$$\Delta\sigma_V = 1.1\ \sigma_V \tag{9}$$

Our result is close to real value but has some error.One possible reason of error for this experiment is the equipment which used during the experiment.The potential difference applied by Franck-Hertz Supply Unit may not be accurate.In addition to this Cathode tube can have slope. Even if the result is different from the theoretical value, in this experiment we observed that the collisions between electrons and mercury is elastic up to a certain point, where energy from electron is transfered to mercury, which excites mercury atom. This is a good evidence that the energy levels in atoms are quantized.

References

[1] Frank-Hertz Experiment $http://hyperphysics.phy-astr.gsu.edu/hbase/FrHz.html$

[2] Frank-Hertz Experiment $https://www.maths.tcd.ie/\ robinson/labs/franck_hertz.pdf$

[3] Constants `https://physics.nist.gov/cuu/Constants/`

[4] Advanced Physics Experiments, Erhan Gulmez, Bogazici University Publications, 1999, ISBN 975-518-129-6

[5] Frank-Hertz Experiment `https://astro.uni-bonn.de/ joachimi/fp/Franck-Hertz-Versuch.pdf`

7 Appendix

8 Codes

```
import matplotlib.pyplot as plt
import numpy as np
from scipy.optimize import curve_fit


from numpy import *
import pylab
import pandas

colnames = ['A1', 'B1']
data = pandas.read_excel("169.xlsx", names=colnames)
Current = data.A1.tolist()
Voltage = data.B1.tolist()
xval=np.array(Voltage)
yval=np.array(Current)

def local_maxima(xval, yval):
    xval = np.asarray(xval)
    yval = np.asarray(yval)

    sort_idx = np.argsort(xval)
    yval = yval[sort_idx]
    gradient = np.diff(yval)
    maxima = np.diff((gradient > 0).view(np.int8))
    return np.concatenate((([0],) if gradient[0] < 0 else ()) +
                          (np.where(maxima == -1)[0] + 1,) +
                          (([len(yval)-1],) if gradient[-1] > 0 else ()))
print(local_maxima(xval, yval))

plt.plot(Voltage,Current,'bo', label='data')
plt.plot(Voltage,Current,'k', label='data')
plt.title('Voltage vs Current')
plt.xlabel('Voltage (Volt)')
plt.ylabel('Current(nA)')
print(local_maxima(xval, yval))
plt.show()


from numpy import *
import pylab
import pandas
colnames = ['A1', 'B1']
data = pandas.read_excel("175.xlsx", names=colnames)
Current = data.A1.tolist()
Voltage = data.B1.tolist()
xval=np.array(Voltage)
yval=np.array(Current)
```

```
def local_maxima(xval, yval):
    xval = np.asarray(xval)
    yval = np.asarray(yval)

    sort_idx = np.argsort(xval)
    yval = yval[sort_idx]
    gradient = np.diff(yval)
    maxima = np.diff((gradient > 0).view(np.int8))
    return np.concatenate((([0],) if gradient[0] < 0 else ()) +
                          (np.where(maxima == -1)[0] + 1,) +
                          (([len(yval)-1],) if gradient[-1] > 0 else ()))
print(local_maxima(xval, yval))
plt.plot(Voltage,Current,'bo', label='data')
plt.plot(Voltage,Current,'k', label='data')
plt.title('Voltage vs Current')
plt.xlabel('Voltage (Volt)')
plt.ylabel('Current(nA)')



plt.show()
```

```
from numpy import *
import pylab
import pandas
colnames = ['A1', 'B1']
data = pandas.read_excel("181.xlsx", names=colnames)
Current = data.A1.tolist()
Voltage = data.B1.tolist()
def local_maxima(xval, yval):
    xval = np.asarray(xval)
    yval = np.asarray(yval)

    sort_idx = np.argsort(xval)
    yval = yval[sort_idx]
    gradient = np.diff(yval)
    maxima = np.diff((gradient > 0).view(np.int8))
    return np.concatenate((([0],) if gradient[0] < 0 else ()) +
                          (np.where(maxima == -1)[0] + 1,) +
                          (([len(yval)-1],) if gradient[-1] > 0 else ()))
print(local_maxima(xval, yval))

plt.plot(Voltage,Current,'bo', label='data')
plt.plot(Voltage,Current,'k', label='data')
plt.title('Voltage vs Current')
plt.xlabel('Voltage (Volt)')
plt.ylabel('Current(nA)')

plt.show()
```

```
from numpy import *
import pylab
import pandas
colnames = ['A1', 'B1']
data = pandas.read_excel("185.xlsx", names=colnames)
Current = data.A1.tolist()
Voltage = data.B1.tolist()
def local_maxima(xval, yval):
    xval = np.asarray(xval)
    yval = np.asarray(yval)

    sort_idx = np.argsort(xval)
    yval = yval[sort_idx]
    gradient = np.diff(yval)
    maxima = np.diff((gradient > 0).view(np.int8))
    return np.concatenate((([0],) if gradient[0] < 0 else ()) +
                          (np.where(maxima == -1)[0] + 1,) +
                          (([len(yval)-1],) if gradient[-1] > 0 else ()))
print(local_maxima(xval, yval))

plt.plot(Voltage,Current,'bo', label='data')
plt.plot(Voltage,Current,'k', label='data')
plt.title('Voltage vs Current')
plt.xlabel('Voltage (Volt)')
plt.ylabel('Current(nA)')

plt.show()
```

```
from numpy import *
import pylab
import pandas
colnames = ['A1', 'B1']
data = pandas.read_excel("189.xlsx", names=colnames)
Current = data.A1.tolist()
Voltage = data.B1.tolist()
def local_maxima(xval, yval):
    xval = np.asarray(xval)
    yval = np.asarray(yval)

    sort_idx = np.argsort(xval)
    yval = yval[sort_idx]
    gradient = np.diff(yval)
    maxima = np.diff((gradient > 0).view(np.int8))
    return np.concatenate((([0],) if gradient[0] < 0 else ()) +
                          (np.where(maxima == -1)[0] + 1,) +
                          (([len(yval)-1],) if gradient[-1] > 0 else ()))
print(local_maxima(xval, yval))
```

```
plt.plot(Voltage,Current,'bo', label='data')
plt.plot(Voltage,Current,'k', label='data')
plt.title('Voltage vs Current')
plt.xlabel('Voltage (Volt)')
plt.ylabel('Current(nA)')


plt.show()

from numpy import *
import pylab
import pandas
colnames = ['A1', 'B1']
data = pandas.read_excel("194.xlsx", names=colnames)
Current = data.A1.tolist()
Voltage = data.B1.tolist()
plt.plot(Voltage,Current,'bo', label='data')
plt.plot(Voltage,Current,'k', label='data')
plt.title('Voltage vs Current')
plt.xlabel('Voltage (Volt)')
plt.ylabel('Current(nA)')

plt.show()




from numpy import *
import pylab
import pandas
colnames = ['A1', 'B1']
data = pandas.read_excel("200.xlsx", names=colnames)
Current = data.A1.tolist()
Voltage = data.B1.tolist()
plt.plot(Voltage,Current,'bo', label='data')
plt.plot(Voltage,Current,'k', label='data')
plt.title('Voltage vs Current')
plt.xlabel('Voltage (Volt)')
plt.ylabel('Current(nA)')


plt.show()



from numpy import *
import pylab
import pandas
colnames = ['A1', 'B1']
data = pandas.read_excel("205.xlsx", names=colnames)
Current = data.A1.tolist()
Voltage = data.B1.tolist()
```

```
plt.plot(Voltage,Current,'bo', label='data')
plt.plot(Voltage,Current,'k', label='data')
plt.title('Voltage vs Current')
plt.xlabel('Voltage (Volt)')
plt.ylabel('Current(nA)')


plt.show()
```

```
from numpy import *
import pylab
import pandas
colnames = ['A1', 'B1']
data = pandas.read_excel("210.xlsx", names=colnames)
Current = data.A1.tolist()
Voltage = data.B1.tolist()
plt.plot(Voltage,Current,'bo', label='data')
plt.plot(Voltage,Current,'k', label='data')
plt.title('Voltage vs Current')
plt.xlabel('Voltage (Volt)')
plt.ylabel('Current(nA)')

plt.show()
```

```
import numpy as np

a=np.array(Voltage)
b=np.array(sigma)
c= 1/b
sum(a/b)
x=sum(a/b)/sum(c)
print(x)

sigma = [ 0.02, 0.08, 0.07,0.15,0.02,0.01,0.04]
k=(1/np.array(sigma))
l=sum(k)
y=1/l

print(y)
```