Overview of BiotSavart

M.W. Reynolds*
Ripplon Software Inc., 4794 Fernglen Drive, Burnaby, BC, Canada V5G 3V7
www.ripplon.com

October 2, 2007

1 Introduction

BiotSavart is a magnetic field calculating program developed by Ripplon Software Inc. It has been available for Macintosh computers since 1991 and for Windows computers since 2007. This note describes version 4.0 of BiotSavart, which was released July 2007.

BiotSavart can calculate the magnetic field, force, and linked flux, for a three-dimensional system of conductors with specified current. It does this in an intuitive and interactive way that makes it valuable for designers who wish to explore various possibilities. BiotSavart has been used for diverse applications including magnetic traps for neutral atoms and molecules (e.g., [1]), magnetic resonance imaging [2], and antennae for RFID [3].

The field calculations in BiotSavart are founded on closed-form expressions for the magnetic vector potential $\boldsymbol{A}$ and the magnetic flux density $\boldsymbol{B}$ sourced by loops and line segments. These fundamental sources are summed to represent more complicated conductor objects. These conductors may be arbitrarily placed in three dimensions. The conductors are connected to current supplies so that the current through several conductors may be varied together. A bias magnetic field can be superimposed on the configuration. The resulting magnetic field may be sampled and displayed in various ways, using probe objects. In the next sections describe the available conductors and probes.

*E-mail: meritt@ripplon.com

2 Conductor types

The available conductor types are Loop, Solenoid, Revolved, Wire, and Racetrack.

The **Loop** object is a collection of an arbitrary number of coaxial loops. These loops share a common wire diameter.

The **Solenoid** object is a coil of rectangular cross section. The shape of the solenoid is specified by giving its inner radius, outer radius, and length. Setting the inner radius equal to the outer radius gives a thin solenoid. Setting the length equal to zero gives a pancake solenoid. The number of turns is specified, or it can be calculated from a given wire diameter. Conversely, it is possible to calculate the wire diameter required to give a desired number of turns.

The **Revolved** object is surface created by revolving an arbitrary path about an axis. The current that flows on the surface of this object is represented by current loops coincident with the surface. The surface current density can be chosen to be distributed as the current density of an axially magnetized body or to be distributed uniformly along the surface.

The **Wire** object is a sequence of straight line segments that approximates current flow along an arbitrary path through space. A simple path description language is used to describe the path, which can include gentle turns and spirals.

The **Racetrack** object is a bulky conductor shaped like an elongated solenoid. It is used in particle accelerators and magnetic traps.

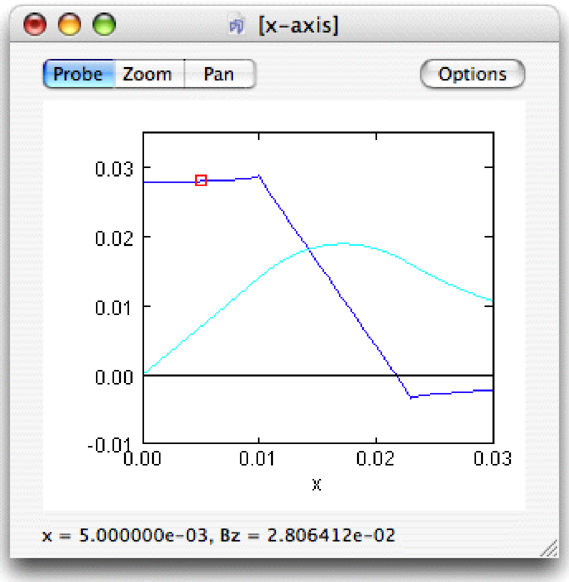

Figure 1: Linear probe window showing the profile of magnetic flux density component B_z and vector potential component A_y through the thickness of a solenoid. The numbers at the bottom describe the selected point.

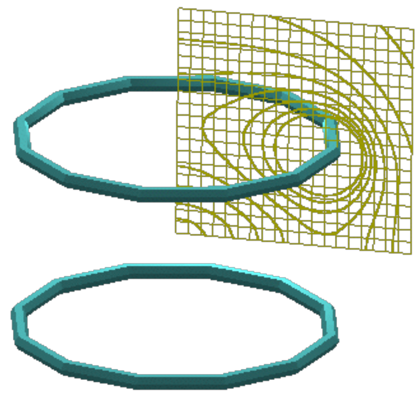

Figure 2: Loop object (with two loops of opposing current) modeling a magnetic trap. A Planar probe is being used here to plot contours of the trapping potential.

3 Probe types

The objects that calculate magnetic fields are called probes. The available probe types are Tracer, Linear, Planar, and Volumetric.

The **Tracer** probe object calculates the path of a magnetic field line starting from a specified point in space.

The **Linear** probe object generates a plot of the field versus coordinate along an arbitrary path (usually taken to be a straight line). Quantities that can be plotted are A_x, A_y, A_z, $|\boldsymbol{A}|$, B_x, B_y, B_z, $|\boldsymbol{B}|$, or a function specified by the macro language. The plots are interactive, in the sense that by clicking with the mouse on any point the program will display the value of the quantity plotted. Scrolling through the values with the arrow keys is also possible.

The **Planar** probe object generates a contour plot of the field on a specified planar surface. Any of the quantities described above may be plotted. The contour plot appears superimposed on the graphical display of the conductors, as shown in figure 2. It may also be plotted in a window of its own.

The **Volumetric** probe object calculates the magnetic field in a rectangular volume. From this data it can display arrows indicating the direction and magnitude of either the vector potential $\boldsymbol{A}$ or the magnetic flux density $\boldsymbol{B}$. It can also display a level-set surface of any quantity. A slider control lets the value of the level-set be adjusted continuously in real time.

4 Field line tracing

To trace field lines, the **Tracer** probe object uses a Runge-Kutta scheme to integrate the fundamental equation

$$\frac{d\boldsymbol{x}}{ds} = \frac{\boldsymbol{B}}{|\boldsymbol{B}|}$$

where s is the arc length along the field line. The length of the field line and the integration step size are under user control. This produces a field line that is superimposed on the configuration display as shown in figure 3.

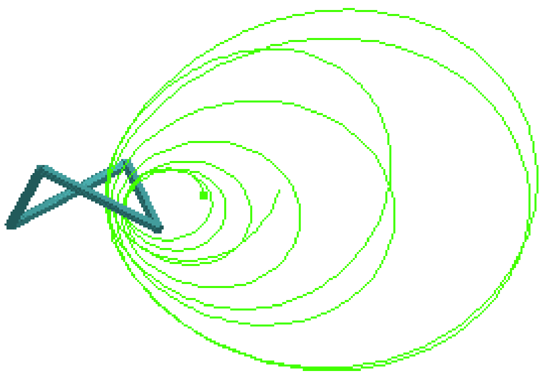

Figure 3: Magnetic field line of a square loop bent out of its plane. The field line is traced by a Tracer object.

5 Inductance calculations

The flux Φ linked by a conductor is obtained from the vector potential by the volume integral

$$\Phi = \int dV\, \boldsymbol{A} \cdot \boldsymbol{W}$$

where $\boldsymbol{W}$ is the winding density vector (if the current in the conductor were set to I the current density would be $I\boldsymbol{W}$). The integral is represented by a sum over what are known as the M points, whose density can be adjusted by the user. For a filamentary conductor the M points are distributed along the length of the conductor. For a solenoid the M points are distributed across the cross section in a rectangular grid, and also along the current path in angular steps through the full circle of rotation. The grid density in the cross section and angular direction can be specified by the user. For an axially symmetric conductor, like a solenoid, also defined are the L points which are the M points of one radial section; the L points are sufficient to calculate self-inductance.

The self-inductance of a conductor is found by calculating the linked flux Φ when only that conductor has current in it; the self inductance L is then given by $L = \Phi/I$ where I is the current in the conductor. There is a button for this calculation in BiotSavart.

The mutual inductance between two conductors 1 and 2 can be calculated by setting the current in all conductors except conductor 1 equal to zero, setting $I_1 = 1\,\mathrm{A}$, and calculating the flux Φ_{21} linked by conductor 2. The mutual inductance M_{21} is then given by $M_{21} = \Phi_{21}/I_1$. BiotSavart provides a button to perform the linked-flux calculation. In the case of coaxial solenoids, mutual inductance calculated by BiotSavart agrees to within 0.5% with the values calculated for all examples given by Babic and Akyel [4].

BiotSavart calculates linked flux flux (and hence mutual inductance) even in geometries that are not axially symmetric.

6 Force and torque

A conductor in magnetic field experiences a force $\boldsymbol{F}$ and torque $\boldsymbol{N}$ given by

$$\boldsymbol{F} = \int dV\, \boldsymbol{J} \times \boldsymbol{B}$$

and[1]

$$\boldsymbol{N} = \int dV\, (\boldsymbol{x} - \boldsymbol{x}_0) \times (\boldsymbol{J} \times \boldsymbol{B})$$

where dV is a volume element of the conductor, $\boldsymbol{J}$ is the current density in the volume element, and $\boldsymbol{B}$ is the magnetic flux density there. Since a conductor can not exert a force or torque on itself, the integrals above are performed using the magnetic field generated by other conductors. Numerically the integration is performed in a manner analogous to the calculation of linked flux, by summing over the M points.

[1] The magnetic torque acting on a conductor is given by BiotSavart calculates the torque about the "current center" defined by

$$\boldsymbol{x}_0 = \int dV\, \boldsymbol{x}|\boldsymbol{J}| \Big/ \int dV\, |\boldsymbol{J}|$$

For symmetrical conductors $\boldsymbol{x}_0$ is the center of symmetry.

7 Macro language

The macro or calculator language in BiotSavart is used to plot user-specified functions of position and field. It is an operator language characterized by the absence of any hierarchy. Operations are grouped right-to-left, as in the language APL. Built-in operators include most of the C math library functions, with the same name (`sin`, `sqrt`, etc.). An example of a mathematical expression and its equivalent macro expression:

$\sqrt{xA_y - yA_x}$ `sqrt (x*Ay)-y*Ax`

This produces field lines in an axially symmetric system if plotted as contours on a surface that is a radial section (e.g., the x-z plane). The square root ensures that the contours are uniformly distributed where the field is uniform.

Quantities available for use in macro expressions are the components and magnitudes of the magnetic field (`Bx`, `By`, `Bz`, `B`, `Ax`, `Ay`, `Az`, `A`) and position (`x`, `y`, `z`), as well as arc length (`s`) for linear probes.

Macro expressions of more than one line evaluate line by line, and the final line is the result that is used. This allows setup of variables to be used in the final line. For example, a macro to plot the potential energy of a Cs-133 atom in a magnetic trap (including gravity) in units of micro-Kelvin is:

```
m=132.905*1.67e-27
g=9.81
mu=9.27e-24
k=1.38e-23
1e6*((mu*B)-m*g*z)/k
```

8 Development

Development of BiotSavart is focusing on magnetic materials, a feature that will be released at version 4.1. This represents a major leap in the evolution of the code, because to solve magnetic materials requires a self-consistent calculation of the magnetization induced in the magnetic material (magnetization generates magnetic field that, in turn, influences magnetization). The model we have adopted initially is to dice the space containing the magnetic material into cubic cells, some partially filled by a polyhedron. If in each cell the magnetization is uniform then the generated fields $\boldsymbol{A}$ and $\boldsymbol{B}$ can be calculated in closed form. At long range a dipole approximation is adequate. The field at any point is calculated by summing over all cells, which is done efficiently using a hierarchical multipole method. The self-consistency requirement is that the magnetization $\boldsymbol{M}$ should be equal to the equilibrium $\boldsymbol{M}(\boldsymbol{B})$ for the calculated $\boldsymbol{B}$. A conjugate gradient algorithm is used to find the self-consistent solution, also in the presence of non-linear materials.

9 About the author

Meritt Reynolds is the author of the software products of Ripplon Software Inc. He has a Ph.D. in experimental physics and has extensive experience working in both academia and industry.

References

[1] Jonathan David Weinstein. *Magnetic Trapping of Atomic Chromium and Molecular Calcium Monohydride.* PhD thesis, Harvard University, 2001.

[2] R. W. Mair, M. I. Hrovat, S. Patz, M. S. Rosen, I. C. Ruset, G. P. Topulos, L. L. Tsai, J. P. Butler, F. W. Hersman, and R. L. Walsworth. ^{3}He lung imaging in an open access, very-low-field human magnetic resonance imaging system. *Magnetic Resonance in Medicine*, 53:745–749, 2005.

[3] Ben Tuppen. Simulation of antennae for radio frequency identification (RFID). 2005.

[4] Slobodan I. Babic and Cevdet Akyel. New analytic-numerical solutions for the mutual inductance of two coaxial circular coils with rectangular cross section in air. *IEEE Transactions on Magnetics*, 42:1661, 2006.