Scattering in Two Dimensions Experiment

Burak Kara
Boğaziçi University • Physics Department

***Abstract*—The experiment aims to study the theory of scattering in two dimensions. We have conducted the experiment to find the total cross section of a cylindrical target, therefore we have calculated its diameter. We have measured the diameter of the target as 5.7 $\pm$ 0.1 cm. With the scattering experiment, we have found it to be 5.3 $\pm$ 0.8 cm and we have calculated the diameter with flux as 5.7 $\pm$ 0.4 cm. Therefore, we are 0.5 σ away from the true value with 7.0$\%$ relative error.**

I. History and Motivation

Scattering is the process of reflecting or redirecting particles, waves or radiation when they interact with other objects. The scattering is essential for us to understand the structure of objects. The Gold Foil Experiment is one of the best examples of the scattering. Ernest Rutherford directed the experiment and published the results in 1911. They have used alpha particles which deflects from the gold foil. The results was not consistent with the Thomson atom model. So, Rutherford concluded that if there is a charged nucleus at the center, this effect can be explained by scattering. Therefore, this experiment changed our understanding of atom. [1].

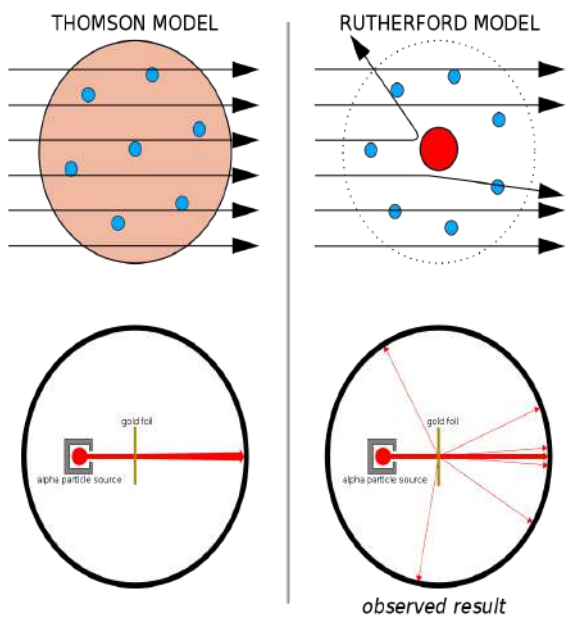


Fig. 1. Thomson Model vs Rutherford Model. [2]

We use a similar technique in the experiment. We shoot metal balls, which are analogue of the alpha particles in Rutherford's experiment. The balls deflect from the cylindrical object at the center, which is the analogue of nucleus. With the help of scattering theory we hope to find the diameter of the cylindrical target.

II. Theory

In the following theory we assume that the metal balls make an ideal reflection and their speed is enough to make a successive collision.

$$I = \frac{\text{Number of Shots}}{\text{Length}} \tag{1}$$

Where I is the flux (Shots/cm)

Taking Derivative:

$$I db = dN \tag{2}$$

where, db is infinitesimal length.

dN is the infinitesimal shots.

Using Ideal Reflection:

$$cos(\frac{\theta}{2}) = \frac{b}{r} \tag{3}$$

where, θ is reflection angle.

b is the vertical distance from center to gun.

r is the radius of the target.

Taking the derivative:

$$-\frac{r}{2}sin(\frac{\theta}{2})d\theta = db \tag{4}$$

Using Eq.2:

$$-I\frac{r}{2}sin(\frac{\theta}{2})d\theta = dN \tag{5}$$

Converting Infinitesimal to Discrete:

$$-I\frac{r}{2}sin(\frac{\theta_i}{2})\Delta\theta_i = \Delta N_i \tag{6}$$

Taking the Integral of Eq.4:

$$-\int_0^{2\pi}\frac{r}{2}sin(\frac{\theta}{2})d\theta = 2r = \int_0^{2r} db \tag{7}$$

Using Eq.2:

$$\int_0^{2r} \frac{dN}{I} = 2r \tag{8}$$

Converting to the Riemann Sum:

$$\sum \frac{\Delta N_i}{I} = 2r \tag{9}$$

With Eq.6 and Eq.9 we have obtained two different methods of calculating the radius of the cylindrical target. The theory suggests that there is a linear relation between number of shots and corresponding sine function. The slope is determined by flux, radius and change in angle. We know the flux and change in angle; therefore, we have obtained a reasonable theory to find the radius of the target.

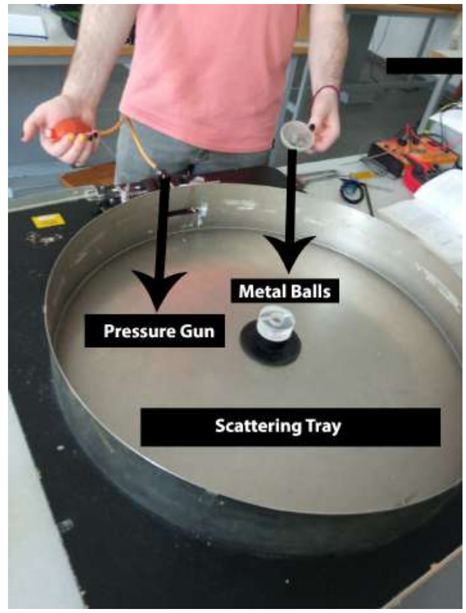


Fig. 2. The Apparatus.

III. Method

We have used a pressure gun which shoots metal balls, and we have aimed it to a cylindrical target. We have changed the position of the gun to aim every point of the cylindrical target. Therefore, we have shot to the cross section of the target which is the diameter. With the theory, we aim to find the diameter of the target from successive scatterings.

To obtain successive scatterings' count we have followed the procedure:

1) We have marked the positions of the gun which shoots the ball to the sides of cylindrical target.
2) The scattering tray is covered with pressure sensitive paper.
3) The gun is set to right-most position.
4) Shot 20 balls.
5) The position is changed after every 20 balls to the left.
6) After 20 balls are shot at left-most position, the angles are marked on the pressure sensitive paper.
7) Counted the successive shoots.

IV. The Experimental Setup

- Scattering Tray.
- Pressure Gun.
- Metal Balls.
- Pressure Sensitive Paper.
- Ruler

Fig. 3. Covering the tray with pressure sensitive paper.

V. The Data

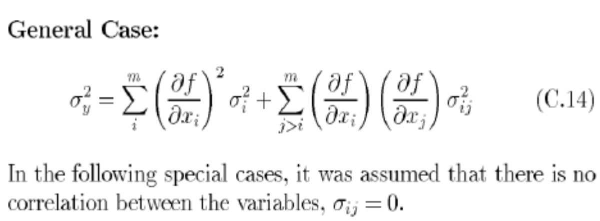
TABLE I
NUMBER OF COUNTS OF SOURCES

Angle(Degree)	Count
10-30	12
30-50	7
50-70	15
70-90	14
90-110	23
110-130	22
130-150	27
150-170	37
170-190	0
190-210	38
210-230	31
230-250	46
250-270	37
270-290	45
290-310	30
310-330	15
330-350	130

The counts has an uncertainty of $\sqrt{N}$.

VI. The Analysis

We have used Root's built-in functions for plotting histograms and fitting the data. The explanation of the least squares method used for linear fit can be found at appendix. Moreover, we have propagated the errors. There is an error of $\sqrt{N}$ for counts, and according to the following formula we have made the uncertainty analysis:



General Case:

$$\sigma_y^2 = \sum_i^m \left(\frac{\partial f}{\partial x_i}\right)^2 \sigma_i^2 + \sum_{j>i}^m \left(\frac{\partial f}{\partial x_i}\right)\left(\frac{\partial f}{\partial x_j}\right)\sigma_{ij}^2 \qquad \text{(C.14)}$$

In the following special cases, it was assumed that there is no correlation between the variables, $\sigma_{ij} = 0$.

Fig. 4. [3]
(See appendix for error propagation macros)

We have plotted the raw data as a histogram. As can be seen from Fig.5, there is accumulation at 330-350 degrees. This is because, the gun we have used is intended to shoot to the right. We have also plotted the count vs sine graph to calculate the radius of the target. As can be seen from Fig.6 the only point that not obeying the theory is the data at 330-350 degrees. Without modifying the data and using Eq.6-9, we have found the diameter to be:

1) Measured Diameter: 0.057 +- 0.001
2) Fitted Diameter: 0.021 +- 0.004
3) Calculated Diameter: 0.038 +- 0.002

So, the raw data is not consistent with the theory. Therefore, we have concluded to drop the data at 330-350 degrees. However, the flux changes significantly due to the data dropping. Therefore, we also modified the flux to be consistent with the theory.

We have made the following modifications:

1) The data at 330-350 degrees are dropped.
2) The flux is modified to be 399/2r.

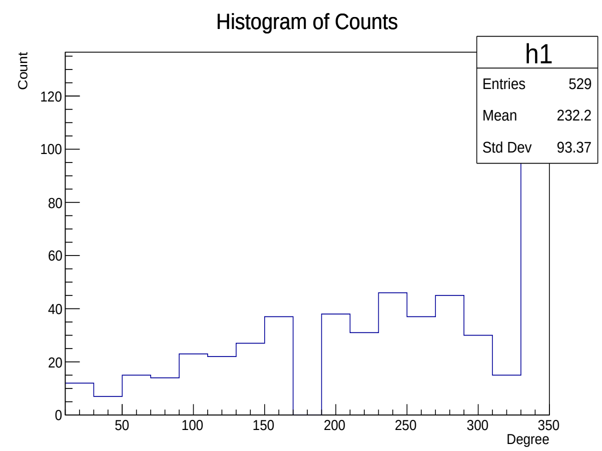


Fig. 5

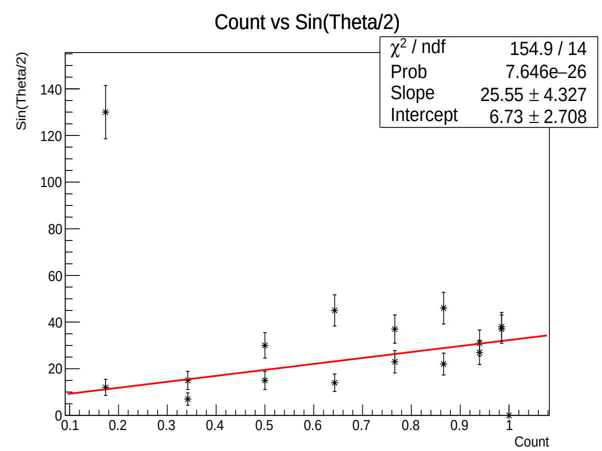


Fig. 6

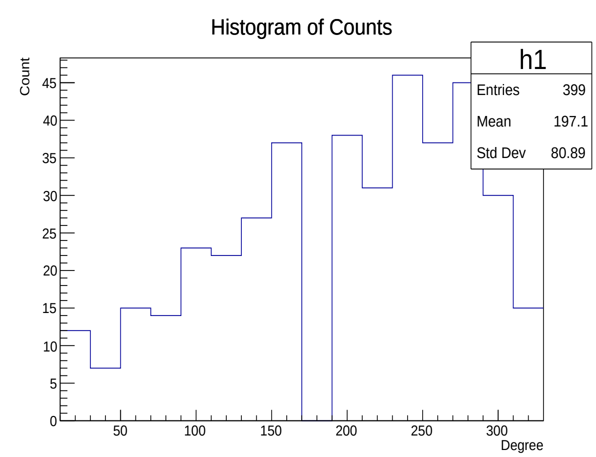


Fig. 7

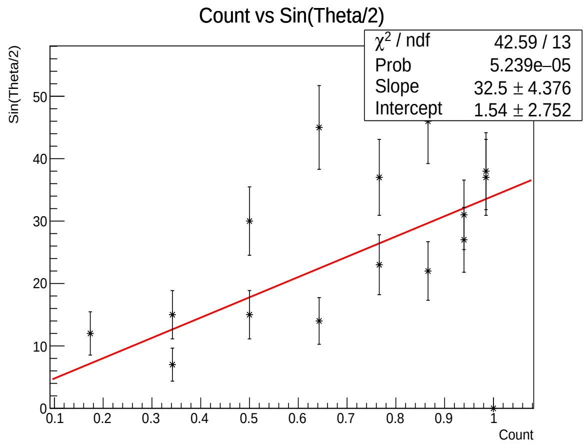


Fig. 8

Using Eq.6-9, we have found the fitted and calculated diameter to be:

1) Measured Diameter: 0.057 +- 0.001 m
2) Fitted Diameter: 0.053 +- 0.008 m
3) Calculated Diameter: 0.057 +- 0.004 m

VII. The Result

We have measured the cylindrical target's diameter as 5.7 +- 0.1 cm. With the scattering experiment, we have found the diameter to be 5.3 +- 0.8 cm. Also, we have calculated the radius with flux, and we have found it to be 5.7 +- 0.4 cm. Therefore, we have concluded that the modifications we made are reasonable. We are 0.5σ away from the true value with 7.0 $\%$ relative error.

VIII. The Conclusion

We have investigated the scattering in two dimensions. We have used a cylindrical object as a target which has a diameter of 5.7 +- 0.1 cm. We have used two different method of calculations as Eq.6 Eq.9 suggests. With the Eq.6 we have measured the radius to be 5.3 +- 0.8 cm. As Eq.9 suggests, we have found the diameter to be 5.7 +- 0.4 cm. We are 0.5σ away with 7.0$\%$ relative error. Therefore, we have concluded that the experiment consistent with the theory that we are tested. However, the modification that we made leads to a "perfect" calculations for the Eq.9 which is 0σ away from the true value. Therefore, we have concluded that the experiment was partly successful.

We have used a gun which shoots a metal ball with the pressure. However, the pressure we have given to the system is not constant. Moreover, at many instances, the gun stuck and the steel ball lost its momentum. This effect changes the mean free path of the balls, so it changes successive collisions.

Another problem we have experienced is that the gun we have used was intended to shoot to the right, so we could not seen a purely symmetric behavior in the plots. The gun could be fixed before the scattering experiment.

References

[1] *The Scattering of and Particles by Matter and the Structure of the Atoms*. URL: https://www.lawebdefisica.com/arts/structureatom.pdf (visited on 04/23/2023).

[2] *Geiger-Marsden experiment expectation and result*. URL: https://commons.wikimedia.org/w/index.php?curid=32215297 (visited on 04/23/2023).

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

IX. Appendix

The fit has been done with Root's built-in function (See Fig.5-8). We have used root release 6.28/00 for Ubuntu22.

Macro for non-modified analysis

```
{
 //Initializing Variables
 float r_m = 0.057/2;
 float r_c = 0;
 float r_f = 0;

 //Initializing Arrays

 // First Case - no modifications
 const int n = 17;
 float I = 20 / (r_m*2 / 40);
 float sd_I = sqrt(pow(20/(r_m*r_m*2 /
     40),2)*pow(0.001,2));
 float count[n] = {12,7,15,14,23,22,27,37,0,
 38,31,46,37,45,30,15,130};
 float angle[n] =
     {20,40,60,80,100,120,140,160,
 180,200,220,240,260,280,300,320,340};
 float sin[n] =
     {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};
 float sx[n] =
     {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};
 float sy[n] =
     {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};

 //Filling Arrays and Histogram
 TH1F *h1 = new TH1F("h1","Count",n, 10, 350);
   for (int i=0; i<n; i++){
     sin[i] =
         TMath::Sin(angle[i]*TMath::Pi()/360);
     sy[i] = sqrt(count[i]);
     for (int j=0; j<count[i]; j++)
       h1->Fill((20*i)+20);
   }

 //Ploting Histogram and Linear Fit
 gStyle->SetOptFit(1111);
 TCanvas *c2 = new TCanvas();
 h1->SetTitle("Histogram of Counts");
 h1->GetXaxis()->SetTitle("Degree");
 h1->GetYaxis()->SetTitle("Count");
 h1->Draw();

 TCanvas *c1 = new TCanvas();
 TGraphErrors *graph = new
     TGraphErrors(n,sin,count,sx,sy);
 TF1 *f1 = new TF1("f1", "[0]*x + [1]",0,1);
 f1->SetParameters(40,0);
 f1->SetParNames("Slope","Intercept");
```

```
  graph->SetTitle("Count vs
      Sin(Theta/2);Count;Sin(Theta/2)");
  graph->Draw("A*");
  graph->Fit("f1","EX0");

  //Calculating Radius
  float del_theta = 20*TMath::Pi()/180;
  float slope = f1->GetParameter(0);
  r_f = slope*2/ (I*del_theta);
  r_c = h1->GetEntries() / (2*I);

  //Calculating Errors
  float sd_n = sqrt(h1->GetEntries());
  float sd_slope = f1->GetParError(0);

  float sd_r_f = sqrt(pow(2/(I*del_theta),2) *
      pow(sd_slope,2)
    +
       pow(2*slope/(I*I*del_theta),2)*pow(sd_I,2));
  float sd_r_c =
      sqrt(pow((1/(2*I)),2)*pow(sd_n,2)
    +
       pow(h1->GetEntries()/(2*I*I),2)*pow(sd_I,2));

  //Printing Results
  cout << "Measured Radius: " << r_m << " +- "
      << "0.001" << endl;
  cout << "Fitted Radius: " << r_f << " +- "
      << sd_r_f << endl;
  cout << "Calculated Radius: " << r_c << " +-
      " << sd_r_c << endl;

  //Saving Plots
  c1->Print("Fit-2.pdf");
  c2->Print("Hist-2.pdf");
}
```

Macro for modified analysis

```
 {
  //Initializing Variables
  float r_m = 0.057/2;
  float r_c = 0;
  float r_f = 0;

  //Initializing Arrays
  const int n = 16;
  float count[n] = {12,7,15,14,23,22,27,37,0
  ,38,31,46,37,45,30,15};
   float angle[n] =
       {20,40,60,80,100,120,140,160,
 180,200,220,240,260,280,300,320};
  float sin[n] =
      {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};
  float sx[n] =
      {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};
  float sy[n] =
      {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};

  //Filling Arrays and Histogram
  TH1F *h1 = new TH1F("h1","Count",n, 10, 330);
    for (int i=0; i<n; i++){
      sin[i] =
          TMath::Sin(angle[i]*TMath::Pi()/360);
      sy[i] = sqrt(count[i]);
      for (int j=0; j<count[i]; j++)
       h1->Fill((20*i)+20);
    }

  //Ploting Histogram and Linear Fit
  gStyle->SetOptFit(1111);
  TCanvas *c2 = new TCanvas();
  h1->SetTitle("Histogram of Counts");
  h1->GetXaxis()->SetTitle("Degree");
  h1->GetYaxis()->SetTitle("Count");
  h1->Draw();

  TCanvas *c1 = new TCanvas();
  TGraphErrors *graph = new
      TGraphErrors(n,sin,count,sx,sy);
  TF1 *f1 = new TF1("f1", "[0]*x + [1]",0,1);
  f1->SetParameters(40,0);
  f1->SetParNames("Slope","Intercept");
  graph->SetTitle("Count vs
      Sin(Theta/2);Count;Sin(Theta/2)");
  graph->Draw("A*");
  graph->Fit("f1","EX0");

  //Calculating Radius
  float del_theta = 20*TMath::Pi()/180;
  float slope = f1->GetParameter(0);
  float I = h1->GetEntries() / (r_m*2);
  r_f = slope*2/ (I*del_theta);
  r_c = h1->GetEntries() / (2*I);

  //Calculating Errors
  float sd_n = sqrt(h1->GetEntries());
  float sd_slope = f1->GetParError(0);
  float sd_I =
      sqrt(pow((1/(r_m*2)),2)*pow(sd_n,2)
    +
       pow(h1->GetEntries()/(r_m*r_m*2),2)*pow(0.001,2))


  float sd_r_f = sqrt(pow(2/(I*del_theta),2) *
      pow(sd_slope,2)
    +
       pow(2*slope/(I*I*del_theta),2)*pow(sd_I,2));

  float sd_r_c =
      sqrt(pow((1/(2*I)),2)*pow(sd_n,2)
    +
       pow(h1->GetEntries()/(2*I*I),2)*pow(sd_I,2));

  //Printing Results
  cout << "Measured Radius: " << r_m << " +- "
      << "0.001" << endl;
  cout << "Fitted Radius: " << r_f << " +- "
      << sd_r_f << endl;
  cout << "Calculated Radius: " << r_c << " +-
      " << sd_r_c << endl;

  //Saving Plots
  c1->Print("Fit-1.pdf");
   c2->Print("Hist-1.pdf");
}
```

Least Squares Method: The least squares method is used for finding the optimum function that fits the data. The method tries to minimize the sum of the squared differences of predicted and observed values. In this experiment, we are using a linear fit, therefore it is sufficient to find the intercept and slope of the best line. After this parameters are found, we have the equation $y = mx + n$. Therefore, the uncertainty can be found by using error propagation showed at Fig.4.