Q/M Ratio of Electron

Zekeriya Gökhan Evelek
Boğaziçi University • Physics Department

***Abstract*—In the experiment the purpose is to find the ratio between the charge and the mass of the electron. First, electrons are given energy and accelerated by applying voltages. Those accelerated electrons are applied a magnetic force which is due to a Helmholtz Coil. Since the direction of the electrons and the magnetic force are perpendicular to each other, accelerated beams move in a circular motion inside the tube. Hence, the Q/m ratio of the electron is found to be FOUND relative to the recommended value RECOMMENDED**

I. Theory

The charge of the electron was already known thanks to Millikan's oil drop experiment. Having measured the same charge-to-mass ratio each time consistently, Thomson named it 'corpuscles', which is 2000 times lighter than the lightest atoms existing. It was going to be helpful to find the charge to the mass ratio of the electron so that it is possible to extract the mass of the electron. It was crucial because measuring the mass of the electron is a difficult task to do, even today, considering its significantly low mass.

Accelerated electrons have the energy:

$$\mathrm{V}.q = \frac{m_e.v_e^2}{2} \tag{1}$$

Force due to electric and magnetic fields:

$$\vec{F} = q(\vec{E} + \vec{v_e} \times \vec{B}) \tag{2}$$

For uniform circular motion, the force is:

$$m_e.\frac{v_e^2}{r} \tag{3}$$

Electric and magnetic fields are perpendicular to each other so are

$$\vec{v_e} \ \ \& \ \ \vec{B}$$

$$v_e = \frac{q.B.r}{m_e} \tag{4}$$

equation(1) & equation(4) yields:

$$\frac{q}{m_e} = \frac{2V}{r.B} \tag{5}$$

B has the general form:

$$B_z = \mu_0.I.R^2((R^2 + z - (\frac{a}{2})^2)^{\frac{3}{2}} + (R^2 + z + (\frac{a}{2})^2)^{\frac{3}{2}})$$

However, for the Helmholtz Coil arrangement used in the experiment, one gets:

$$B = (\frac{4}{5})^{\frac{3}{2}}.\mu_0.n.\frac{I}{R} \tag{6}$$

where

$$\mu_0 = 1.257 \times 10^-6\frac{Vs}{Am}, R = 0.2m, n = 154$$

Hence, combining equation(4) with equation(6), we get:

$$\frac{q}{m_e} = \frac{2V.R^2.5^3}{I^2.r^2.\mu_0^2.n^2.4^3} \tag{7}$$

Here the assumption is that accelerated electrons' velocity is perpendicular to a uniform magnetic field, which is not the case because the magnetic field is lower at the edges. Electron beams make a circular motion according to Laurentz Force law. The expectation is that the same q/m ratio is calculated regardless of the voltage and current we control.

Inside the tube, there is low-pressure Neon Gas which is going to be excited owing to the motion of the electrons. Excited gas molecules are responsible for the blue light that is observed during the experiment.

II. Method

In the first place, filament is heated to get electron beams, which are going to be accelerated by an increasing electric field. Then, magnetic field resulting from the current in the Helmholtz Coil changes the direction of the motion of the electrons. Depending on the V and I values, a circular path is to be created.

1) Power supply is made on to give energy to electrons
2) Electron beams are capable of being seen, first a straight path is observed
3) Current is supplied in order to bend the path of the electrons so that we get a circular path at the end
4) Current and voltage values are taken which enable us to observe the circular path

III. The Experimental Setup

- Power Supply for Current Creating Magnetic Field
- Power Supply for Electric Field
- Ammeter for Current Creating Magnetic Field
- Voltmeter for Power Supply Creating Electric Field
- Helmholtz Coil
- Fine-Beam Tube

IV. The Data

Measured V and I values that caused circular motion of the electrons are listed:

TABLE I
MEASURED RADIUS FOR VOLT AND CURRENT VALUES

Radius(cm)	Volt(V)	Current(A)
2.0 ± 0.2	173.3 ± 0.1	1.21 ± 0.01
3.0 ± 0.2	123.7 ± 0.1	1.21 ± 0.01
4.0 ± 0.2	97.2 ± 0.1	1.21 ± 0.01
5.0 ± 0.2	55.8 ± 0.1	1.21 ± 0.01
2.0 ± 0.2	195.8 ± 0.1	1.32 ± 0.01
3.0 ± 0.2	140.0 ± 0.1	1.32 ± 0.01
4.0 ± 0.2	97.6 ± 0.1	1.32 ± 0.01
5.0 ± 0.2	61.6 ± 0.1	1.32 ± 0.01
2.0 ± 0.2	137.2 ± 0.1	1.03 ± 0.01
3.0 ± 0.2	137.2 ± 0.1	1.30 ± 0.01
4.0 ± 0.2	137.2 ± 0.1	1.83 ± 0.01
5.0 ± 0.2	137.2 ± 0.1	2.93 ± 0.01
2.0 ± 0.2	153.0 ± 0.1	1.10 ± 0.01
3.0 ± 0.2	153.0 ± 0.1	1.41 ± 0.01
4.0 ± 0.2	153.0 ± 0.1	1.98 ± 0.01
5.0 ± 0.2	153.0 ± 0.1	3.14 ± 0.01

V. The Analysis

We measured 4 sets of data, two of them came from different constant currents,s and the other two came from different constant voltages. As it can be seen, eq(7)suggests a linear relationship between $\frac{V}{r^2}$ and B^2. Uncertainty propagation is done in the following way:

$$B^2 * (\frac{q}{m}) = \frac{2V}{r^2} \tag{8}$$

$$f = I^2.r^2 \tag{9}$$

$$\sigma_f = \sqrt{(\frac{\partial f}{\partial I})^2\sigma_I^2 + (\frac{\partial f}{\partial r})^2\sigma_r^2} \tag{10}$$

$$\sigma_f = \sqrt{4I^2r^4\sigma_I^2 + 4I^4r^2\sigma_r^2} \tag{11}$$

where f is the function
σ_I and σ_r are corresponding uncertainties

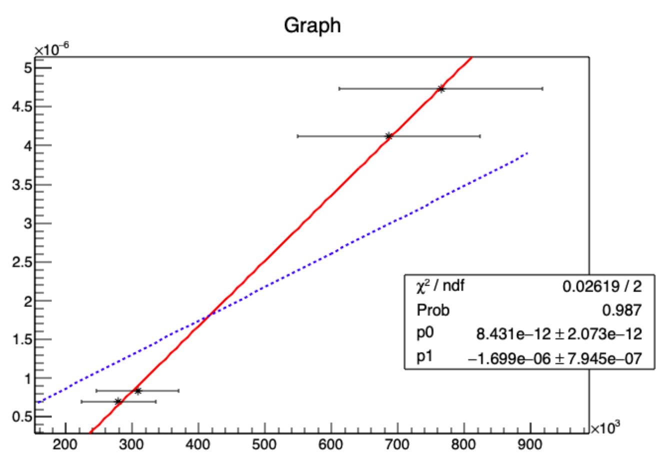

Fig. 1. $\frac{2V}{r^2}$ versus B^2 for r=2 cm

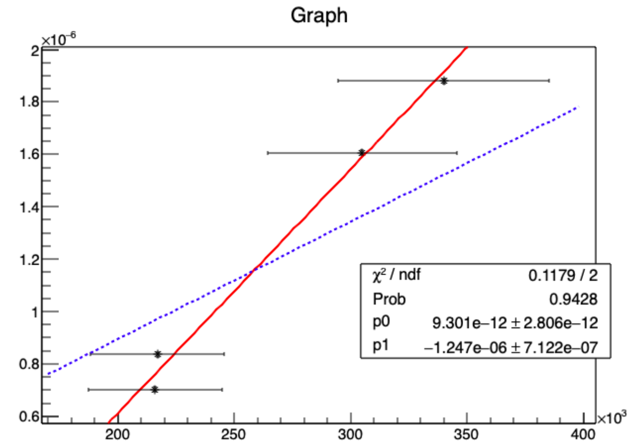

Fig. 2. $\frac{2V}{r^2}$ versus B^2 for r=3 cm

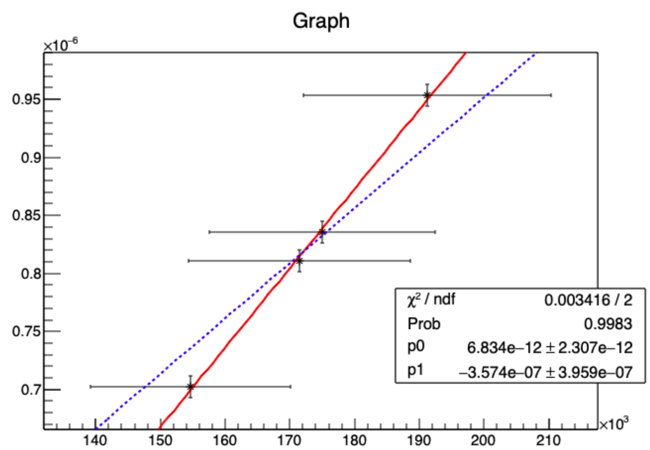

Fig. 3. $\frac{2V}{r^2}$ versus B^2 for r=4 cm

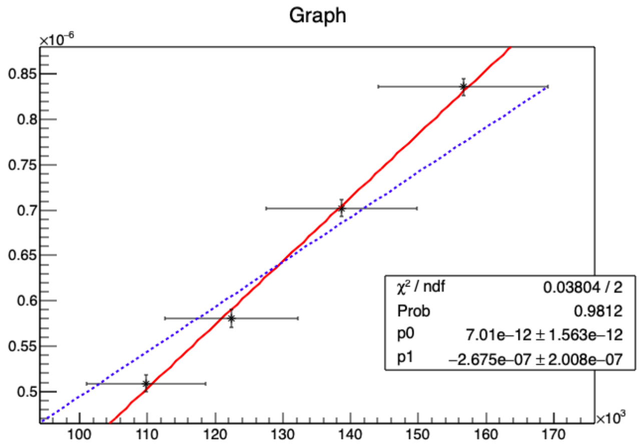

Fig. 4. $\frac{2V}{r^2}$ versus B^2 for r=4 cm

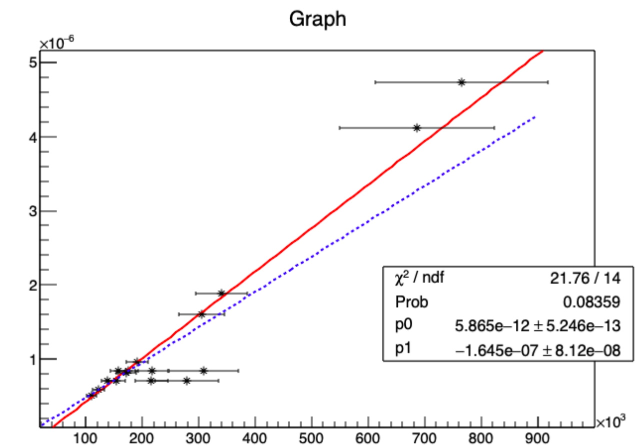

Fig. 5. $\frac{2V}{r^2}$ versus B^2 for all r values

After dealing with the uncertainties, the data points are calculated in order for us to fit a line for each radius separately first, then for combined values for all radius values. The graphs are done by using Root's built-in function and weighted average. Weighted average of the q/m value for different I,V and r values is calculated according to the Erhan Gülmez's lab book(see references).

VI. The Result

We have found the q/m ratio of electron to be $2.06 * 10^11$ with the uncertainty of $5.74 * 10^9$. Owing to the fact that our electron's direction is towards positive, we conclude that it has a negative charge so the result would be $-2.06 * 10^11 \pm 5.74 * 10^9$. CODATA recommended value for this ratio was $-1.76 * 10^11$ which is close to our experimental result.

VII. The Conclusion

After calculating the weighted average for our sixteen different sets of data, our final result seems to be in a range that is reasonably close to the theoretical expected value. Therefore it can be said that our experiment is a success. During the experiment, due to the fact that there are other people in the same room conducting different experiment and in need of light, there are some difficulties in finding out the radius of the motion of the electrons, especially for lower V values because lower the energy given to the beams, the less bright the light caused by the gas in the tube, which makes it difficult what is the exact voltage or current value for the circular motion of electron beams having the radius of 2 cm. It may be due to this problem that data diverges slightly from the expected ones since sometimes it is hard to be sure that it is exactly the correct I and V values for given radius measurements on account of low low energies of the electrons. The problem can be seen by some of our graphs that the red fitted line is far from overlapping with the blue dashed-line. However, in the graph including all the measurements, line-fit seems to be ok. In short, for conducting this experiment, one had better to do it in a darker room.

References

[1] *CODATA Value: electron charge to mass quotient.* URL: https://physics.nist.gov/cgi-bin/cuu/Value?esme.

[2] E. Gülmez. *Advanced Physics Experiments*. 1st. Boğaziçi University Publications, 1999.

VIII. Appendix

Here is my code for $\frac{2V}{r^2}$ versus B^2 linefit via root. This one includes only the measured values for the r=2cm but as it can easily be seen, the plots for 3,4, or 5 cm have exactly the same structure so I only gave one of them. Furthermore, the plot that has all the radius values only differs in that it has 16 instead of 4 different values in each of its initial arrays.

```
{
 const int ndata = 4;
 float amp[ndata] = {1.21 , 1.32, 2.93, 3.14};
 float volt[ndata] = {55.8, 61.6, 137.2,
     153.0};
 float radius[ndata]= {0.02, 0.02, 0.02,
     0.02};
 x= new float[ndata];
 y = new float[ndata];
 sx = new float[ndata];
 sy = new float [ndata];

 float del_r = 0.002;
 float del_v = 0.1;
 float del_a = 0.01;

 float constant = 0.0006925657;
 float constant2 = constant * constant;
 for (int i=0; i<ndata; ++i){ // Error
     Propagation
   float r2 = radius[i]*radius[i];
   float a2 = amp[i]*amp[i];
   float del_r2 = r2 * 2 * del_r / radius[i];
   x[i] = 2 * volt[i] / r2;
   y[i] = constant2 * a2;
   sx[i] = x[i] * sqrt((del_v /
       volt[i])*(del_v / volt[i]) + (del_r2 /
       r2)*(del_r2 / r2));
   sy[i] = y[i] * 2 * del_a / a2;
 }


 float weight = 0;
```

```
  float totw = 0; // Total weight
  float xybar = 0, xbar = 0, ybar = 0, x2bar =
     0, y2bar = 0; // weighted averages

  for (int i=0; i<ndata; ++i) {
   weight = 1./((sy[i]*sy[i]) + (sx[i]*sx[i]));
   totw += weight;
   xybar += (x[i]*y[i]*weight);
   xbar += (x[i]*weight);
   ybar += (y[i]*weight);
   x2bar += (x[i]*x[i]*weight);
   y2bar += (y[i]*y[i]*weight);
   }


  float sy2bar = ndata / totw; // weighted
     average error squared
  float slope = (xybar - xbar*ybar) / (x2bar -
     xbar*xbar);
  float itcpt = ybar - slope * xbar;
  float slopeerr = sqrt ( sy2bar / (ndata *
     (x2bar - xbar*xbar) ) );
  float itcpterr = sqrt ( x2bar ) * slopeerr;
  cout << "slope of fit line = " << slope << "
     +- " << slopeerr << endl;
  cout << "intercept of fit line = " << itcpt
     << " +- " << itcpterr << endl;

  TGraphErrors *mygraph = new
     TGraphErrors(ndata,x,y,sx,sy);
  mygraph->Draw("A*");

  gStyle->SetOptFit(1111);
  //double yPosition = 0.6e-6;
  //gStyle->SetStatY(yPosition); // Adjust
     x-position of the fit results table

  TF1 *ffitline = new
     TF1("ffitline","[0]*x+[1]",0,900e3);
  ffitline->SetParameter(0,slope);
  ffitline->SetParameter(1,itcpt);
  ffitline->SetLineColor(kBlue); // draw in
     blue color
  ffitline->SetLineStyle(2); // draw dotted
     line
  ffitline->Draw("same");


  // let's also try the same with ROOT's own
     fitter
  TF1 *fnew = new
     TF1("fnew","[0]*x+[1]",0,900e3);
  fnew->SetParameters(slope, itcpt); //
     arbitrary starting parameters
  mygraph->Fit(fnew);



}
```