

PHY 4210-01 Senior Lab  
Lab M-1: Magnetic Field Mapping

Sarah Arends  
Jacquelyne Miksanek  
Ryan Wojtyla

Instructor: Jerry Collins II

February 7, 2019

**Abstract**

In this experiment the magnetic field inside a Helmholtz coil was measured and compared to theoretical calculations determined from the Smythe derivation of the Biot-Sarvat Law for a plane displaced from the central axis, with coordinates  $z$ ,  $\rho$ , and  $\phi$ . When determining the magnetic field inside a Helmholtz coil, a Hall probe is used to obtain the magnitude of the magnetic field at varying positions inside the coil. Theoretically the axial component of the magnetic field that is produced inside the Helmholtz coil is, to some extent, of uniform magnitude.

# Contents

<b>1</b>	<b>Objective of the Experiment</b>	<b>3</b>
<b>2</b>	<b>Theory of the Experiment</b>	<b>3</b>
<b>3</b>	<b>Equipment Utilized</b>	<b>3</b>
3.1	The Helmholtz coil . . . . .	3
3.2	The Hall Effect Probe . . . . .	4
3.3	Position Controls . . . . .	4
<b>4</b>	<b>Procedure</b>	<b>4</b>
4.1	Measuring the External Field . . . . .	5
4.2	Procedural Modifications . . . . .	5
4.3	Additional Sources of Error . . . . .	6
<b>5</b>	<b>Data Analysis</b>	<b>6</b>
5.1	Calculating Supply Voltage . . . . .	6
5.2	Determining the Number of Turns in a Coil . . . . .	6
<b>6</b>	<b>Results</b>	<b>6</b>
6.1	Comparing the directions of the Magnetic Field . . . . .	6
<b>7</b>	<b>Conclusion</b>	<b>7</b>
<b>8</b>	<b>Appendices</b>	<b>7</b>
8.1	Appendix A: Data . . . . .	7
8.2	Appendix B: Source Code . . . . .	7

# 1 Objective of the Experiment

During this lab, the number of turns of wire inside a Helmholtz coil was determined for use in theoretical calculations. Then a 3-dimensional and 2-dimensional mapping of the magnetic field inside the Helmholtz coil was created in order to investigate the presence of a uniform field, running along its axial direction.

# 2 Theory of the Experiment

Recall for a straight current-carrying wire, circular magnetic field lines are generated around the wire in accordance with the curling right-hand rule. The Helmholtz coil contains two regions of circularly wound wires. Due to the the circular symmetry, all components of each infinitesimal segment of the wire will cancel *except* for that in the axial direction. In summary, a circular current produces a linear magnetic field.

The field point of the system has before been typically placed along the axis of the direction of the magnetic field, we will call this the z-direction. This was due to the ease of solving the Biot-Savart Law under these simple conditions, as the direction and strength of the magnetic field will follow along the z-axis of the system, which is where the field point is placed. When this is applied to the co-axial coils of the Helmholtz apparatus the evaluation of the Biot-Savart Law becomes too trivial. One then chooses the field point to be placed off of the z-axis as more information about the magnetic field of the coils can be determined. This is the more general scenario and thus more complex. The off axis form can be used for any point that is off of the z-axis, while the on axis is a specific and simplified form of the general case. The general form is best represented by Smythe's derivation of the Biot-Savart Law.

$$B_z = \frac{\mu_0 IN}{2\pi} \left[ \frac{1}{\sqrt{(a+\rho)^2 + (a-z)^2}} \left[ K_1 + \left( \frac{a^2 - \rho^2 - (a-z)^2}{(a-\rho)^2 + (a-z)^2} \right) E_1 \right] \right. \\ \left. + \frac{1}{\sqrt{(a+\rho)^2 + z^2}} \left[ K_2 + \left( \frac{a^2 - \rho^2 - z^2}{(a-\rho)^2 + z^2} \right) E_2 \right] \right] \quad (1)$$

# 3 Equipment Utilized

## 3.1 The Helmholtz coil

The Helmholtz coil consists of two concentric sets of coils, each with the same radius and separated by a distance equal to their radius. This configuration allows the contribution of each set of coils to produce a uniform field in the center of the coils. The current in each set of coils must be oriented in a particular direction so that their contributions constructively interfere. The circuit is shown in figure 1.

Figure 1: Flow of current through the Helmholtz coil, oriented such that the produced fields are constructive.

### 3.2 The Hall Effect Probe

A DC Gaussmeter (AlphaLab Model GM-1-HS) was connected to a Hall Effect Probe in order to measure the field strength inside the Helmholtz coil. The Hall Effect Probe contains a semiconductor junction that, when exposed to a magnetic field, produces a voltage proportional to the field strength.

### 3.3 Position Controls

The position of the Hall Effect Probe can be modified in the  $\rho$  direction by sliding the ruler bar through the acrylic cube shown in figure 2. The position can be modified in the  $\phi$  direction by rotation the ruler bar about the central pole. However, for the sake of this experiment, this did not have to be modified because measurements were taken in a single  $\rho, z$  plane. The  $z$  coordinate was modified by sliding the acrylic cube and ruler bar up and down the central pole.

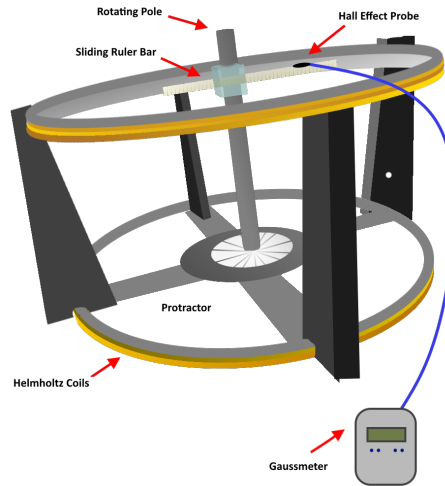


Figure 2: Two concentric Helmholtz coils separated by a distance equal to their radius. Rotating pole and sliding ruler allow for modification of the probe's position.

## 4 Procedure

Note that, per suggestion of the laboratory manual, the procedural steps of this experiment have been omitted. The discussion section provides sufficient detail on what actions were taken.

## 4.1 Measuring the External Field

The Helmholtz coil is oriented such that the Earth's magnetic field is parallel to the z-axis of the coils. This allows us to produce an applied magnetic field that is exactly anti-parallel to the Earth's field. From there, we can compute the applied field by subtracting the Earth's field from the total resultant field.

Note that there was an apparent offset in the Gaussmeter reading, as the 0.36G measurement for Earth's field was consistently higher than the expected value for Earth's field of 0.24G. However, if there truly exists such an offset in the measurement device, it would appear in both the measurement of Earth's field and in the measurement of the total field inside the Helmholtz coil. Subtracting these two to obtain the strength of the applied field would cancel any contribution from such an offset.

## 4.2 Procedural Modifications

Upon initial inspection of the equipment, it appeared the center pole running along the z axis of the Helmholtz coil was misaligned. In order to mitigate this error and ensure that coordinates were modified independently, a chord was used to realign the pole as closely as possible to the true z axis. However, since this alignment was not quantified, it is possible that there the pole is misaligned to some degree. This would result in a systematic error intrinsic to the experimental set-up. If the pole deviates from the z-axis, the experimentally recorded z-values are underestimated, causing the experimental field strengths to trend lower than the theoretical field strengths.

The majority of field strength measurements collected for the 3-dimensional mapping were taken on the same day of experimentation. After resuming this data collection on the next day, the values appeared to be systematically higher. Possible causes of this offset were investigated. Before taking measurements and intermittently during the data collection, the hall effect probe was zeroed and observed with the power supply off in order to ensure a consistent reading of the Earth's magnetic field. The reference measurement taken at the start of this lab session was similar to those taken during the previous session (zeroed field measurements were between 0.36G and 0.4G on both days), so a discrepancy in the Earth's field strength measurement was eliminated as the source of this error. Note that any small variation in the Earth's field measurement could be due to misalignment of the probe (a systematic error in measurement that would under-report the field strength) or simply a random error in measurement due to the limited performance of the probe.

An ammeter was also used to ensure a 2A current was consistently applied on both days of data collection, thus a change in the applied current was eliminated as a source of error. Because the source of this error was ultimately not determined and eliminated, the effect had to be compensated for with a procedural modification. In order to recreate the data points from the previous lab session, the current from the power supply was modified until the field strength matched previous measurements in several locations. This ultimately required lowering the applied current from 2000mA to 1790mA.

Upon further investigation, it appeared the current from the power supply was unstable, as it would decrease and increase every few minutes. This produced a source of random intrinsic error, which was mitigated by fine tuning

the current value before each measurement after the issue was discovered.

### 4.3 Additional Sources of Error

Because the experimental set-up was restricted to a small area, the contribution from the field produced by the power supply may be non-negligible. From the perspective of the experimenter, the power supply sits behind and to the right of the Helmholtz coil. Therefore, by the curling right hand rule, this would produce an upward magnetic field on the side of the wire nearest the Helmholtz coil. This would produce a systematic intrinsic error that causes the external field measurements to be overestimated. Similarly, the power supply itself may be producing a small field that could also contribute a systematic error, although the exact effect could not be determined without knowing the orientation of such a field.

## 5 Data Analysis

### 5.1 Calculating Supply Voltage

Using a multimeter, the resistance of a set of coils was measured to be  $3.4\Omega$ . In order to determine the necessary voltage to send 3A of current through the coils, we make a simple calculation using Ohm's law.

$$\begin{aligned} V &= IR \\ &= (3\text{ A})(3.4\Omega) \\ &= 10.2V \end{aligned}$$

### 5.2 Determining the Number of Turns in a Coil

Further calculations will require knowing the number of turns of wire in each set of coils. MORE TEXT TO COME HERE

Theoretical values for what the strength of the magnetic field ought to be within the Helmholtz coils were determined by translating Equation 1 into Julia code, provided in Appendix B ??.

## 6 Results

### 6.1 Comparing the directions of the Magnetic Field

When measuring at a probe height of  $a/2$  (16cm), where 'a' is the separation distance between the coils, the strength of the magnetic field in the 'z' direction was measured to be -3.13 Gauss. When measuring the magnetic field in the 'z' direction at a probe height of 5cm, the magnetic field strength was measured to be -3.28 Gauss. These results follow with the theory as it is expected that the magnetic field is propagated in the 'z' direction. The measured magnetic field strength for the  $\rho$  direction was -0.46 and -0.05 Gauss for a probe height

of 16cm and 5cm respectfully. The measured magnetic field strength for the  $\phi$  direction was -0.51 and -0.31 Gauss for a probe height of 16cm and 5cm respectfully. For a probe height of 16cm the percentage for the magnitude of the magnetic field that is measured to be in the  $\rho$  direction is 14% while the percentage for the magnitude of the magnetic field that is measured to be in the  $\phi$  direction is 16%. For a probe height of 5cm the percentage for the magnitude of the magnetic field that is measured to be in the  $\rho$  direction is 1% while the percentage for the magnitude of the magnetic field that is measured to be in the  $\phi$  direction is 9%. The magnetic field produced by the Helmholtz coils should be directed along the 'z' axis. These small measured values follow the aforementioned theory and we can determine that the magnetic field produced by the Helmholtz coil is indeed axial. Furthermore, we can determine that the magnetic field is axial along the 'z' direction.

## 7 Conclusion

## 8 Appendices

### 8.1 Appendix A: Data

### 8.2 Appendix B: Source Code

---

```

1  # Source code for calculating the theoretical data and plotting it in both 2D
2  # and 3D.
3
4  # -----BEGIN PACKAGES-----
5
6  # enable the addition of packages
7  using Pkg
8
9  # integration
10 Pkg.add("QuadGK")
11 using QuadGK
12
13 # plotting
14 Pkg.add("Plots")
15 using Plots
16
17 # displaying data
18 Pkg.add("Printf")
19 using Printf
20
21 # -----END PACKAGES-----
22
23 # -----BEGIN CONSTANTS-----
24
25 # Permeability of Free Space  $\frac{Tm}{A}$ 
26  $\mu_0 = 1.256e-6$ 

```

```

27
28 # -----END CONSTANTS-----
29
30 # -----BEGIN BIOT-SAVART ON AXIS-----
31
32 # Biot-Savart (one loop, on axis)
33 B1(z,I,a) = ((μ0 * I * (a ^ 2)) / 2) * (1 / (((a ^ 2) + (z ^ 2)) ^ (3/2)))
34
35 # Biot-Savart (two loops, on axis)
36 B2(z,I,a) = ((μ0 * I * (a ^ 2)) / 2) *
37     ((1 / (((a ^ 2) + (z ^ 2)) ^ (3/2))) +
38     (1 / (((a ^ 2) + ((a - z) ^ 2)) ^ (3/2))))
39
40 # -----END BIOT-SAVART ON AXIS-----
41
42 # -----BEGIN BIOT-SAVART OFF AXIS-----
43
44 # the full Biot-Savart equation (two loops, off axis)
45 B(z,I,N,a,ρ) =
46     ((μ0 * I * N) / (2 * pi) ) *
47     (
48         (1 / sqrt((a + ρ) ^ 2 + ((a - z) ^ 2))) *
49         (
50             K(1,a,ρ,z) +
51             (
52                 (((a ^ 2) - (ρ ^ 2) - ((a - z) ^ 2)) /
53                 (((a - ρ) ^ 2) + ((a - z) ^ 2))) *
54                 E(1,a,ρ,z)
55             )
56         ) +
57         (
58             (1 / sqrt(((a + ρ) ^ 2) + (z ^ 2))) *
59             (
60                 K(2,a,ρ,z) +
61                 (
62                     (((a ^ 2) - (ρ ^ 2) - (z ^ 2)) /
63                     (((a - ρ) ^ 2) + (z ^ 2))) *
64                     E(2,a,ρ,z)
65                 )
66             )
67         )
68     )
69
70 # elliptic integrals of first kind (series expansion)
71 K(j,a,ρ,z, precision = 4) =
72     (pi / 2) * (
73         1 + sum([
74             ((reduce(*, [n - 1 for n in 2:2:precision]) /
75             reduce(*, [n for n in 2:2:precision])) ^ 2) *
76             (k(j,a,abs(ρ),z) ^ n)

```



```

77         for n in 2:2:precision
78     ])
79 )
80
81 # elliptic integrals of the second kind (series expansion)
82 E(j,a,ρ,z, precision = 4) =
83     (pi / 2) * (
84         1 - foldl(-,[
85             ((reduce(*, [n-1 for n in 2:2:precision]) /
86                 reduce(*, [n for n in 2:2:precision])) ^ 2) *
87             ((k(j,a,abs(ρ),z) ^ n) / (n-1))
88             for n in 2:2:precision
89         ])
90     )
91
92 # the collections of variables k1 and k2
93 k(j,a,ρ,z) = (j == 1) ?
94     sqrt((4 * a * ρ) / (((a + ρ) ^ 2) + ((a - z) ^ 2))) :
95     sqrt((4 * a * ρ) / (((a + ρ) ^ 2) + (z ^ 2)))
96
97 # -----END BIOT-SAVART OFF AXIS-----
98
99 # -----BEGIN CONDITIONS-----
100
101 # heights at which measurements were taken (meters)
102 heightLevels = 0.03:0.02:0.33
103
104 # current applied to the coils (amperes)
105 current = 2
106
107 # turns of wire in each coil
108 turns = 73
109
110 # radius of the coils (meters)
111 radius = 0.332
112
113 # radial distances at which measurements were taken (meters)
114 rhos = -0.22:0.02:0.22
115
116 # -----END CONDITIONS-----
117
118 # -----BEGIN CALCULATIONS-----
119
120 # list of Biot-Savart functions for each height level
121 BHeights = [ futureRho -> B(heightLevel, current, turns, radius, futureRho) * 1e6
122             for heightLevel in heightLevels ]
123
124 # apply the list of radial distances to each height level's function (rhos are rows)
125 allBs = [ map(heightLevelFunction, rhos) for heightLevelFunction in BHeights ]
126

```

```

127 # -----END CALCULATIONS-----
128
129 # -----BEGIN MAKE DATA POINTS-----
130
131 # convert column number (1:23) to radial distance value (-22:2:22)
132 colNum2RadialDist(col) = col + (col - 24)
133
134 # convert row number (1:16) to height value (3:2:33)
135 rowNum2Height(row) = row + (row + 1)
136
137 # data points for a row of data at the same height level
138 getRow(row) = [ (rowNum2Height(row), colNum2RadialDist(col), allBs[row][col])
139                 for col in 1:23 ]
140
141 # data points as triples with rows as heights and columns as radial distances
142 allPts = map(getRow, 1:16)
143
144 # -----END MAKE DATA POINTS-----
145
146 # -----BEGIN FORMAT DATA-----
147
148 # split up coords into their own lists
149 formatRow(row) = ( [ coords[1] for coords in row ],
150                   [ coords[2] for coords in row ],
151                   [ coords[3] for coords in row ] )
152
153 # format all rows into form ([zs], [rs], [Bs])
154 listOfRows = map(formatRow, allPts)
155
156 # all these lists must be combined
157 tripleData = ( foldl(hcat, map(x -> x[1], listOfRows)),
158               foldl(hcat, map(x -> x[2], listOfRows)),
159               foldl(hcat, map(x -> x[3], listOfRows)) )
160
161 # print the data in neat columns
162 function displayFormattedData(data)
163
164     # print header
165     @printf("z ρ B\n")
166
167     # iterate through each data point
168     for ii in 1:length(data[1])
169         @printf("%f %f %f\n", data[1][ii], data[2][ii], data[3][ii])
170     end
171
172 end
173
174 # apply the data printing function onto our data
175 displayFormattedData(tripleData)
176

```

```

177 # -----END FORMAT DATA-----
178
179 # -----BEGIN GENERATE 2D PLOT-----
180
181 # extract the Bs from the list of row triples
182 rowsBs = [ row[3] for row in listOfRows ]
183
184 # generate height level labels for each line
185 heightLevelLabels = map(string, 3:2:33)
186
187 # generate the plot
188 theoPlot2d = Plots.plot(listOfRows[1][2], rowsBs, label = heightLevelLabels,
189                        title = "Theoretical Magnetic Field 2D",
190                        size = (550,550),
191                        legendtitle = "Height Levels (cm)",
192                        legend = :top,
193                        xlabel = "Radial Distance from Axis (cm)",
194                        ylabel = "Magnetic Field Strength (uT)"
195                        )
196
197 # save the plot to disk
198 savefig(theoPlot2d, "2DPlotTheoretical")
199
200 # -----END GENERATE 2D PLOT-----
201
202 # -----BEGIN GENERATE 3D PLOT-----
203
204 # generate the plot
205 theoPlot3d = Plots.plot(tripleData[3], tripleData[2], tripleData[1],
206                        seriestype = :scatter, legend = :topright, legendtitle = "Rows",
207                        title = "Theoretical Magnetic Field 3D", size = (550,550),
208                        xlabel = "Magnetic Field Strength (uT)",
209                        ylabel = "Height Relative to Bottom Coil (cm)",
210                        zlabel = "Radial Distance from Axis (cm)"
211                        )
212
213 # save the plot to disk
214 savefig(theoPlot3d, "3DPlotTheoretical.png")
215
216 # -----END GENERATE 3D PLOT-----

```

---

```

1 # Generate 2D and 3D plots of the experimental data collected during the
2 # experiment.
3
4 # -----BEGIN PACKAGES-----
5
6 # add other packages
7 using Pkg
8

```

```

9  # interpreter for reading CSV files
10 Pkg.add("CSV")
11 using CSV
12
13 # convenient container for data extracted from CSV files
14 Pkg.add("DataFrames")
15 using DataFrames
16
17 # plotting
18 Pkg.add("Plots")
19 using Plots
20
21 # format data output
22 Pkg.add("Printf")
23 using Printf
24
25 # -----END PACKAGES-----
26
27 # -----BEGIN EXTRACT DATA-----
28
29 # read data from CSV into a DataFrame
30 data = CSV.File("2dMappingTesla.csv") |> DataFrame
31
32 # convert row number (1:16) to height value (3:2:33)
33 rowNum2Height(row) = row + (row + 1)
34
35 # reverse the order of the incoming heights
36 heightReversal(z) = z + (16 - (z - 1) - z)
37
38 # extract the data from a row into a triple representing that row
39 extractRow(row) = (
40     fill(rowNum2Height(row), 23),
41     -22:2:22,
42     [(Bval + 0.000024) * 1e6 for Bval in [data[ii][heightReversal(row)] for ii in 2:24]]
43 )
44
45 # list of triples, each representing a row, of the form ([zs], [rs], [Bs])
46 listOfRows = map(extractRow, 1:16)
47
48 # -----END EXTRACT DATA-----
49
50 # -----BEGIN FORMAT DATA-----
51
52 # now, a "row" is of the form ([zs], [rs], [Bs]); all these lists must be combined
53 tripleData = ( foldl(hcat, map(x -> x[1], listOfRows)),
54                foldl(hcat, map(x -> x[2], listOfRows)),
55                foldl(hcat, map(x -> x[3], listOfRows)) )
56
57 # print the data in neat columns (display only, does not need to be included)
58 function displayFormattedData(data)

```

```

59
60     # print header
61     @printf("z  $\rho$  B\n")
62
63     # iterate through each data point
64     for ii in 1:length(data[1])
65         @printf("%f %f %f\n", data[1][ii], data[2][ii], data[3][ii])
66     end
67
68 end
69
70 # apply the display function to the data
71 displayFormattedData(tripleData)
72
73 # -----END FORMAT DATA-----
74
75 # -----BEGIN GENERATE 2D PLOT-----
76
77 # extract the Bs from the list of row triples
78 rowsBs = [ row[3] for row in listOfRows ]
79
80 # generate height level labels for each line
81 heightLevelLabels = map(string, 3:2:33)
82
83 # generate the plot
84 expPlot2d = Plots.plot(listOfRows[1][2], rowsBs, label = heightLevelLabels,
85                        title = "Experimental Magnetic Field 2D",
86                        size = (550,550),
87                        legendtitle = "Height Levels (cm)",
88                        legend = :bottom,
89                        xlabel = "Radial Distance from Axis (cm)",
90                        ylabel = "Magnetic Field Strength (uT)"
91                        )
92
93 # save the plot to disk
94 savefig(expPlot2d, "2DPlotExperimental")
95
96 # -----END GENERATE 2D PLOT-----
97
98 # -----BEGIN GENERATE 3D PLOT-----
99
100 # generate the plot
101 expPlot3d = Plots.plot(tripleData[3], tripleData[2], tripleData[1],
102                        seriestype = :scatter, legend = :topleft,
103                        legendtitle = "Rows",
104                        title = "Experimental Magnetic Field", size = (550,550),
105                        xlabel = "Magnetic Field Strength (uT)",
106                        ylabel = "Height Relative to Bottom Coil (cm)",
107                        zlabel = "Radial Distance from Axis (cm)"
108                        )

```

```

109
110 # save the image to disk
111 savefig(expPlot3d, "3DPlotExperimental.png")
112
113 # -----END GENERATE 3D PLOT-----

```

---

```

1 # -----BEGIN PACKAGES-----
2
3 # enable the addition of packages
4 using Pkg
5
6 # interpreter for reading CSV files
7 Pkg.add("CSV")
8 using CSV
9
10 # convenient container for data extracted from CSV files
11 Pkg.add("DataFrames")
12 using DataFrames
13
14 # plotting
15 Pkg.add("Plots")
16 using Plots
17
18 # -----BEGIN GENERATE THEORETICAL DATA-----
19
20 # Permeability of Free Space (T*m)/A
21  $\mu_0 = 1.256\text{e-}6$ 
22
23 # the full Biot-Savart equation (two loops, off axis)
24 B(z,I,N,a, $\rho$ ) =
25     (( $\mu_0$  * I * N) / (2 * pi) ) *
26     (
27         (1 / sqrt((a +  $\rho$ ) ^ 2) + ((a - z) ^ 2))) *
28         (
29             K(1,a, $\rho$ ,z) +
30             (
31                 (((a ^ 2) - ( $\rho$  ^ 2) - ((a - z) ^ 2)) /
32                 (((a -  $\rho$ ) ^ 2) + ((a - z) ^ 2))) *
33                 E(1,a, $\rho$ ,z)
34             )
35         ) +
36         (
37             (1 / sqrt(((a +  $\rho$ ) ^ 2) + (z ^ 2))) *
38             (
39                 K(2,a, $\rho$ ,z) +
40                 (
41                     (((a ^ 2) - ( $\rho$  ^ 2) - (z ^ 2)) /
42                     (((a -  $\rho$ ) ^ 2) + (z ^ 2))) *
43                     E(2,a, $\rho$ ,z)

```

```

44         )
45     )
46 )
47 )
48
49 # elliptic integrals of first kind (series expansion)
50 K(j,a,ρ,z, precision = 4) =
51     (pi / 2) * (
52         1 + sum([
53             ((reduce(*, [n - 1 for n in 2:2:precision]) /
54              reduce(*, [n for n in 2:2:precision])) ^ 2) *
55             (k(j,a,abs(ρ),z) ^ n)
56             for n in 2:2:precision
57         ])
58     )
59
60 # elliptic integrals of the second kind (series expansion)
61 E(j,a,ρ,z, precision = 4) =
62     (pi / 2) * (
63         1 - foldl(-,[
64             ((reduce(*, [n-1 for n in 2:2:precision]) /
65              reduce(*, [n for n in 2:2:precision])) ^ 2) *
66             ((k(j,a,abs(ρ),z) ^ n) / (n-1))
67             for n in 2:2:precision
68         ])
69     )
70
71 # the collections of variables k1 and k2
72 k(j,a,ρ,z) = (j == 1) ?
73     sqrt((4 * a * ρ) / (((a + ρ) ^ 2) + ((a - z) ^ 2))) :
74     sqrt((4 * a * ρ) / (((a + ρ) ^ 2) + (z ^ 2)))
75
76 # heights at which measurements were taken (meters)
77 heightLevels = 0.03:0.02:0.33
78
79 # current applied to the coils (amperes)
80 current = 2
81
82 # turns of wire in each coil
83 turns = 73
84
85 # radius of the coils (meters)
86 radius = 0.332
87
88 # radial distances at which measurements were taken (meters)
89 rhos = -0.22:0.02:0.22
90
91 # list of Biot-Savart functions for each height level
92 BHeights = [ futureRho -> B(heightLevel, current, turns, radius, futureRho) * 1e6
93             for heightLevel in heightLevels ]

```

```

94
95 # apply the list of radial distances to each height level's function (rhos are rows)
96 # list of lists of Bs where each internal list is a row
97 theoBs = [ map(heightLevelFunction, rhos) for heightLevelFunction in BHeights ]
98
99 # -----END GENERATE THEORETICAL DATA-----
100
101 # -----BEGIN EXTRACT EXPERIMENTAL DATA-----
102
103 # read data from CSV into a DataFrame
104 data = CSV.File("2dMappingTesla.csv") |> DataFrame
105
106 # convert row number (1:16) to height value (3:2:33)
107 rowNum2Height(row) = row + (row + 1)
108
109 # reverse the order of the incoming heights
110 heightReversal(z) = z + (16 - (z - 1) - z)
111
112 # apply corrections to data (Earth's magnetic field & correct units)
113 corrB(b) = ((b + 0.000024) * 1e6)
114
115 # extract the data from a row into a triple representing that row
116 extractRow(row) = (
117     fill(rowNum2Height(row), 23),
118     -22:2:22,
119     [ corrB(Bval) for Bval in [ data[ii][heightReversal(row)] for ii in 2:24 ] ]
120 )
121
122 # list of triples, each representing a row
123 expRows = map(extractRow, 1:16)
124
125 # -----END EXTRACT EXPERIMENTAL DATA-----
126
127 # -----BEGIN PLOT THE DIFFERENCE-----
128
129 # list of triples, each representing a row, where the Bs are (theoB-expB)
130 diffRows = [ (expRows[ii][1], expRows[ii][2], theoBs[ii] .- expRows[ii][3])
131     for ii in 1:length(expRows) ]
132
133 # now, a "row" is of the form ([zs], [rs], [Bs]); all these lists must be combined
134 tripleData = ( foldl(hcat, map(x -> x[1], diffRows)),
135     foldl(hcat, map(x -> x[2], diffRows)),
136     foldl(hcat, map(x -> x[3], diffRows)) )
137
138 # plot the difference between the theoretical and experimental data
139 diffPlot = Plots.plot(tripleData[3], tripleData[2], tripleData[1],
140     seriestype = :scatter, legend = :topright, legendtitle = "Rows",
141     title = "Difference Between Theoretical and Experimental Magnetic Field",
142     size = (550,550),
143     xlabel = "Magnetic Field Strength (uT)",

```



```
144         ylabel = "Height Relative to Bottom Coil (cm)",
145         zlabel = "Radial Distance from Axis (cm)"
146     )
147     savefig("3DPlotDiff.png")
148
149     # -----END PLOT THE DIFFERENCE-----
```

---