Python Week 6: Using Biot Savart to compute $\overline{B}$ Fields

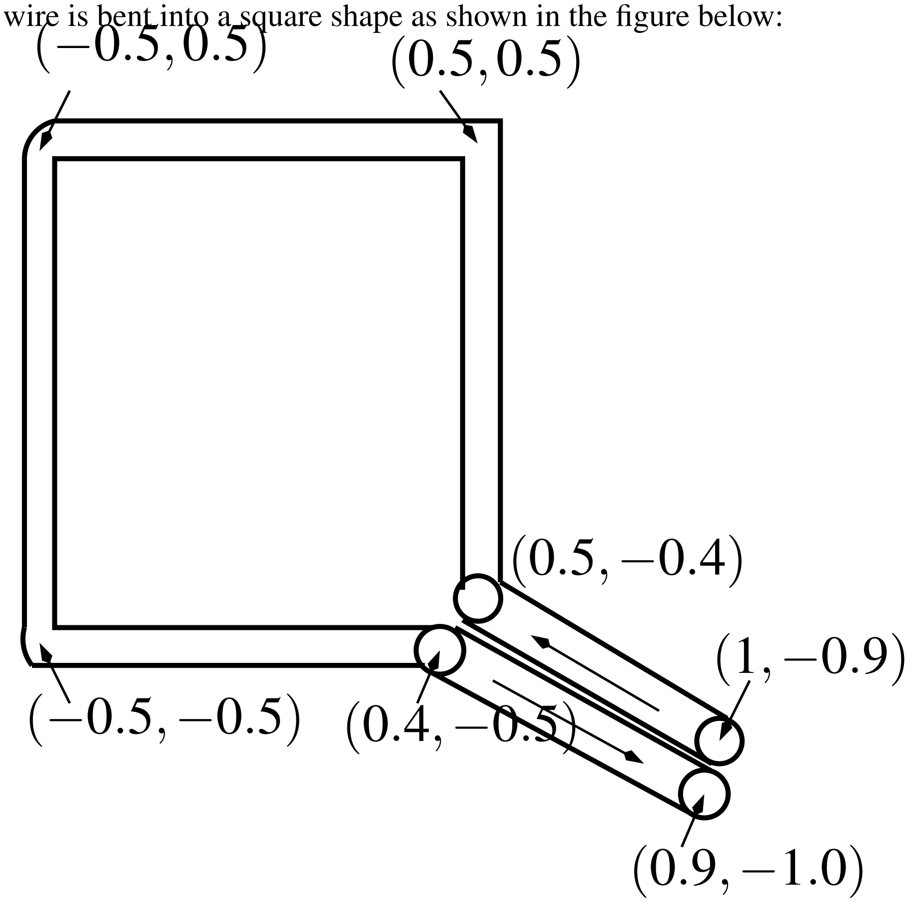
February 19, 2017

Introduction

In this assignment, we compute the Magnetic Field due to current in a wire. We break up the wire into little pieces, and for each, we use the Biot Savart Law to compute $\vec{B}$ at different points. We sum up all the contributions to get the $\vec{B}$ field due to the coil.

Geometry of Problem

A wire is bent into a square shape as shown in the figure below:



The locations are shown in cm. The radius of the wire is 0.05 cm. A current of 1 Amp flows through the wire. The problem is to compute the magnetic field in a cube that goes from $-1 < x, y, z < 1$.

Solution Process

- ▷ Define the line passing through the centre of the wire where it enters at $(1, -0.9)$ going round the square till it leaves at $(0.9, -1)$. This length is $0.9 + 1 + 1 + 0.9 + 0.5/\sqrt{2} + 0.5/\sqrt{2}$ cm.

- ▷ Divide the line into pieces of length 0.1 cm. The leads can have lengths slightly longer to make them integer number of pieces.

- ▷ Create the points corresponding to the ends of the pieces into an array in Python. For instance, along the vertical wire it would be

$$\begin{pmatrix} \cdots & \cdots & \\ 0.6 & -0.5 & \text{on bent wire} \\ 0.5 & -0.4 & \text{on vertical wire} \\ 0.5 & -0.3 & \text{on vertical wire} \\ \cdots & \cdots & \\ 0.5 & 0.4 & \text{on vertical wire} \\ 0.5 & 0.5 & \text{on vertical wire} \\ 0.4 & 0.5 & \text{on horizontal wire} \\ \cdots & \cdots & \end{pmatrix}$$

  A single array should go from $(1, -0.9)$ to $(0.9, -1)$

- ▷ Locate the centres of the individual pieces. For the fragment above, these would be

$$\begin{pmatrix} \cdots & \cdots & \\ 0.55 & -0.45 & \text{on bent wire} \\ 0.5 & -0.35 & \text{on vertical wire} \\ 0.5 & -0.25 & \text{on vertical wire} \\ \cdots & \cdots & \\ 0.5 & 0.45 & \text{on vertical wire} \\ 0.45 & 0.5 & \text{on horizontal wire} \\ \cdots & \cdots & \end{pmatrix}$$

Note that there will be one less piece than the number of ends - your array row dimension will be one less. **Use vector operations!**

▷ Pad the centres with a zero column, representing the z coordinate

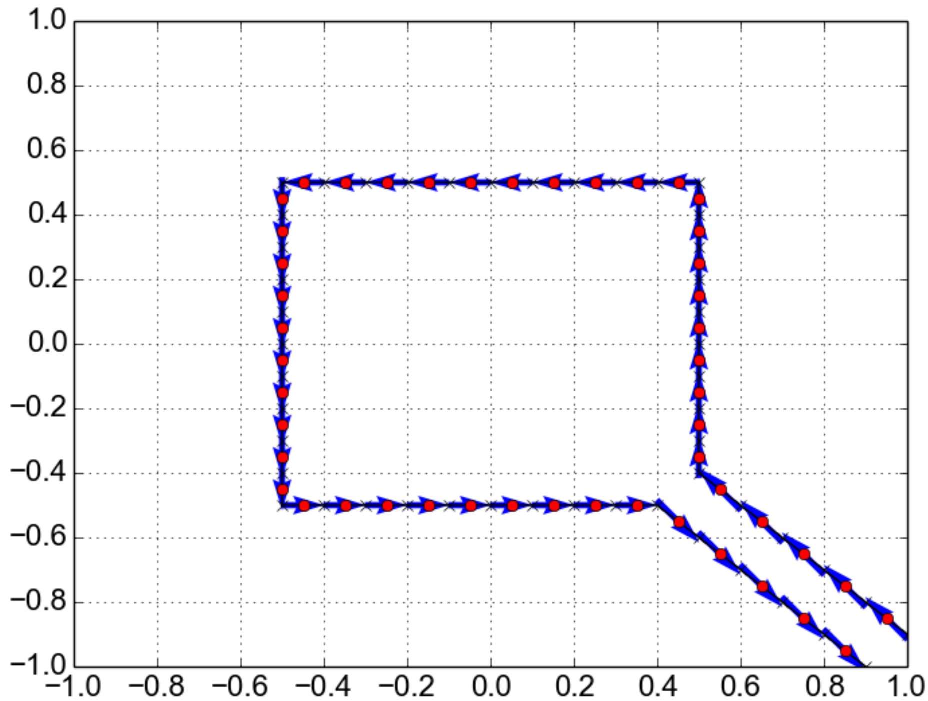
$$\begin{pmatrix} \cdots & \cdots & \cdots \\ 0.55 & -0.45 & 0 \\ 0.5 & -0.35 & 0 \\ 0.5 & -0.25 & 0 \\ \cdots & \cdots & \cdots \\ 0.5 & 0.45 & 0 \\ 0.45 & 0.5 & 0 \\ \cdots & \cdots & \cdots \end{pmatrix}$$

These are the locations of the centres of the wire pieces.

▷ Create a vector of current directions (the amplitude is unity)

$$\begin{pmatrix} \cdots & \cdots & \cdots & \\ -1/\sqrt{2} & 1/\sqrt{2} & 0 & \text{on bent wire} \\ 0 & 1 & 0 & \text{on vertical wire} \\ 0 & 1 & 0 & \text{on vertical wire} \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 1 & 0 & \text{on vertical wire} \\ -1 & 0 & 0 & \text{on horizontal wire} \\ \cdots & \cdots & \cdots & \end{pmatrix}$$

You should plot the end locations (with crosses), the centres with red dots and the current arrows in blue (use quiver, right over the plot) and get the following figure:



- ▷ Now create an array of points in (x,y,z) that are 0.1 spaced. this will be a $3-D$ array for each of x, y and z, and each array will have 21^3points. To do this, create each as a 1-Dimensional vector and use *meshgrid*.

- ▷ Create arrays to hold B at these points. (Dimension will be $21\times21\times21\times3$)

- ▷ Now implement the Biot Savart Law at each point (x_i, y_j, z_k), due to the m^{th} wire element:

$$d\vec{B} = \frac{\mu_0}{4\pi}\frac{I\vec{dl}\times\vec{R}}{R^3}$$

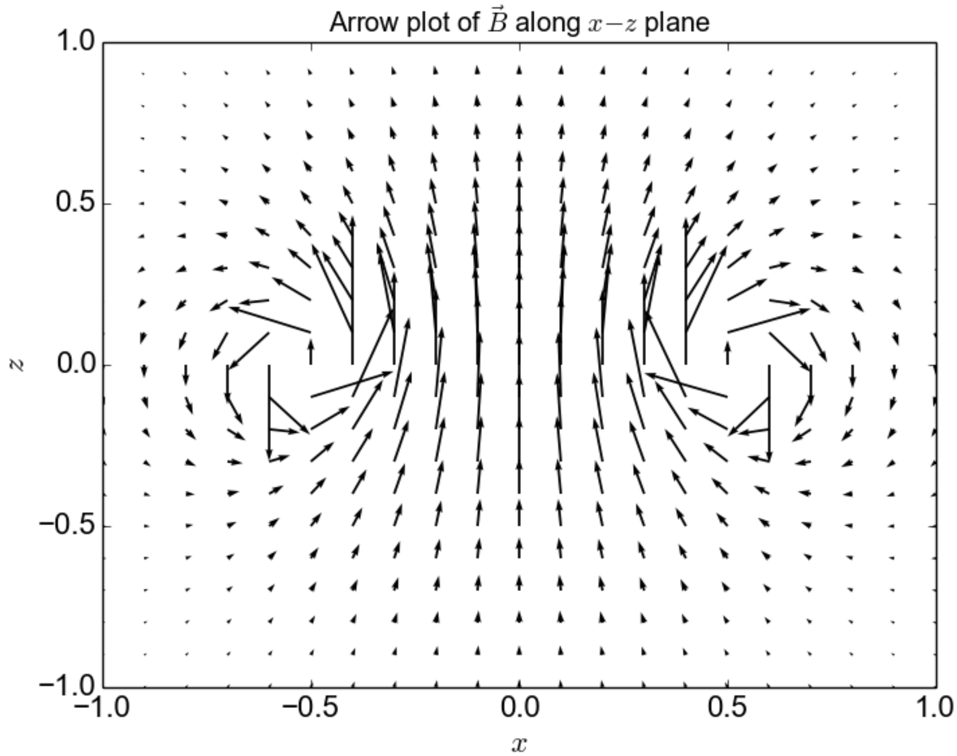
where

$$\vec{R} = (x_i - x_m)\hat{x} + \left(y_j - y_m\right)\hat{y} + (z_k - z_m)\hat{z}$$

This should be implemented using vector operations, or your code will take far too long to run. Remember that you have about 50 wire pieces, and 9261 places where you will compute $\vec{B}$. You can look up the *cross* command or else implement the cross product yourself. If you do it yourself, you can take advantage of the fact that the current is purely in the $x-y$ plane.

Note: First compute the array of $\vec{R}$ and R values. Find where they are not zero, so that when you divide by R^3 you compute the $d\vec{B}$ only where R is not zero.

▷ Do an arrow plot of the field in the $x-z$ plane, and in vertical planes corresponding to $y=-0.3$, $y=-0.5$ and $y=-0.7$. Explain what you see. Below is what I got for $y=0$:



This method can be used in general. Usually the computation is speeded up by having pre-computed formulae for straight pieces of wire and arcs.

Since the computation is parallel across all the points for a given $I\vec{dl}$ element, this algorithm is easily speeded up in a graphics processor using CUDA.