APPLIED PROGRAMMING LAB
(Week - 6)
Modeling Magnetic Field due to a current carrying wire

Shubhangi Ghosh
EE15B129
Department of Electrical Engineering

March 4, 2017

1 INTRODUCTION:

This report presents the computation of magnetic field, $\vec{B}$, using Biot-Savart's Law

$$d\overrightarrow{B} = \frac{\mu_0}{4\pi}\frac{I\overrightarrow{dl}x\overrightarrow{R}}{R^3} \tag{1}$$

due to current carrying wire(direction of current is known) with given dimensions.

2 BRIEF CODE EXPLANATION:

2.1 CODE:

```
#Importing header files
import numpy as np
from scipy import *
from matplotlib.pyplot import *
from scipy.integrate import quad
from math import *
from pylab import *
import mpl_toolkits.mplot3d.axes3d as p3
import sys
#returns index where value of y-index is i/10
def ret_idx(i):
return np.where((10*np.array(Y[:,0,0])).astype(int)==i)[0][0]
# declaring array of points
#top part of bent wire
points_top_bent = zeros((8,2)) # allocate a matrix with zeroes
points_top_bent[:,0] = np.arange(1.0,0.5,-0.1/np.sqrt(2)) #assign values
points_top_bent[:,1] = np.arange(-0.9,-0.4,0.1/np.sqrt(2))
#right part of wire
points_rgt = zeros((10,2)) # allocate a matrix with zeroes
points_rgt[:,0] = 0.5 #assign values
points_rgt[:,1] = np.arange(-0.4,0.6,0.1)
#top part of wire
```

```
points_top = zeros((10,2)) # allocate a matrix with zeroes
points_top[:,0] = np.arange(0.4,-0.6,-0.1)#assign values
points_top[:,1] = 0.5
#left part of wire
points_lft = zeros((10,2)) # allocate a matrix with zeroes
points_lft[:,0] = -0.5#assign values
points_lft[:,1] = np.arange(0.4,-0.6,-0.1)
#bottom part of wire
points_btm = zeros((9,2)) # allocate a matrix with zeroes
points_btm[:,0] = np.arange(-0.4,0.5,0.1)#assign values
points_btm[:,1] = -0.5
#bottom part of bent wire
points_btm_bent = zeros((8,2)) # allocate a matrix with zeroes
points_btm_bentx = np.arange(0.9,0.4,-0.1/np.sqrt(2))#assign values
points_btm_benty = np.arange(-1.0,-0.5,0.1/np.sqrt(2))
points_btm_bent[:,0] = points_btm_bentx[::-1] #reversing array since I've started count
# from bottom for simplicity whereas it is supposed to be counted the other way round
points_btm_bent[:,1] = points_btm_benty[::-1]
#concatenate all points
points = np.concatenate((points_top_bent,points_rgt,points_top,points_lft,points_btm,po
#find centers
centers = (points[:-1]+points[1:])/2
#concatenate with zero column z-axis
centers = np.concatenate((centers,np.array([zeros(len(centers))]).transpose()),axis=1)
#find current direction by subtarcting
curr_dir=(points[1:]-points[:-1])*10
#concatenate with zero column z-axis
curr_dir = np.concatenate((curr_dir,np.array([zeros(len(curr_dir))]).transpose()),axis=
#plot points, centers and current direction
fig = figure()
title('Current direction in wire')
xlim(-1.0,1.1)
ylim(-1.0,1.1)
xticks(np.arange(-1.0,1.1,0.2))
yticks(np.arange(-1.0,1.1,0.2))
grid()
plot(points[:,0],points[:,1],'kx')
plot(centers[:,0],centers[:,1],'ro')
quiver(points[:-1,0],points[:-1,1],curr_dir[:,0],curr_dir[:,1],color='b')
 #quiver plot for current direction
legend(('Points on wire','Centers'))
savefig('current.pdf',format='pdf')
show()
x=np.arange(-1.0,1.1,0.1) # create x and y and z axes
y=np.arange(-1.0,1.1,0.1)
z=np.arange(-1.0,1.1,0.1)
X,Y,Z=meshgrid(x,y,z) # creates arrays out of x, y and z
# Field at point due to itself is assumed to be zero
#creating array of the size of meshgrid to store B in directions x,y and z
Bx=np.array(zeros((len(x),len(y),len(z))),dtype=float)
By=np.array(zeros((len(x),len(y),len(z))),dtype=float)
Bz=np.array(zeros((len(x),len(y),len(z))),dtype=float)
#evaluating field due to all the centers of wires one by one
```

```
for i,j in zip(centers,curr_dir):
#position vector R
Rx = X-i[0]
Ry = Y-i[1]
Rz = Z-i[2]
#magnitude of R
R = (Rx**2+Ry**2+Rz**2)**(0.5)
#masking values where R is nearly 0
R=np.ma.masked_where(R<0.00001, R)
Rx=np.ma.masked_where(R<0.00001, Rx)
Ry=np.ma.masked_where(R<0.00001, Ry)
Rz=np.ma.masked_where(R<0.00001, Rz)
#dB due to current element
dBx=(10.0**(-5)*j[1]*Rz)/(R**3)
dBy=(10.0**(-5)*-j[0]*Rz)/(R**3)
dBz=(10.0**(-5)*(j[0]*Ry-j[1]*Rx))/(R**3)
#adding all dB to get total field
Bx=Bx+dBx
By=By+dBy
Bz=Bz+dBz
#plotting B at y=0.0
title(r"Arrow plot of $\vec{B}$ along the $x-z$ plane")
xlabel('$x$')
ylabel('$z$')
quiver(X,Z,Bx[ret_idx(0)],Bz[ret_idx(0)],scale=0.01)
savefig('B(y=0.0).pdf',format='pdf')
show()
#plotting B at y=-0.3
title(r"Arrow plot of $\vec{B}$ along the plane $y=-0.3cm$")
xlabel(r'$x$')
ylabel(r'$z$')
quiver(X,Z,Bx[ret_idx(-3)],Bz[ret_idx(-3)],scale=0.01)
savefig('B(y=-0.3).pdf',format='pdf')
show()
#plotting B at y=-0.5
title(r"Arrow plot of $\vec{B}$ along the plane $y=-0.5cm$")
xlabel(r'$x$')
ylabel(r'$z$')
quiver(X,Z,Bx[ret_idx(-5)],Bz[ret_idx(-5)],scale=0.01)
savefig('B(y=-0.5).pdf',format='pdf')
show()
#plotting B at y=-0.7
title(r"Arrow plot of $\vec{B}$ along the plane $y=-0.7cm$")
xlabel(r'$x$')
ylabel(r'$z$')
quiver(X,Z,Bx[ret_idx(-7)],Bz[ret_idx(-7)],scale=0.01)
savefig('B(y=-0.7).pdf',format='pdf')
show()
```

2.2 FUNCTIONS USED IN CODE:

1. $ret\_idx(i)$:
   This function returns the meshgrid points of Y(*i.e.* y-axis points) where $y = 0.1 * i$.

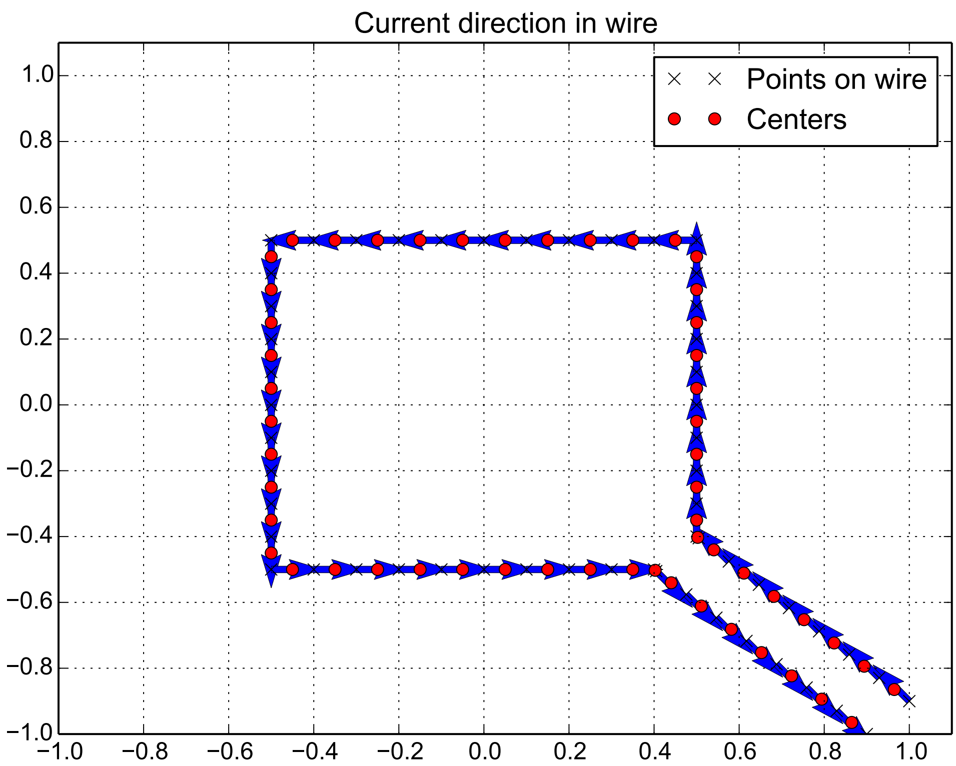
2.3 BRIEF CODE EXPLANATION:

The following steps have been followed in the code:

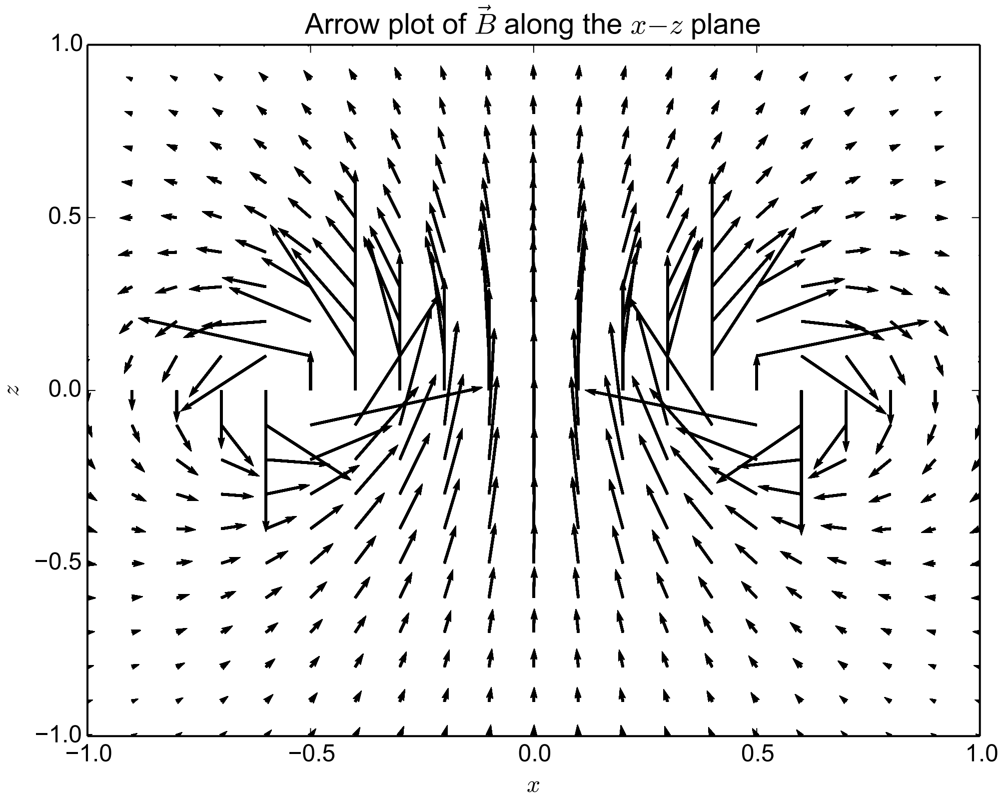
1. First, we define points on the wire spaced at a length of $0.1cm$ in an array.
2. At the bent portion of the wire, we assume the wire length to be a little shorter than $0.1cm$.
3. Next, we define an array consisting of centers of the wire segments, by taking average of the coordinates of the points.
4. We concatenate the centers with the z-axis column, $i.e. \, z = 0$ column.
5. We define an array consisting of current directions in the wire segments, by taking difference of the coordinates of the points, and multiplying by 10 to make the magnitude of the vector $=1$.
6. We concatenate the current direction vector with the z-axis column, $i.e. \, z = 0$ column.
7. We construct the quiver plot of the points, centers and current directions in the wire.
8. Then, we create a meshgrid of $x-$, $y-and\, z-$axes, with $x = -0.1cm\, to\, 0.1cm$, $y = -0.1cm\, to\, 0.1cm\, and\, z = -0.1cm\, to\, 0.1cm$.
9. We evaluate flux density or magnetic field intensity, $\overrightarrow{B}$ at all the points in the meshgrid, using Biot-Savart's Law, as in Eqn. (1), by iterating through all the wire segments.
10. The points which lie on the centers of the wire segments $i.e. \, \overrightarrow{r} = 0$, are masked in the meshgrid to prevent division-by-zero error.
11. We then construct quiver plots of $\overrightarrow{B}$ on the $x-z\, plane$, $y = -0.3cm$, $y = -0.5cm\, and\, y = -0.7cm$.

2.4 PLOTS:

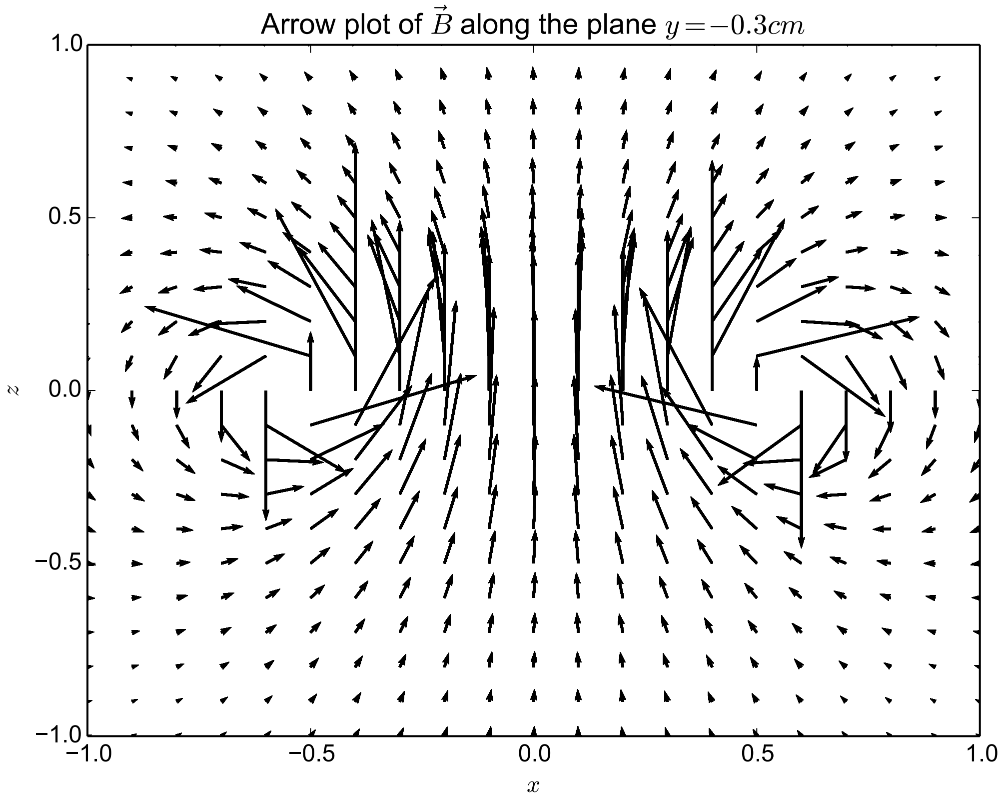
2.4.1 Quiver Plot of Current in the wire



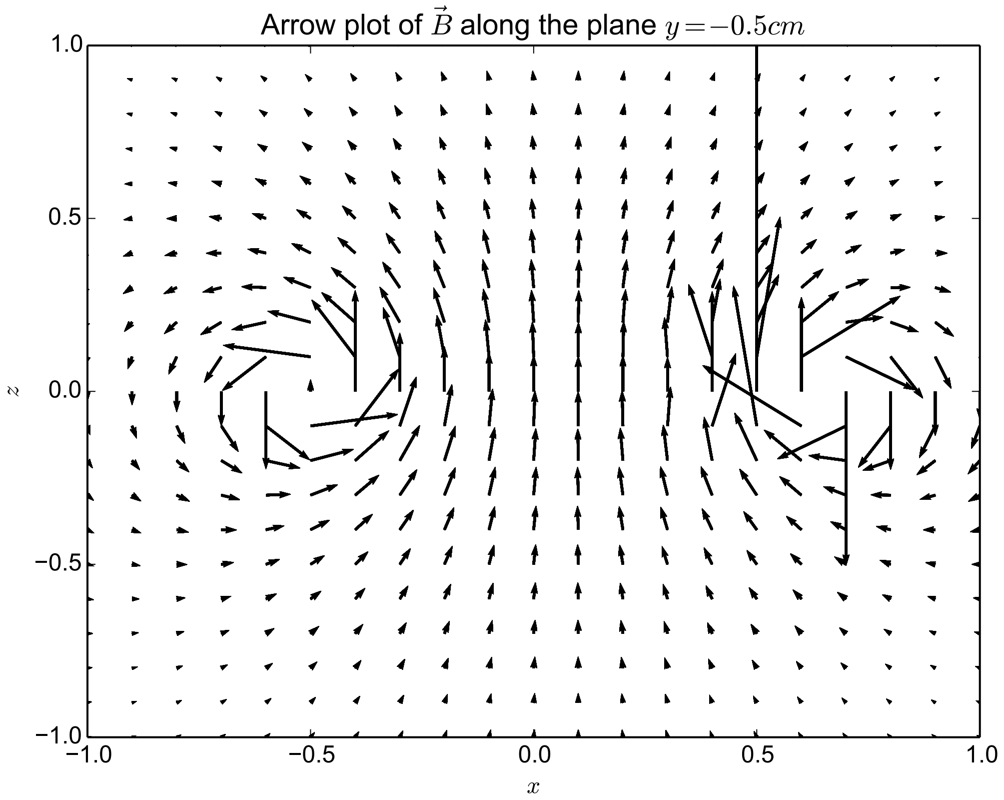
2.4.2 Quiver Plot of $\overrightarrow{B}$ on $x - z$ plane



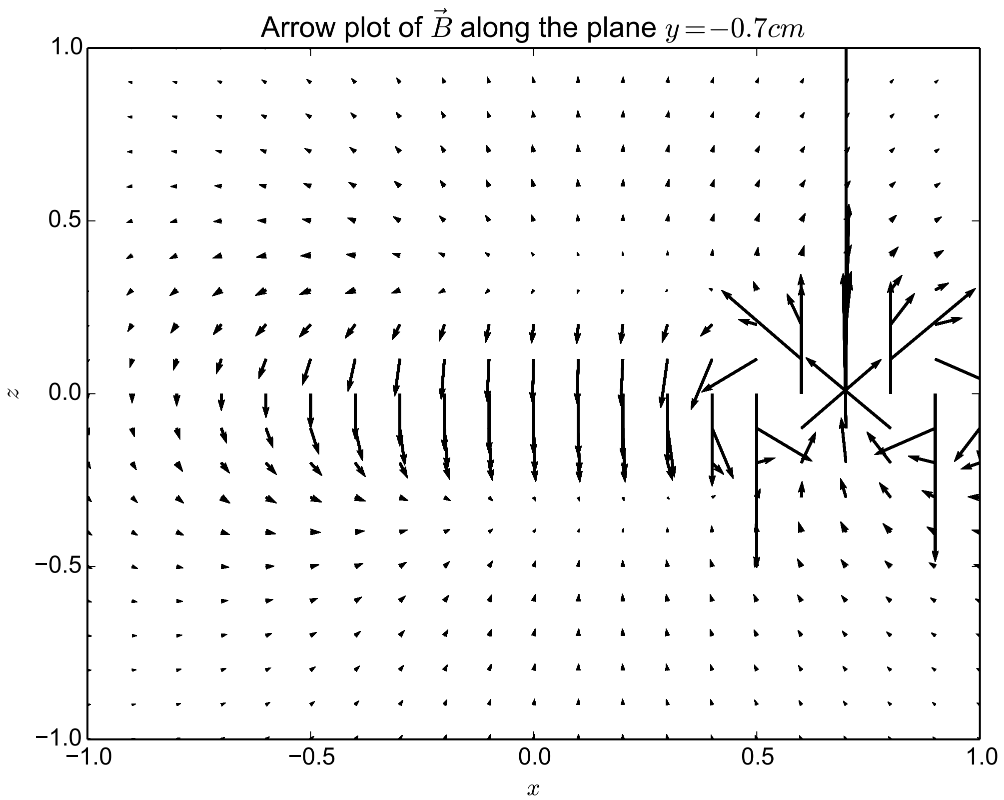
2.4.3 Quiver Plot of $\overrightarrow{B}$ on $y = -0.3cm$ plane



2.4.4 Quiver Plot of $\overrightarrow{B}$ on $y = -0.5cm$ plane



2.4.5 Quiver Plot of $\overrightarrow{B}$ on $y = -0.7cm$ plane



3 RESULTS AND DISCUSSION:

When we construct the quiver plot of $\overrightarrow{B}$, we clearly observe that Ampere's circuital law is obeyed *i.e.* the field forms loops around the point where the wire cuts the plane(observe all the plots of field, $\overrightarrow{B}$ and compare with the first plot which indicates the position of the wire.