



**PerMag**  
**Finite-element Magnetostatic Field  
Solutions with Permanant Magnets**

**Field Precision**  
Copyright 2003

PO Box 13595  
Albuquerque, New Mexico 87192 U.S.A.  
Telephone: 505-220-3975  
FAX: 505-294-0222  
E Mail: [techinfo@fieldp.com](mailto:techinfo@fieldp.com)  
Internet: <http://www.fieldp.com>

# PerMag 5.0 Manual - Contents

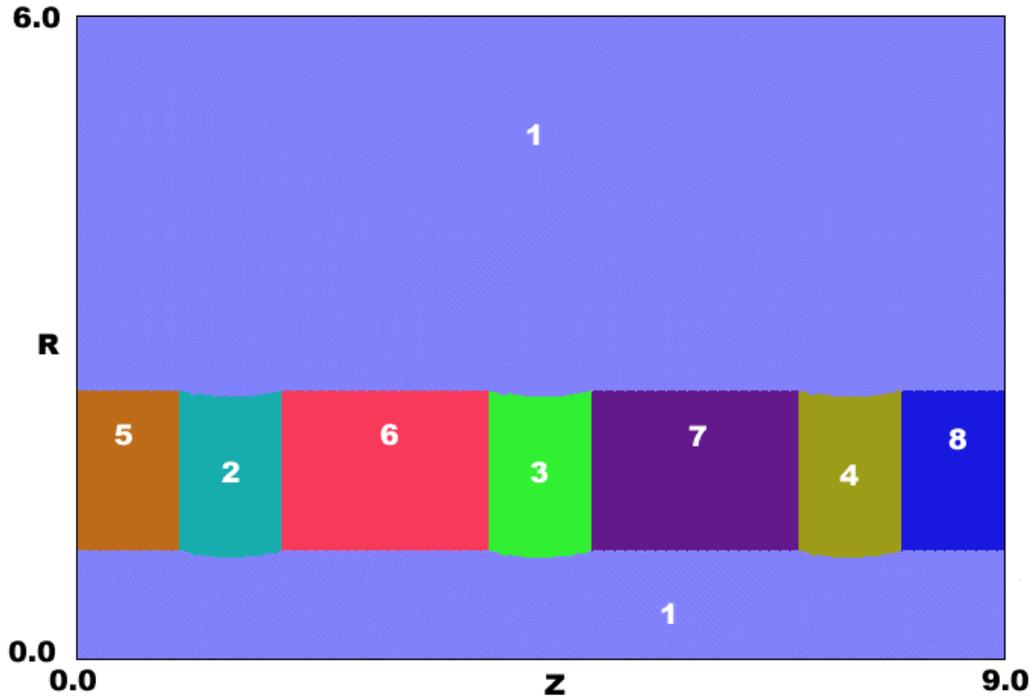
- Section 1. Program function
- Section 2. Walkthrough example
- Section 3. Organizing runs
- Section 4. Structure of the command file
- Section 5. Commands for general program control
- Section 6. General region commands
- Section 7. Region commands for magnetic materials
- Section 8. Modeling non-linear magnetic materials
- Section 9. Physical properties of permanent magnet materials
- Section 10. Region commands for permanent magnet materials
- Section 11. Vector potential and boundary conditions
- Section 12. Boundary and superposition commands
- Section 13. Running PerMag interactively
- Section 14. Running PerMag from the command prompt
- Section 15. PerView - file operations
- Section 16. PerView - spatial plots
- Section 17. PerView - analysis functions
- Section 18. PerView - scan plots
- Section 19. PerView - script operation
- Section 20. PerMag output file format
- Section 21. QuikPlot utility

## 1. Program function

**PerMag** is a versatile numerical tool for the design of permanent magnet devices. The program calculates magnetostatic fields in complex geometries with coils, linear or non-linear ferromaterials, anisotropic materials and permanent magnets. **PerMag** handles the full spectrum of permanent magnet materials including those with non-linear demagnetization curves, such as Alnico and ceramics. **PerMag** applies finite-element methods on a variable-resolution conformal triangular mesh. The program has dynamic memory allocation so that the number of nodes is limited only by the installed memory of the computer. **PerMag** handles three-dimensional cylindrical problems (azimuthal symmetry) and two-dimensional planar problems (variation in  $x$  and  $y$  with infinite extent in  $z$ ). Using a simple command language, you can define up to 127 regions to represent materials or coils. The **PerView** interactive graphical analysis program provides full information on fields and magnetic permeability. Other features include automatic calculation of field energy and energy-product arranged by region. The program makes screen and hardcopy plots of magnetic field lines, field magnitude and relative permeability.

**PerMag** includes all functions of the **TriComp BStat** program and has several additional capabilities.

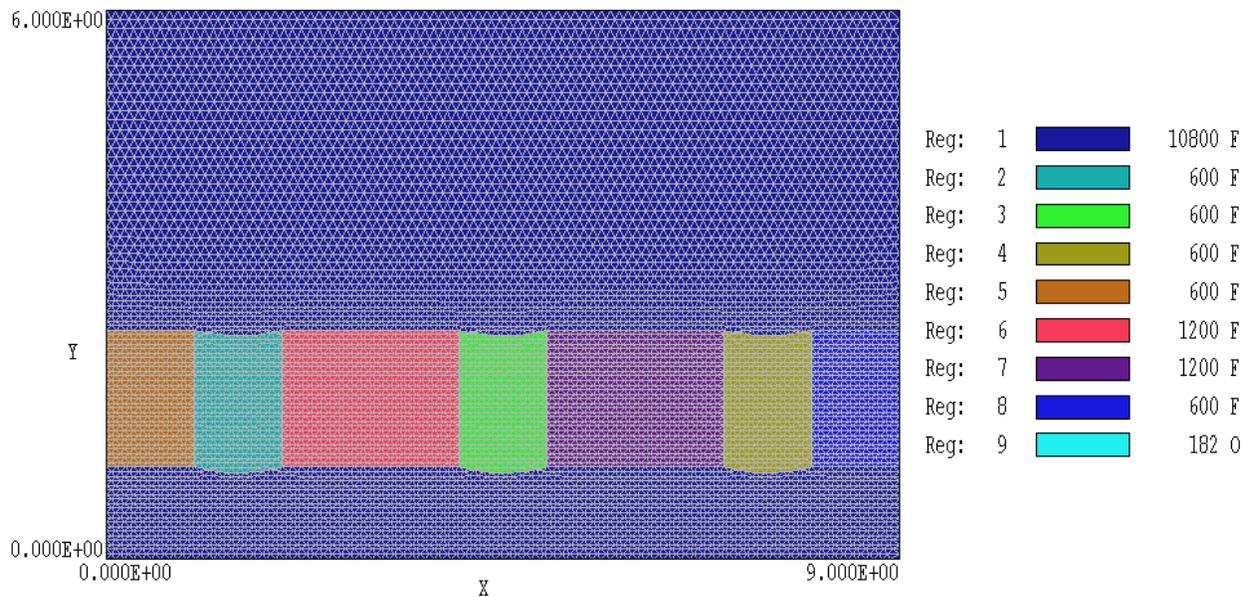
- Handles anisotropic magnetic materials with different values of magnetic permeability along two perpendicular axes. Includes the option for non-linear variations along each axis.
- Treats permanent magnets in three limits: ideal materials, two-point demagnetization curves and complete non-linear demagnetization curves.
- Includes routines for automatic analysis of average energy-product for permanent magnet regions.



**Figure 1.** Periodic permanent magnet stack.

## 2. Walkthrough example

To test the installation and to get acquainted with **PerMag** procedures we shall follow an example that demonstrates many of the program features. We want to calculate the magnetic fields generated by a *periodic-permanent magnet* (PPM) array. This configuration, used for electron beam transport in high-power traveling wave tubes, consists of a set of toroidal permanent magnets with the magnetization axis along  $z$ . The magnetization direction reverses between adjacent magnets, producing a solenoidal field on-axis that varies periodically. Figure 1 shows the geometry. Regions 5, 6, 7 and 8 are magnets with magnetization in  $-z$ ,  $+z$ ,  $-z$  and  $+z$  respectively. To model an infinitely long system, we represent Regions 5 and 8 as half magnets and apply symmetry conditions on the left and right boundaries. Regions 2, 3 and 4 are shaped soft-iron inserts to conduct field lines radially. Region 9 (not shown) is composed of lines along the upper and lower boundaries. This region is used to set a condition on boundary nodes.



**Figure 2.** Mesh for the PPM program

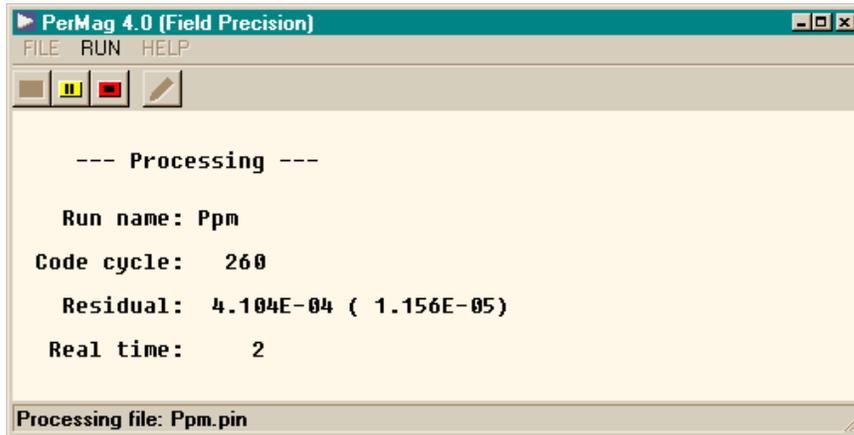
**Note.** In cylindrical systems the **TriComp** programs plot the geometry in  $z$  and  $r$  (where  $r$  must be greater than or equal to zero). The plot of Figure 1 is not a cross-section cut in the  $x$  or  $y$  plane, so there is no mirror set of objects below the  $z$ -axis.

We need four input data files for the run. The file `PPM.MIN` (**M**esh **I**nput) describes the geometry of Fig. 1 and `PPM.PIN` (**P**erMag **I**nput) sets material properties and controls the field solution. A tabular data file (`ALNICO5.MAT`) is required to define the properties of the permanent magnets. Move the files from the `/TRICOMP/EXAMPLE` directory to a working directory such as `/TRICOMP/BUFFER`. The **Mesh 5.0** instruction manual describes the syntax and commands of the MIN file. Run **Mesh** and pick the command *Load script (MIN)*. In the dialog, move to working directory (if necessary) and pick the file `PPM.MIN`. Next, choose the command *Process*. Click the right mouse button when the program is finished and then choose the *Save mesh (MOU)* command. This program creates the file `PPM.MOU` (**M**esh **O**utput) which is required for the **PerMag** run. Figure 2 shows the resulting mesh. At this point you can experiment with the plot capabilities of **Mesh** or minimize or close the program.

**Table 1. File PPM.PIN**

```
* FILE PPM.PIN
SET Geometry Cylin
SET DUnit 100.0
SET ResTarget 5.0E-7
SET Omega 1.80 1.90
SET MaxCycle 2500
SET Update 20 100
SET Avg 0.05
* Main solution volume
REGION 1 Mu 1.0
* Iron01
REGION 2 Mu 500.0
* Iron02
REGION 3 Mu 500.0
* Iron03
REGION 4 Mu 500.0
* Alnico -
REGION 5 PerMag Table Alnico5.Mat 180.0
* Alnico +
REGION 6 PerMag Table Alnico5.Mat 0.0
* Alnico -
REGION 7 PerMag Table Alnico5.Mat 180.0
* Alnico +
REGION 8 PerMag Table Alnico5.Mat 0.0
* Dirichlet boundary
REGION 9 VecPot 0.0
ENDFILE
```

Next start PerMag .EXE (Figure 3). To begin we shall inspect the input control file. Click on *File* and then *Edit input files*. Change directories if necessary and pick the file PPM .PIN. The editor shows the text file reproduced in Table 1. The first group of commands that begin with the keyword *SET* control program operation. The first two specify that the geometry is cylindrical and that dimensions in the mesh input file were specified in centimeters. The next five commands set



**Figure 3.** PerMag running in a window

parameters for the relaxation procedure to ensure that the solution with non-linear materials converges to stable, self-consistent values. The remaining commands specify the physical properties associated with regions. In this case Region 1 represents air or vacuum with  $\mu_r = 1.0$  and Regions 2, 3 and 4 represent the unsaturated iron poles pieces with constant values  $\mu_r = 500.0$ . If saturation effects are important, we could associate the poles with a table to define  $\mu(B)$ . Regions 5, 6, 7 and 8 are permanent magnets with non-linear demagnetization curves. The table ALNICO5 .MAT gives the demagnetization curve along the axis of magnetization. The final quantity in the permanent magnet commands is the orientation (in degrees) of the magnetization axis relative to the  $z$  direction. Note that the direction reverses between adjacent magnets. Region 9 corresponds to a constant value of radially-weighted vector potential ( $rA_\theta = 0.0$ ) on the axis and outer boundary. This condition (a Dirichlet boundary) constrains magnetic flux to lie within the radial boundaries of the solution volume. There is no boundary specification on the left and right sides. In this case, the finite-element method applied in **PerMag** automatically enforces the specialized Neuman condition,  $\partial(rA_\theta)/\partial z = 0.0$ . This condition ensures that magnetic field lines are normal to the boundaries, equivalent to the symmetry condition of an infinite array.

Close the editor and choose *Run* then *Start run*. If the four required input files are present in the working directory **PerMag** begins the relaxation solution and displays information

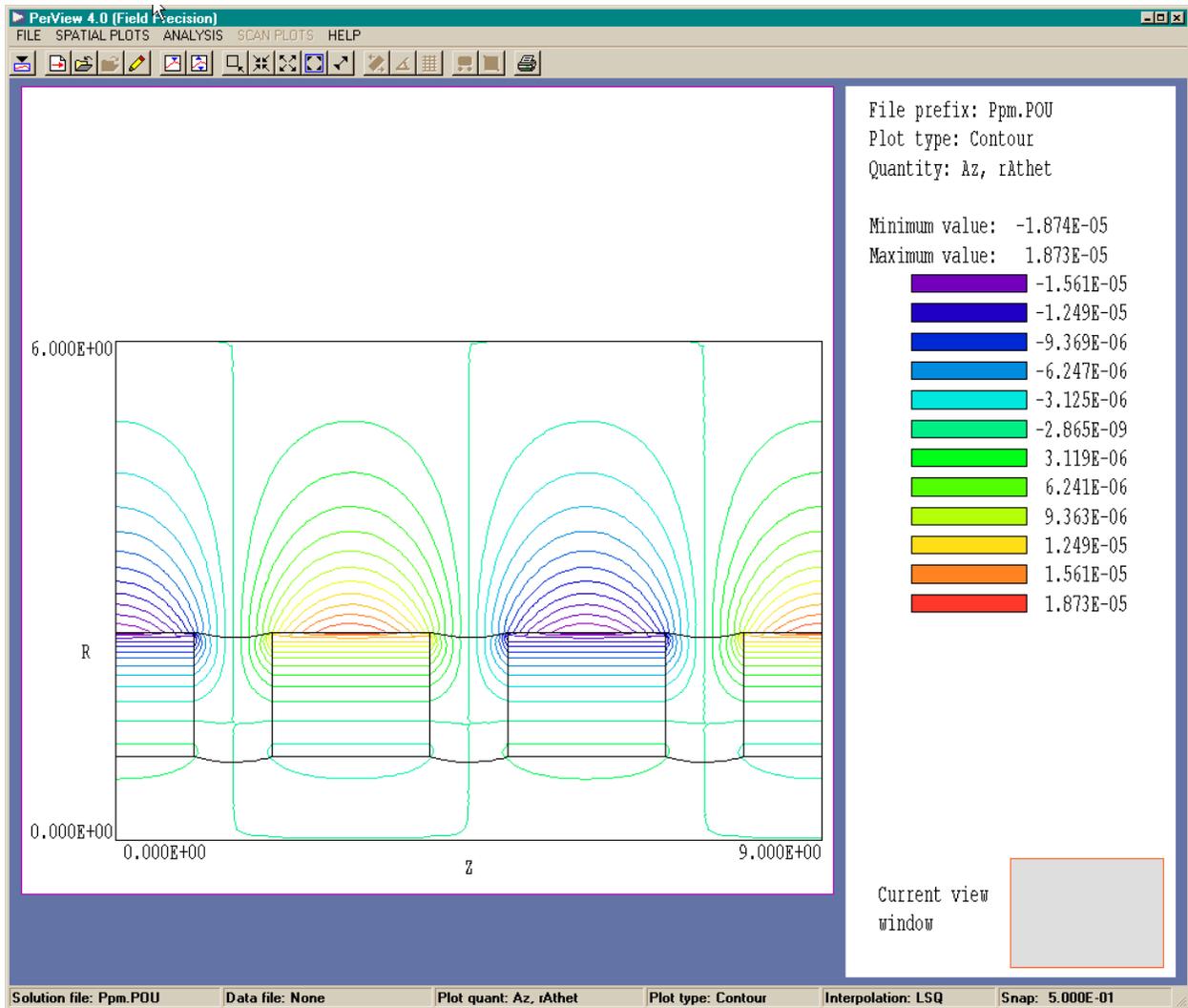
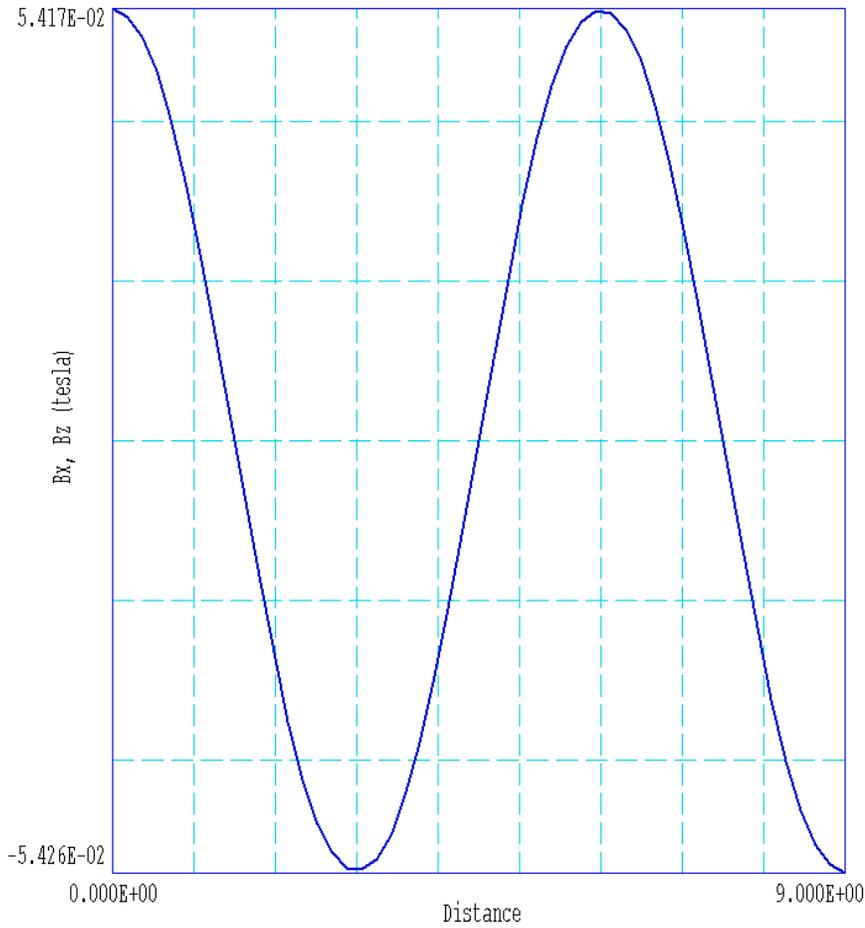


Figure 4. PerView screen display

on the screen (Fig. 3). The relaxation process and adjustment of material properties take place simultaneously to minimize the run time. Depending on the speed of your computer, the solution should be complete in 5-20 seconds. **PerMag** creates the output file PPM.POU (**PerMag** **OU**tput). Like all files in the **TriComp** series, the output file is in text format. It consists of an ordered list of vector potential at all nodes as well as the physical properties of regions. The program also creates a list of volume-averaged energy product in each permanent magnet region. This quantity, defined as the product  $BH$  along the axis of magnetization, gives an idea of how effectively a permanent magnet material is used. The listing output for the PPM example is shown below.



**Figure 5.** Example PPM, scan of  $B_z$  along the axis.

--- ENERGY PRODUCT ANALYSIS OF PERMANENT MAGNET REGIONS ---  
 UNITS

Int(BH) in J/m (planar) or J (cylindrical)  
 Volume in m3/m (planar) or m3 (cylindrical)  
 Avg(BH) in J/m3

NReg	Int(BH)	Volume	Avg(BH)
5	6.33261E-02	1.64934E-05	3.83949E+03
6	1.26503E-01	3.29867E-05	3.83496E+03
7	1.26499E-01	3.29867E-05	3.83484E+03
8	6.31737E-02	1.64934E-05	3.83025E+03

The final step is to run the **PerView** graphical post-processor to analyze the results. After starting `PERVIEW.EXE`, choose *File* then *Load solution file*. In the dialog change directories if necessary and pick `PPM.POU`. The file should load and display the default magnetic field line plot of Figure 4. At this point, you can experiment with the plot and analysis functions of the program. As an example, Figure 5 shows a scan of  $B_z$  along the axis. The plot illustrates the periodic field variation and shows a peak field value of about 540 gauss.

### 3. Organizing runs

The **PerMag** package consists of a program that computes the physical solution (`PERMAG.EXE`) and a dedicated post-processor for analyses (`PERVIEW.EXE`). The solution program can run in two modes: interactively in a window or autonomously in the background with the option for batch file control. Similarly, **PerView** can run as an interactive application under user control or autonomously under script file control. The autonomous modes allow automatic processing of large or repetitive data sets.

**PerMag** requires two input files.

- A **Mesh** output file that describes the conformal triangular mesh. The file contains node coordinates and the region numbers of elements and vertices.
- A command script file that sets control parameters for the solution program and describes the physical properties associated with region numbers.

The mesh output file always has a name of the form `RUNNAME.MOU`, where `RUNNAME` is a valid file prefix (1 to 20 characters). The command script must have a name of the form `RUNNAME.PIN` where `RUNNAME` is the same prefix as that of the mesh file. **PerMag** issues an error message if both input files are not available in the working directory. To organize data, the resulting output file has the name `RUNNAME.POU`.

A **PerMag** run consists of several steps that involve three different programs: mesh generation, field solution, and analysis. The programs communicate through data files. Sometimes, you may calculate several solutions from the same mesh by changing the characteristics of materials. Each run includes the following steps.

- Prepare a **Mesh** input script with a name of the form RUNNAME . MIN following the instructions in the **Mesh** manual. You can create the file directly with a text editor or graphically using the Drawing Editor of **Mesh**.
- Run **Mesh** either interactively from the **TC** program launcher or from the Windows Command Prompt to create the file RUNNAME . MOU.
- Prepare a command script (RUNNAME . PIN) using the *Create command file* menu entry in the solution program or your own text editor. The allowed file commands are described in this manual and the manual for the specific solution program.
- Run **PerMag** to create the output file RUNNAME . POU. It is in text format and contains information on the mesh geometry, the physical properties of regions and values of computed node quantities. In magnetostatic solutions the region properties are the relative magnetic permeability and current. The computed quantity is the vector potential  $A_z$  for planar problems and  $rA_\theta$  for cylindrical problems.
- Analyze the solution using **PerView**. You can also transfer the information in RUNNAME . POU to your own analysis programs.

## 4. Structure of the command file

The input script for all **TriComp** solution programs is a text file with data lines containing commands and parameters. The file must end with the EndFile command. The entries on a line can be separated by the following delimiters

Space, Comma, Forward slash,  
Colon, Tab, Equal sign

Any number of delimiters may appear on a line. Blank lines and comment lines are ignored. Comment lines begin with an asterisk (\*). Most parameters are real numbers. The following formats are valid.

1.250  
5.67E6  
6.8845E+09  
5

The bottom number is interpreted as 5.0.

The program will accept commands in any order. The following example illustrates a complete control file for **PerMag**:

```
SET ResTarget 5.0E-8
SET Geometry Rect
SET DUnit 100.0
SET Omega 1.90 1.95
SET MaxCycle 1000

REGION 1 Mu 1.0
REGION 2 PerMag 1.12 90.0
REGION 3 PerMag 1.12 -90.0
REGION 4 PerMag 1.12 180.0
REGION 5 VecPot 0.0

ENDFILE
```

There are two types of commands that begin with different keywords:

```
SET
REGION
```

Commands that begin with the word **SET** control program settings. The keyword that follows determines the function of the command. One or more parameters follow the keyword. For example, the command

```
SET GEOMETRY CYLIN
```

instructs the program to use element weighting for cylindrical coordinates. The available **SET** commands depend on the solution program.

**REGION** commands set physical properties associated with elements and vertices. They all start with the word **REGION** and have same structure:

```
Region   RegNo   Keyword   Value
```

Here, **RegNo** (an integer) is the region number defined in the **Mesh** input file. The string **Keyword** specifies the physical property. The keyword is followed by one or more numbers. As an example, the command

```
REGION 2 VecPot 0.0
```

sets nodes with region number 2 as fixed vector-potential points with the value  $A_z = 0.0$  or  $rA_\theta = 0.0$ . In some instances multiple `REGION` commands may refer to the same mesh volume and set different properties.

## 5. Commands for general program control

Commands in the **PerMag** control script divide into two groups: those that control the program operation and those that set the physical properties of regions. In this section we shall consider the control commands. They begin with the keyword *SET*. Each command is written as it might appear in the `FPREFIX.PIN` file.

### **SET DUnit 100.0**

Informs the program how to interpret spatial coordinates from the **Mesh** input file. The parameter should equal the number of **Mesh** units per meter. For example, if use used units of microns in the definition of the mesh, set  $DUnit = 1.0 \times 10^6$ .

### **SET Geometry Rect**

Informs the program whether the solution geometry should be interpreted as rectangular or cylindrical. The options are *Rect* and *Cylin*. Note that **PerMag** issues an error message if negative values of  $r$  ( $y$  in **Mesh**) are detected in a cylindrical solution.

### **SET MaxCycle 2500**

Controls the maximum number of iteration cycles in the matrix solution of the node equations for vector potential. The default value is 2500. The number of relaxation cycles should be large enough to ensure that the solution has converged. More cycles may be necessary in solution with non-linear materials.

### **SET ResTarget 5.0E-6**

Sets a target for the root-mean-squared residual for the matrix solution, an indicator of convergence. The final residual should be small compared to unity. The default value is  $1.0 \times 10^{-7}$ .

### **SET Omega 1.95**

This command sets a value for the parameter that controls the iterative matrix solution. Values must be in the range  $0.0 < \textit{Omega} < 2.0$ . High values usually give the fastest convergence. Lower values may be necessary in solutions with highly non-linear materials. The default is  $\textit{Omega} = 1.95$ .

### **SET GammaTarget 5.0E-6**

Sets a target for the root-mean-squared error in the correction of the quantity  $\textit{Gamma} = 1/\mu_r$ , where  $\mu_r$  is the relative magnetic permeability in non-linear materials. The default value is  $10^{-5}$ . In runs with non-linear materials,, **PerMag** will continue iterations until both *ResTarget* and *GammaTarget* have reached the desired values or until *MaxCycle* is exceeded.

### **SET Avg 0.5**

The quantity *Avg* controls adjustment of non-linear materials during the iterative matrix solution. You can enter values between 0.0 and 1.0. The default is 0.1. A low value corresponds to a gradual adjustment of magnetic permeability. Values in the range 0.05 or less may be necessary to achieve convergence when there are highly saturated materials. **PerMag** lists the root-mean-squared error in material values (*EpsCorrect*) during the solution.

### **SET Update 20 300**

This command controls adjustment of the magnetic permeability in non-linear materials during the matrix solution. The first parameter after the keyword (*NUpdate*) is the number of iteration cycles per material update. The default value is 10. The second optional integer parameter (*NDelay*) equals the number of interaction cycles performed before material updating begins. The default is  $\textit{NDelay} = 150$ . It is essential to delay non-linear material adjustment because the initial distribution of vector potential is an arbitrary guess that could give unrealistically large values of magnetic field in some elements. The delay allows the solution to relax to a state closer to a valid solution. You may need to increase *NDelay* for runs that converge slowly.

### **SET Interp Linear**

Set the interpolation method for tables of non-linear materials and permanent magnets. The options are *Spline* (default) and *Linear*. Use *Linear* for noisy or discontinuous data.

## 6. General region commands

The **PerMag** command set for defining the material properties of regions is simple. The commands to set the physical properties of materials all start with the word *Region* and have the same format.

**Region RegNo Keyword Value**

Here, *RegNo* (an integer) is the region number defined in the **Mesh** input file. The *Keyword* is one of the words listed below and *Value* is a real number.

### **REGION 4 VecPot 2500.0**

This command sets a region of fixed vector potential. Such regions generally represent Dirichlet boundaries. In planar problems the fixed value applies to  $A_z$  in units of tesla-m. In cylindrical problems the constant value applies to the quantity  $rA_\theta$  in tesla-m<sup>2</sup>. Note that **PerMag** automatically creates a region with  $VecPot = 0.0$  along the axis when a cylindrical solution includes  $r = 0$ . Non-zero values of vector potential or stream function can be used to set up solutions with prescribed flux. Section 11 describes the physical meaning of the vector potential in more detail.

### **REGION 10 Current 2500.0**

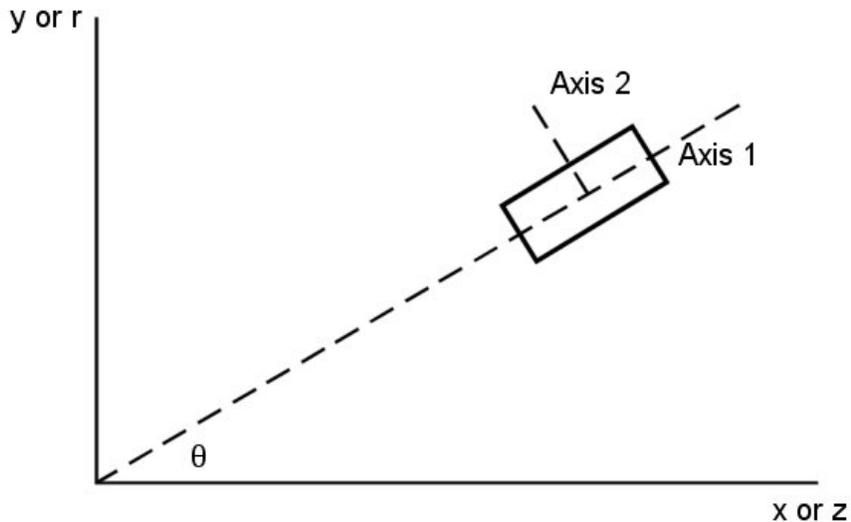
This command assigns current to a region that represents a magnet coil. The example command distributes a current of 2500 uniformly over the cross-section area of Region 10. In planar solutions the current points along  $z$  (out of the page in an  $x$ - $y$  plot). In cylindrical solutions the current is directed along  $\theta$  (out of the page in an  $r$ - $z$  plot).

## 7. Region commands for magnetic materials

The following commands apply to soft magnetic materials, isotropic or anisotropic. Figure 6 shows the geometric definitions for anisotropic materials.

### **REGION 3 Mu 12.6**

Sets a uniform value of the relative magnetic permeability in a linear,



**Figure 6.** Geometry definitions for anisotropic materials.

isotropic magnetic material. The example command sets  $\mu_r = 12.6$  in all elements of Region 3.

#### **REGION 7 Mu 7.8 3.5 45.0**

This command sets uniform values of relative magnetic permeability along two normal axes of a linear, anisotropic material. The parameters following the keyword *Mu* have the following interpretations.

Param1: Value of relative magnetic permeability along Axis 1.

Param2: Value of relative magnetic permeability along Axis 2 (normal to Axis 1).

Param3: Angle  $\theta$  between Axis 1 and the *x* (or *z*) axis (degrees).

#### **REGION 5 Mu Table Steel1014.Dat**

This command defines properties of a non-linear, isotropic magnetic material (*i.e.*, soft iron or transformer steel). The string following the keywords *Mu Table* is the full name of a data file in the current directory. Section 8 describes the format of **PerMag** tables for non-linear materials.

#### **REGION 8 Mu Table Axis1.Dat Axis2.DAT 30.0**

This form of the *Mu Table* command defines non-linear variations of the relative magnetic permeability along two normal axes. The first

string parameter is the name of a B- $\mu_r$  data file in the current directory for Axis 1. The second parameter is the name of the data file for Axis 2. The final real number parameter is the angle  $\theta$  between Axis 1 and the  $x$  (or  $z$ ) axis in degrees.

## 8. Modeling non-linear magnetic materials

**PerMag** can represent soft non-linear magnetic materials like transformer steel. A *soft* material has a narrow hysteresis curve so that the magnetization process is almost reversible. A common reason to include soft, non-linear materials is to calculate saturation effects in magnet poles. The magnetization curve of a material is entered as a table of values of the magnetic field magnitude (in tesla) and the corresponding relative permeability. On each iteration cycle of the relaxation solution, **PerMag** calculates the magnetic field amplitude in elements associated with a tabulated material. The program then uses the interpolated value of the magnetic permeability to update the coupling coefficients for all connected nodes. The two relaxation processes proceed in tandem, leading to self-consistent values of the magnetic permeability and the field. For maximum speed, **PerMag** updates coupling coefficients only in non-linear regions and skips the process altogether when no tables have been entered.

**Note.** **PerMag** solutions are limited to static fields. The **TriComp** solution program **Nelson** finds solutions with AC magnetic fields and eddy currents. The **Pulse** program solves the general problem of magnetic field diffusion into resistive materials with arbitrary temporal variations of coil current.

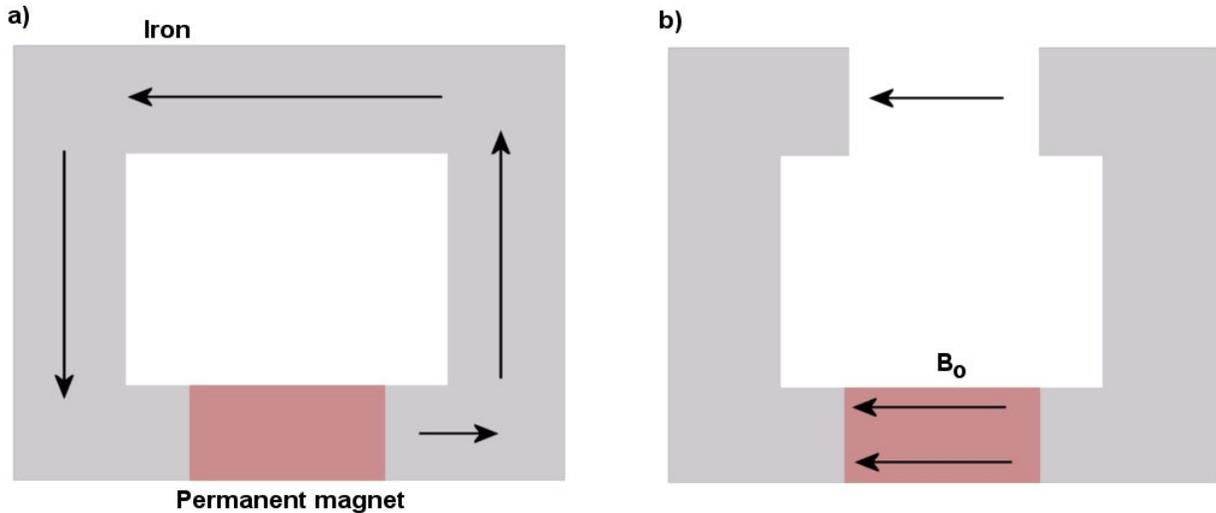
The key to flexibility in modeling non-linear materials is program input through *tabular functions*. A magnetization tabular function is a text file consisting of up to 256 data lines containing values of magnetic field and corresponding relative magnetic permeability ( $\mu/\mu_0$ ). You can prepare tabular function files with a text editor or spreadsheet. You can also use published data or digitized experimental traces. Table 2 shows a portion of the tabular function `SOFTIRON.MAT` supplied with the **PerMag** package. Note that the file syntax conforms to the rules of the command file. The free-form parser accepts real numbers in any format with a choice of delimiters. You can add documenting comment lines starting with an asterisk. The end of the data is marked with the *EndFile* command. You can add any text after *EndFile*.

**Table 2. Excerpts from the file SOFTIRON.MAT**

```
* File SOFTIRON.DAT
* Generic soft iron table
* Adapted from Superfish/Poisson Code Group
* Los Alamos National Laboratory
* Checked for cubic splines, 11/95
  0.0000      4075.45105
  0.8944      3768.28312
  1.2000      3166.80993
  1.4000      2380.64690
      ...
  2.8498      3.94592075
  3.2074      3.15384615
  3.5644      2.45501166
  4.2782      2.02377622
  4.8134      1.76596737
  5.7052      1.60606061
  6.4186      1.48298368
  7.4887      1.42983683
ENDFILE
```

Although the lines in the above example are ordered by increasing value of  $|\mathbf{B}|$ , the ordering is not required. **PerMag** sorts the list before use and records the final order in the listing file. Furthermore, the intervals need not be uniform. Note that the entries in `SOFTIRON.MAT` are clustered near the saturation transition. The maximum number of tables for all regions is 32. A table requires a minimum of 5 entries. You must ensure that the tabular functions extend over the full range of magnetic field that will be encountered in the solution. The interpolation routines return the value  $\mu_r = 1.0$  for values of  $|\mathbf{B}|$  that are off the table.

**PerMag** uses cubic splines for interpolation of the tables. This method minimizes recalculation time and gives smooth interpolations that aid convergence. It is important to note that the quality of the interpolation depends on the nature of the numerical data. The dependent quantity and its first derivative should vary smoothly over the table range. The cubic spline routines will faithfully match values at the entry points and preserve continuity of the first and second derivatives. If the data has discontinuities, the cubic spline values may be quite inaccurate between points. You can check the fidelity of the interpolations by inspecting the listing file `RUNNAME.PLS`. After recording the sorted table, **PerMag** writes a sample set of interpolated values.

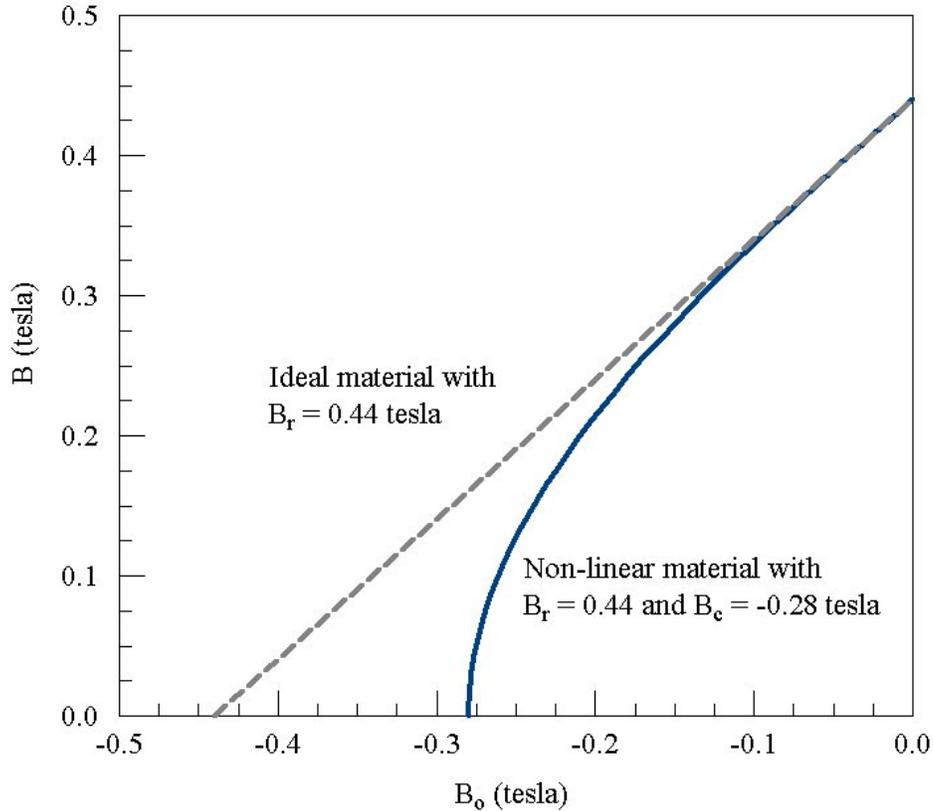


**Figure 7.** Permanent magnet in a magnetic circuit. a) Zero-reluctance circuit, all flux circulates through the iron. b) With an air gap, some of the flux is forced to return through the permanent magnet.

## 9. Physical properties of permanent magnet materials

This section gives a brief review of permanent magnets and the quantities used to characterize their properties. A more complete discussion is given in the references S. Humphries, **Principles of Charged Particle Acceleration** (Wiley, New York, 1986), Sect. 5.8 and S. Humphries, **Field Solutions on Computers** (CRC Press, Boca Raton, 1997), Sect. 9.6. A PDF version of the first reference is included with the **PerMag** package.

Figure 7a shows a test setup with a permanent magnet in a zero-reluctance circuit. All magnetic flux produced by the aligned domains of the permanent magnet returns through the high- $\mu$  iron structure. The total magnetic flux density  $B$  inside the magnetic material equals the flux density generated by the aligned domains. This value is called the *remanence* magnetic flux density,  $B_r$ . Now suppose we add an air gap as shown in Fig. 7b. Because of the added reluctance, some of the magnetic flux is forced to return through and around the permanent magnet. We shall denote the component of return magnetic flux density parallel to the direction of magnetization as  $B_o$ . The magnitude of  $B_o$  increases with increasing air-gap width (reluctance). The quantity depends on the geometry of the magnetic circuit and generally varies with location in the permanent magnet.

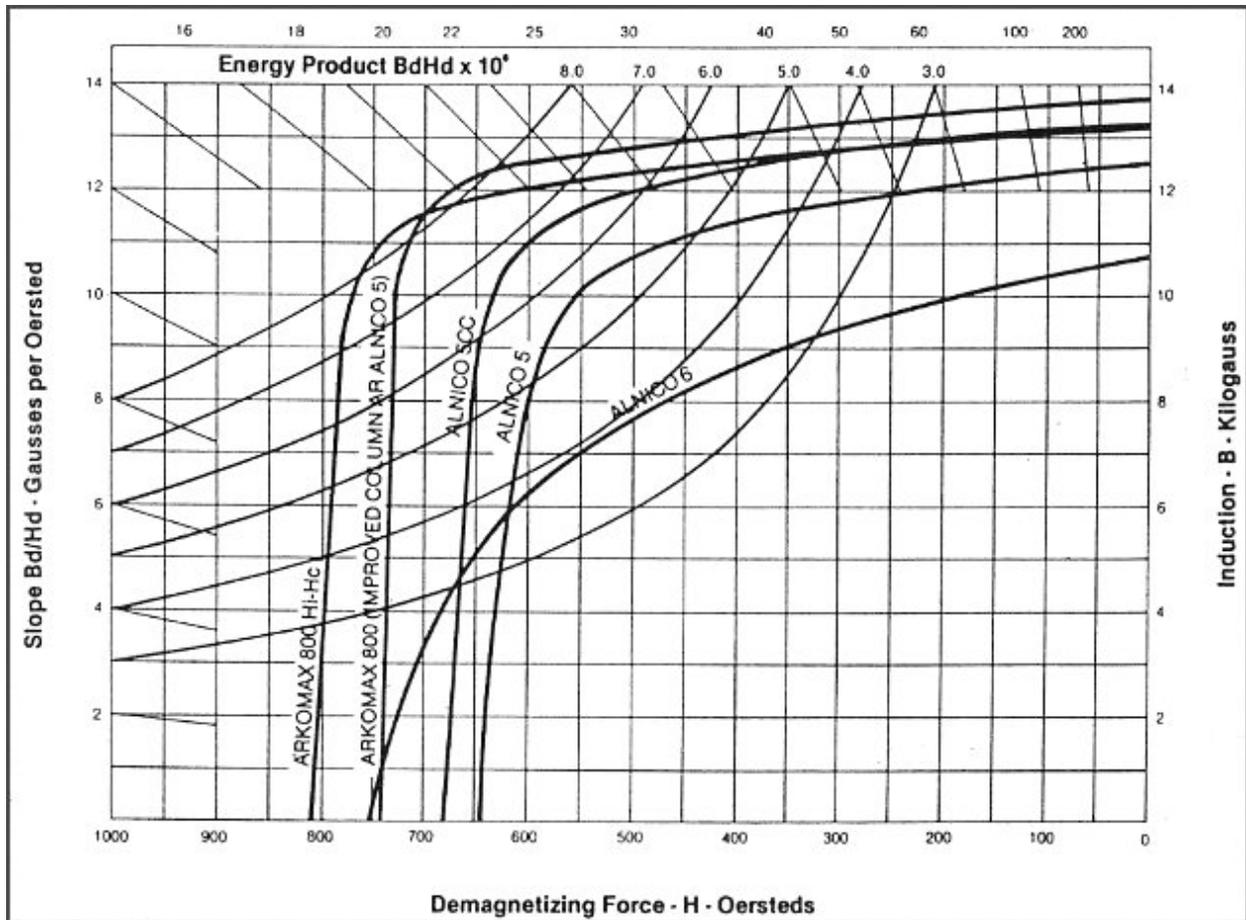


**Figure 8.** Demagnetization curves for ideal and non-linear permanent-magnet materials

The *demagnetization curve* is a plot of the total magnetic flux density inside the magnet as a function of  $B_o$ . (**Note:** curves for commercial materials are usually plotted in terms of the magnetic field intensity,  $H_o = B_o/\mu_o$ .) In an ideal material, domains are tightly locked so that their alignment is unaffected by the return flux. In this case, the total magnetic flux density inside the magnet is given by

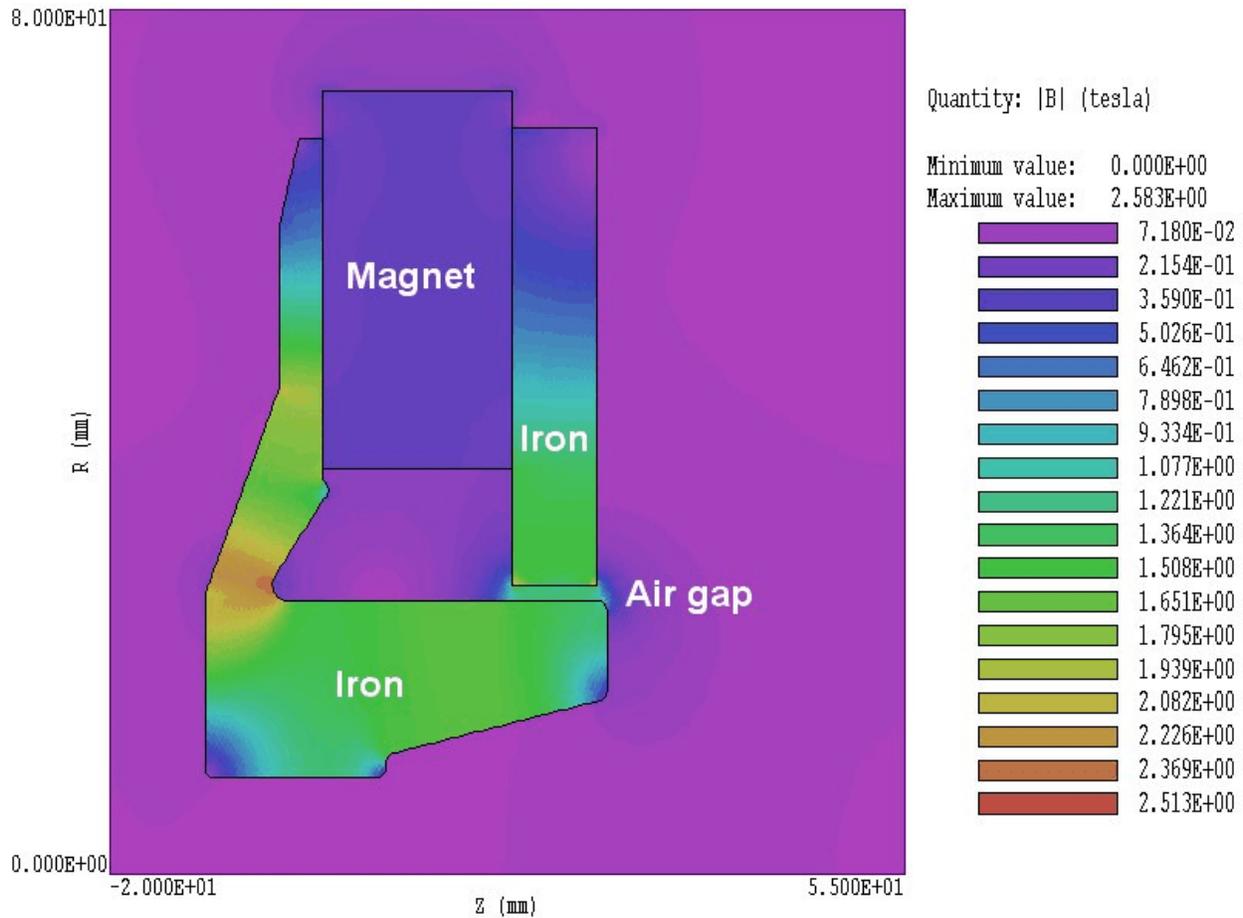
$$B = B_r - B_o.$$

The ideal demagnetization curve is a line with slope  $-45^\circ$  (Fig. 8). The total magnetic flux density inside the magnet drops to zero when  $B_o = -B_c = -B_r$ . The quantity  $H_c = B_c/\mu_o$  is called the *coercive field*. Modern magnet materials like neodymium iron and samarium cobalt approach ideal behavior (Table 3).



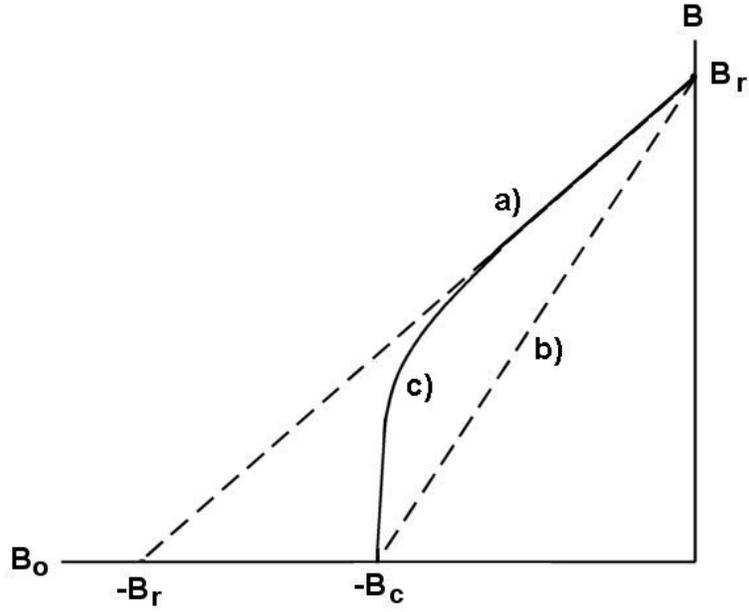
**Figure 9.** Demagnetization curves for several non-linear permanent-magnet materials (Courtesy of Arnold Engineering)

The return flux may reduce the degree of domain alignment in conventional materials like alnico. In this case, the magnet contribution to the total flux density is less than  $B_r$  and the demagnetization curve drops below the 45° line (Fig. 8). For these materials the demagnetization curve is non-linear and the condition  $B_c < B_r$  holds (Table 3). **PerMag** accurately represents non-linear materials if you enter the non-linear demagnetization curve as a table (Sect. 10). Figure 9 shows some examples of commercial demagnetization curves for non-linear materials.



**Figure 10.** Benchmark calculation – non-linear material with an unknown demagnetization curve.

A practical problem of working with non-linear magnetic materials is that often you do not have the full demagnetization curve. The manufacturer may provide only values for  $B_r$  and  $H_c$  (as in Table 3). If you simply model the curve as a straight line between  $B_r$  and  $B_c$  (Fig. 11b), the predicted flux values will be quite low. For example, the magnet of Figure 10 uses BEC material Y-38 with  $B_r = 0.44$  tesla and  $B_c = 0.28$  tesla, and produces a measured gap flux density of 1.22 tesla. Approximating the demagnetization curve as a straight line from  $B_r$  to  $B_c$  in **PerMag** gives a predicted gap field of only 0.64 tesla. Figure 11c illustrates a better approach when you have incomplete knowledge of the material. The approximating demagnetization curve defined by the following equation passes through the points  $(-B_c, 0)$  and  $(0, B_r)$  with infinite slope at  $(-B_c, 0)$  and 45° slope at  $(0, B_r)$ :



**Figure 11.** Non-linear material with an unknown demagnetization curve. *a)* Ideal material. *b)* Straight-line between  $-B_c$  and  $B_r$ . *c)* Approximation used in the **PerMag** two-point model.

$$B_o = -B_r \left[ \left( 1 - \frac{B}{B_r} \right) - \left( 1 - \frac{B_c}{B_r} \right) \left( 1 - \frac{B}{B_r} \right) \left[ \frac{1}{1 - B_c/B_r} \right] \right].$$

The equation is the basis of the PerMag two-point model discussed in the following section. In the benchmark example, the revised calculation gives an improved value of 1.27 tesla for the gap flux density.

**Table 3. Properties of some permanent magnet materials**

Material	$B_r$ (tesla)	$\mu_0 H_c$ (tesla)	Material	$B_r$ (tesla)	$\mu_0 H_c$ (tesla)
17 % Cobalt steel	0.95	0.015	NdFeB 27	1.05	1.00
36% Cobalt steel	0.95	0.024	NdFeB 27H	1.085	1.00
Alnico 1	0.72	0.044	NdFeB 28	1.08	1.01
Alnico 12	0.58	0.095	NdFeB 30	1.10	1.06
Alnico 2	0.72	0.055	NdFeB 30H	1.12	1.07
Alnico 2 (sintered)	0.69	0.052	NdFeB 32	1.16	1.08
Alnico 4	0.55	0.070	NdFeB 32H	1.15	1.09
Alnico 5	1.25	0.055	NdFeB 35	1.23	1.13
Alnico 5 (sintered)	1.09	0.062	NdFeB 35H	1.21	1.16
Alnico 6	1.00	0.075	NdFeB 37	1.24	1.18
Alnico 8	0.82	0.165	Neo Flex6	0.545	0.49
Alnico 8 (sintered)	0.74	0.150	Platinum-cobalt	5.90	0.36
Carbon steel	1.00	0.005	Remalloy or Comol	1.05	0.025
Ceramic 1	0.23	0.186	Silmanal	0.055	0.60
Ceramic 5	0.38	0.24	SmCo 16	0.80	0.76
Ceramic 8	0.385	0.295	SmCo 18	0.85	0.80
Chromium steel	0.97	0.0065	SmCo 20	0.905	0.83
Cunico	0.34	0.066	SmCo 22	0.95	0.86
Cunife (wire)	0.54	0.055	SmCo 22B	0.96	0.78
Flexible regular	0.16	0.137	SmCo 24	0.995	0.72
HF1	0.22	0.20	SmCo 26	1.05	0.92
HF2	0.245	0.22	SmCo 28	1.075	0.84
HF3	0.265	0.22	Tungsten steel	1.03	0.007
Hyflux Fine powder	6.60	0.039	Vectolite	1.60	0.10
NdFeB 24	0.98	0.95	Vicalloy 1	0.88	0.030
NdFeB 26	1.04	1.00	Vicalloy 2 (wire)	1.00	0.051

## 10. Region commands for permanent magnet materials

**PerMag** has three methods to represent permanent magnets. The ideal material model with a 45° demagnetization curve gives a good approximation for modern materials like neodymium-iron and samarium-cobalt. Non-linear materials like Alnico and ceramics follow a more complex variation. If you have access to the non-linear demagnetization curve, you can enter it as a tabular function. Otherwise, you can simply supply values for  $B_r$  and  $B_c$  and **PerMag** will apply the equation of Sect. 9.

### REGION 9 PerMag 1.27 45.0

This command defines an ideal permanent magnet with a straight line demagnetization curve with  $B_r = -B_c$ . Here,  $B_r$  is the *remanence* value of magnetic flux density and  $B_c$  is the *coercive* value of flux density ( $B_c = \mu_0 H_c$ ). Both values are in tesla. The final parameter  $\theta$  is the angle in degrees between the axis of magnetization (easy axis) and the  $x$  (or  $z$ ) axis (Fig. 6).

### REGION 3 PerMag 1.10 -0.96 45.0

This command defines a two-point permanent magnet material where the demagnetization curve follows the equation of Sect. 9. The three parameters that follow the *PerMag* keyword have the following meanings.

Param1:  $B_r$  in tesla

Param2:  $B_c$  in tesla (the sign can be omitted)

Param3: The angle in degrees between the direction of magnetization and the  $x$  (or  $z$ ) axis.

Table 3 lists values of  $B_r$  and  $B_c$  for several materials. Specifications for commercial materials often give the remanence magnetic flux density in kG and the coercive magnetic field  $H_c$  in kOe. To convert from kG to tesla, multiply by 10<sup>-1</sup>. To convert  $H_c$  in kOe to  $B_c$  in tesla, multiply by 0.1.

### REGION 9 PerMag Table Alnico8.MAT 30.0

This command defines a material with a non-linear demagnetization curve. The first string parameter following the *PerMag Table* keywords is the full name of the data file that defines the demagnetization curve. The file must be available in the working directory. The second parameter is the angle (in degrees) between the direction of magnetization and the  $x$  (or  $z$ ) axis.

Table 4 shows an example of a demagnetization table. Data lines have the entries

B      B0

where both quantities are given in tesla. The first quantity is the total flux density in the material and the second is the return flux density (equal to  $\mu_0 H$ ). The first data line must contain the entries:

0.0 BC

Lines should be arranged in ascending order of  $B$  and stop at

BR 0.0

The file may contain comments, blank lines and up to 256 data lines.

To employ the method described in S. Humphries, **Field Solutions on Computers** (CRC Press, Boca Raton, 1997), Sect. 9.7, **PerMag** converts the table to an internal format of  $B_{||}-\mu_{||}$  where the quantity  $\mu_{||}$  is defined by

$$B_{||} = \mu_{||} (|B_c| - |B_o|).$$

The program records a copy of the converted table in the listing file RUNNAME . PLS.

Table 4. File ALNICO9.MAT	
* Alnico 9	
* B(tesla)	B0(tesla)
* =====	
0.000	-0.1360
0.100	-0.1355
0.200	-0.1350
0.300	-0.1344
0.400	-0.1335
0.500	-0.1320
0.600	-0.1304
0.700	-0.1278
0.800	-0.1232
0.900	-0.1135
0.950	-0.0972
1.000	-0.0758
1.050	-0.0495
1.100	-0.0212
1.125	-0.0000
ENDFILE	

## 11. Vector potential and boundary conditions

A boundary condition is the specification of the calculated field quantity on the outer edge of the solution region. To understand boundaries, it is important to realize that **PerMag** determines values for the magnetic vector potential at node points. The analysis program **PerView** then takes spatial derivatives of these values to determine the magnetic fields. The boundary conditions therefore apply to the vector potential.

In rectangular problems (variation in  $x$  and  $y$  with no variation in  $z$ ) **PerMag** determines the vector potential  $A_z$  created by currents that move in the  $z$ -direction. The magnetic fields are given by

$$B_y = - \frac{\partial A_z}{\partial x}.$$

and

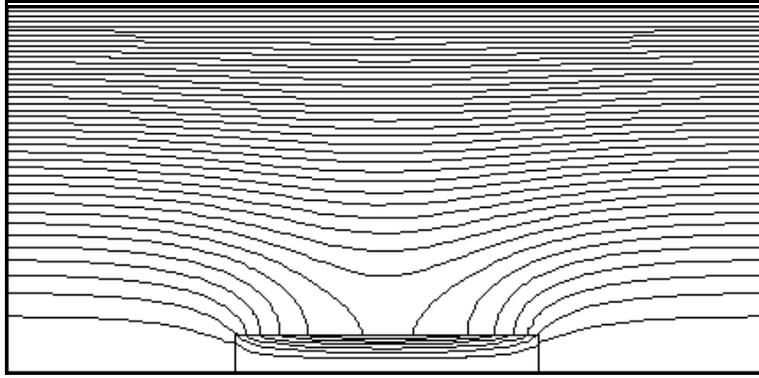
$$B_x = \frac{\partial A_z}{\partial y}.$$

For problems in cylindrical coordinates, **PerMag** calculates the quantity  $rA_\theta$ , where  $A_\theta$  is the vector potential created by azimuthal currents. The quantity  $rA_\theta$  is called the *stream function* and denoted by  $\psi$ . In this case the magnetic fields are given by

$$B_r = - \frac{\partial A_\theta}{\partial z},$$

and

$$B_z = \frac{1}{r} \frac{\partial}{\partial r} rA_\theta .$$



**Figure 12.** Setting boundary values of the vector potential to create a region of uniform axial flux. Cylindrical geometry. Vertical:  $r$  from 0.0 to 0.1 m. Horizontal:  $z$  from 0.0 to 0.2 m. Lower boundary:  $rA_\theta = 0.0$  tesla-m<sup>2</sup>. Upper boundary:  $rA_\theta = 5.0 \times 10^{-4}$  tesla-m<sup>2</sup>.

The above equations show that differences in values of the vector potential are related to the flux of magnetic field lines (the surface integral of magnetic field). For example, in a rectangular problem with boundaries at  $y_1$  and  $y_2$ , the difference in the boundary values of the vector potential at position  $x_0$  is given by

$$A_z(x_0, y_2) - A_z(x_0, y_1) = \int_{y_1}^{y_2} B_x(y', x_0) dy'$$

A similar equation holds for integrals in the  $x$ -direction. Therefore, if we set  $A_z$  equal to a constant value (usually zero) around the complete solution boundary, all magnetic flux must be trapped inside the region since the integral of  $B_x$  or  $B_y$  over any plane is zero. This is equivalent to generating the magnetic field inside a perfectly-conducting metal box. Another way to see this is to note that because  $\mathbf{B} = \nabla \times \mathbf{A}$ , the magnetic field must be parallel to a surface of constant  $\mathbf{A}$ .

In cylindrical geometry, the symmetry of magnetic field lines dictates that the axis ( $r = 0$ ) must always be a line with  $\psi = rA_\theta = 0.0$ . **PerMag** automatically sets the condition for cylindrical problems that include  $r = 0.0$  by adding a constant potential line region on the axis. The value of the stream function on the outer boundary at the outer solution radius is related to the flux of axial field by

$$(rA_{\theta})\Big|_{r_o} = \left(\frac{1}{2\pi}\right) \int_0^{r_o} 2\pi r' dr' B_z(r,z) = \frac{Flux}{2\pi} .$$

As an example, Fig. 12 shows the magnetic field lines around an iron cylinder immersed in a uniform solenoidal field of 0.1 tesla. The uniform field was created by setting  $\psi = 5.0 \times 10^{-4}$  tesla-m<sup>2</sup> on the outer boundary at radius 0.10 m.

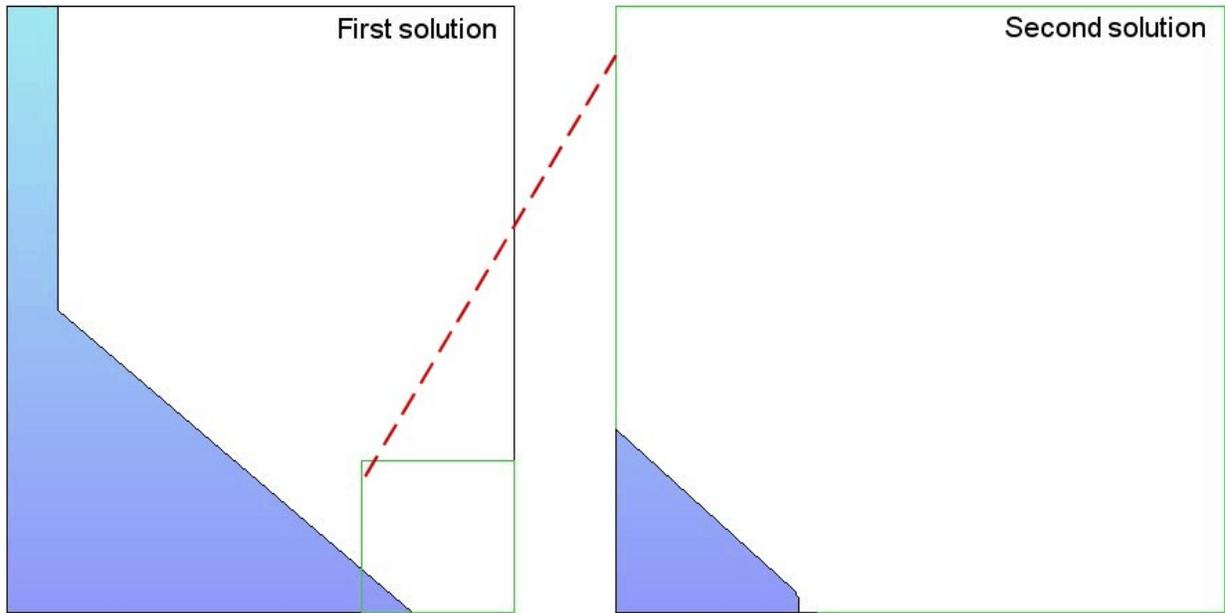
With this background, we can now discuss the implications of the two types of boundary conditions used in **PerMag**:

### **Dirichlet**

Dirichlet boundary points have a fixed value of vector potential that does not change during the interactive matrix solution. A line-region of uniform vector potential represents a perfectly conducting metal surface with magnetic field lines parallel to the boundary. An internal volume region at fixed vector potential represents a metal body that excludes the magnetic field.

### **Neumann**

A Neumann boundary is one where the normal derivative of the vector potential is specified. The boundaries in **PerMag** are limited to the special case  $\partial A/\partial n = 0$ . The special Neumann condition implies that the magnetic field is normal to the boundary. One of the strong points of the finite-element method is that all boundaries that are not fixed automatically satisfy the special Neumann condition, even if they are slanted or curved. Neumann boundaries are often used to reduce computation time for symmetric systems, as in the example of Sect. 2.



**Figure 13.** Interpolated Dirichlet boundary for microscopic solution

## 12. Boundary and superposition commands

Version 5.0 of **PerMag** and the other **TriComp** solution programs incorporate a powerful new feature that enables accurate calculations of fields near small features in a large solution space. Figure 13 illustrates the procedure for an electrostatic solution. Suppose we wanted a precision calculation of the fields on a field emission tip. The radius of the tip is much smaller than the scale size of the electrodes that create the field and hold the tip. We need to do a large-scale calculation to find the macroscopic fields (Solution 1), but we would need very small elements near the tip to resolve the curvature. One approach is to use variable mesh resolution (**Mesh** manual, Section 6) to create small elements near the tip. A limitation to this technique arises from the structured mesh used in **TriComp** – the region of small elements must extend the full length of the solution volume.

Figure 13 illustrates an alternate approach. We create a second solution that extends over a small subregion of the original solution (green outline). The microscopic solution contains any electrodes that may enter the subregion. The difference is that the electrode shapes may be resolved in much greater detail. The question is how to incorporate the macroscopic fields correctly into the microscopic solution. One resolution is to enclose the second solution in a variable-potential Dirichlet boundary. Values of  $\phi$  on the boundary are calculated by interpolation at the corresponding point

in the macroscopic solution (dashed red line). The total solution will be approximately correct as long as new features (such as the rounded tip) are well-removed from the variable-potential boundary.

The variable resolution boundary is controlled by the following command that appears in the control script of Solution 2:

### **SET BOUNDARY FPrefix [BndScale]**

#### **SET BOUNDARY FEmitMacro**

The quantity *FPrefix* (a string) is the prefix of the output file for Solution 1. The quantity *BndScale* (a real number) is a scaling factor applied to vector potential values transferred from Solution 1 to Solution 2. The default is *BndScale* = 1.0.

**PerMag** issues an error message under the following conditions: 1) the output file for Solution 1 (*FPREFIX.POU*) is not available in the working directory, 2) Solution 2 does not fit completely inside Solution 1, 3) Solution 1 and Solution 2 have different symmetries, or 4) an interpolation failed. The program makes no further validity checks. You must ensure that the geometry of Solution 2 represents a correct microscopic section of Solution 1.

A node in Solution 2 is on the boundary under two conditions:

- It has indices  $K = 1$ ,  $K = K_{\max}$ ,  $L = 1$  or  $L = L_{\max}$ . In solutions with cylindrical symmetry, points on the axis ( $L = 1$  and  $r = 0.0$ ) are automatically not set to the Dirichlet condition.
- One of the adjacent elements has *RegNo* = 0. This condition means that the boundary of Solution 2 need not be rectangular.

Figure14 illustrates the boundary point criteria for an electrostatic solution. Solution 1 (*BoundaryVal01*) represents electric fields between two spherical-section electrodes separated by a 3.00" gap. We want to find the fields on a small dielectric body inserted at the center of the gap. Solution 2 (*BoundaryVal02*) is a spherical region (radius 1.00") at the midpoint. The microscopic solution contains a dielectric sphere of radius 0.10". Note the assignment of potential values on the circular outer boundary of Solution 2. The lower boundary of the cylindrically-symmetric solution has not been set to the Dirichlet condition; therefore, the potential can adjust to the presence of the dielectric.

In response to the *Superposition* command, **PerMag** superimposes values from a large-scale solution (Solution 1) on a small-scale solution (Solution 2). The following statement may appear in the command script for Solution 2.

**SET SUPERPOSITION FPrefix [SuperScale]  
SET SUPERPOSITION UniField**

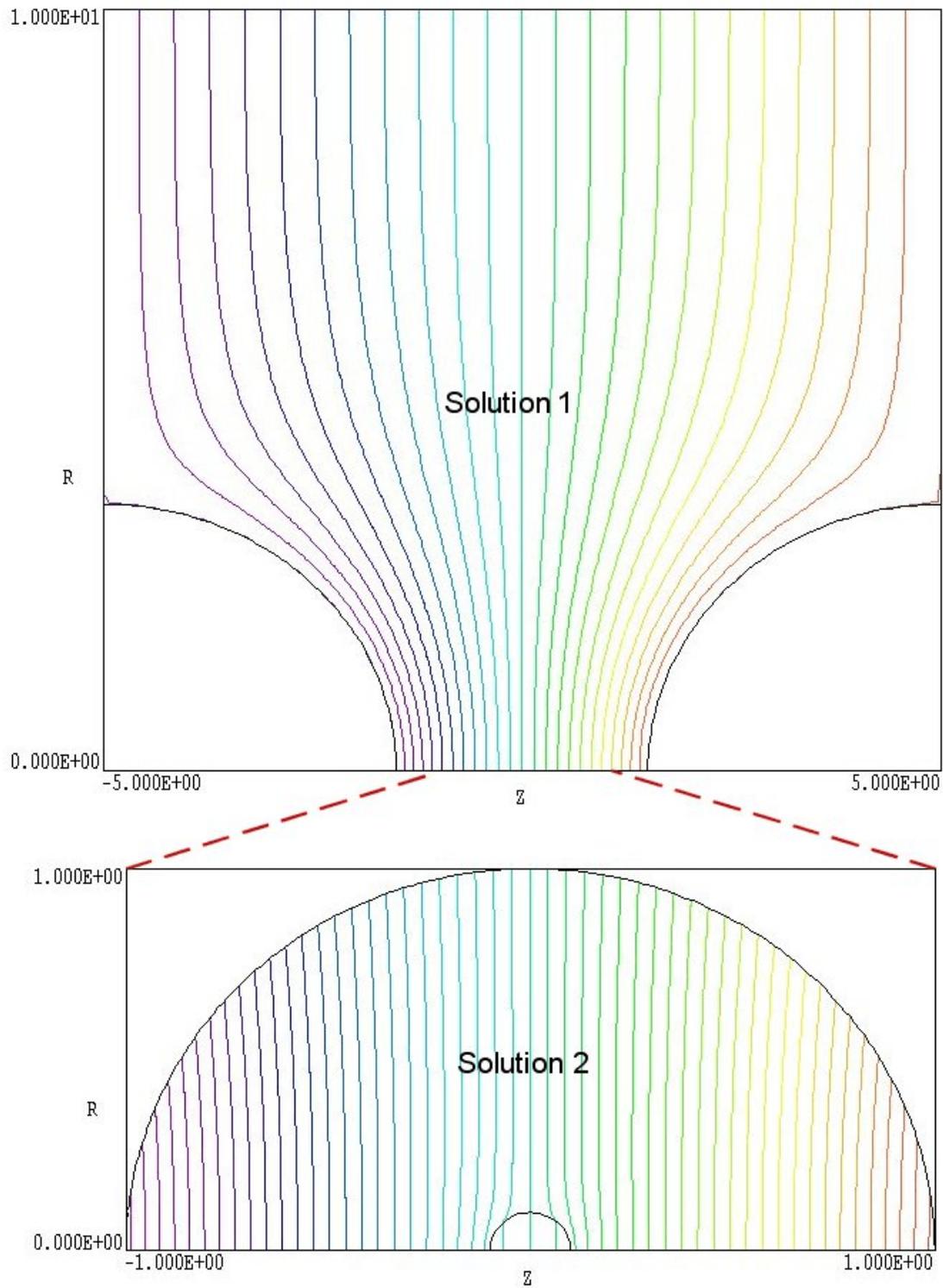
The quantity *FPrefix* (a string) is the prefix of the output file for Solution 1. The quantity *BndScale* (a real number) is a scaling factor applied to potential values transferred from Solution 1 to Solution 2. The default is *BndScale* = 1.0.

When the *Superposition* command is issued, **PerMag** opens the file *FPrefix.POU* after Solution 2 has been completed. The program performs an interpolation in the space of Solution 1 to determine the vector potential or stream function at each node location in Solution 2 and adjusts values according to

$$A'_{z2} = A_{z2} + \textit{SuperScale} \times A_{z1} , \quad (11)$$

before writing the output file.

You must ensure that the superposition is physically correct. The procedure is usually invalid in solutions with non-linear materials.



**Figure 9.** Application of the *BOUNDARY* capability in an electrostatic solution.

## 13. Running PerMag interactively

The program PERMAG . EXE can run interactively in a Window (Figure 3). In this mode you can carry out several solutions in a session or leave the program while you work in other Windows applications.

The most convenient way to run solution programs and post-processors is from the **TC** program launcher. Operation and setup of the launcher is described in the *TC Manual*. You can also run the executable programs individually from Windows Explorer or add shortcuts to the desktop or the Windows Start Menu.

The program has three popup menus: *File*, *Run* and *Help*. The following commands appear in the *File* menu.

### Create input file

This command calls up a full-featured internal Windows editor to create an input script. Supply a run prefix (1-20 characters). The allowed commands for the specific solution program appear as a set of comment lines. Save or abandon the file to return to the solution program.

### Edit input file

### Edit listing file

### Edit file

The commands call the editor to inspect or to modify ASCII input and output files for **PerMag**. Choosing a file from an alternate directory does not change the working directory. The *Edit input file* command shows a list of all files with names of the form FPREFIX . PIN. The *Edit listing file* command displays files with names FPREFIX . PLS.

The *Run* menu has four commands.

### Start run

Pick an input file (such as FPREFIX . PIN) to start a solution. The working directory is changed if you pick a file from an alternate directory. The run begins if the file FPREFIX . MOU and any data files specified in the command script are present. The small text window displays information on the run status.

### Pause run

The intensive calculations of the solution program demand the full resources of your computer, causing other tasks to run slowly. If you need to perform critical work, you can pause the solution program and reactivate it later without loss of data.

### Stop run

This command terminates the program and saves the current state of the relaxation solution. For example, you may want to stop a relaxation solution at a moderate value of convergence to check whether the problem has been correctly defined.

## 14. Running PerMag from the command prompt

To run **PerMag** in the background, go to the Command Prompt from Windows and log to the data directory that contains the required MOU and PIN files.. For example, suppose the input data files SOLENOID.MOU and SOLENOID.PIN are stored in \TRICOMP\BUFFER and that the program PERMAG.EXE is in the directory \TRICOMP. From \TRICOMP\BUFFER type

```
..\PERMAG SOLENOID <Enter>
```

The program runs silently, writing detailed information in the listing file SOLENOID.PLS. If the solution is successful, the program writes the output file SOLENOID.POU to the data directory. During lengthy runs you can perform other tasks in Windows. Note that the response of the computer may be slowed considerably.

The main function of the command mode is autonomous operation under batch file control. This feature is useful if you want to make an extended series of solutions overnight or in the background. As an example, suppose you have prepared the input files SWT01.MIN, . . . , SWT08.MIN and SWT01.PIN, . . . , SWT08.PIN in the directory \TRICOMP\BUFFER. Furthermore, you have written the following batch file SWRUN.BAT in the data directory using a text editor.

```
@ECHO OFF
ECHO Main switch data run
START ..\MESH.EXE SWT01
START ..\PERMAG.EXE SWT01
START ..\MESH.EXE SWT02
START ..\PERMAG.EXE SWT02
...
START ..\MESH.EXE SWT08
START ..\PERMAG.EXE SWT08
```

Type

```
SWRUN <Enter>
```

to generate all solutions without the need for further user input.

Microsoft has released over thirty versions of its 32-bit operating system since Windows 95. There is considerable inconsistency in DOS emulation between versions. To ensure consistent batch file operation we supply the utility **GCon** with all our software. The program emulates many DOS commands and has advanced features to organize and to analyze large data runs. To avoid programs, we advise running batch scripts from **GCon** rather than from the Command Prompt.

## 15. PerView file menu

**PerView** (for View PerMag) is the interactive graphical analysis program for output files from **PerMag**. The program automatically adjusts labels and calculated quantities depending on whether the solution is dielectric or conductive:

### **Program units**

Spatial dimensions: meters or units set by *DUnit*.

Magnetic fields: tesla

Vector potential: tesla-m

Current: amperes

Forces: newtons

Torques: newton-m

**PerView** has the following popup menus: *File*, *Spatial plots*, *Analysis*, *Scan plots* and *Help*. When the program starts only the *File* and *Help* menus are active. You must load a data file in order to create plots or to perform analyses. This section reviews options in the *File* menu.

### **Load solution file**

The *Load solution file* command displays a dialog with a list of solution files (suffix *POU*). Changing the directory in the dialog changes the program working directory. Pick an available file and click *OK*. The program loads the solution and updates the status bar. If data retrieval is successful, the *Spatial plots* and *Analysis* menus become active.

### **Open data listing file**

Several of the analysis commands like *Point calculation* and *Line scan* generate quantitative information. You can automatically record the data generated during an analysis session by opening a data file. Supply a file prefix in the dialog or accept the default. The data file has a name of the form *FPREFIX.DAT* and will be stored in the working directory. The file is in ASCII format. You can use a text editor to view the file or to extract information to send to mathematical analysis programs or spreadsheets.

### **Close data listing file**

Use this command if you want to start a new file to record data. You must close the file before attempting to view or edit it using the built-in editor of the post-processor.

### **Run script**

Sometimes you may want to perform complex or repetitive analyses on a set of similar solutions. Script file operation is a powerful feature of the **TriComp** postprocessing programs. This command displays a dialog with a list of analysis files that you have created with the suffix *SCR*. Pick a file and click *OK*. The script file can load data files, open and close history files, and perform any of the quantitative analysis functions described in this manual. The script command language is described in Section 16. Note that the script file should be in the same directory as the data files.

### Create script

This command allows you to create script files using the internal editor of the post-processor. A dialog box requests a file prefix. The resulting script file will be saved as `FPREFIX.SCR`. Next, the program opens the file in the editor and writes a list of allowed commands for reference. This list follows the *EndFile* command and will be ignored. Enter analysis commands above the *EndFile* command. After saving the file, you can run it using the *Run script* command.

### Edit script

Use this command to change an existing script file. The dialog lists files in the current directory with the subscript `SCR`. Changing directories does not change the working directory of the program.

### Edit data file

Use this command to view or to modify files with names of the form `FPREFIX.DAT`.

### Edit file

Use this command to view or to modify any ASCII file.

## 16. PerView spatial plot menu

Spatial plots show variations of quantities over the two-dimensional space of the simulation. **PerView** makes a default plot when a data file is loaded.

### Plot type

**PerView** can create a variety of colored screen and hardcopy plots.

**Mesh.** Outline of the elements in the computational mesh.

**Region.** Computational mesh with elements color-coded by region number.

**Contour.** Lines that follow constant values of a calculated quantity. In two-dimensional magnetostatic solutions constant values of  $A_z$  or  $rA_\theta$  are parallel to magnetic field lines. Contours are separated by equal

increments of magnetic flux. Therefore, in cylindrical coordinates the line spacing increases near the axis even if the field magnitude  $|\mathbf{B}|$  is uniform.

**Element.** Elements of the solution space color-coded according to a computed quantity (such as the magnetic field magnitude). You can add or remove element boundaries with the *Toggle element outline* command.

**Vector.** An element plot with orientation lines included in each element to show the local direction of a vector quantity.

**Surface.** A three-dimensional plot where a computed quantity is displayed along  $z$  over a region in the  $x$ - $y$  or  $z$ - $r$  plane. The spatial limits of the plot correspond to the current view window for *Mesh*, *Region*, *Contour*, *Element* or *Vector* plots.

A plot type may not support some plotted quantities. For example, a vector plot of magnetic permeability is undefined. If you receive a message when you switch plot types that the current quantity is not allowed, use the *Plotted quantity* command to pick a valid option.

### Plotted quantity

A dialog shows a list of available quantities consistent with the current plot type. Note that the list will be empty for *Mesh* and *Region* plots.

**PerView** supports the following plot quantities. Note that available quantities depend on the solution type (dielectric or conductive).

### Contour plot quantities

- Vector potential ( $A_z$ ) or stream function ( $rA_\theta$ )
- Amplitude of the magnetic flux density,  $|\mathbf{B}|$
- Relative magnetic permeability,  $\mu_r$

### Element plot quantities

- Amplitude of the magnetic flux density,  $|\mathbf{B}|$
- Magnetic permeability,  $\mu$
- Magnetic field,  $\mathbf{H}$

### Vector plot quantities

- Magnetic flux density,  $\mathbf{B}$

### Surface plot quantities

Amplitude of the magnetic flux density,  $|\mathbf{B}|$   
Horizontal field,  $B_x$  or  $B_z$   
Vertical field,  $B_y$  or  $B_r$   
Relative magnetic permeability,  $\mu$   
Vector potential,  $A_z$  or  $A_\theta$   
Magnetic field,  $\mathbf{H}$

### Plot limits

In the default *autoscale* mode the program picks limits in *Contour*, *Element*, *Vector* and *Surface* plots that span the full range of the current quantity. With this command you can set manual limits. In the dialog uncheck the *Autoscale* box and fill in minimum and maximum values. Note that the program does not check that the values you supply are physically reasonable. This operation will not affect scaling of other plot quantities. Check the box to return to *autoscale* mode.

### Toggle grid

A set of dashed grid lines in  $x$ - $y$  or  $z$ - $r$  can be superimposed on *Mesh*, *Region*, *Contour*, *Element* and *Vector* plots. The program automatically chooses intervals and positions so that lines occur at convenient values of  $x$  or  $y$  (for example, 0.01 rather than 0.01153). Grids corresponding to the axes ( $x = 0.0$  or  $y = 0.0$ ) are plotted as solid lines.

### Set contour plot style

This command is active only when the current plot type is *Contour*. There are four choices: monochrome, monochrome with labels, colored and colored with labels. In the colored mode, the lines are color-coded according to the value of the plotted quantity. A legend is included in the information window to the right of the plot. In the labeled modes, contour lines are numbered according to their values. Overlapping labels on closely-spaced lines may look better in a zoomed view.

### Toggle element outline

This command determines whether the element boundaries are included in *Element* and *Vector* plots. It may be necessary to deactivate outlines for large meshes.

### **Toggle mouse/keyboard**

By default the program uses interactive mouse entry of coordinates for commands like *Line scan* and *Zoom*. This command switches between mouse and keyboard input. Enter keyboard coordinates in the distance units used in **Mesh**. In other words, if the solution program has  $DUnit = 1.0 \times 10^6$ , then enter dimensions in microns.

### **Toggle snap mode**

The mouse snap mode is a convenient feature of **PerView**. When snap mode is active, the mouse returns the coordinate values closest to an integral multiple of the quantity  $DSnap$ . In other words, if  $DSnap = 0.5$  and the mouse position is [5.4331,-2.6253], the returned coordinates are [5.5,-2.5]. By default, snap mode is *ON*. Snap mode is automatically turned off for coordinate input to the commands *Point calculation* and *Element properties*. Otherwise, the program would pick a location closest to the snap point rather than the point at the tip of the mouse arrow, giving misleading information.

### **Set DSnap**

Set the distance scale for the mouse snap mode.

The following commands change the view limits in *Mesh*, *Region*, *Contour*, *Element* and *Vector* plots. The current view limits of the two-dimensional plots are used when creating three-dimensional *Surface* plots.

### **Zoom window**

You can zoom in on any area of the plot by specifying the two corners of a view box with the mouse. The coordinates are displayed in the window below the plot. The returned coordinate values depend on whether snap mode is active. You can also enter view coordinates from the keyboard by issuing the *Toggle mouse/keyboard* command.

### **Zoom in**

Magnifies the view about the center of the current plot.

### **Expand view**

Expands the view area about the center of the current plot.

**Global view**

Returns the plot boundaries to the full solution area.

**Pan**

Moves the center of the plot. Enter two points to define a displacement vector.

The following commands control the appearance of *Surface* plots. The commands are active only when a *Surface* plot is displayed. You may notice that there is a delay if you choose the *Surface* option in the *Plot type* command or if you change the plotted quantity when a *Surface* plot is active. To create the plot, **PerView** must map the current quantity to a rectangular grid, a process that requires a large number of interpolations.

**Rotate 3D image**

Rotate the *Surface* plot 90° in the spatial plane.

**View angle 3D**

Change the angle of view for the *Surface* plot.

**Set grid 3D**

To create the plot a quantity is mapped to a rectangular grid with dimensions  $NX \times NY$ . These numbers also determine the total number of grid lines in the *Surface* plot. The default values are  $NX = NY = 40$ . You can change values with this command. Although higher numbers give plots with more detail, the regeneration time is longer and the screen display may be unattractive. The command causes a program delay because values must be recalculated.

**Default printer**

Spatial and scan plots can be ported to any installed Windows printer (including network printers, postscript drivers,...). You can generate colored plots if you have a color printer. When you issue the *Default printer* command, the current screen plot is sent to the default Windows printer. If necessary, change the default using the *Settings* command of Windows.

### Plot file (PostScript)

Use this command to make a plot file of the current screen plot in Encapsulated PostScript format. Supply a file prefix in the dialog box. The plot file will be created in the current directory with the name `FPREFIX.EPS`.

### Plot file (BMP)

Use this command to make a plot file of the current screen plot in Windows Bitmap format. Supply a file prefix in the dialog box. The plot file will be created in the current directory with the name `FPREFIX.BMP`.

### Plot file (PNG)

Use this command to make a plot file of the current screen plot in Portable Network Graphics format. Supply a file prefix in the dialog box. The plot file will be created in the current directory with the name `FPREFIX.PNG`.

### Copy to clipboard

Copy the current plot to the clipboard in Windows MetaFile format. The plot can then be pasted into a compatible graphics program.

## 17. PerView analysis menu

The commands in the Analysis menu generate numerical data. Most of the functions require coordinate input from the user, usually through the mouse. Therefore, the *Analysis menu* is active only when a *Mesh*, *Region*, *Contour*, *Element* or *Vector* plots is displayed.

### Point calculation

**PerView** employs a sophisticated interpolation technique that preserves discontinuities at material boundaries. The program gives correct values for magnetic field on both sides of a material boundary. Click on the command and then point to any position in a *Mesh*, *Region*, *Contour*, *Element* or *Vector* plot. Note that snap mode is turned *OFF* for coordinate input. The program writes a subset of interpolated quantities to the window below the plot and also records complete information if a data file is open. To enter point coordinates by keyboard, use the *Toggle mouse/keyboard* command.

### Toggle interpolation

The default interpolation method for the *Point calculation* and *Line scan* commands is a second-order least-squares fit with intelligent collection of data points. For example, only points on the side of a dielectric boundary that contains the target point are included to give the correct field discontinuity at the boundary. The least-squares fit may fail in very small regions or enclosed areas if the program cannot identify enough data points. In this case, toggle to the linear mode. Here, field values are determined by a first order fit in the element that contains the target point.

### Line scan

The line scan is one of the most useful functions of **PerView**. After clicking on the command, supply two points with the mouse in a view of a *Mesh*, *Region*, *Contour*, *Element* or *Vector* plot to define a scan line. The snap mode is useful in this application (for example, you may want a scan to extend from 0.000 to 5.000 rather than 0.067 to 4.985.) The program computes a series of values of field quantities at equal intervals along the line. The information is recorded if a history file is open. The program also makes a screen plot of the currently selected quantity versus distance along the scan and activates the commands in the *Scan plot* menu (see Section 15). The program adds fiducial lines to the plot using intelligent grid selection. This means that the plot is adjusted to fill the screen and grids are drawn at useful intervals (like 0.05 or 2.00).

### Scan plot quantity

With this command you can choose the quantity to display in screen and hardcopy plots of line scans. Pick the quantity from the list box and click *OK*. This setting has no effect on the history file listing which includes all field quantities. The available line scan quantities depend on the solution type. The options are: vector potential ( $A_z$  or  $A_\theta$ ), horizontal component of magnetic flux density ( $B_x$  or  $B_z$ ), vertical component of magnetic flux density ( $B_y$  or  $B_r$ ), magnetic flux density amplitude ( $|\mathbf{B}|$ ), relative magnetic permeability ( $\mu_r$ ) and magnetic field amplitude  $|\mathbf{H}|$ .

### Set number of scan points

This command sets the number of line scan points used for the screen plot and recorded in the history file. The default value is 50 and the maximum number is 500.

## Element properties

After you pick a triangular element with the mouse (or keyboard), **PerView** writes material and field properties of the element to the screen. The information is also recorded if a history file is open. It is usually best to start from a *Mesh*, *Region*, *Element* or *Vector* plot to identify the element.

## Matrix file

**PerView** can make matrix files of field values to help you create your own analysis routines. Although information is also available in the output file of the solution program, it is difficult to deal with the conformal triangular mesh. The *Matrix file* command uses the interpolation capabilities of the program to create an ASCII data file of field quantities on a rectangular grid in x-y or z-r. The command displays a dialog box where you set the matrix file prefix, the dimensions of the box and the number of intervals along *x* and *y* (or *z* and *r*). The program creates the file `FPREFIX.MTX` in the current directory.

## Line integrals

**PerView** calculates line integrals that are useful for calculations of enclosed current and force. The quantity listed as *Total Current* is the Ampere's law integral ,

$$\int \mathbf{B} \cdot d\mathbf{l} / \mu_o .$$

If you take line scans around a closed path in the solution space, the sum of the integrals gives the enclosed total current (free current plus the contribution from magnetic materials). The quantity *Free Current* is the integral

$$\int \mathbf{B} \cdot d\mathbf{l} / \mu_r \mu_o .$$

The integral of the magnetic stress tensor (see, for instance, S. Humphries, **Field Solutions on Computers** (CRC Press, Boca Raton, 1997), Sect. 10.5) is also recorded. Multiple scans defining a closed

path can be used to find the force on a region. The results are invalid if the path passes through a region with  $\mu \neq \mu_o$ .

### Volume integrals

The following analyses are performed in response to the *Volume integrals* command.

- 1) Magnetostatic field energy density,  $u = B^2/2\mu,\mu_o$  over the solution space and region volumes. The output units are J/m in rectangular solutions and J in cylindrical solutions.
- 2) Coil forces, the volume integrals of  $\mathbf{j} \times \mathbf{B}$  for all regions with non-zero current. The units are newtons/m for rectangular problems and newtons for cylindrical problems.
- 3) Coil torques, the volume integrals of  $\mathbf{x} \times \mathbf{j} \times \mathbf{B}$  for all regions with non-zero current. The vector  $\mathbf{x}$  is taken relative to the origin, ( $x = 0.0$ ,  $y = 0.0$ ). This calculation is valid only for planar simulations. The units are N-m/m.
- 4) The location and value of the maximum magnetic field in the solution space and in each region.
- 5) Flux integrals to find the self-and mutual inductances of coils. In a planar simulation, the inductance factor for a region is given by

$$IndFact = \frac{\int dx \int dy A_z(x,y)}{\int dx \int dy} .$$

The integral is taken over the region area. In a cylindrical simulation, the inductance factor is

$$IndFact = \frac{\int dz \int dr 2\pi r A_\theta(z,r)}{\int dz \int dr} .$$

A discussion on the use of the vector potential integrals to find self-

inductance and mutual inductance is given in S. Humphries, **Field Solutions on Computers** (CRC Press, Boca Raton, 1997), Sect. 10.4.

### Region properties

In response to the *Region properties* command, **PerView** calculates volume integrals of field energy density and power density over the region. Several quantities are calculated from line integrals around the region boundary.

1) Region current. Automatic identification of region surfaces and calculation of surface integrals of Ampere's law over each region to determine enclosed current. The total current is the integral  $\int \mathbf{B} \cdot d\mathbf{s} / \mu_o$ , while the free current is the integral  $\int \mathbf{B} \cdot d\mathbf{s} / \mu$ . The free current should equal the current assigned to the region. The quantity is included to check the accuracy of the surface integrals. Note that the free current integral is invalid when two ferromagnetic materials ( $\mu \neq \mu_o$ ) share a common boundary. The surface integral includes the total atomic surface current along the shared boundary and therefore may give non-zero free current for a ferromagnetic region.

2) Region forces, given by a surface integral of magnetic stress over a filled region. For coil regions, this result can be compared to that of the volume integral method. Generally, the surface integral is less accurate for coils, particularly on coarse meshes with discontinuous region boundaries (*i.e.*, square cross section). The forces computed by the surface integral method are generally accurate for magnetic materials with surface currents. Check the accuracy by running two or more solutions with different mesh resolution. Because of ambiguities in resolving atomic surface current layers, the results are invalid when two ferromagnetic materials ( $\mu \neq \mu_o$ ) share a common boundary.

3) Region torques. This computation applies only to filled regions in rectangular problems and uses the surface integral method. The torque is calculated with respect to an axis at the point (0,0). Pick the values of  $x_{\min}$ ,  $x_{\max}$ ,  $y_{\min}$  and  $y_{\max}$  in **Mesh** so that this point corresponds to the correct physical axis. The results are given in newton-m/m. Again, the results are invalid when ferromagnetic materials share a boundary.

## 18. PerView scan plot menu

The commands of the *Scan plot* menu become active when a plot is created following the line scan command. The commands to export plots were described in Section 12. This section describes unique commands of the Scan plot menu.

### Oscilloscope mode

In the oscilloscope mode, a scan plot assumes many of the characteristics of a digital oscilloscope. The program superimposes a cross-hair pattern on the graph. Plot values at the intersection are displayed in the information window. Move the marker along the plot by moving the mouse. If you click the left mouse button at a point, the program displays the plot values along with the numerical derivative and integral of the curve. The definite integral is taken from the left-hand side of the plot to the current point. Values are displayed on the screen and written to the data file if open. Press the right mouse button to exit the oscilloscope mode.

### Toggle scan symbols

The setting determines whether plot symbols are added to the scan plot showing calculated points.

### Toggle grid

The setting determines whether grid lines are added to the screen and hardcopy scan plots.

### Close scan plot

The scan plot must be closed before you can use the *File* and *Analysis* functions of **PerView**. This command closes the scan plot and returns the program to the previous spatial plot.

## 19. PerView script file commands

Script files to control **PerView** have a name of the form `FPREFIX.SCR`. They should be in the same directory as the data files. Scripts are ASCII files that follow the **TriComp** syntax conventions. The program ignores blank lines and indentations. Data lines use the standard delimiters and comment lines begin with ‘\*’ (asterisk). Processing ends when the *EndFile* command is encountered.

To run a script, click on the *Run script* command in the *File* menu. The program shows a list of available script files. Pick a file and click *OK*. The script operates on the presently loaded data file or you can load other files from within the script. You can also sequentially open one or more data files.

**PerView** can run autonomously under script file control from the Windows Command Prompt. Suppose you have a file `GTest.SCR` in the directory `\TRICOMP\BUFFER` and that `PerView.EXE` is in the directory `\TRICOMP`. From `\TRICOMP\BUFFER`, type

```
..\PerView GTEST <Enter>
```

The main application of the Command Prompt mode is to generate data files and to perform extensive analyses under batch file control.

The following commands can appear in a script:

### Input **SOLENOID.POU**

Close the current data file and load a file for analysis. The parameter is the full name of the data file. For the command illustrated, the post-processor would load the file `SOLENOID.POU`. You can load several files for sequential analysis.

### Output **SW02**

Close the current history file and open an output file `SW02.DAT`.

### Point **5.65 10.68**

Perform interpolations at the specified point and write the results to the data file. The two real number parameters are the coordinates of the point ( $x$ - $y$  or  $z$ - $r$ ) in **Mesh** units.

**Scan 0.00 0.00 10.00 0.00**

Write the results of a line scan between the specified points to the data file. The four real number parameters are the starting and end coordinates in **Mesh** units:  $[x_{\text{start}} y_{\text{start}} x_{\text{end}} y_{\text{end}}]$  or  $[z_{\text{start}} r_{\text{start}} z_{\text{end}} r_{\text{end}}]$ .

**Interpolation LSQ**

Set the interpolation method for subsequent *Point*, *Line scan* and *Matrix* commands. The options are *LSQ* (least-squares fit) and *Linear*.

**Element 5.65 10.68**

Write the properties of the element at the specified point to the data file. The two real number parameters are the coordinates of the point (*x-y* or *z-r*) in **Mesh** units.

**NScan 150**

Set the number of points in a line scan. The default is 50 and the maximum number is 500.

**Region 5**

Write volume and surface integrals for a region to the data file. The integer parameter is the region number.

**LineInt 0.00 0.00 10.00 0.00**

Write line integrals along a scan line to the data file. The four real number parameters are the starting and end coordinates in **Mesh** units:  $[x_{\text{start}} y_{\text{start}} x_{\text{end}} y_{\text{end}}]$  or  $[z_{\text{start}} r_{\text{start}} z_{\text{end}} r_{\text{end}}]$ .

**VolumInt**

Write volume integrals for the full solution and regions to the data file.

### **Matrix Switch1 10 20 0.00 0.00 5.00 10.00**

Create a matrix file and record values. This command must have the following seven parameters.

Param 1: The prefix of the matrix file `FPREFIX.MTX` (string).

Param 2: Number of intervals along the  $x$  (or  $z$ ) direction (integer).

Param 3: Number of intervals along the  $y$  (or  $r$ ) direction (integer).

Param 4-7: The coordinates of the corners of a box in the solution volume,  $[x1,y1,x2,y2]$  or  $[z1,r1,z2,r2]$  (real).

### **Endfile**

Terminate execution of the script file. You can add descriptive text in any format after this command.

The following is an example of a script file to compare field values along the axes of four different solutions and to write the results in a file `COMP.DAT`.

```
OPEN OUTPUT COMP
OPEN INPUT SWITCH01.POU
SCAN 0.00 -50.00 0.00 50.00
OPEN INPUT SWITCH02.POU
SCAN 0.00 -50.00 0.00 50.00
OPEN INPUT SWITCH03.POU
SCAN 0.00 -50.00 0.00 50.00
OPEN INPUT SWITCH04.POU
SCAN 0.00 -50.00 0.00 50.00
ENDFILE
```

## **20. Format of the PerMag output file**

The **PerMag** output file `FPrefix.POU` is in text format. You can inspect it with any text editor. The file has three sections:

- Header with general information on the run
- Node and element information
- Region information

The header section consists of a title line and 9 data lines:

```

-- Run parameters ---
XMin: -1.270003E-01
XMax:  1.270003E-01
KMax:  101
YMin:  0.000000E+00
YMax:  2.540005E-01
LMax:  101
DUnit:  3.937000E+01
NReg:   5
ICylin:  1

```

Lines 2 and 3 list  $x_{\min}$  and  $x_{\max}$ , the limits along horizontal axis ( $x$  or  $z$ ) of the solution volume. Dimensions are given in meters. The quantity  $K_{\max}$  in Line 4 is the number of nodes along the horizontal direction. Lines 5-7 describe the vertical axis ( $y$  or  $r$ ). Line 8 contains the quantity  $DUnit$ , the conversion factor from dimensions used in **Mesh** to meters. Line 9 gives the number of regions in the solution, while Line 10 specifies the symmetry (0: planar, 1: cylindrical).

The node section consists of 4 title lines and  $K_{\max} \times L_{\max}$  data lines, one for each node of the solution space.

```

--- Vertices ---
k    l  RgNo RgUp RgDn      x          y          Pot
=====
  1   1    5   1   0  0.000000E+00  0.000000E+00  0.000000E+00
  2   1    5   1   0  6.250087E-04  0.000000E+00  0.000000E+00
  3   1    5   1   0  1.250009E-03  0.000000E+00  0.000000E+00
  4   1    5   1   0  1.875005E-03  0.000000E+00  0.000000E+00
  5   1    5   2   0  2.500000E-03  0.000000E+00  0.000000E+00
...

          Mu1Up      Mu1Dn      Mu2Up      Mu2Dn
=====
  1.000000E+00  1.000000E+00  1.000000E+00  1.000000E+00
  1.000000E+00  1.000000E+00  1.000000E+00  1.000000E+00
  1.000000E+00  1.000000E+00  1.000000E+00  1.000000E+00
  1.000000E+00  1.000000E+00  1.000000E+00  1.000000E+00
  3.473353E+03  1.000000E+00  3.473353E+03  1.000000E+00
...

```

Each data line contains the following quantities:

- The indices of the node ( $K, L$ )
- The region number of the node ( $RgNo$ ) and region numbers for two associated elements ( $RgUp$  and  $RgDn$ ). The upper element lies about the line between nodes ( $K, L$ ) and ( $K+1, L$ ) and the lower element lies below the line.

- The coordinates of the element in meters ( $x, y$ )
- The vector potential  $A_z$  at the node in tesla-m or the stream function  $rA_\theta$  in tesla-m<sup>2</sup>.
- The values of relative magnetic permeability along Axis 1 and Axis 2 in the upper and lower associated elements.

The region section consists of four title lines following by  $NReg$  data lines, one for each region. The region section has the following appearance:

```

--- Regions ---
RegNo Fix Aniso                Mu1                Mu2
  1    0    0    0    0    0  1.000000E+00  1.000000E+00
  2    0    0    0    0    0  1.000000E+00  1.000000E+00
  3    0    1    0    0    0  1.000000E+00  1.000000E+00
  4    1    0    0    0    0  1.000000E+00  1.000000E+00
...
      RegAngle      RegCurrent      Potential      Area      Bc
0.000000E+00  0.000000E+00  0.000000E+00  3.500001E-04  0.000000E+00
0.000000E+00  0.000000E+00  0.000000E+00  3.250000E-04  0.000000E+00
0.000000E+00  0.000000E+00  0.000000E+00  7.499999E-05 -1.360000E-01
0.000000E+00  0.000000E+00  0.000000E+00  0.000000E+00  0.000000E+00
...

```

An entry of unity in the second column designates a fixed-potential region such as a Dirichlet boundary. A unity value in the third column indicates an anisotropic material with different values of  $\mu_r$  along two normal axes. The real number parameters have the following values: relative magnetic permeability along Axes 1 and 2, the angle of Axis 1 with respect to the  $x$  or  $z$  direction, coil current, vector potential or stream function, cross-section area in m<sup>2</sup>, and the coercive value of magnetic flux density. Note that not all values may be used for a particular region.

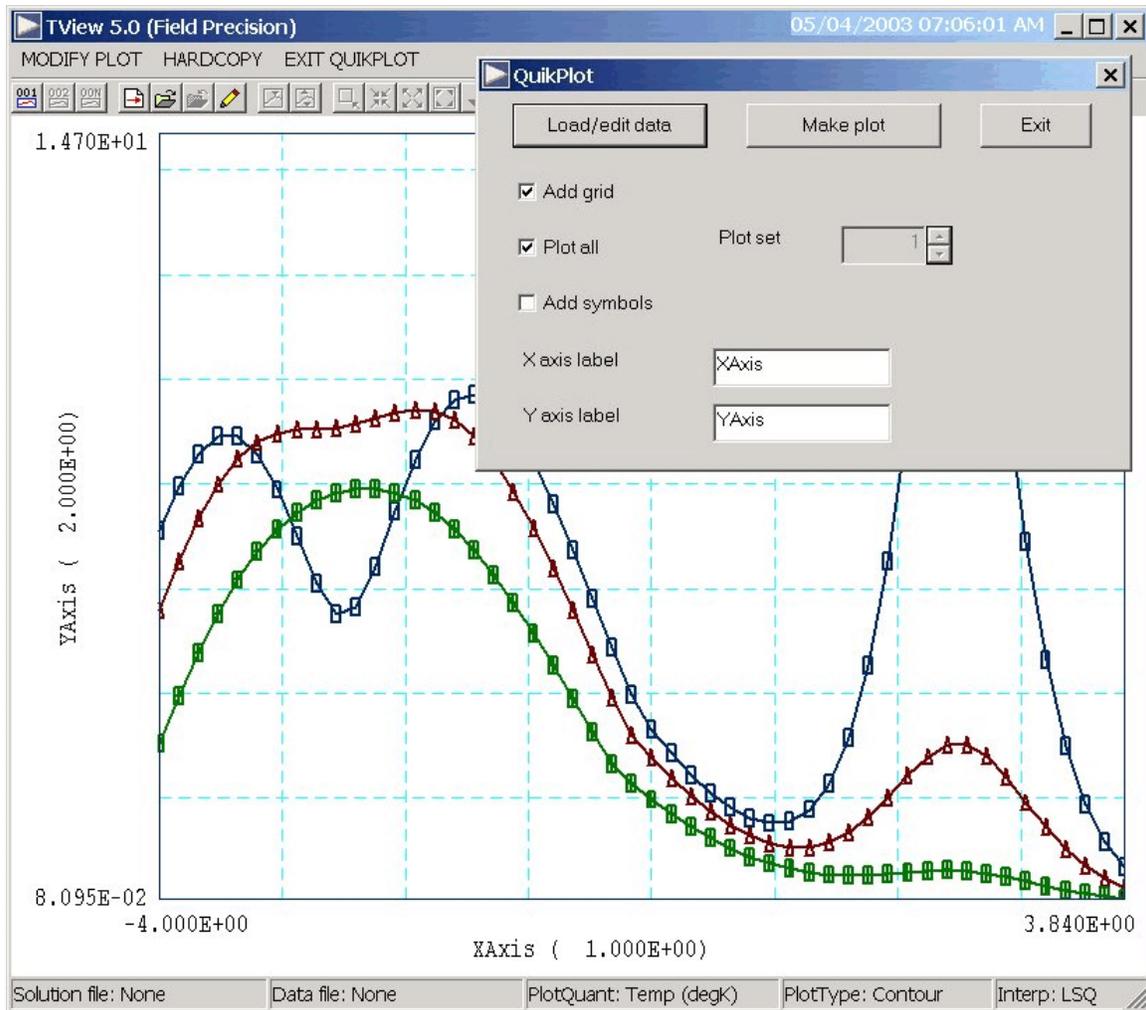


Figure 10. Control dialog superimposed on QuikPlot screen.

## 21. QuikPlot utility

Postprocessors for **TriComp** and **Amaze** programs contain a utility for making fast, simple plots of numerical data. The feature may be useful, for example, to check a table that defines non-linear magnetic permeability. Click on *QUIK PLOT* in the *TOOLS* menu to bring up a dialog (Fig. 10). Initially, all commands are deactivated except *LOAD/EDIT DATA*. Clicking this command brings up an editor window. You can paste numerical data that you have copied from text files or spreadsheets into the window. You can also use the editor function to make changes or to save the data to a file.

The data should consist of columns of numbers separated by any of the standard delimiters (space, tab, comma, equal sign, parentheses). The first column contains  $x$ -values. Additional columns contain from 1 to 5 sets of  $y$ -values. All columns must have the same number of rows. Although there is no limit to the number of rows, the program will pick a maximum of 250 evenly-spaced values to make the plots. Click *SAVE DATA* when you are through.

When you return to the main dialog all commands should be active. Click *MAKE PLOT* to see a plot of the data. The program adjusts the ranges and grid intervals automatically. When the plot is active you can export it in various formats or copy it to the clipboard. Click *MODIFY PLOT* to return to the main dialog to make changes. Here you can change the display of grids and plot symbols or add text labels to the  $x$  and  $y$  axes. Note that the  $x$ - $y$  labels also show the grid intervals. By default, **QuikPlot** shows all data sets (columns of  $y$ -values) with different line colors and plot symbols.

You can plot a particular set by unclicking the *PLOT ALL* box and specifying the set number (1-5). The plot range is automatically adjusted to the set. In the main dialog, click on *EXIT QUIK PLOT* to return to the normal functions of **PerView**.