Magneticasoft

Magnetica
User Book

Version 1.0

Short Contents

Table of Contents

■ Chapter 1: Introduction

■ 1-1 About *Magnetica*

The release that comes with this book is the first public release of *Magnetica*, a high precision magnetostatics software. This software is a response to the need for high precision field calculation in MRI design. It was at first developed by Professor Lucien Donadieu for his personal use.

Magnetica was developed under the scientific system *Mathematica* which is a powerful,fully integrated environment for technical computing. It offers the user access to a vast collection of algorithms as well as to a programming language.

Magnetica is designed so that only a very limited knowledge of *Mathematica* is required. A large variety of examples (more than a hundred) is integrated into *Magnetica*, thus reducing learning time. By using a limited set of commands, the user can create and display magnetic systems, compute fields, draw field maps and compute even more complex functions.

Combining *Magnetica* and *Mathematica* allows the user to solve advanced magnetostatics problems.

This release is fully functional for 2D cylindrical axis symmetric systems for all kinds of configurations even with ferromagnetic materials, including field, gradient, flux and inductance calculations.

■ 1-2 Physics and Mathematics in *Magnetica*

Magnetica is based on classical Physics originating from the standard Maxwell Equations restricted to time-independent phenomena.

It is assumed, without restriction, that the magnetic field is amperian i.e. that any magnetic field, whatever its origin, is always produced with an electric current. This hypothesis is very easy to understand for a current carrying generator such as a straight line (Ampere law), or a circular current loop. For magnetic materials, this hypothesis implies surface current. It can be shown, again from the same Maxwell Equations, that the intensity of magnetization and surface currents are two related ways of expressing the capability of matter to generate magnetic field. This unified amperian approach has many advantages for both grasping the physics of magnetism and magnetostatics, as well as for carrying out all related calculations.

For a magnetic system made of many different sources, the total field is simply the vector sum of all the elementary contributions (law of composition). No other physical hypothesis is implied in *Magnetica*.

The physical properties of the materials involved are accepted as such without discussion. For instance, the laws of magnetization of ferromagnetic materials are treated as phenomenological physical laws and used in *Magnetica* as mathematical functions that simulate perfectly the experimental curves. This modeling is a proprietary method of MAGNETICASOFT. The rest is just the application of mathematical methodology to solve, in the best possible ways with regard to both accuracy and calculation time, the fundamental Maxwell equations combined, for magnetic materials, with the relevant laws of magnetization.

■ 1-3 General Features

Magnetica is a high-precision (generally 10^{12}) calculation engine, not only for magnetic fields but also for various related functions including field gradients, magnetic potential, fluxes, inductances, magnetic energies (forces and torques will be implemented in a future version). Results can be shown as numerical values, tables and graphics. Taking advantage of *Mathematica*, offers the possibility of parametrizing data and easy post-processing.

Magnetica has 2 different basic modes of calculation for the magnetic field. They are based on the derivation from either the vector potential or the scalar potential processed using spherical harmonics. For practical reasons, another operating mode is implemented (the automatic mode). The automatic mode chooses simply the most efficient basic mode in terms of speed and accuracy. It is the default mode of calculation but it can be overridden either selectively or generally for any of the basic modes.

Magnetica defines a magnetic system as a set of any number (at least 1) of primary magnetic generators (called magnetic components). In the initial version 1.00, only cylindrical axis symmetric components are admitted. They are made of current carrying windings (cylindrical loop or solenoid) or cylinders, rings or bars of magnetic materials set on the same axis. All types of magnetic materials are admitted including paramagnetic, diamagnetic and ferromagnetic, both soft and hard (permanent magnet).

Magnetica is concerned primarily with steady state phenomenon. However, beyond these straightforward situations, *Magnetica* can also be used for special cases for the analysis and design of apparatuses and devices involving transient or periodic phenomena. This occurs when the relevant dimensions of the magnetic system are such that the magnetic coupling is much more important than the electromagnetic coupling (which implies wavelength). If the dimensions are much less than the relevant wavelength, then the magnetic coupling is dominant and time-independent. Secondary phenomenon related to the properties of the materials (eddy currents, skin effects) may interfere with the time-independent calculations, resulting in incorrect evaluations. But the appropriate choice of materials, to a large extent, allows avoiding these deleterious secondary effects. It is considered that the high frequency value of the inductance is within 5 %, of the DC value (Terman, Radio Engineering Handbook).

Moreover, the combined uses of the specific method for magnetic systems with ferromagnetic parts has led to the definition of a new concept of inductance namely ferro-inductance. The application of this concept to the analysis of behavior or design of magnetically coupled devices (inductor or transformer) gives very valuable data on their behavior under all load conditions.

■ 1-4 *Magnetica* Practice and Help

Magnetica, like the parent software, *Mathematica*, is organized in work sessions. The first thing to do to start a new work session is to

1. Launch *Mathematica*,

2. Load *Magnetica* with the initial command:

```
Get["Magnetica`"]
```

or

```
<< Magnetica`
```

Don't forget the character back quote, see appendix B.4: Practical Usage and Miscellaneous for common Help and tips.

You normally interact with *Magnetica* through computer objects called notebooks. Notebooks can have many forms, but typically they consist of cells that can contain input, text, calculations, or graphics. After working in a Notebook,

you can give it a name and save it as a file with the extension ".nb". You can also reopen it and reuse it later. You can have several notebooks within a session.

It should be noted that *Magnetica* can be used without enterring into the intricacies of *Mathematica* programming. You have just to know a few elementary rules that are explained in Appendix B.4. However all the capabilities of *Mathematica* are available at any time within a session.

Then in the next step, you must define a magnetic system. This is the basic computer object of *Magnetica* and you can give it a name and then work with it. You can have several magnetic systems within a single notebook.

Magnetics, magnetic physics and magnetic fields involve complex phenomena. Even Magnetostatics which is restricted to time-independent situations, leads to intricate mathematical relations, particularly with ferromagnetic materials. *Magnetica* offers a large number of functions that are able to deal with very complicated magnetic systems involving dozens of components. In order to facilitate the understanding of the science of Magnetostatics, you can use the *Magnetica* User Book as a guide to approach through its 13 chapters, increasingly more complex situations. In addition, a Reference Guide or immediate Help (using ? followed by the name of the function) provides direct information on any *Magnetica* function.

And finally, a set of magnetic system cases shows a wide variety of examples where you can find topics such as: 1) learning and training on basic magnetics, 2) many typical applications related to magnet and coil analysis and design, magnetic shielding, electromagnetic compatibility and inductance applications, including inductors and transformers and 3) magnetic materials (see Appendix A1).

All subjects of these various help tools are directly available at any time during a session through the Help Browser at the section Add-ons ▷ Magnetica.

■ Chapter 2: The Basic Circular Current Loop

■ 2-1 Introduction

The single circular current loop is the simplest cylindrical magnetic field generator for axis symmetric systems. All calculations of more elaborate cylindrical magnetic components are derived from the basic circular loop shown in the following figure:

AXSCircularLoop
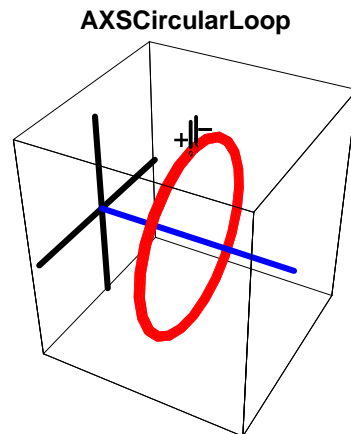
■ 2-2 Entering Data

■ 2-2-1 First example

For example, let us assume a single turn of a 1 millimeter diameter copper wire wound on a circular path of 10 centimeters in diameter and fed by a steady current of 1 ampere. These data can be entered in various ways, such as:

- copying from previous data,

- taking one *Magnetica* example case (see appendix A.1),

- using palettes (see Appendix B.3),

- or typing directly the data as follows:

```
loop = AXSMagneticSystem[
{AXSCircularLoop[CopperOFHC, 100,
     1, 0, 1, 1, ConductorDiameter → 1, Temperature → 27]}];
```

- The first term on the left hand side is the call name of the magnetic system. The name and its spelling are totally free, but it is recommended to use only low case letters to avoid possible interference or conflict with *Magnetica* or *Mathematica* reserved names.

- The right-hand side is the *Magnetica* expression which is constructed as follows:

■ 2-2-2 Definition of the Cylindrical System

The first term, `AXSMagneticSystem`, defines the type of magnetic system used, i.e., a 2D AXiSymmetric system, (the other type of magnetic system is the 3D Cartesian System). A magnetic system is made of at least one magnetic axis symmetric component.

- The arguments for any function are always given between square brackets [...], not between parentheses.
- The first argument of the `AXSMagneticSystem` function is a list of components given between curly braces {...} (see below).
- The following arguments are optional.

■ 2-2-3 Definition of the Cylindrical Loop Component

Component reserved symbolic name : `AXSCircularLoop`

The arguments for this component are, as always, given between square brackets. Here is the description of each argument of the example:

N.B. the example uses the default units (see appendix B.2)

- The material name: `CopperOFHC` this is a reserved *Magnetica* name. It defines a set of physical properties that will be used for the calculation. (see the MagneticaMaterials palette and the Appendix B.1 about Materials)
- The loop middle diameter: 100(millimeters) this is the distance from the axis to the center of the winding. The dimensions given are the nominal dimensions, that is, at the reference temperature (20° Celsius or 68°Fahrenheit)
- The winding diameter: 1millimeter: this is the diameter of the loop winding section (nominal dimension at 20° Celsius or 68°Fahrenheit)
- The loop position: 0 millimeter this is the position of the center of the loop on the symmetrical axis. The coordinate position is assumed to be temperature-independent.
- The number of turns: 1 (dimensionless) the number of turns of the winding is a signed value according to the standard "right hand rule" orientation.
- The current intensity: 1 Ampere the current per conductor. This number is a signed value according to the direction of the current (+,-).
- The wire diameter: 1 millimeter that is an optional item of information among other possibilities. It is specified by typing `ConductorDiameter->1`.
- The temperature: 27° Celsius this is also an optional argument, it is necessary to indicate it only if it is different from the reference temperature 20°. The actual temperature will be used in the calculation to evaluate the exact dimensions and other temperature-dependent properties. It is specified by typing `Temperature->27`.

These data can also be entered using palettes (see Appendix B.3). This approach is very instructive and is recommended for new *Magnetica* users.

■ 2-2-4 Another example: the Helmholtz coil pair

The Helmholtz pair of circular coils is a couple of identical coils set symmetrically with respect to the central radial plane and with a special gap length between the coils.

The next example is for 2 coils made of 10 turns of the same copper wire mentioned above and loaded with a 4 amperes current. It is obtained by typing:

```
helmholtzloop = AXSMagneticSystem[{
     AXSCircularLoop[CopperOFHC, 100, 10, -25, 10, 4, ConductorDiameter → 1],
     AXSCircularLoop[CopperOFHC, 100, 10, 25, 10, 4, ConductorDiameter → 1]}];
```

■ 2-3 Immediate Help

On line direct information, including options, on any *Magnetica* function can be obtained selectively and at any time during a session by typing Question mark (?) followed by the name of the function.

```
?AXSMagneticSystem
```

```
AXSMagneticSystem[{comp1,comp2,...},options]  defines
   a new axis symmetric system made with the specified components.
```

```
Options: SystemLabel, MeshingLevel,
   LegendrePointPosition, LegendreRadius, LegendreOrder Details…
```

```
? AXSCircularLoop
```

```
AXSCircularLoop[material,nominalMidDiameter,
   windingNominalDiameter,position,numberOfTurns,intensity,options]
```

```
builds a Magnetica object that represents a current-carrying
   winding, of circular cross section, in 2D cylindrical axis symmetric space
```

```
Options: FillingFactor, ConductorDiameter, Temperature, FilamentMode Details…
```

```
? ConductorDiameter
```

```
ConductorDiameter considers that the solenoid or the loop given is made of a conductor with
   circular section and sets the conductor diameter to the specified value Details…
```

■ 2-4 Data Validation

Warning: Data are validated (as they always are in *Mathematica*) by typing Shift-Return and not only Return. Typing semi colon (;) at the end of the expression hides the results of the function called.

Once input data are entered, the validation command triggers the processing of the data. If data can be processed a simple generic message is displayed as shown next:

```
loop = AXSMagneticSystem[
{AXSCircularLoop[CopperOFHC, 100,
     1, 0, 1, 1, ConductorDiameter → 1, Temperature → 27]}]
```

```
- AXSMagneticSystem :{-loop-} -
```

Then, all further calculations and drawings can be made using the system name, here the symbol "loop"

In the case of abnormal data, an error message is displayed, such as:

```
wrongloop = AXSMagneticSystem[
{AXSCircularLoop[Copper, 100, 1, 0, 1, 1, ConductorDiameter → 1]}]
```

AXSCircularLoop:
  Argument 1 (Copper)
 must be a current-carrying material (resistive or superconductor)

```
Details…
```

```
$Aborted
```

■ 2-5 System Drawing

■ 2-5-1 Drawing the Single Coil

A first visual control of the magnetic components can be obtained with a 3D or 2D drawing (to scale) of the system.

- 3D View

DrawSystem[loop]

System name: **loop**

3D view ZYX
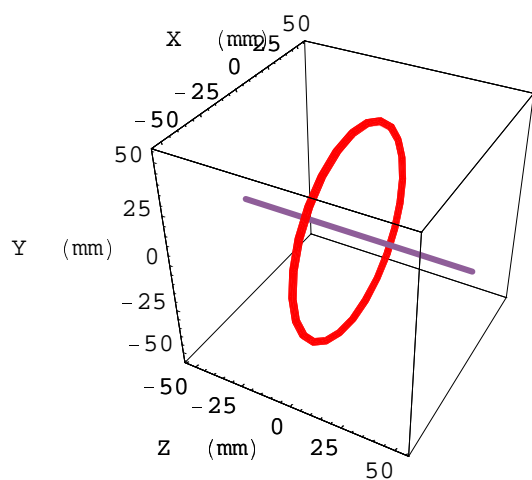
- Graphics3D -

- 2D drawing in the XY plane and in the ZX plane

DrawSystem2DXY[loop]

System name: **loop**

2D view XY
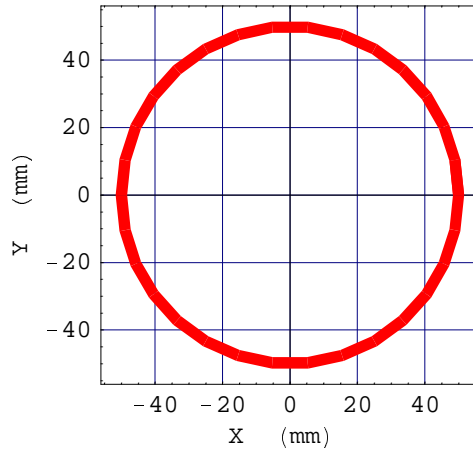
- Graphics -

DrawSystem2DZX[loop]

System name: **loop**

2D view ZX
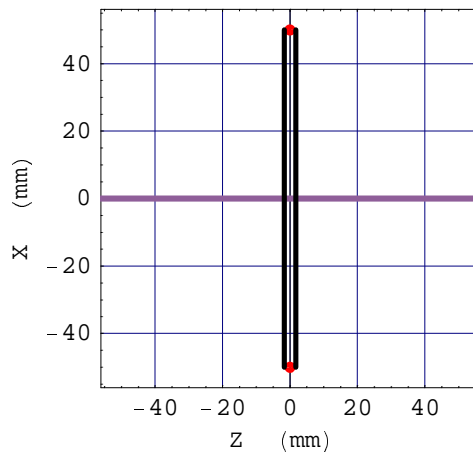
- Graphics -

■ 2-5-2 Drawing the Helmholtz Coil Pair

3D View and 2D View

```
DrawSystem[helmholtzloop];
DrawSystem2D[helmholtzloop];
```

System name: **helmholtzloop**

3D view ZYX
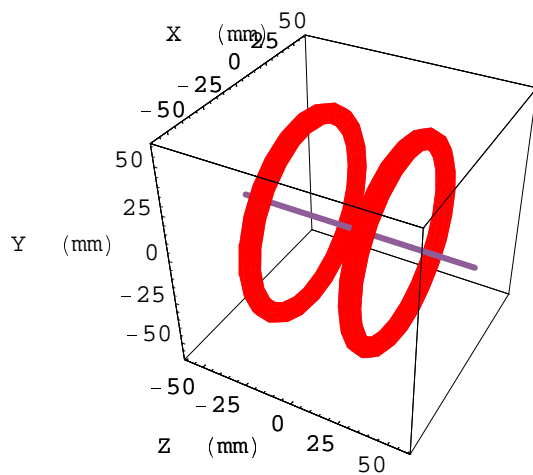
System name: **helmholtzloop**

2D view ZR
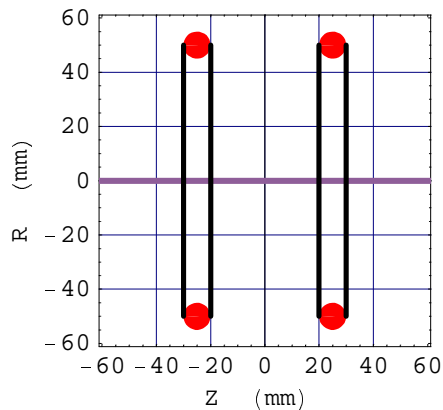
■ 2-6 Field Calculation

The Magnetic Field can be calculated at any point in space: the target point.

The coordinates of the target point are given as a list of two numbers {r,z}, the first representing the r-coordinate (distance from the axis), and the second representing the z-coordinate (distance between the origin and the projection of the point on the axis).

The output is given as a list {Br,Bz}, representing respectively the field vector in the radial direction, and the field vector in the axial direction.

The default units are Milli Meter for length units and Gauss (0.0001 Teslas) for magnetic field. But any other units can be used (See Appendix B.2 on units)

Standard output precision is 6/7 digits but greater output precision can be obtained using the *Mathematica* command `NumberForm`.

Standard internal number precision is to 16 digits. However, through *Mathematica* functionalities, greater internal precision can be obtained.

1. Magnetic field of the single loop at the target field point 30 mm radial and 10 mm axial, standard output

   ```
   FieldPoint[loop, {30, 10}]
   ```

   {0.0404502, 0.147779}

2. `NumberForm` is a *Mathematica* function that allows displaying greater or smaller number of digits. `FieldPoint` output can be given as an argument to this function to obtain, for instance, an output to 10 decimal places.

   ```
   NumberForm[FieldPoint[loop, {30, 10}], 10]
   ```

   {0.04045022074, 0.1477785966}

3. It is possible to have a more detailed and formatted output for `FieldPoint` (and for many *Magnetica* functions) by prefixing the function name with the word **Verbose**. However, this output cannot be used as an argument for other calculation functions.

Here is an example for the Magnetic field of the Helmholtz loops pair at the target point 30 mm radial and 10 mm axial

   ```
   VerboseFieldPoint[helmholtzloop, {30, 10}]

   System name: helmholtzloop
   The magnetic field (in Gauss) at the point {30, 10}(in Milli Meter)
   Calculation mode: Automatic Mode
   Br          -0.625297 G
   Bz           7.232273 G
   Bm           7.259254 G
   ```

4. Other magnetic field functions can be obtained the same way using the proper function name (see chapter 4 for more details). For instance, the magnetic flux in a circular area defined by its radius and its axial coordinate is obtained as follows (the default unit is Micro Weber).

   - Magnetic flux of the single loop through a circle with a radius of 30 mm centered on the axis, and set at z = 10 mm axial, standard output

   ```
   FluxArea[loop, {30, 10}]
   ```

   {0.0374576}

   - Magnetic flux of the Helmholtz loops pair through a circle with a radius of 30 mm at the axial coordinate 10 mm axial, standard output

   ```
   FluxArea[helmholtzloop, {30, 10}]
   ```

   {2.05361}

■ **2-7 Field Map Graphics**

■ **2-7-1 Introduction**

Magnetica offers many possibilities of showing 2D and 3D graphics for the field map and related functions. (See chapter 4 for more details).

■ **2-7-2 Field Map of the single loop**

▪ Field map of the single loop on a line parallel to the axis for the magnetic field axial and radial components `Bz` and `Br`.

```
FieldMap[loop, {Bz, Br}, Line[{30, -50}, {30, 50}]];
```

      System name: **loop**

      Bz (G) (Continuous, Automatic Mode)
      between {30., -50.} and {30., 50.}
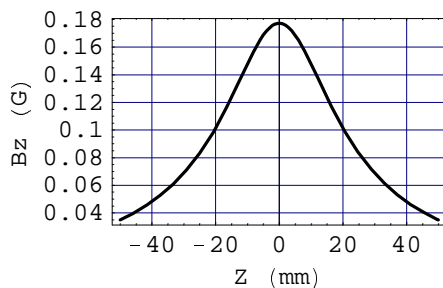
      System name: **loop**

      Br (G) (Continuous, Automatic Mode)
      between {30., -50.} and {30., 50.}
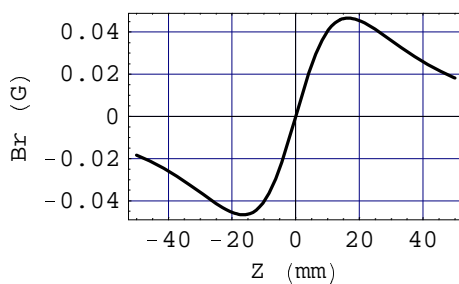
▪ 3D field map for the single loop on a cylindrical shell with a 30 mm radius and 100 mm length for the Bz field component, using standard graphics parameter definition (15 x 15 points).

```
FieldMap[loop, {Bz}, Plane[{-30, -50}, {30, 50}]];
```

```
System name: loop

Bz (G) (Continuous, Automatic Mode)
between {-30., -50.} and {30., 50.}
```
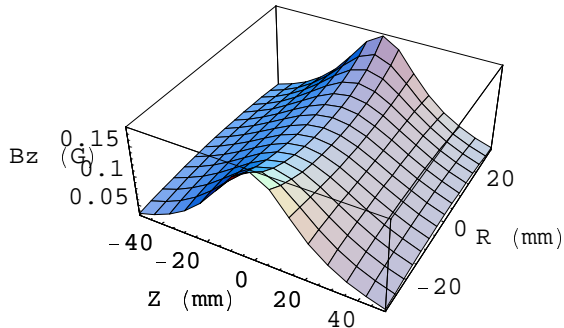

- 3D vector field map for the single coil of the magnetic field vector B on a plane through the axis with a 200 mm diameter and 180 mm length using standard graphics parameter definition (15 x 15 points).

```
VectorFieldMap[loop, Plane[{-100, 0, -90}, {100, 0, 90}]];
```

```
System name: loop

vector field (Discrete, Automatic
Mode)
between {-100., 0., -90.} and {100., 0., 90.}
```
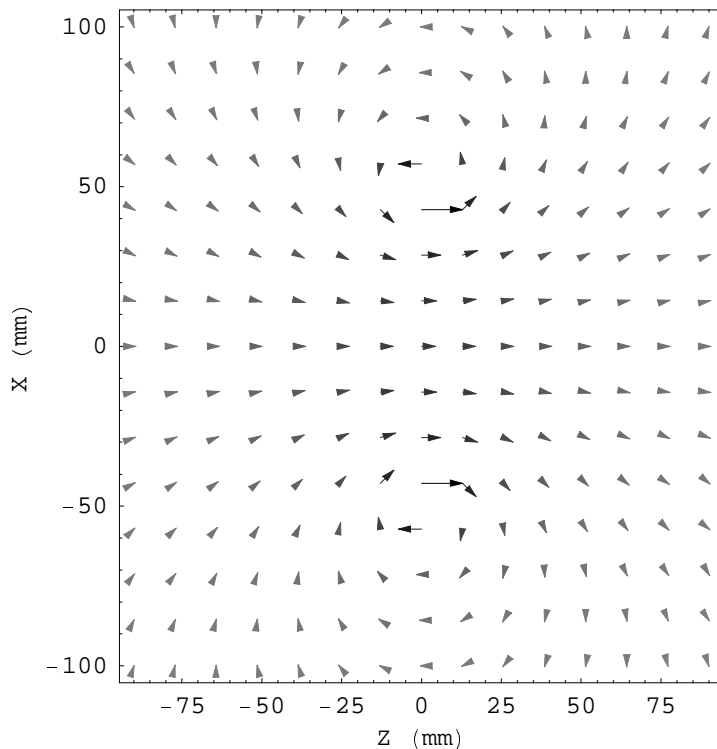
■ **2-7-3 Field Map of the Helmholtz Coil Pair**

■ 2D field map for the Helmholtz coil pair on a line parallel to the axis for the Bz and Br field components.

```
FieldMap[helmholtzloop, {Bz, Br}, Line[{30, -100}, {30, 100}]]
```

System name: **helmholtzloop**

Bz (G) (Continuous, Automatic Mode)
between {30., -100.} and {30., 100.}
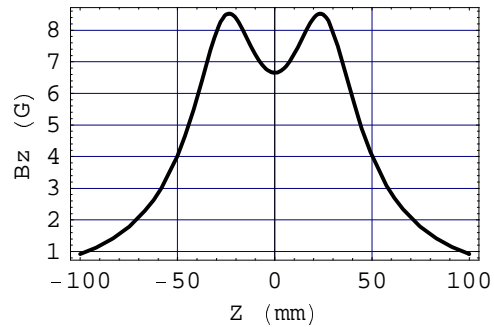
System name: **helmholtzloop**

Br (G) (Continuous, Automatic Mode)
between {30., -100.} and {30., 100.}
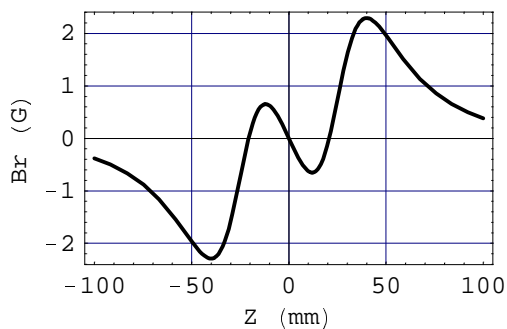
{- Graphics -, - Graphics -}

■ 3D contour field map for the Helmholtz coil pair on a 60 mm long × 60 mm wide plane passing through the axis, of the Bz and Br field components using the graphic parameter NumberOfPoints to obtain 25 x 25 points (greater than the standard).

```
ContourFieldMap[helmholtzloop, {Bz},
  Plane[{-30, -30}, {30, 30}], NumberOfPoints → 25];
```

System name: **helmholtzloop**

Bz (G) (Continuous, Automatic Mode)
between {-30., -30.} and {30., 30.}
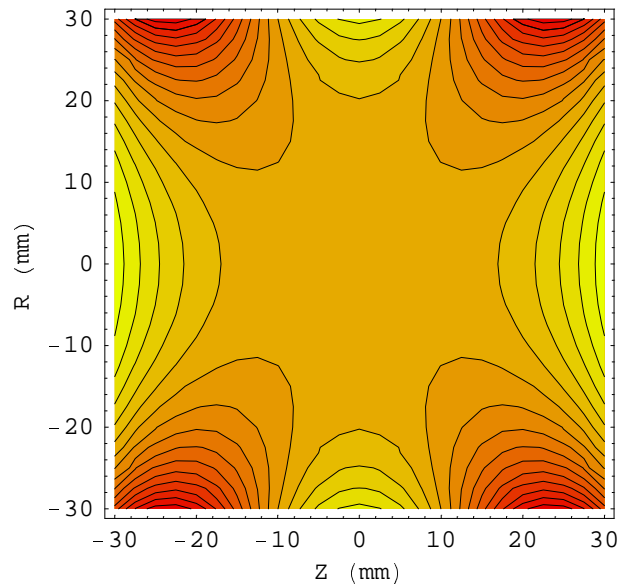
■ Chapter 3: Solenoids and Large Magnetic Sources

■ 3-1 Introduction

The solenoid is the most widely used current-carrying magnetic field generator. By definition, the term Solenoid means a component of circular shape with a winding of rectangular cross section. The cross section can have any shape ranging from a very thin cylindrical winding (current sheet), through a very thick or very long shape to a quasi disk winding. Moreover, in this chapter, it is assumed that the system is axis symmetric, i.e., that all the solenoids are set on the same axis.

The large Magnetic Source component is a design facility that implies a magnetic field made by an external generator whose dimensions are very large, quasi infinite, with respect to the dimensions of the other components of the system. An example is the Earth's magnetic field used in conjunction with a device or component whose dimensions are of the order of a few meters or less.

The 2 following drawings illustrate these cases.

{AXSSolenoid}

{AXSLargeMagneticSource}
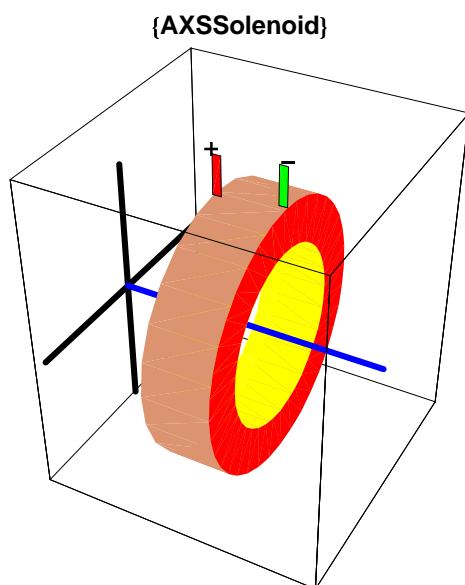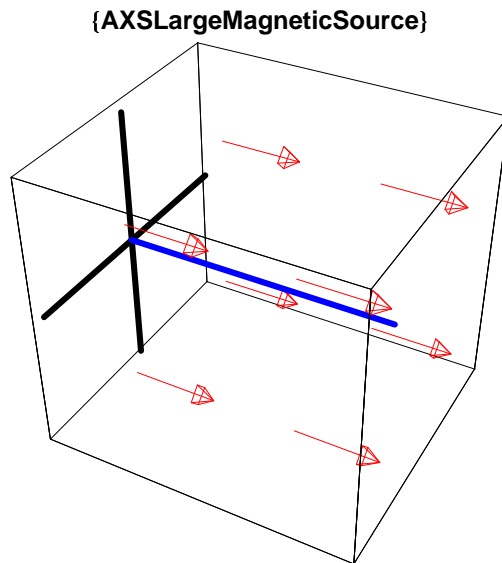
■ 3-2 Immediate Help

■ 3-2-1 The Magnetic System

The basic function for a magnetic system can be complemented with various options. The option SystemLabel is an option that allows adding text to the component list as seen below. The label text is displayed with most of the user's functions.

```
? AXSMagneticSystem

AXSMagneticSystem[{comp1,comp2,...},options]  defines
  a new axis symmetric system made with the specified components.

Options: SystemLabel, MeshingLevel,
  LegendrePointPosition, LegendreRadius, LegendreOrder Details…

? SystemLabel

SystemLabel is an option for AXSMagneticSystem that
  specifies the label to be associated with the system Details…
```

■ 3-2-2 Standard Solenoid Component

AXSSolenoid stands for constant current density winding. There are several ways to finalize the specific data necessary for calculation. The choice is made with the following specific options:

1. using the global winding filling factor: **FillingFactor** option

2. winding made of round wire: **ConductorDiameter** option

3. winding made of rectangular section conductor: **ConductorHeightWidth** option

4. winding made of superconducting wire: no option is necessary since the properties of this type of material include the wire diameter. This is therefore the **default case.**

These options cover a large spectrum of windings ranging from resistive, with all types of materials and conductor shapes and dimensions, to most commercially available superconducting materials.

```
? AXSSolenoid
```

```
AXSSolenoid[material,nominalInnDiameter,nominalOutDiameter,
  nominalLength,position,numberOfTurns,intensity,options]
```

```
builds a Magnetica object that represents a current-carrying winding, constant
  current density cylindrical solenoid, in 2D  cylindrical axis symmetric space.
```

```
Options: FillingFactor, ConductorDiameter, ConductorHeightWidth, Temperature Details…
```

```
? FillingFactor
? ConductorDiameter
? ConductorHeightWidth
```

```
FillingFactor sets the filling factor, and considers
  that the solenoid or the loop is of the general type Details…
```

```
ConductorDiameter considers that the solenoid or the loop given is made of a conductor with
  circular section and sets the conductor diameter to the specified value Details…
```

```
ConductorHeightWidth->{height,width}: considers that the solenoid given is made of a
  conductor with rectangular section and with dimensions height and width Details…
```

■ 3-2-3 Bitter Solenoid Component

This type of solenoid is made of resistive disks. The current density varies with the radius (variation in $1/r$)

```
? AXSBitterSolenoid
```

```
AXSBitterSolenoid[material,nominalInnDiameter,nominalOutDiameter,
  nominalLength,position,numberOfTurns,intensity,diskWidth,option]
```

```
builds a Magnetica object which represents a current-carrying winding, 1/r
  current density cylindrical solenoid, in 2D  cylindrical axis symmetric space
```

```
Option: Temperature Details…
```

■ 3-3 Generic Solenoid, Winding Filling Factor Given

■ 3-3-1 Introduction

The generic solenoid represents a general way of defining a solenoid without entering the specific composition of the winding but only the filling factor.

■ 3-3-2 Entering and Drawing the Data

This can be done either by entering the data through palettes (see appendix B.3) or typing the data as shown below. The capability of Magnetica of adding a system label will be used to give pertinent information.

The following case is a thick solenoid made as follows:

- winding of multiple layers of electrical quality aluminium, *Magnetica* reference **AlumElec**.

- inner diameter: 600 mm,

- outer diameter: 1200 mm,

- length: 300 mm,

- number of turns: 500

- position on the z-axis: 0 mm,

- average filling factor of the winding: 60%.

- The solenoid carries a DC current of 160 amperes (per conductor).

```
solenoid1 = AXSMagneticSystem[
  {AXSSolenoid[AlumElec, 600, 1200, 300, 0, 500, 160, FillingFactor → 0.6]},
  SystemLabel -> "generic solenoid"]

- AXSMagneticSystem :{-solenoid-} -
```

Drawing as a 3D view

```
DrawSystem[solenoid1];
```

System name: **solenoid1**
generic solenoid
3D view ZYX
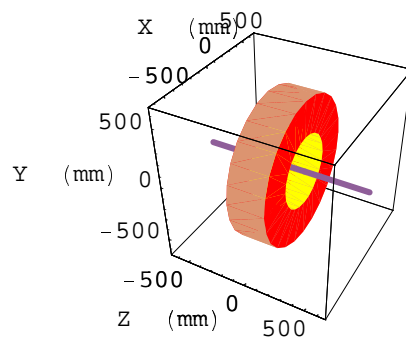
The same in 2D view

```
DrawSystem2D[solenoid1]
```

System name: **solenoid1**
generic solenoid
2D view ZR
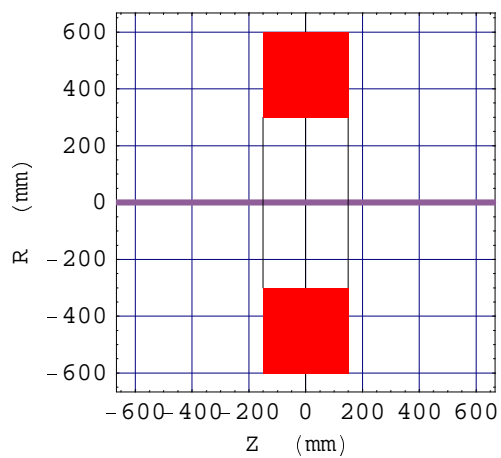
- Graphics -

■ 3-3-3 Field Calculation

The magnetic field is calculated at a target point of coordinates r=100 mm and z=50 mm. using the *Magnetica* **Field-Point** function. This is the standard function to calculate a magnetic field.

The units are the default units Milli Meter and Gauss (see appendix B.2).

The 2 values for the field correspond to the radial and the axial field vector coordinates: Br and Bz.

```
FieldPoint[solenoid1, {100, 50}]
```

```
{38.5438, 1108.12}
```

- The above output accuracy is the standard screen output *Magnetica* accuracy: 6 digits (internal accuracy is still 16 digits).

- The output (i.e. Br, Bz values with the internal accuracy) can be used as argument input for any other *Magnetica* or *Mathematica* function.

- For instance, the screen output accuracy can be set to another number of digits using the **NumberForm** function in front of the FieldPoint function as illustrated below.

```
NumberForm[FieldPoint[solenoid1, {100, 50}], 9]
```

```
{38.5438499, 1108.11815}
```

The next two chapters, 4 and 5, will provide more complete treatment of field and related function calculations.

The function **VerboseFieldPoint** (see appendix B.4) prints detailed and formatted data for field calculation but it cannot be used as input argument for further calculation.

```
VerboseFieldPoint[solenoid1, {100, 50}]
```

```
System name: solenoid1
The magnetic field (in Gauss) at the point {100, 50}(in Milli Meter)
Calculation mode: Automatic Mode
Br         38.543850 G
Bz       1108.118151 G
Bm       1108.788287 G
```

■ 3-4 Round wire winding, thin solenoid

■ 3-4-1 Introduction

Round copper wire of high electrical conductivity is the most common material used to construct a solenoid. The wire diameter is given as an option.

■ 3-4-2 Entering and Drawing the Data

The example is a specific case of a thin solenoid made as follows:

- winding on a marble cylinder of a single layer of copper wire, *Magnetica* reference **CopperOFHC**

- wire diameter: 0.634 mm

- inner diameter: 541.09 mm

- length: 305.51 mm.

- number of turns: 440

- position on the z-axis: 0 mm,

- The solenoid carries a DC current of 200 Milli Amperes.

This solenoid has been used as a standard of inductance as we will see in chapter 10.

case 1: solenoid standing alone

```
solenoid2 = AXSMagneticSystem[
  {AXSSolenoid[CopperOFHC, 541.09,
    542.358, 305.51, 0, 440, 0.2, ConductorDiameter → 0.634]},
  SystemLabel -> "solenoid single layer (self inductance standard)"]
```

```
- AXSMagneticSystem :{-solenoid-} -
```

case 2: solenoid whose axis is oriented in the same direction as the Earth's magnetic field

```
solenoid2earth = AXSMagneticSystem[
  {AXSSolenoid[CopperOFHC, 541.09,
    542.358, 305.51, 0, 440, 0.05, ConductorDiameter → 0.634],
   AXSLargeMagneticSource[0.459]},
  SystemLabel -> "solenoid single layer and Earth's magnetic field"]
```

```
- AXSMagneticSystem :{-solenoid-, -l. source-} -
```

3D drawing case 1

```
DrawSystem3D[solenoid2];
```

```
System name: solenoid2
solenoid single layer (self inductance
standard)
3D view ZYX
```
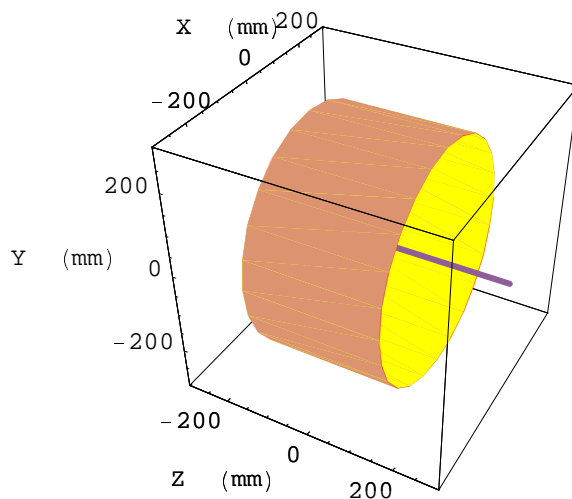

2D drawing case 1

```
DrawSystem2D[solenoid2];
```

System name: **solenoid2**
solenoid single layer (*self inductance*
standard)
2D view ZR
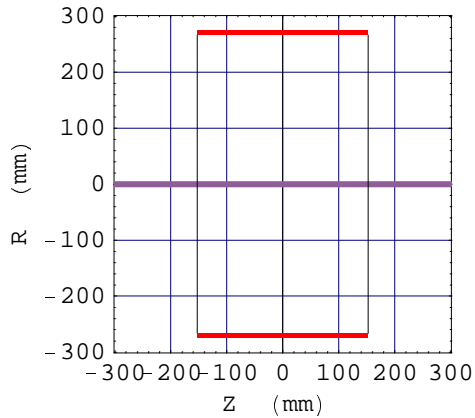
■ 3-4-3 Field Calculation

1. Field calculation case 1. The target point coordinates (100 mm, 50 mm) are the same as in the previous example but the coordinates as well as the accuracy level are now parameterized (screen output accuracy to 9 digits).

```
rc = 100; zc = 50; precis = 9;
NumberForm[FieldPoint[solenoid2, {rc, zc}], precis]
```

{0.110574281, 1.82687316}

```
VerboseFieldPoint[solenoid2, {rc, zc}]
```

System name: **solenoid2**
The magnetic field (in Gauss) at the point {100, 50}(in Milli Meter)
Calculation mode: Automatic Mode
Br   0.110574 G
Bz   1.826873 G
Bm   1.830216 G

2. Field calculation case 2. Same target point coordinates but the field unit is set in Milli Gauss.

```
MagneticaUserUnits[FieldUnit → Milli Gauss]
```

{FieldUnit → Gauss Milli}

```
rc = 100; zc = 50; precis = 9;
NumberForm[FieldPoint[solenoid2earth, {rc, zc}], precis]
```

{27.6435703, 915.71829}

```
VerboseFieldPoint[solenoid2earth,{rc,zc}]

System name: solenoid2earth
The magnetic field (in Milli Gauss) at the point {100, 50}(in Milli Meter)
Calculation mode: Automatic Mode
Br            27.643570 mG
Bz           915.718290 mG
Bm           916.135445 mG
```

■ 3-5 Rectangular ribbon, solenoid

■ 3-5-1 Introduction

This case concerns a large resistive solenoid used as in early type MRI magnet technology.

■ 3-5-2 Entering and Drawing the Data

The example is a specific case of a thick solenoid made as follows:

- winding of multiple layers of high conductivity copper ribbon 2.5mm x 20mm,

- inner diameter: 1170 mm,

- outer diameter: 1530 mm,

- length: 102 mm.

- number of turns: 272

- The solenoid carries a DC current of 149.839 Amperes

- The actual running temperature of the winding is then 45° Celsius.

```
solenoid3 = AXSMagneticSystem[
  {AXSSolenoid[CopperOFHC, 1170, 1530, 102, 0, 272, 149.839,
    ConductorHeightWidth → {2.5, 20},
    Temperature → 45]},
  SystemLabel -> "copper ribbon solenoid"]

- AXSMagneticSystem :{-solenoid-} -
```

2D drawing

```
DrawSystem2D[solenoid3];
```

System name: **solenoid3**
copper ribbon solenoid
2D view ZR
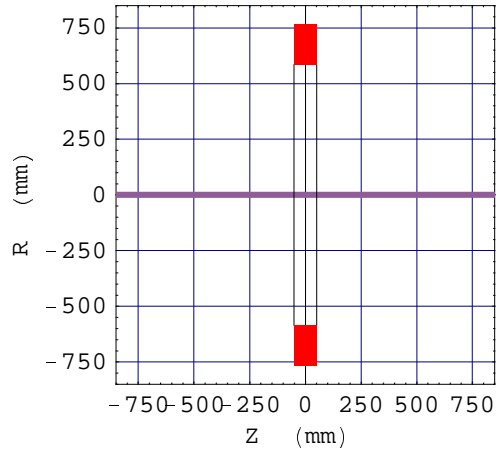
3-5-3 Field calculation

The field calculation is made at the target point coordinates (400 mm, 48 mm) and the field unit is set in Milli Tesla with a screen output precision to 7 digits.

```
MagneticaUserUnits[FieldUnit → Milli Tesla]
rc = 400; zc = 48; precis = 7;
NumberForm[FieldPoint[solenoid3, {rc, zc}], precis]
```

{FieldUnit → Milli Tesla}

{5.568122, 52.22557}

```
VerboseFieldPoint[solenoid3,{rc,zc}]
```

System name: **solenoid3**
The magnetic field (in Milli Tesla) at the point {400, 48}(in Milli Meter)
Calculation mode: Automatic Mode
Br   5.568122 mT
Bz   52.225565 mT
Bm   52.521554 mT

3-6 Superconducting Solenoid

3-6-1 Introduction

This case is a typical superconducting solenoid used in the current MRI magnet technology.

3-6-2 Entering and Drawing the Data

The example is a specific case of a sperconducting solenoid made as follows:

- winding of multiple layers of Niobium Titanium superconductor wire, 1.5 mm in diameter, *Magnetica* reference SuperNbTi5L

- inner diameter: 1000 mm,

- outer diameter: 1050 mm,

- length: 200 mm.

- number of turns: 1000

- position on the z-axis: 0 mm,

- The solenoid carries a DC current of 830.901 Amperes.

- The coil operates at the cryogenic temperature of 4.2 ° Kelvin (-269 ° Celsius) (implicit data)

```
solenoid4 =
 AXSMagneticSystem[{AXSSolenoid[SuperNbTi5, 1000, 1050, 200, 0, 1000, 830.901]},
  SystemLabel -> "Niobium Titanium superconductor solenoid"]

- AXSMagneticSystem :{-solenoid-} -
```

2D drawing

```
DrawSystem2D[solenoid4];
```

System name: **solenoid4**
Niobium Titanium superconductor
solenoid
2D view ZR
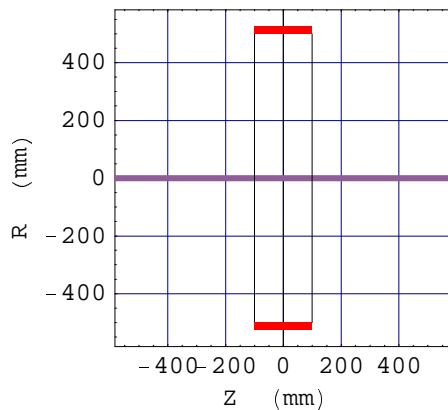
■ 3-6-3 Field Calculation

The field calculation is made at the target point coordinates (250 mm, 15 mm) and the field unit is set in Tesla with a precision to 9 digits.

```
MagneticaUserUnits[FieldUnit → Tesla]
rc = 250; zc = 15; precis = 9;
NumberForm[FieldPoint[solenoid4, {rc, zc}], precis]

{FieldUnit → Tesla}

{0.0305707668, 1.20330744}
```

```
VerboseFieldPoint[solenoid4, {rc, zc}]

System name: solenoid4
The magnetic field (in Tesla) at the point {250, 15}(in Milli Meter)
Calculation mode: Automatic Mode
Br          0.030571 T
Bz          1.203307 T
Bm          1.203696 T
```

■ 3-7 Bitter Solenoid

■ 3-7-1 Introduction

This case is a 15 teslas Bitter type solenoid

■ 3-7-2 Entering and Drawing the Data

The example is a specific case of a Bitter solenoid made as follows:

- winding of multiple disks of high mechanical strength Copper 0.7 mm thick, *Magnetica* reference CopperFHard,

- inner diameter: 60.4 mm,

- outer diameter: 406 mm,

- length: 184 mm.

- number of disks: 150

- position on the z-axis: 0 mm,

- The solenoid carries a DC current of 20 000 Amperes.

- The actual running temperature of the winding is then 40° Celsius.

This coil has been used at the MIT National Magnet laboratory

```
solenoid5 = AXSMagneticSystem[
  {AXSBitterSolenoid[CopperFHard, 60.4, 406, 184, 0, 150, 20000, 0.7,
    Temperature → 40]}, SystemLabel -> "MIT - 15 Teslas BITTER solenoid"]

- AXSMagneticSystem :{-solenoid-} -
```

3D drawing

```
DrawSystem3D[solenoid5]
```

```
          System name: solenoid5
          MIT - 15 Teslas BITTER solenoid
          3D view ZYX
```
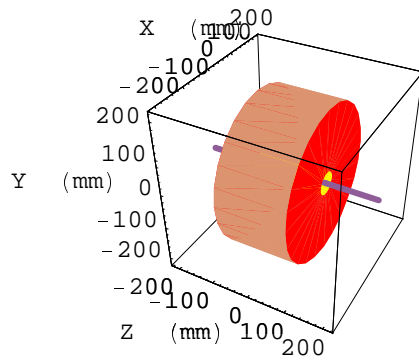
- Graphics3D -

2D drawing

```
DrawSystem2D[solenoid5]
```

```
          System name: solenoid5
          MIT - 15 Teslas BITTER solenoid
          2D view ZR
```
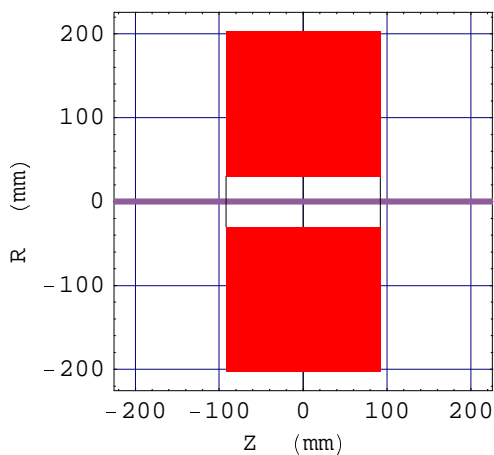
- Graphics -

■ 3-7-3 Field Calculation

The field calculation is made at the target point coordinates (5 mm, 35 mm) and the field unit and precision are unchanged with respect to the previous case. It is not necessary to type these data parameters unless changed.

```
rc = 5; zc = 35;
NumberForm[FieldPoint[solenoid5, {rc, zc}], precis]

{0.0975265866, 14.3472775}
```

VerboseFieldPoint[solenoid5, {rc, zc}]

```
System name: solenoid5
The magnetic field (in Tesla) at the point {5, 35}(in Milli Meter)
Calculation mode: Automatic Mode
Br          0.097527 T
Bz         14.347278 T
Bm         14.347609 T
```

We reset the default units for future use.

MagneticaUserUnits[]

```
{}
```

■ Chapter 4: Magnetic Field Calculation Using Vector Potential

■ 4-1 Fundamentals

■ 4-1-1 *Magnetica* modes for calculations

Magnetica offers 2 different basic calculation modes:

- The vector potential mode (suffix VP): the magnetic field is derived from the vector potential assuming the Lorentz gauge (the arbitrary, space-independent vector potential is null as is its divergence). This leads to Poisson's equation, which is solved for the appropriate magnetic component geometry.

- The scalar potential processed using spherical harmonics (suffix SH): this mathematical approach to field space relations is possible only in space without any magnetic source (i.e. neither electrical current, nor magnetic materials). This approach leads to Laplace's equation, which can be solved using series expansion in spherical harmonics represented by Legendre's polynomials.

We will develop an extended explanation of this last method in Chapter 5: ways, means and applications.
Besides the fact that the SH mode cannot be used in some particular regions of space, the basic VP and SH modes have different and somewhat opposite and complementary properties in terms of calculation speed and precision.

Magnetica provides another practical mode, called Automatic, that automatically switches between the 2 basic modes depending on the position of the target point. The Automatic mode is the global default mode. It is the most efficient mode in terms of speed and accuracy but it can be overridden either by changing the global calculation mode, or by selectively using the suffixes VP / SH or the option CalculationMode.

■ 4-1-2 The Magnetic Functions in *Magnetica*

The outcome of field calculation is the magnetic field itself or any of the associated functions such as magnetic potential, magnetic flux and magnetic gradient.

- For axis symmetric systems, **the magnetic field** is a vector materialized by 2 components: the radial and the axial components. *Magnetica* uses the standard definitions and abbreviations:

  - **Bz** is the axial component

  - **Br** is the radial component.

  Br is positive when its direction is away from the Z axis and negative otherwise. The field modulus **Bm** is the absolute value of the magnetic field. In case of 3D graphics, the vector field components Bx and By could be used.

  $$Bm(r,z) = \sqrt{Br^2 + Bz^2} \ \text{ or } \ Bm(x,y,z) = \sqrt{Bx^2 + By^2 + Bz^2} \tag{1}$$

- **The magnetic vector potential** Pv is a vector whose direction is given by the current that generates the magnetic field. Consequently, in the case of an axis symmetrical cylindrical system, since all the current is azimuthal (turning around the axis), the magnetic potential is also only azimuthal and its value is unique for a given couple (r,z) of coordinates. It can be therefore treated as a scalar

f(r,z), the azimuthal angle being implicit. This is the common practice and it will be used in *Magnetica.*

- **The magnetic flux** Φ is the cumulative (integrated) product of the field in a specific direction multiplied by the area normal to the field. In an axis symmetric system taking a circular area centered on the axis, the only global flux is in the axial direction. For axisymmetic systems a very simple relation exists between the magnetic flux and the magnetic potential:

$$\phi(r, z) = Pv(r, z) * 2\pi * r \tag{2}$$

The calculation of the flux is fundamental to many applications such as inductance calculation.

- **The gradient of the magnetic field** is a tensor which can be expressed by a 2 x 2 matrix in the axis symmetric case.

$$Gr\,(r, z) = \begin{pmatrix} \frac{\partial Bz}{\partial z} & \frac{\partial Bz}{\partial r} \\ \frac{\partial Br}{\partial z} & \frac{\partial Br}{\partial r} \end{pmatrix} \tag{3}$$

The Maxwell equations imply specific relationships between the elements of this matrix.

■ 4-1-3 Space Geometry Considerations for Field Calculations

From the point of view of mathematics, magnetic field calculation is solved with analytical geometry tools within the physical framework of the Maxwell equations. For any axis symmetric magnetic generator, 3 types of situations arise, depending on the position of the target point:

- *target point on the axis:* the field source relationships are simple, analytically defined, but completely specific to this situation. They are easily derived from the Biot and Savart law. The formulas do not apply to any other points in space.

- *target point off the axis but out of the winding:* the field source relationships are quite complex and in most cases, the resolution of the field calculation requires the use of numerical methods. Only the filament loop proceeds from regular mathematical functions : elliptic integrals. The off-axis formulas do not apply to target points on the axis. Pseudo mathematical singularities occur at the border line.

- *target point in the winding:* once we have passed through the threshold of the winding, we are faced with mathemal singularity problems. This applies to any winding topology starting at the infinitely long wire of given radius. Indeed, the Ampere law shows dependence on 1/r outside the wire radius and in r inside, assuming a constant current density. The application of these considerations to any winding shape leads to very elaborate algorithms with only numerical solutions.

The calculation engine of *Magnetica* automatically takes care of all 3 of these situations providing an excellent continuity at the interface of these topological regions. Illustrations of field calculations are shown in the following examples for the 2 basic types of current-carrying circular windings: loop and solenoid.

Magnetica offers considerable versatility in field calculation and graphics field mapping. The examples shown below are not exhaustive. For a complete definition see the Reference Guide.

■ 4-1-4 Filament Loop / Tubular Loop

Magnetica uses 2 different types of modelization of the circular loop: **filament** and **tubular**. They are selected by setting the option **FilamentMode** respectively on True or False. By definition, a filament is a pure theoretical entity seen as a wire of null radius which implies an infinite current density. This situation is physically impossible, and, as a consequence, a filament circular loop cannot materially exist. Nevertheless, most of the books on magnetism deal, implicitly, with the filament loop. For instance the field solution off axis is presented as a linear combination of Elliptic Integrals even if the drawing shows a definite size winding. However, the implicit (and physically incorrect) assumption that the dimension of the filament is null is of little consequence when the target point is distant from the winding particularly if the winding size is small in relation to the loop diameter. Naturally, the differences increase when the target point becomes closer and even more so when it is inside the winding. This last case occurs in internal inductance calculations as we will see in Chapter 10.

Consequently, the accurate calculation of the field at any position implies the use of the exact dimension of the winding as well as current density distribution inside the winding (*Magnetica* assumes for the tubular mode a constant current density). However given the common usage of the filament loop particularly in education, *Magnetica* provides 2 choices of field calculation using the appropriate option FilamentMode→True (default case) or FilamentMode→False.

Examples in the following sections of this chapter will show the cases of a finite radius loop treated either as a filament (implicitly at the center of the winding) or as a constant current density.

■ 4-2 Immediate Help

■ 4-2-1 Immediate Help for Field Point Calculation

```
? MagneticaCalculationMode
```

```
MagneticaCalculationMode[m] sets the global calculation mode to m,
  possible values being VectorPotential, SphericalHarmonics or Automatic.
```

```
MagneticaCalculationMode[] displays the current calculation mode Details…
```

```
? FieldPoint
? FieldPointVP
? FieldPointSH
```

```
FieldPoint[system,p] returns the field value
  at the point p defined by the coordinates, using FieldPointVP or
  FieldPointSH, according to the global field calculation mode Details…
```

```
FieldPointVP[system,p] returns the field value at the point p
  defined by the coordinates, using the vector potential method. Details…
```

```
FieldPointSH[system,p] returns the field value at the point p
  defined by the coordinates, using the spherical harmonics method. Details…
```

? PotentialPoint
? PotentialPointVP
? PotentialPointSH

PotentialPoint[system,p] returns the magnetic potential value
  at the point p defined by the coordinates, using PotentialPointVP or
  PotentialPointSH, according to the global field calculation mode. Details…

PotentialPointVP[system,p] returns the magnetic potential value at the point
  p defined by the coordinates, using the vector potential method. Details…

PotentialPointSH[system,p] returns the magnetic potential value at the point
  pdefined by the coordinates, using the spherical harmonics method. Details…

? FluxArea
? FluxAreaVP
? FluxAreaSH

FluxArea[system,p] returns the flux through the area bordered
  by the circle lying in the plane normal to the z-axis, centered on the
  z-axis and containing the point p. This is done using FluxAreaVP or
  FluxAreaSH according to the global field calculation mode Details…

FluxAreaVP[system,p] uses the vector potential method to determine the
  flux through the area bordered by the circle lying in the plane normal to
  the z-axis, centered on the z-axis and containing the point p. Details…

FluxAreaSH[system,p] uses the spherical harmonics method to determine
  the flux through the area bordered by the circle lying in the plane normal
  to the z-axis, centered on the z-axis and containing the point p. Details…

? GradientPoint
? GradientPointVP
? GradientPointSH

GradientPoint[system,p] returns the field gradient
  at the point p defined by the coordinates, using GradientPointSH
  or GradientPointVP, according to the global field calculation mode.
Output form is: {{GZz,GZr},{GRz,GRr}}. Details…

GradientPointVP[system,p] returns the field gradient at the
  point p defined by the coordinates, using the vector potential method.
Output form is: {{GZz,GZr},{GRz,GRr}}. Details…

GradientPointSH[system,p] returns the field gradient at the
  point p defined by the coordinates, using the spherical harmonics method.
Output form is: {{GZz,GZr},{GRz,GRr}}. Details…

? FieldPointOrigin

FieldPointOrigin[s] is a shortcut for VerboseFieldPoint[s,{0,0,0}] Details…

This is a useful *Mathematica* function:

? NumberForm

NumberForm[expr, n] prints with approximate real numbers in expr given to n-digit precision.
  Plus…

This is the list of the units used in *Magnetica*:

? MagneticaUserUnits

```
MagneticaUserUnits[LengthUnit -> Meter, ...]
   overrides the default Magnetica units with user defined units.
The available units are:

LengthUnit            (for all lengths)
FieldUnit             (for the magnetic field)
PotentialUnit         (for the magnetic potential)
FluxUnit              (for the magnetic flux)
GradientUnit          (for the magnetic field gradient)
InductanceUnit        (for the inductance)
MagneticEnergyUnit    (for the energy)
MagneticMomentUnit    (for the magnetic moment)
CurrentIntensityUnit  (for the current intensity)
PowerUnit             (for the power)
VoltageUnit           (for the current voltage)
ResistanceUnit        (for the electrical resistance). Details…
```

Most of the calculation functions can be displayed with a formatted output by prefixing with the words "Verbose". For instance FieldPoint gives rise to VerboseFieldPoint

? VerboseFieldPoint

```
VerboseFieldPoint[system, target, option]
   displays a detailed and formatted output of FieldPoint. Details…
```

Each Verbose function has the same behavior as its raw version (same syntax and at least the same options)

■ 4-2-2 Immediate Help for Field Map Drawing

? FieldMap

```
FieldMap[system, fieldFunctions, Line[{r1,z1},{r2,z2}], options]
   plots each of the fieldFunctions, for the system given, on the line
   between points p1 and p2. fieldFunctions is an element or a sublist of
   {Bz, Br, Bm, Flux, Potential, GradRR, GradRZ, GradZR, GradZZ}. Details…
FieldMap[system, fieldFunctions, Line[{x1, y1, z1},{x2, y2, z2}], options] plots
   each of the fieldFunctions, for the system given, on the line between points
   p1 and p2. fieldFunctions is an element or a sublist of {Bx, By, Bz, Bm, Flux,
   PotentialX, PotentialY, GradXX, GradYY, GradZX, GradZY, GradXZ, GradYZ, GradZZ}.

FieldMap[system, fieldFunctions, Plane[{r1,z1},{r2,z2}], options] plots
   each of the fieldFunctions, for the system given, on the rectangle defined
   by the two opposite corners {r1,z1} and {r2, z2}. fieldFunctions is an
   element or a sublist of {Bz, Br, Bm, Flux, Potential, GradRR, GradRZ, GradZR,
   GradZZ}. p1 and p2 must define a plane parallel to one of the canonical planes

FieldMap[system, fieldFunctions, Plane[{x1,y1,z1},{x2,y2,z2}], options] plots each of the
   fieldFunctions, for the system given, on the rectangle defined by the two opposite corners
   {x1,y1,z1} and {x2,y2,z2}. fieldFunctions is an element or a sublist of {Bx, By, Bz, Bm,
   Flux, PotentialX, PotentialY, GradXX, GradYY, GradZX, GradZY, GradXZ, GradYZ, GradZZ}.
   {x1,y1,z1} and {x2,y2,z2} must define a plane parallel to one of the canonical planes

options: NumberOfPoints, CalculationMode, PlotScale
```

? FieldMapPoints

```
FieldMapPoints[system, fieldFunction Line[p1,p2], options] or FieldMapPoints[
   system, fieldFunctions, Plane[p1,p2], options] (see FieldMap for details) plots
   the field functions asked for, using a linear distribution of points. Details…
```

? FieldMapContinuous

FieldMapContinuous[system, fieldFunctions, Line[p1, p2], options]
  or FieldMapContinuous[system, fieldFunctions, Plane[a, b], options] (see
  FieldMap for details) plots the field functions asked for, using Mathematica
  internal algorithm to determine the distribution of points. Details…

? NumberOfPoints

NumberOfPoints is an option for drawing
  functions that specifies how many points to use. Details…

? CalculationMode

CalculationMode is an option for field drawing, and field calculations
  functions that allows changing the field computation mode locally.
    Possible values: VectorPotential, SphericalHarmonics, Automatic Details…

? PlotScale

PlotScale is an option for FieldMap and other field drawing functions.
    Possible values: LinearMap, LinearLogMap, LogLinearMap LogLogMap Details…

Options[FieldMap]

{PlotScale → LinearMap, CalculationMode → Inherited, NumberOfPoints → Automatic}

? ContourFieldMap

ContourFieldMap[system, fieldFunctions, Plane[p1,p2] , options] displays the
  contour of each fieldFunctions, for the system given, on the rectangle defined
  by the two opposite corners p1 and p2. fieldFunctions is an element or a sublist
  of {Bx, By, Bz, Br, Bm, Flux, Potential, GradRR, GradRZ, GradZR, GradZZ}. p1
  and p2 must define a plane parallel to one of the canonical planes Details…

options: NumberOfPoints, CalculationMode

? VectorFieldMap

VectorFieldMap[system, Plane[p1,p2], options] displays the magnetic field of the
  system given, within the rectangle defined by the two opposite corners p1 and
  p2. These must define a plane parallel to one of the canonical planes Details…
VectorFieldMap[system, Box[p1,p2], options] displays the magnetic field of the
  system given, within the cubic space defined by two opposite corners p1 and p2.

options: NumberOfPoints, CalculationMode,

? DrawIsoFieldLines

DrawIsoFieldLines[system,fieldValues,options]
Displays the magnetic system with its iso field
  lines corresponding to the values given in the list fieldValues.
Suffixing DrawIsoFieldLines with XY,XZ,YX,YZ,ZX,ZY,RZ, or ZR, shows the chosen point of view.

options: NumberOfPoints, AngularRange, CalculationMode Details…

■ 4-3 Punctual Field Calculation - Syntax

■ 4-3-1 Input Syntax for Punctual Field Calculation

The functions provided for accessing punctual information are:

- **FieldPoint[***system, targetPoint***]**: for magnetic field

- **PotentialPoint[***system, targetPoint***]**: for vector potential.

- **FluxArea[***system, targetPoint***]**: for magnetic flux

- **GradientPoint[***system, targetPoint***]**: for the gradient of the magnetic field

For each of these functions, the arguments are the following:

- the magnetic system, designated by with its symbol name.

- the spatial coordinates of the target point

The coordinates are specified as a list of one of the following forms:

- {r, z}: the first for the radial (r) coordinate and the second for axial (z) coordinate.

- {x, y, z}: the usual Cartesian specification.

■ 4-3-2 Output Precision and Formatting

The normal displayed output of these functions is to 6 digits.
But the internal precision standard is to 16 digits.

The output can easily be used as argument for other *Magnetica* or *Mathematica* functions. One can use this capability to display the output to a greater or smaller number of digits using the *Mathematica* function **NumberForm**.

Moreover, for most of calculation functions, *Magnetica* provides a Verbose version that displays a formatted output with detailed information (units, calculation mode,...), but whose output cannot be used for further calculation. These functions are obtained by simply prefixing the word "Verbose" at the beginning of the original functions: **VerboseField-Point**, **VerbosePotentialPoint**, **VerboseFluxArea**, **VerboseGradientPoint**.

■ 4-3-3 Switching to Other Calculation Modes

The default setting for calculation mode is Automatic.

The calculation mode can be controlled in one of the following ways:

- The suffix **VP** or **SH** can be appended to any of the preceding function to direct the selective use of a specific basic mode

- The option **CalculationMode** can be used by giving to it one of the permitted values:
  VectorPotential, **SphericalHarmonics** or **Automatic**. (this will also work for drawing functions)

- The global setting of the calculation mode can be modified by using the function

  MagneticaCalculationMode[*mode***]**, mode being one of the previous values.

Differences between modes and details about meaning and related functions will be given in Chapter 5.

■ 4-3-4 Unit Behavior

Without any local specification, input and output are expressed in the global current units. These units can be consulted with the function **MagneticaCurrentUnits[]**

```
MagneticaCurrentUnits[]
```

```
Length                    Milli Meter
Magnetic field            Gauss
Magnetic potential        Micro Weber/Meter
Magnetic flux             Micro Weber
Magnetic field gradient   Gauss/Meter
Inductance                Micro Henry
Magnetic energy           Joule
Magnetic moment           Micro Ampere Meter²
Current intensity         Ampere
Power                     Watt
Voltage                   Volt
Electrical Resistance     Ohm
```

One can globally change some current units by using the command: **MagneticaUserUnits[***unitsSpecifications***]**

Here is an example:

```
MagneticaUserUnits[LengthUnit → Meter, FieldUnit → Micro Tesla]
```

```
{LengthUnit → Meter, FieldUnit → Micro Tesla}
```

From now on, all lengths will be understood in meters, and magnetic fields (output) in micro tesla. The other units are set at the default settings.

Using MagneticaUserUnits without any argument resets all the units to their default settings

```
MagneticaUserUnits[];
```

One can also modify only locally some units, by adding the specification as an additionnal option to the concerned function.

■ 4-4 Examples of Field Calculation with Circular Loops

■ 4-4-1 Building Systems for the Examples

The following loop will be used for all basic demonstrations on computation functions

```
loopFilament = AXSMagneticSystem[
    {AXSCircularLoop[CopperOFHC, 100, 30, 0, 20, 0.5, FillingFactor → 0.7]}];
```

```
DrawSystem[loopFilament];
```

System name: **loopFilament**

3D view ZYX


```
DrawSystem2D[loopFilament];
```
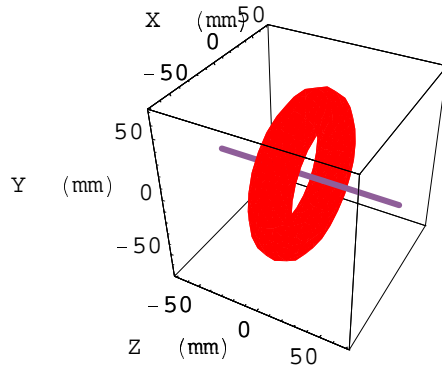
System name: **loopFilament**

2D view ZR
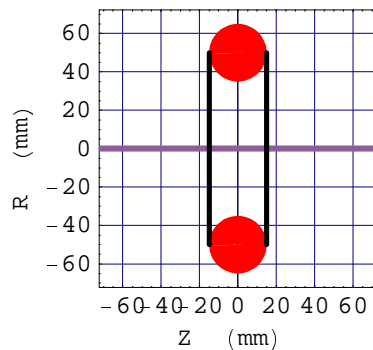
This loop will be used to show the difference between a filament loop and a tubular loop

```
loopTubular = AXSMagneticSystem[{AXSCircularLoop[CopperOFHC,
    100, 30, 0, 20, 0.5, FillingFactor → 0.7, FilamentMode → False]}];
```

■ 4-4-2 Basic Calculations

Field at the isocenter point (R=0, Z=0) for the filament loop and the tubular loop. The two modes give close values.

```
rc = 0; zc = 0;
FieldPoint[loopFilament, {rc, zc}]
FieldPoint[loopTubular, {rc, zc}]
```

{0, 1.25664}

{0, 1.24234}

Same target point but output with a greater precision (12 digits) using NumberForm:

```
rc = 0; zc = 0; precis = 12;
NumberForm[FieldPoint[loopFilament, {rc, zc}], precis]
NumberForm[FieldPoint[loopTubular, {rc, zc}], precis]
```

```
{0, 1.25663706144}
```

```
{0, 1.24233619336}
```

Target point near the winding:

```
rc = 34; zc = 5; precis = 12;
NumberForm[FieldPoint[loopFilament, {rc, zc}], precis]
NumberForm[FieldPoint[loopTubular, {rc, zc}], precis]
```

```
{0.375281174675, 1.88938007184}
```

```
{0.350238376999, 1.84736497195}
```

Let us use the Verbose version of FieldPoint to display more information.

```
VerboseFieldPoint[loopFilament, {rc, zc}]
VerboseFieldPoint[loopTubular, {rc, zc}]
```

```
System name: loopFilament
The magnetic field (in Gauss) at the point {34, 5}(in Milli Meter)
Calculation mode: Automatic Mode
Br          0.375281 G
Bz          1.889380 G
Bm          1.926290 G
```

```
System name: loopTubular
The magnetic field (in Gauss) at the point {34, 5}(in Milli Meter)
Calculation mode: Automatic Mode
Br          0.350238 G
Bz          1.847365 G
Bm          1.880272 G
```

■ 4-4-3 Modifying Input and Output Units

The existing current units:

```
MagneticaCurrentUnits[]
```

```
Length                    Milli Meter
Magnetic field            Gauss
Magnetic potential        Micro Weber/Meter
Magnetic flux             Micro Weber
Magnetic field gradient   Gauss/Meter
Inductance                Micro Henry
Magnetic energy           Joule
Magnetic moment           Micro Ampere Meter²
Current intensity         Ampere
Power                     Watt
Voltage                   Volt
Electrical Resistance     Ohm
```

Magnetic field set in Micro Tesla (warning: there must be a space between the prefix and the unit)

Warning: from now on, the field unit will always be in Micro Tesla, until explicitly changed.

```
MagneticaUserUnits[FieldUnit → Micro Tesla];
NumberForm[FieldPoint[loopFilament, {rc, zc}], precis]
NumberForm[FieldPoint[loopTubular, {rc, zc}], precis]
```

{37.5281174675, 188.938007184}

{35.0238376999, 184.736497195}

Verbose version to get a better insight:

```
VerboseFieldPoint[loopFilament, {rc, zc}]
VerboseFieldPoint[loopTubular, {rc, zc}]
```

```
System name: loopFilament
The magnetic field (in Micro Tesla) at the point {34, 5}(in Milli Meter)
Calculation mode: Automatic Mode
Br          37.528117 μT
Bz         188.938007 μT
Bm         192.628996 μT


System name: loopTubular
The magnetic field (in Micro Tesla) at the point {34, 5}(in Milli Meter)
Calculation mode: Automatic Mode
Br          35.023838 μT
Bz         184.736497 μT
Bm         188.027239 μT
```

Changing coordinate units (and all other length arguments) into Inches and also the field unit.

Warning: from now on, all lengths will be in Inches, and all fields in Milli Gauss, until explicitly changed

```
rc = 1; zc = 0.1;
MagneticaUserUnits[FieldUnit → Milli Gauss, LengthUnit → Inch];
NumberForm[FieldPoint[loopFilament, {rc, zc}], precis]
NumberForm[FieldPoint[loopTubular, {rc, zc}], precis]
```

{84.29468977, 1564.46423157}

{80.1301301162, 1537.48792788}

```
VerboseFieldPoint[loopFilament, {rc, zc}]
VerboseFieldPoint[loopTubular, {rc, zc}]
```

```
System name: loopFilament
The magnetic field (in Milli Gauss) at the point {1, 0.1}(in Inch)
Calculation mode: Automatic Mode
Br           84.294690 mG
Bz         1564.464232 mG
Bm         1566.733521 mG


System name: loopTubular
The magnetic field (in Milli Gauss) at the point {1, 0.1}(in Inch)
Calculation mode: Automatic Mode
Br           80.130130 mG
Bz         1537.487928 mG
Bm         1539.574606 mG
```

Resetting all the units to their default settings

```
MagneticaUserUnits[];
```

■ **4-4-4 Potential, Flux, Gradient**

Punctual calculation of the magnetic vector potential, the magnetic flux, the gradient, at a target point off the axis (R=20mm, Z=10mm) for the filament loop and the tubular loop. Using `MatrixForm`, the gradient can be shown as a matrix.

Default units are:
- Micro Weber / meter for the magnetic potential
- Micro Weber for the magnetic flux
- Gauss / meter for the magnetic field gradient

```
rc = 20; zc = 10;
PotentialPoint[loopFilament, {rc, zc}]
PotentialPoint[loopTubular, {rc, zc}]
FluxArea[loopFilament, {rc, zc}]
FluxArea[loopTubular, {rc, zc}]
GradientPoint[loopFilament, {rc, zc}]
GradientPoint[loopTubular, {rc, zc}]
MatrixForm[GradientPoint[loopFilament, {rc, zc}]]
MatrixForm[GradientPoint[loopTubular, {rc, zc}]]
```

{1.24333}

{1.22985}

{0.156241}

{0.154547}

{{-23.4458, 12.9928}, {12.9928, 14.4071}}

{{-22.436, 12.6559}, {12.6559, 13.7413}}

$$\begin{pmatrix} -23.4458 & 12.9928 \\ 12.9928 & 14.4071 \end{pmatrix}$$

$$\begin{pmatrix} -22.436 & 12.6559 \\ 12.6559 & 13.7413 \end{pmatrix}$$

Each of the previous functions has a "Verbose" version:

```
VerbosePotentialPoint[loopFilament, {rc, zc}]
VerbosePotentialPoint[loopTubular, {rc, zc}]
```

```
System name: loopFilament
The vector potential  (in Micro Weber/Meter) at the point {20, 10}(in Milli Meter)
Calculation mode: Automatic Mode
Potential              1.243329 μWb/m

System name: loopTubular
The vector potential  (in Micro Weber/Meter) at the point {20, 10}(in Milli Meter)
Calculation mode: Automatic Mode
Potential              1.229845 μWb/m
```

```
VerboseFluxArea[loopFilament, {rc, zc}]
VerboseFluxArea[loopTubular, {rc, zc}]
```

```
System name: loopFilament
The FluxArea (in Micro Weber) at the point {20, 10}(in Milli Meter)
Calculation mode: Automatic Mode
Flux            0.156241 µWb
```

```
System name: loopTubular
The FluxArea (in Micro Weber) at the point {20, 10}(in Milli Meter)
Calculation mode: Automatic Mode
Flux            0.154547 µWb
```

```
VerboseGradientPoint[loopFilament, {rc, zc}]
VerboseGradientPoint[loopTubular, {rc, zc}]
```

```
System name: loopFilament
The gradient of the field (in Gauss/Meter) at the point {20, 10}(in Milli Meter)
Calculation mode: Automatic Mode
∂Bz/∂z          -23.445806 G/m
∂Bz/∂r           12.992833 G/m
∂Br/∂z           12.992833 G/m
∂Br/∂r           14.407110 G/m
```

```
System name: loopTubular
The gradient of the field (in Gauss/Meter) at the point {20, 10}(in Milli Meter)
Calculation mode: Automatic Mode
∂Bz/∂z          -22.435978 G/m
∂Bz/∂r           12.655928 G/m
∂Br/∂z           12.655928 G/m
∂Br/∂r           13.741288 G/m
```

■ 4-5 Examples of Field Calculations with Solenoids

■ 4-5-1 Entering and Drawing the Data

This is a MRI magnet of 1.5 Teslas made of 3 pairs of superconducting solenoids.

```
unshieldedMRI =
AXSMagneticSystem[
    {AXSSolenoid[SuperNbTi4, 1240, 1280, 300, 180.05374474797077, 635, 735.2202],
     AXSSolenoid[SuperNbTi4, 1240, 1280, 300, -180.05374474797077, 635, 735.2202],
     AXSSolenoid[SuperNbTi4, 1240, 1280, 250, 550.1194910091477, 598, 735.2202],
     AXSSolenoid[SuperNbTi4, 1240, 1280, 250, -550.1194910091477, 598, 735.2202],
     AXSSolenoid[SuperNbTi4, 1240, 1320, 250, 874.8480174899299, 1015, 735.2202],
     AXSSolenoid[SuperNbTi4, 1240, 1320, 250, -874.8480174899299, 1015, 735.2202]},
    SystemLabel → "unshielded MRI 1.5 T superconducting magnet"];
```

3D drawing

DrawSystem[unshieldedMRI]

System name: **unshieldedMRI**

unshielded MRI 1.5 T superconducting
magnet

3D view ZYX
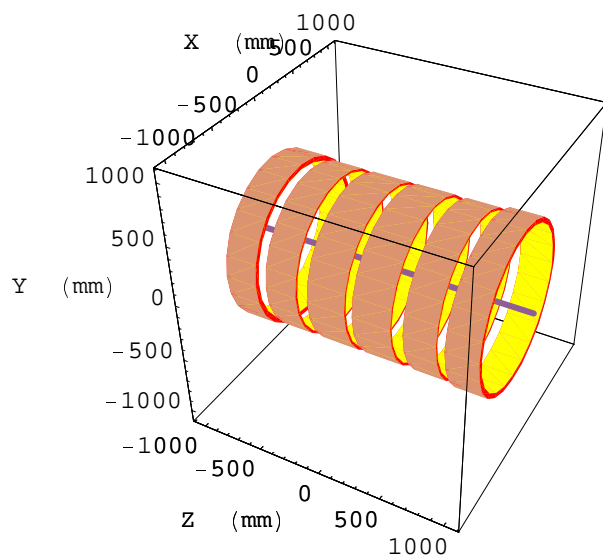
- Graphics3D -

DrawSystem2D[unshieldedMRI];

System name: **unshieldedMRI**

unshielded MRI 1.5 T superconducting
magnet

2D view ZR
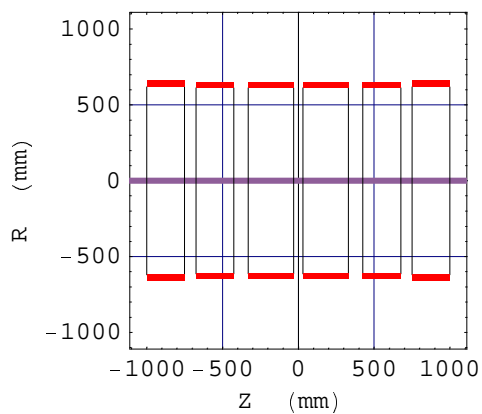
■ 4-5-2 Calculations with Solenoids

Resetting the units to the default settings

MagneticaUserUnits[];

The FieldPointOrigin function is a shortcut for VerboseFieldPoint at the isocenter point (0,0)

FieldPointOrigin[unshieldedMRI]

```
System name: unshieldedMRI
The magnetic field (in Gauss) at the origin
Calculation mode: Automatic Mode
Br          0.000000 G
Bz      15000.000068 G
Bm      15000.000068 G
```

Field on the axis with z = 250mm

NumberForm[FieldPoint[unshieldedMRI, {0, 250}], 9]

{0, 15000.7816}

VerboseFieldPoint[unshieldedMRI, {0, 250}]

```
System name: unshieldedMRI
The magnetic field (in Gauss) at the point {0, 250}(in Milli Meter)
Calculation mode: Automatic Mode
Br          0.000000 G
Bz      15000.781583 G
Bm      15000.781583 G
```

Field off the axis: lengths in inches (Inch) and field in Telsa:

rc=10;zc=8;
MagneticaUserUnits[FieldUnit→Tesla,LengthUnit→Inch];
NumberForm[FieldPoint[unshieldedMRI,{rc,zc}],9]

{0.000413420999, 1.49968351}

VerboseFieldPoint[unshieldedMRI, {rc, zc}]

```
System name: unshieldedMRI
The magnetic field (in Tesla) at the point {10, 8}(in Inch)
Calculation mode: Automatic Mode
Br          0.000413 T
Bz          1.499684 T
Bm          1.499684 T
```

Potential and Flux off the axis (R=8 In, Z=4 In), Potential in Milli Weber/Meter, Flux in Milli Weber:

rc=8;zc=4;
MagneticaUserUnits[PotentialUnit-> Milli Weber/Meter,FluxUnit→Milli Weber];
NumberForm[PotentialPoint[unshieldedMRI,{rc,zc}],9]
NumberForm[FluxArea[unshieldedMRI,{rc,zc}],9]

{152.401168}

{194.577163}

```
VerbosePotentialPoint[unshieldedMRI,{rc,zc}]
VerboseFluxArea[unshieldedMRI,{rc,zc}]

System name: unshieldedMRI
The vector potential  (in Milli Weber/Meter) at the point {8, 4}(in Inch)
Calculation mode: Automatic Mode
Potential        152.401168 mWb/m


System name: unshieldedMRI
The FluxArea (in Milli Weber) at the point {8, 4}(in Inch)
Calculation mode: Automatic Mode
Flux         194.577163 mWb
```

■ 4-6 Field Map Graphics of Magnetic Field functions

■ 4-6-1 Input Syntax

The results of field calculation can be displayed as 2 dimensional (2D) or 3 dimensional (3D) field maps using the graphics capabilities of *Mathematica.*
Here are the different kind of plots that *Magnetica* can display:

▪ **FieldMap[**_system, fieldFunctions, spacePlot, options_**]**: allows plotting each basic field function on a given line or a given plane.
▪ **ContourFieldMap[**_system, fieldFunctions, spacePlot, options_**]**: gives contour graphics of each basic field function on a given plane.
▪ **VectorFieldMap[**_system, spacePlot, options_**]**: draws a view of the magnetic field $\vec{B}$ as a vector field.

Field mapping proceeds from punctual field calculation at several points. The distribution of points can be done through a *Mathematica* internal algorithm that provides the best possible accurate display; this is done with the functions `FieldMapContinuous` or `FieldMap` (and the same for the other functions). But the distribution can be done as a set of points regularly spaced between the limits of the display; this option can be chosen by calling `FieldMap` `Points` instead of `FieldMap`. In some cases, the use of `FieldMapPoints` is the most appropriate, particularly for lengthy calculations.

The next sections explain what field functions are possible, and how to describe the space plot.

■ 4-6-2 Directives for Space Plot

It is necessary, for field map functions, to describe the space where the plot must be done. Some plots can be done on a line, or a plane (in fact, a rectangle), and some others on a 3D box, using the following directives:

- **Line[**_{r1, z1},{r2, z2}_**]** or **Line[**_{x1, y1, z1},{x2, y2, z2}_**]**

- **Plane[**_{r1, z1},{r2, z2}_**]** or **Plane[**_{x1, y1, z1},{x2, y2, z2}_**]**

- **Box[**_{x1, y1, z1},{x2, y2, z2}_**]**

Even with axisymmetric systems, graphics can be obtained not only in abstract 2D coordinate space defined by the coordinates {r,z}, but also in the projection of the coordinates {r,z} on the Cartesian 3D coordinate system, by identifying the coordinate r with the coordinate x (or another).

■ 4-6-3 Magnetic Field Functions

It is also necessary, to indicate the magnetic field functions that are to be drawn. The set of possible functions is different if the space description is given in 2D coordinates {r,z}, or in 3D coordinates {x,y,z}:

- With a 2D space description, the possible functions are:

  Br, Bz, Bm, Flux, Potential, GradZZ, GradRZ, GradZR, GradRR.

- With a 3D space description, the possible functions are:

  Bx, By, Bz, Bm, Flux, PotentialX, PotentialY, GradZZ

N.B. only GradZZ is implemented on the 3D Graphics

■ 4-6-4 Iso Field Lines

It is also possible to have a drawing of lines where modulus of $\vec{B}$ is equal to given constant values:

DrawIsoFieldLines[*system, {fieldValues}, options***]**

gives the polar plot of the Bm around the system isocenter. It displays one line for each field value.

■ 4-7 Examples of Field Map Graphics

■ 4-7-1 Loop: 2D mapping

1. Bz 2D field map on the axis after resetting the units

   ```
   MagneticaUserUnits[];
   ```

   ```
   FieldMap[loopFilament,{Bz},Line[{0,-100},{0,100}]];
   ```

   System name: **loopFilament**

   Bz (G) (Continuous, Automatic Mode)
   between {0., -100.} and {0., 100.}
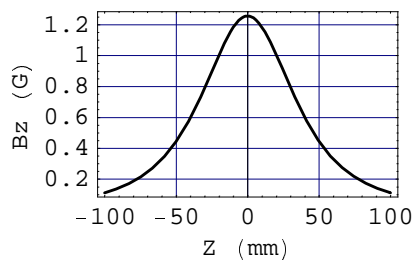
2. Bz, Br, Bm 2D field map for the tubular loop on a radial line at z=20 mm from r=0 to r=100 mm, FieldUnit is expressed in Milli Gauss (for this plot only)

   ```
   FieldMap[loopTubular,{Bz,Br,Bm},Line[{0,20},{100,20}],FieldUnit->Milli Gauss];
   ```

```
System name: loopTubular

Bz (mG) (Continuous, Automatic
Mode)
between {0., 20.} and {100., 20.}
```
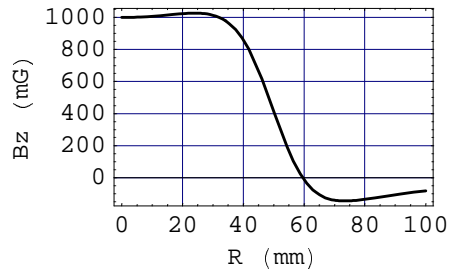


```
System name: loopTubular

Br (mG) (Continuous, Automatic
Mode)
between {0., 20.} and {100., 20.}
```
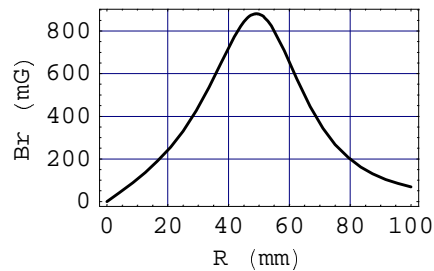


```
System name: loopTubular

Bm (mG) (Continuous, Automatic
Mode)
between {0., 20.} and {100., 20.}
```
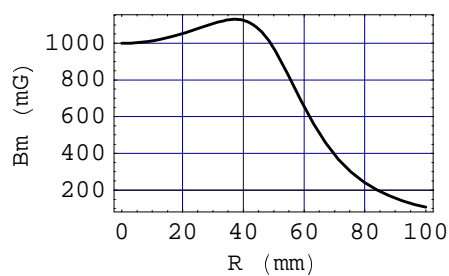

3. Potential, Flux 2D field map for the filament loop on an axial line at r=20 mm, from z=0 to r=50 mm, Flux Unit is in Maxwell i.e. CGS emu unit (Gauss $*$ cm^2)(for this plot only)

    FieldMap[loopFilament,{Potential,Flux},Line[{20,0},{20,50}],FluxUnit->Maxwell];

```
System name: loopFilament
```

```
Potential (μWb/m) (Continuous,
Automatic Mode)
between {20., 0.} and {20., 50.}
```
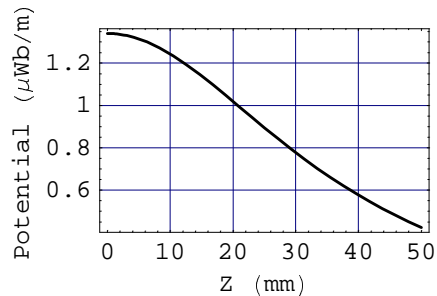


```
System name: loopFilament
```

```
Flux (Mx) (Continuous, Automatic
Mode)
between {20., 0.} and {20., 50.}
```
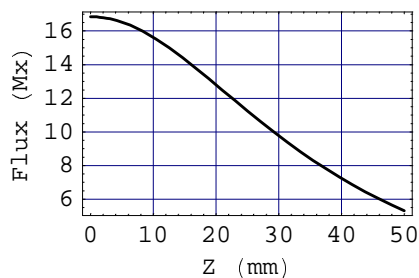

4. The 4 components of the Gradient (i.e. the tensor) map for the tubular loop on an axial line at r=20 mm, from z=-40 to z=40 mm.

N.B. the cross gradient elements ∂Bz/∂r and ∂Br/∂z are identical

```
FieldMap[loopTubular,{GradZZ,GradZR,GradRZ,GradRR},Line[{20,-40},{20,40}]];
```

```
System name: loopTubular
```

```
GradZZ (G/m) (Continuous, Automatic
Mode)
between {20., -40.} and {20., 40.}
```
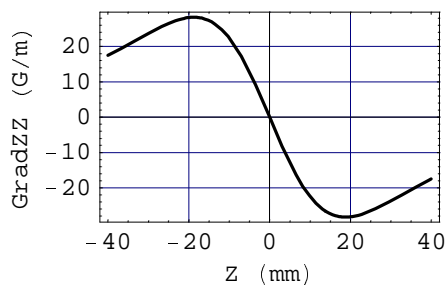
System name: **loopTubular**

GradZR (G/m) (Continuous, Automatic
Mode)
between {20., -40.} and {20., 40.}
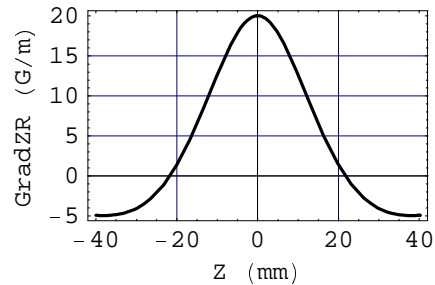
System name: **loopTubular**

GradRZ (G/m) (Continuous, Automatic
Mode)
between {20., -40.} and {20., 40.}
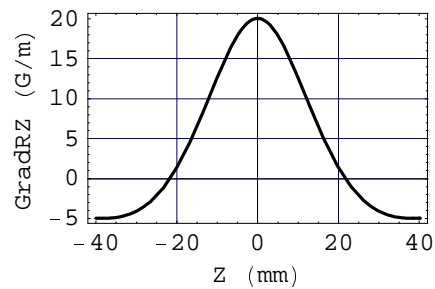
System name: **loopTubular**

GradRR (G/m) (Continuous, Automatic
Mode)
between {20., -40.} and {20., 40.}
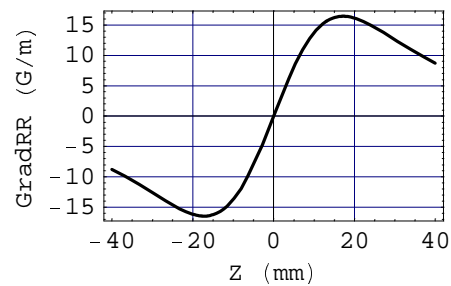
■ 4-7-2 Solenoid: 2D Mapping

This is to reset the units to their default settings

```
MagneticaUserUnits[];
```

1. Bz 2D field map on the axis

```
FieldMap[unshieldedMRI,{Bz},Line[{0,-800},{0,800}]];
```

System name: **unshieldedMRI**

unshielded MRI 1.5 T superconducting
magnet

Bz (G) (Continuous, Automatic Mode)
between {0., -800.} and {0., 800.}
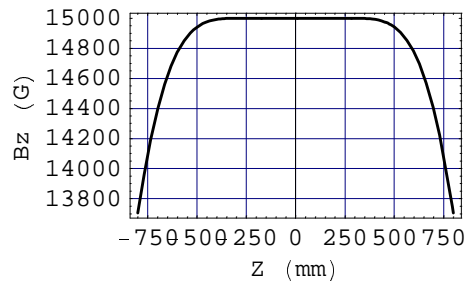
2. Bz, Br, Bm 2D field map for the MRI solenoid on a radial line at z=20 mm, from r=0 to r=100 mm, FieldUnit is in Milli Gauss (for this plot only)

```
FieldMap[unshieldedMRI,{Bz,Br,Bm},Line[{0,20},{750,20}],FieldUnit->Milli Tesla];
```

System name: **unshieldedMRI**

unshielded MRI 1.5 T superconducting
magnet

Bz (mT) (Continuous, Automatic
Mode)
between {0., 20.} and {750., 20.}
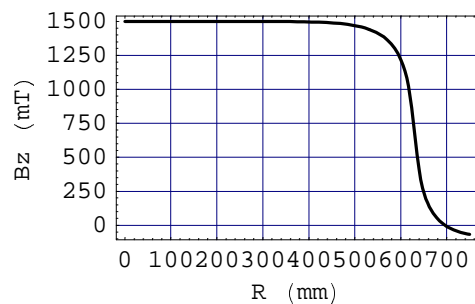
System name: **unshieldedMRI**

unshielded MRI 1.5 T superconducting
magnet

Br (mT) (Continuous, Automatic
Mode)
between {0., 20.} and {750., 20.}
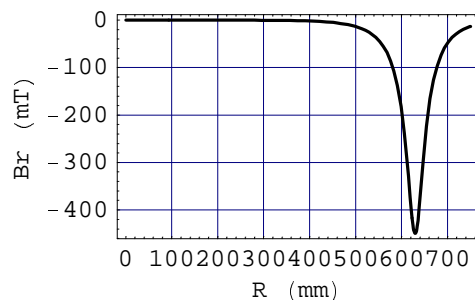
```
System name: unshieldedMRI
unshielded MRI 1.5 T superconducting
magnet
Bm (mT) (Continuous, Automatic
Mode)
between {0., 20.} and {750., 20.}
```
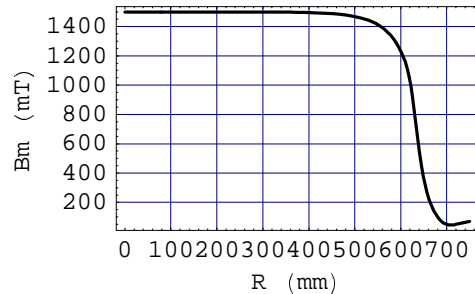

3. Potential, Flux 2D field map for the MRI solenoid on an axial line at r=200 mm, from z=-500 to z=500 mm, FluxUnit is in Maxwell i.e. CGS emu unit (Gauss $*\mathrm{cm}^2$)(for this plot only)

FieldMap[unshieldedMRI,{Potential,Flux},Line[{200,-500},{200,500}],FluxUnit->Milli Weber];

```
System name: unshieldedMRI
unshielded MRI 1.5 T superconducting
magnet
Potential (μWb/m) (Continuous,
Automatic Mode)
between {200., -500.} and {200., 500.}
```
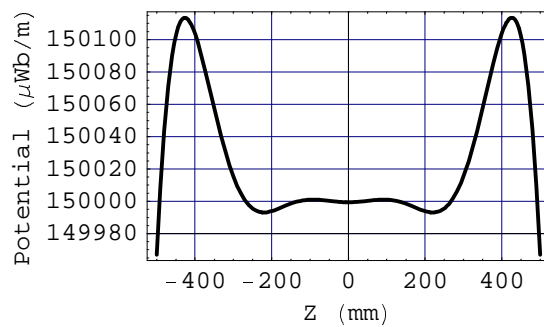
```
System name: unshieldedMRI
unshielded MRI 1.5 T superconducting
magnet
Flux (mWb) (Continuous, Automatic
Mode)
between {200., -500.} and {200., 500.}
```
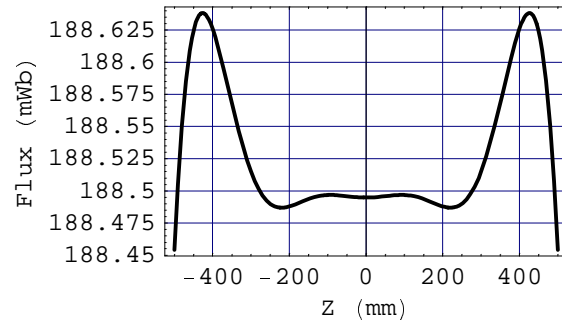

4. The 4 components of the Gradient (i.e. the tensor) map for the MRI solenoid on an axial line at r=200 mm, from z=-400 to z=400 mm.

N.B. the cross gradient elements $\partial Bz/\partial r$ and $\partial Br/\partial z$ are identical

```
FieldMap[unshieldedMRI,{GradZZ,GradZR,GradRZ,GradRR},Line[{200,-400},{200,400}]];
```

```
System name: unshieldedMRI
unshielded MRI 1.5 T superconducting
magnet
GradZZ (G/m) (Continuous, Automatic
Mode)
between {200., -400.} and {200., 400.}
```
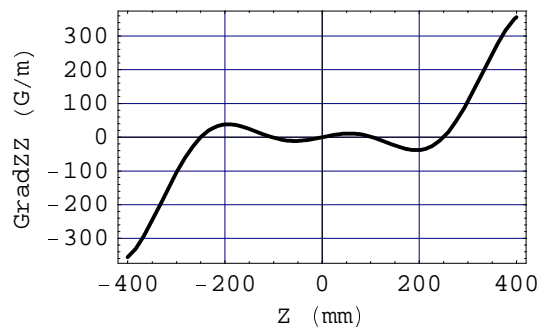
System name: **unshieldedMRI**
unshielded MRI 1.5 T superconducting magnet
GradZR (G/m) (Continuous, Automatic Mode)
between {200., -400.} and {200., 400.}
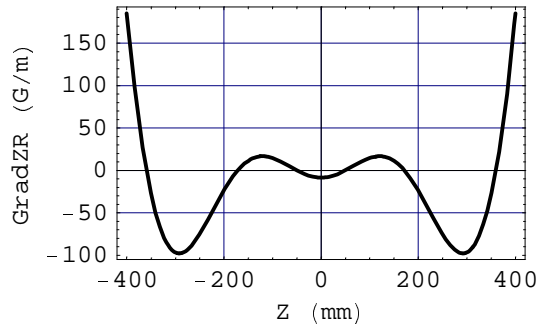
System name: **unshieldedMRI**
unshielded MRI 1.5 T superconducting magnet
GradRZ (G/m) (Continuous, Automatic Mode)
between {200., -400.} and {200., 400.}
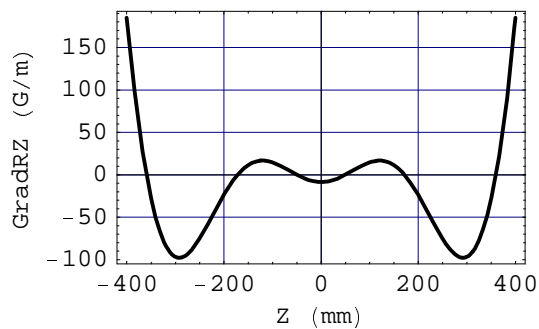
System name: **unshieldedMRI**
unshielded MRI 1.5 T superconducting magnet
GradRR (G/m) (Continuous, Automatic Mode)
between {200., -400.} and {200., 400.}
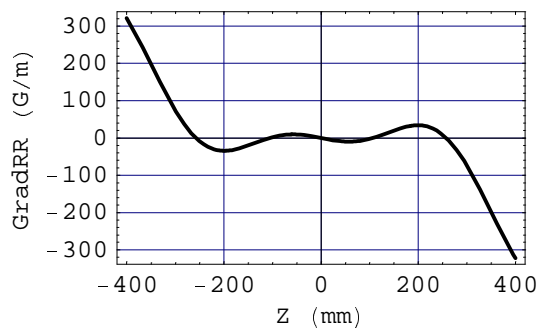
■ **4-7-3 Loop: 3D Mapping**

1. Bz 3D field map on a plane XZ i.e. orthogonal to the Y axis at the coordinate Y=5. The use of True 3D Graphics provides the capability of mapping the field on any plane orthogonal to one of the axes.

   FieldMap[loopFilament,{Bz},Plane[{-20,5,-20},{20,5,20}]];

   System name: **loopFilament**

   Bz (G) (Continuous, Automatic Mode)
   between {-20., 5., -20.} and {20., 5., 20.}
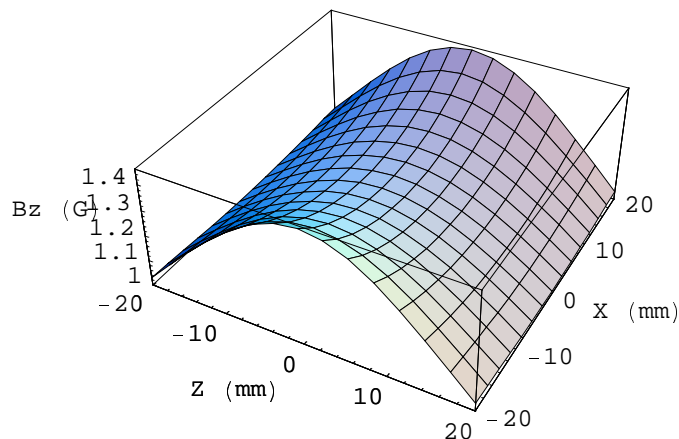
2. Bz 3D Contour Field Map on a plane XZ i.e. orthogonal the Y axis at the coordinate Y=0.

   ContourFieldMap[loopTubular,{Bz},Plane[{-20,0,-20},{20,0,20}]];

   System name: **loopTubular**

   Bz (G) (Continuous, Automatic Mode)
   between {-20., 0., -20.} and {20., 0., 20.}
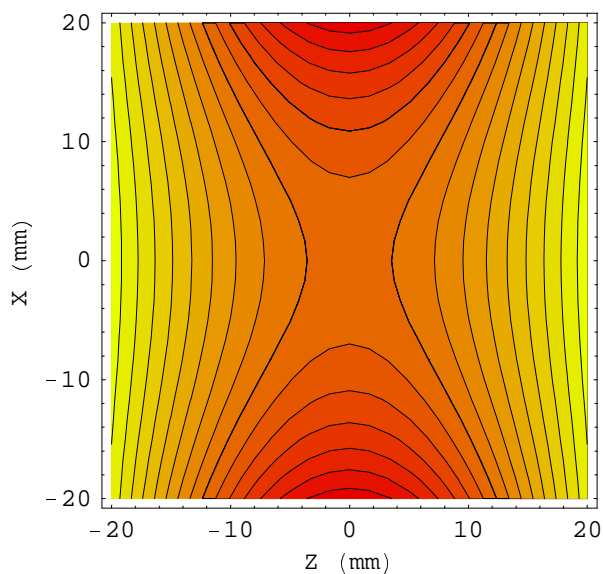
3. Flux 3D Contour Field Map on a plane through the magnetic system axis. This plot gives the true magnetic field lines (lines tangent to the vector field at any point).

```
ContourFieldMap[loopFilament,{Flux},Plane[{-20,-20},{20,20}]];
```

System name: **loopFilament**

Flux (μWb) (Continuous, Automatic
Mode)
between {-20., -20.} and {20., 20.}
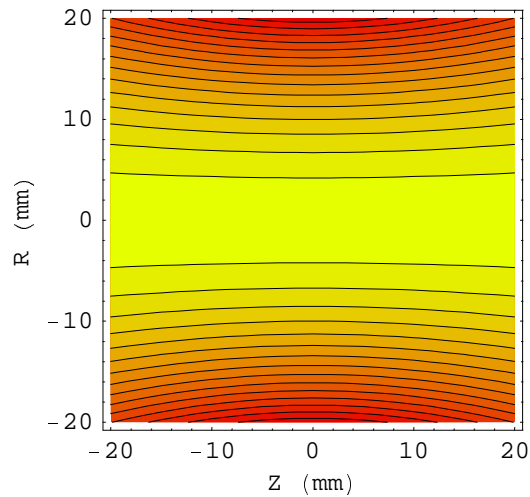
4. Vector plot map of the magnetic field on a plane through the magnetic system axis.

```
VectorFieldMap[loopFilament,Plane[{-20,-20},{20,20}]];
```

System name: **loopFilament**

vector field (Discrete, Automatic
Mode)
between {-20., -20.} and {20., 20.}
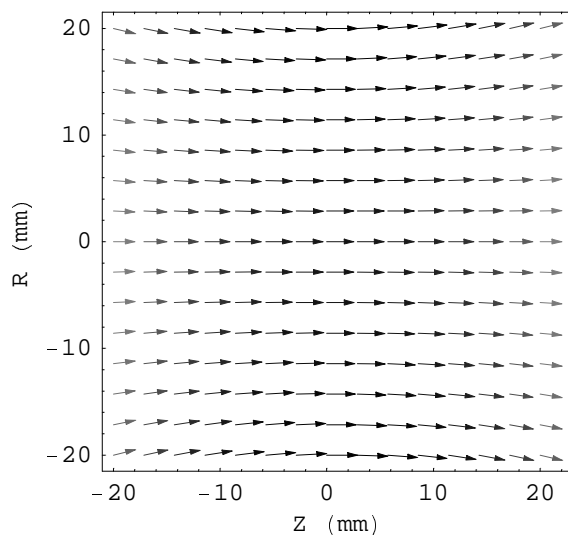
■ 4-7-4 Solenoid: 3D mapping

1. Bz 3D Field Map on a plane XZ i.e. orthogonal to the Y axis at the coordinate Y=100. The use of
   True 3D Graphics provides the capability of mapping the field at any plane orthogonal to one of the
   axes.

FieldMap[unshieldedMRI,{Bz,Bx,By},Plane[{-200,100,-200},{200,100,200}]];

System name: **unshieldedMRI**

unshielded MRI 1.5 T superconducting
magnet

Bz (G) (Continuous, Automatic Mode)
between {-200., 100., -200.} and {200., 100., 200.}
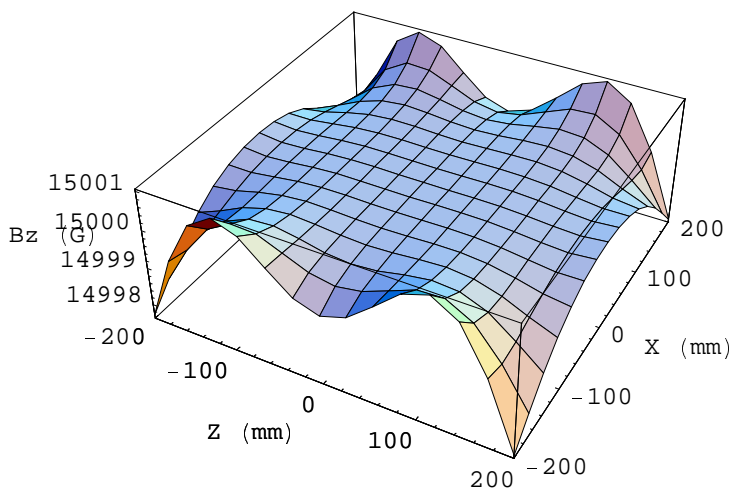
System name: **unshieldedMRI**

unshielded MRI 1.5 T superconducting
magnet

Bx (G) (Continuous, Automatic Mode)
between {-200., 100., -200.} and {200., 100., 200.}
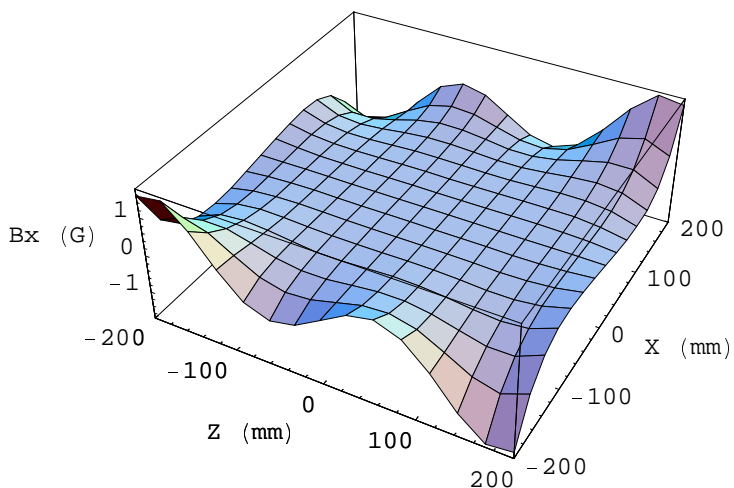
```
System name: unshieldedMRI
unshielded MRI 1.5 T superconducting
magnet
By (G) (Continuous, Automatic Mode)
between {-200., 100., -200.} and {200., 100., 200.}
```
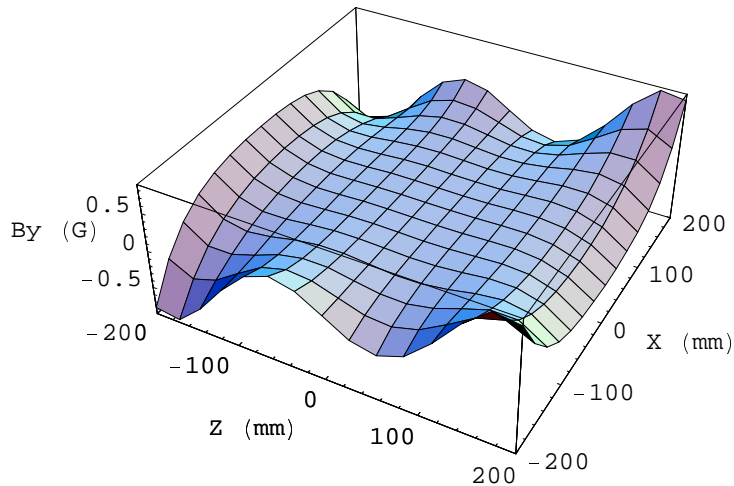

2. Bz 3D Contour Field Map on a plane XZ i.e. orthogonal the Y axis at the coordinate Y=0.

```
ContourFieldMap[unshieldedMRI,{Bz},Plane[{-200,0,-200},{200,0,200}]];
```

```
System name: unshieldedMRI
unshielded MRI 1.5 T superconducting
magnet
Bz (G) (Continuous, Automatic Mode)
between {-200., 0., -200.} and {200., 0., 200.}
```
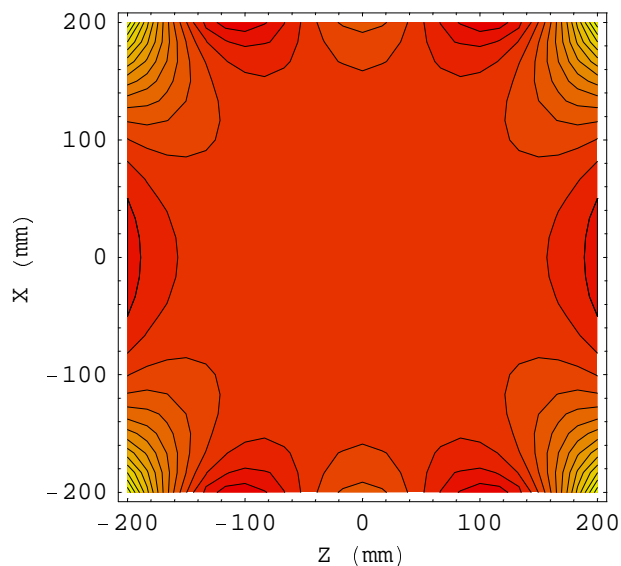

3. Flux 3D Contour Field Map on a plane through the magnetic system axis. This plot gives the true magnetic field lines (lines tangent to the vector field at any point).

```
ContourFieldMap[unshieldedMRI,{Flux},Plane[{-200,-200},{200,200}]];
```

System name: **unshieldedMRI**

unshielded MRI 1.5 T superconducting

magnet

Flux (μWb) (Continuous, Automatic

Mode)

between {-200., -200.} and {200., 200.}
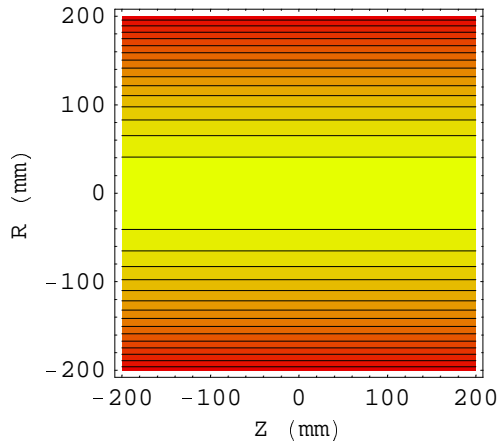
4. Vector Plot Map of the magnetic field on a plane through the magnetic system axis.

```
VectorFieldMap[unshieldedMRI,Plane[{-200,-200},{200,200}],NumberOfPoints->10];
```

System name: **unshieldedMRI**

unshielded MRI 1.5 T superconducting

magnet

vector field (Discrete, Automatic

Mode)

between {-200., -200.} and {200., 200.}
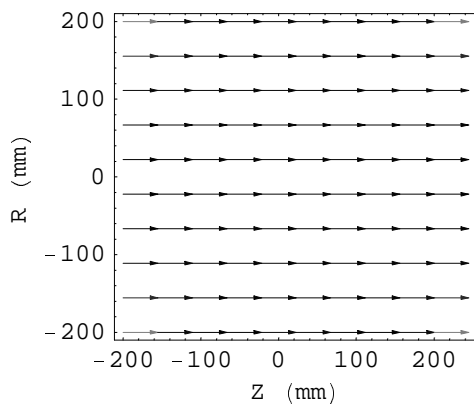
■ Chapter 5: Field Calculation using Spherical Harmonics

■ 5-1 Fundamentals of Spherical Harmonics

■ 5-1-1 The Basic Concepts

Spherical harmonics is the solution of Laplace equations in spherical coordinates that is formalized with Legendre polynomial series. There are many ways to proceed for field calculation. The goal of the methodology used in *Magnetica* is to provide accuracy, swiftness of calculation and versatility of use.

The basic principle is the separation of the complete formula in 2 subsets using appropriate factorization. One subset is specific to the field generator source and the other depends only on the position of the target point. It is necessary to distinguish the central region and the remote region, as they are different both in terms of source coefficients and in terms of target field formulas.

This separation is particularly interesting for computer calculation. Indeed, the source coefficients need to be calculated only once. Moreover, using appropriate normalization, the various system component coefficients can be simply summed to obtain the global source coefficients of the system. It should be noted that appropriate spherical harmonics formulas could be derived for all the field-related functions (field, flux, etc.). These formulas are all of the same type: polynomial series that can be calculated rapidly and easily. For magnetic systems made of several components, the advantage of the swiftness of the SH mode is even greater.

The *Magnetica* spherical harmonics functions are dependent of the 2 following parameters:

1. The position of the spherical coordinate (called the Legendre point). This point can be situated at any place on the axis.

2. The system normalization radius (called the Legendre radius).

Although arbitrary, the choice of these parameters is not indifferent. The position of the Legendre point is basic for setting the region where SH mode can be applied and the accuracy of the final result is related to the Legendre radius.

■ 5-1-2 The Space Limitations for the Spherical Harmonics Mode and the Legendre Point Position

This concept is fundamental for the correct use of the spherical harmonics mode of calculation since, by definition, the method cannot be applied everywhere. For any magnetic system, the central and the remote limiting radius are precisely calculated. They are shown when using the function **SHLimits** (see paragraph 5-3-2).

A graphics display of these limits is generated using the function **DrawSHLimits**. More specifically, this graphic displays shows:

- The strictly forbidden region defined as the region of absolutely no convergence of the Spherical Harmonics series

- On the 2 sides of this region are zones where the accuracy of the SH mode is uncertain due to the limited numbers of terms of the *Magnetica* algorithm. These 2 regions are also excluded in the SH mode.

- On the 2 sides of the previous region, other zones concern the Automatic mode. When this mode is selected, the switch between the SH and the VP modes is made automatically depending on whether the target point is outside or inside these 2 regions.

As we have indicated in the previous Chapter, the use of the field function such as FieldPoint appended with the suffixes VP or SH overrides the Automatic mode.

The position of the Legendre point is arbitrary. For axisymmetric systems, it can be set at any position on the axis. The default position of the Legendre point is the origin. However, it can be set at any position on the axis using the system option **LegendrePointPosition**→value. This circumvents to a certain extent the inherent space limitations.

▪ 5-1-3 Legendre Radius and Legendre Source Coefficients

The first and main reason for using a Legendre radius parameter is to provide a normalization of spherical harmonics dimensions for a magnetic system made of several different dimension components. This parameter is arbitrary and, theoretically, its value has no direct consequence on field calculation. However, the choice is not indifferent. The numerical values of the Legendre source coefficients depend strongly on the Legendre radius, as we will see in the examples. Indeed, the coefficients of order *N* of the series vary with power (N). Due to the limited precision of the computer real numbers, the coefficients of high order may reach numerical values where the computer number truncation error becomes important. *Magnetica* chooses automatically the Legendre radius in appropriate relation with the real size of the magnetic system and the Legendre point position. This minimizes numerical error.

Another reason for the choice of the Legendre Radius is the physical meaning attached to coefficients. Their values represent the contribution to the magnetic field on the surface of the sphere for each of the successive orders of the series. In a sense, they measure in an orderly manner, the quality of the field on the surface of the sphere without doing any field computation.

A good example of this use is its application to an MRI magnet. Indeed, MRI magnet design are requires an efficient way to characterize field homogeneity in the region where the patient image is observed (ROI, region of interest). This is a spherical volume of about 0.5 meters (20 inches) in diameter. Therefore, setting the Legendre radius at this value provides a direct appreciation of the field quality in the ROI. The lower order (up to 6) coefficients must be very small to achieve the necessary homogeneity.

The coefficients are calculated only once for a magnetic system and may be printed out at any time. The results print-out can be done in different ways, either as the internal table (default output) or for other values of the Legendre radius value (without affecting the internal table). Also, the output mode can generate either the actual values of the coefficients or the relative value with respect to the first order coefficient.

The function **LegendreCoefficients** uses the system name as argument and produces the array of all coefficients. The options: **LegendreMode**→*type* and **LegendreRadius**→*value* provide new versatile forms of results.

The first coefficient term stands for the order 0 (space-independent), and the other terms are sorted up to the order 20.

The first term of the remote table starts at order 2.

Edited output can be displayed using the **Verbose** prefix. Various relevant items of information are printed as well as the coefficients for both the central and the remote regions.

■ 5-2 Immediate Help

■ 5-2-1 Options of Magnetic Systems Related to Spherical Harmonics

?AXSMagneticSystem

```
AXSMagneticSystem[{comp1,comp2,...},options]  defines
  a new axis symmetric system made with the specified components.

Options: SystemLabel, MeshingLevel,
  LegendrePointPosition, LegendreRadius, LegendreOrder Details…
```

?LegendreRadius

```
LegendreRadius is an option for AXSMagneticSystem,
  that specifies the Legendre radius value for this system Details…
LegendreRadius as an option for LegendreCoefficients or VerboseLegendreCoefficients
  allows displaying the Legendre coefficients inan alternative Legendre Radius. You may
  specify a single value or a list of 2 values as the central and remote coefficients.
```

?LegendrePointPosition

```
LegendrePointPosition is an option for AXSMagneticSystem,
  specifies the position of the Legendre point on the Z-axis. Details…
```

?LegendreOrder

```
LegendreOrder is an option for AXSMagneticSystem that specifies the number of
  Legendre Coefficients used in the spherical harmonics calculations Details…
```

■ 5-2-2 Functions and Directives for Spherical Harmonics Calculation

?SHLimits

```
SHLimits[system] gives a list of the limits for the convergence region, the effective
  limits for SphericalHarmonics, the limits used for the Automatic mode Details…
```

?DrawSHLimits

```
DrawSHLimits[system] displays the magnetic system with its Spherical Harmonics limits.
Suffixing DrawSHLimits with XY,XZ,YX,... displays the chosen point of view. Details…
```

?VerboseSHLimits

```
VerboseSHLimits[system] prints a detailed and formatted output of SHLimits Details…
```

?LegendreCoefficients

```
LegendreCoefficients[system, options] displays the
  Legendre source coefficients of the system. The available options are:

LegendreMode -> AbsoluteMode | RelativeMode
LegendreRadius -> radius | {centralRadius, remoteRadius} Details…
```

?LegendreMode

```
LegendreMode is an option for LegendreCoefficients or VerboseLegendreCoefficients that
  allows specifying if you want to see the Legendre source coefficients of a system
  as "RelativeMode" or as "AbsoluteMode". By default the mode is AbsoluteMode Details…
```

```
?LegendreRadius
```

```
LegendreRadius is an option for AXSMagneticSystem,
   that specifies the Legendre radius value for this system Details…
LegendreRadius as an option for LegendreCoefficients or VerboseLegendreCoefficients
   allows displaying the Legendre coefficients inan alternative Legendre Radius. You may
   specify a single value or a list of 2 values as the central and remote coefficients.
```

■ 5-3 Circular Loop Field Calculation

■ 5-3-0 Introduction

The first example is the basic circular loop already used in the previous chapter (4-3).

In order to illustrate the effect of changing the position of the Legendre point, 2 cases are shown with the Legendre point set either on or off center.

■ 5-3-1 Entering Data

Case 1: single loop, filament type , Legendre point set at the center. The relevant data are:

- **mean loop diameter**: 100 mm

- **axial position of the loop**: 0 mm (i.e. on the coordinate isocenter)

- **winding diameter**: 30 mm

- **number of turns (right hand rule, i.e. counterclockwise)**: 20

- **current intensity**: 0.5 amperes

```
LoopFilamentOn = AXSMagneticSystem[
    {AXSCircularLoop[CopperOFHC, 100, 30, 0, 20, 0.5, FillingFactor → 0.7]},
    SystemLabel -> "Basic Loop, Legendre point on center"];
```

Case 2: the same loop, Legendre point set off the center using the option:

- **LegendrePointPosition→** 100 (mm).

```
LoopFilamentOff= AXSMagneticSystem[
{AXSCircularLoop[CopperOFHC, 100, 30, 0, 20, 0.5, FillingFactor→0.7]},
SystemLabel -> "Basic Loop, Legendre point off center : 100 mm",
LegendrePointPosition -> 100];
```

■ 5-3-2 Spherical Limit Data and Drawing

1. Case 1: the function **SHLimits** displays the raw data on the limits that can be used for further calculation

```
SHLimits[LoopFilamentOn]
```

```
{{35., 65.}, {32.55, 74.1}, {25.55, 89.05}}
```

2. Case 1: the function **VerboseSHLimits** prints out the limits as edited and formatted output

```
VerboseSHLimits[LoopFilamentOn]
```

	Min	Max
Convergence Region	35.	65.
Limits for using SH	32.55	74.1
Limits for Automatic	25.55	89.05

3. Same as 2 for Case 2. The limits are modified significantly by displacement of the Legendre point

```
VerboseSHLimits[LoopFilamentOff]
```

	Min	Max
Convergence Region	96.8034	126.803
Limits for using SH	90.0272	144.556
Limits for Automatic	70.6665	173.721

4. Drawing the limits for case 1 using the function **DrawSHLimits**. The drawing shows clearly the 3 spherical regions. They are centered at the Legendre Point and at the origin.

```
DrawSHLimits[LoopFilamentOn];
```

System name: **LoopFilamentOn**
Basic Loop, Legendre point on center
2D view ZR
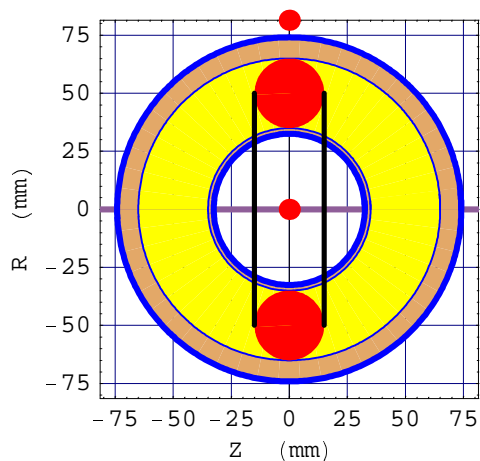
5. Drawing the limits for Case 2. Now the 3 spherical regions centered on the Legendre point off isocenter are very different from those of case 1.

```
DrawSHLimits[LoopFilamentOff];
```

System name: **LoopFilamentOff**
Basic Loop, Legendre point off
center : 100 mm
2D view ZR
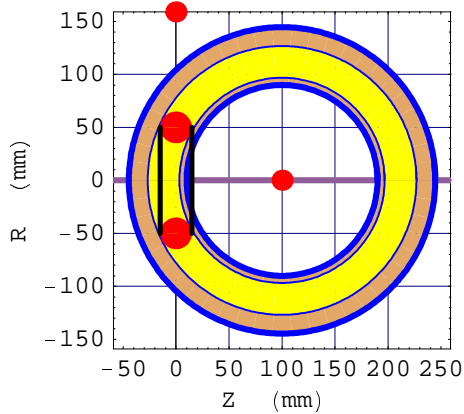
▪ 5-3-3 Table of the Legendre Source Coefficients

The Legendre source coefficients can be displayed through different procedures.

N.B. the unit of coefficients being a magnetic field unit on the proper sphere, either central or remote, all the directives related to changing units can be applied.

- **Array** of 2 lists of 21 coefficients for case 1. The first list is for the central coefficients and the second is for the remote.

  ```
  LegendreCoefficients[LoopFilamentOn]
  ```

  ```
  {{1.25664, 0, -1.88496, 0, 2.35619, 0, -2.74889, 0, 3.09251, 0,
    -3.40176, 0, 3.68524, 0, -3.94847, 0, 4.19525, 0, -4.42831, 0, 4.64973},
   {0, 0, 1.25664, 0, -1.88496, 0, 2.35619, 0, -2.74889, 0, 3.09251,
    0, -3.40176, 0, 3.68524, 0, -3.94847, 0, 4.19525, 0, -4.42831}}
  ```

- The 2 lists are shown as a double vertical table that is easily readable using the *Mathematica* functions TableForm and Transpose:

```
TableForm[Transpose[LegendreCoefficients[LoopFilamentOn]]]
```

```
1.25664        0
0              0
-1.88496       1.25664
0              0
2.35619        -1.88496
0              0
-2.74889       2.35619
0              0
3.09251        -2.74889
0              0
-3.40176       3.09251
0              0
3.68524        -3.40176
0              0
-3.94847       3.68524
0              0
4.19525        -3.94847
0              0
-4.42831       4.19525
0              0
4.64973        -4.42831
```

- Sorting out the first order central coefficient and the second order remote coefficient (the two tables begin at coefficient order zero, so the list number is the order incremented by 1).

```
firstcentral=LegendreCoefficients[LoopFilamentOn][[1,1]]
secondremote=LegendreCoefficients[LoopFilamentOn][[2,3]]
```

```
1.25664
```

```
1.25664
```

- Changing the units of coefficients into Milli Tesla

```
LegendreCoefficients[LoopFilamentOn,FieldUnit→Milli Tesla]
```

```
{{0.125664, 0, -0.188496, 0, 0.235619, 0, -0.274889, 0, 0.309251, 0, -0.340176,
  0, 0.368524, 0, -0.394847, 0, 0.419525, 0, -0.442831, 0, 0.464973},
 {0, 0, 0.125664, 0, -0.188496, 0, 0.235619, 0, -0.274889, 0, 0.309251,
  0, -0.340176, 0, 0.368524, 0, -0.394847, 0, 0.419525, 0, -0.442831}}
```

Edited output using the prefix **Verbose.** The coefficients are formatted and complementary data on the limiting radius are given

Default print-out of Case 1 using the function **VerboseLegendreCoefficients** without options

- The system Legendre radius automatically calculated is 50 mm

- The spherical limits are respectively 32.55 and 74.1 mm

- The magnetic parity system is True, which implies that all coefficients of odd order are null.

```
VerboseLegendreCoefficients[LoopFilamentOn,
 LegendreRadius → {10, 100}, FieldUnit → Milli Tesla]
```

```
System Name:                              LoopFilamentOn
System Legendre radius                     50.000 mm
Printout Legendre radius central           10.000 mm
Printout Legendre radius remote           100.000 mm
System Legendre center position            0.000 mm
Inner spherical radius limit              32.550 mm
Outer spherical radius limit              74.100 mm
System magnetic parity                      True
System magnetic constant field             0.000 mT
Mode: absolute Field (Milli Tesla)


Order         central coeff.         remote coeff.
D0              0.125663706            0.000000000
D1              0.000000000            0.000000000
D2             -0.007539822            0.015707963
D3              0.000000000            0.000000000
D4              0.000376991           -0.005890486
D5              0.000000000            0.000000000
D6             -0.000017593            0.001840777
D7              0.000000000            0.000000000
D8              7.916813000e-7        -0.000536893
D9              0.000000000            0.000000000
D10            -3.483400000e-8         0.000151001
D11             0.000000000            0.000000000
D12             1.509500000e-9        -0.000041525
D13             0.000000000            0.000000000
D14             0.000000000            0.000011246
D15             0.000000000            0.000000000
D16             0.000000000           -3.012440900e-6
D17             0.000000000            0.000000000
D18             0.000000000            8.001796000e-7
D19             0.000000000            0.000000000
D20             0.000000000           -2.111585000e-7
```

Print-out for Case 1, using the function **LegendreSourceCoefficients** with the option **LegendreMode**→RelativeMode. This print out shows:

- The central relative coefficients are the initial value of the Legendre polynomial derivative

- The remote coefficients are identical to the central coefficients but with a difference of order of 2

N.B. These relations are strictly specific to the centered filament loop.

```
VerboseLegendreCoefficients[LoopFilamentOn, LegendreMode → RelativeMode]

System Name:                           LoopFilamentOn
System Legendre radius                    50.000 mm
Printout Legendre radius central          50.000 mm
Printout Legendre radius remote           50.000 mm
System Legendre center position            0.000 mm
Inner spherical radius limit              32.550 mm
Outer spherical radius limit              74.100 mm
System magnetic parity                      True
System magnetic constant field             0.000 G
Mode: relative Field to D=0

Order           central coeff.         remote coeff.
D0               1.000000000            0.000000000
D1               0.000000000            0.000000000
D2              -1.500000000            1.000000000
D3               0.000000000            0.000000000
D4               1.875000000           -1.500000000
D5               0.000000000            0.000000000
D6              -2.187500000            1.875000000
D7               0.000000000            0.000000000
D8               2.460937500           -2.187500000
D9               0.000000000            0.000000000
D10             -2.707031250            2.460937500
D11              0.000000000            0.000000000
D12              2.932617188           -2.707031250
D13              0.000000000            0.000000000
D14             -3.142089844            2.932617188
D15              0.000000000            0.000000000
D16              3.338470459           -3.142089844
D17              0.000000000            0.000000000
D18             -3.523941040            3.338470459
D19              0.000000000            0.000000000
D20              3.700138092           -3.523941040
```

Print-out for Case 2, using the function **LegendreSourceCoefficients** with the option **LegendreRadius**→{50,250}.

This print-out shows that:

- The system Legendre radius automatically calculated is 111.803 mm

- The output Legendre radii are respectively 50 and 250 mm

- The meaning of the coefficients as the field contribution of the various orders on the surface of the sphere, central and remote, can readily been interpreted.

```
VerboseLegendreCoefficients[LoopFilamentOff, LegendreRadius → {50, 250}]

System Name:                              LoopFilamentOff
System Legendre radius                    111.803  mm
Printout Legendre radius central           50.000  mm
Printout Legendre radius remote           250.000  mm
System Legendre center position           100.000  mm
Inner spherical radius limit               90.027  mm
Outer spherical radius limit              144.556  mm
System magnetic parity                      False
System magnetic constant field              0.000  G
Mode: absolute Field (Gauss)


Order            central coeff.           remote coeff.
D0                0.112397036             0.000000000
D1               -0.134876443             0.000000000
D2                0.101157332             0.010053096
D3               -0.058446459            -0.012063716
D4                0.027312480             0.009047787
D5               -0.010007832            -0.005227610
D6                0.002300543             0.002442902
D7                0.000315611            -0.000895128
D8               -0.000785891             0.000205767
D9                0.000593506             0.000028229
D10              -0.000325649            -0.000070292
D11               0.000142869             0.000053085
D12              -0.000048500            -0.000029127
D13               9.520563900e-6          0.000012779
D14               2.504379700e-6         -4.337967600e-6
D15              -4.101340900e-6          8.515451000e-7
D16               2.851425500e-6          2.239985000e-7
D17              -1.479713500e-6         -3.668351000e-7
D18               6.146857000e-7          2.550393000e-7
D19              -1.931707000e-7         -1.323496000e-7
D20               2.931600000e-8          5.497920000e-8
```

■ 5-3-4 Ponctual Magnetic Field Calculation

All the magnetic field functions can be used with spherical harmonics, the only restriction being the forbidden regions of space.

As stated in 5-1, the field functions automatically take advantage of the spherical harmonics mode when the coordinate of the target point are situated in the proper regions. But using the suffix SH works out spherical harmonics calculations in any position and generates an error message when the calculation can not be done correctly.

In the following examples calculation will be carried out with the 2 modes in order to compare the results, all for Case 1 (circular loop centered)

   1. Field on the axis at the isocenter point (R=0,Z=0). The VP and the SH modes give exactly the same
      values as expected.

```
rc = 0; zc = 0;
FieldPointSH[LoopFilamentOn, {rc, zc}]
FieldPointVP[LoopFilamentOn, {rc, zc}]
```

```
{0, 1.25664}
```

```
{0, 1.25664}
```

2. Target point near the winding. Although the target point is close to the winding, the difference between the SH and the VP modes is very small (<0.0001)

N.B. the practical limits of the radius are such that the accuracy is better than 1 % at the limit, whatever the coil geometry.

```
rc = 32.1; zc = 5; precis = 12;
NumberForm[FieldPointSH[LoopFilamentOn, {rc, zc}], precis]
NumberForm[FieldPointVP[LoopFilamentOn, {rc, zc}], precis]
```

```
{0.304110254339, 1.78429083004}
```

```
{0.304058527682, 1.78420370776}
```

3. Target point in the winding.

```
rc = 52; zc = 5; precis = 12;
NumberForm[FieldPointSH[LoopFilamentOn, {rc, zc}], precis]
NumberForm[FieldPointVP[LoopFilamentOn, {rc, zc}], precis]
```

```
FieldPointSH: target point {52, 5} not in
    the meaningful domain. Use VectorPotential or Automatic mode
```

```
Details…
```

```
$Aborted
```

```
{0.401264093924, 0.459497176316}
```

4. Target point near the winding but with changing the length units to inches and field units to Milli Gauss.

Warning: from now on, lengths are in Inches, and fields are in Milli Gauss, until explicitly changed.

```
rc = 1.23; zc = 0.2; precis = 12;
MagneticaUserUnits[FieldUnit → Milli Gauss, LengthUnit → Inch];
NumberForm[FieldPointSH[LoopFilamentOn, {rc, zc}], precis]
NumberForm[FieldPointVP[LoopFilamentOn, {rc, zc}], precis]
```

```
{281.68064577, 1739.24531252}
```

```
{281.64619312, 1739.20232348}
```

Resetting the units at the default values:

```
MagneticaUserUnits[];
```

5. Magnetic Potential, Flux and Gradient at a target point on axis (R=20 mm, Z=10 mm) for the filament loop and the tubular loop. using MatrixForm, the gradient can be shown as a matrix default units are:

- Micro Weber / Meter for the magnetic potential

- Micro Weber for the magnetic flux

- Gauss / Meter for the magnetic field gradient

```
rc = 20; zc = 10;
PotentialPointSH[LoopFilamentOn, {rc, zc}]
FluxAreaSH[LoopFilamentOn, {rc, zc}]
GradientPointSH[LoopFilamentOn, {rc, zc}]
MatrixForm[GradientPointSH[LoopFilamentOn, {rc, zc}]]
```

{1.24333}

{0.156241}

{{-23.4458, 12.9928}, {12.9928, 14.4071}}

$$\begin{pmatrix} -23.4458 & 12.9928 \\ 12.9928 & 14.4071 \end{pmatrix}$$

■ 5-3-5 Field Map Graphics

As regards punctual calculations, all the field Map Graphics functions can be used in the spherical harmonics mode, naturally with the same restrictions regarding the forbidden regions of space. The existing mode may be overridden and the spherical harmonics mode is selected by using the option `CalculationMode → SphericalHarmonics`, as an additional argument to the concerned function.

1. This example is for the Case 1 of the circular loop centered for the Bz field map along the space plot, defined by a radial line at z = 75 mm from r = 0 to r = 100 mm. Field unit is in Milli Gauss (for this plot only) using the option `FieldUnit→Milli Gauss`.

```
FieldMap[LoopFilamentOn,{Bz},Line[{0,75},{100,75}],
CalculationMode->SphericalHarmonics,
FieldUnit->Milli Gauss];
```

```
System name: LoopFilamentOn
Basic Loop, Legendre point on center
Bz (mG) (Continuous, Spherical
Harmonics)
between {0., 75.} and {100., 75.}
```
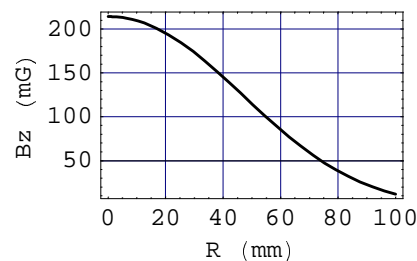

2. ∂Bz/∂z component of Gradient Map for the same circular loop on an axial line at r = 5 mm from z = -30 to r = 30 mm

```
FieldMap[LoopFilamentOn,{GradZZ},Line[{5,-30},{5,30}],
CalculationMode->SphericalHarmonics];
```

System name: **LoopFilamentOn**
Basic Loop, Legendre point on center
GradZZ (G/m) (Continuous, Spherical
Harmonics)
between {5., -30.} and {5., 30.}
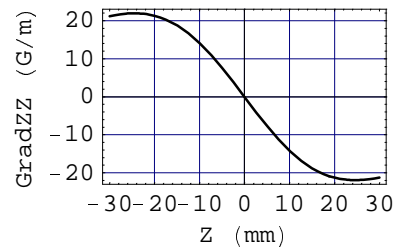
3. Field Plot Map of the magnetic field for the same case on a plane through the magnetic system axis

```
FieldMap[LoopFilamentOn,{Bz},Plane[{-20,-20},{20,20}],
CalculationMode->SphericalHarmonics];
```

System name: **LoopFilamentOn**
Basic Loop, Legendre point on center
Bz (G) (Continuous, Spherical Harmonics)
between {-20., -20.} and {20., 20.}
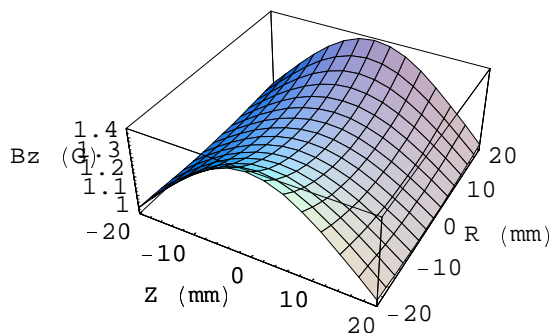
4. Flux 3D Contour Field Map on a plane through the magnetic system axis

N.B. this plot gives the true magnetic field lines (line tangent to the vector field at any point).

```
ContourFieldMap[LoopFilamentOn,{Flux},Plane[{-20,80},{20,120}],
CalculationMode->SphericalHarmonics];
```

System name: **LoopFilamentOn**
Basic Loop, Legendre point on center
Flux (μWb) (Continuous, Spherical
Harmonics)
between {-20., 80.} and {20., 120.}
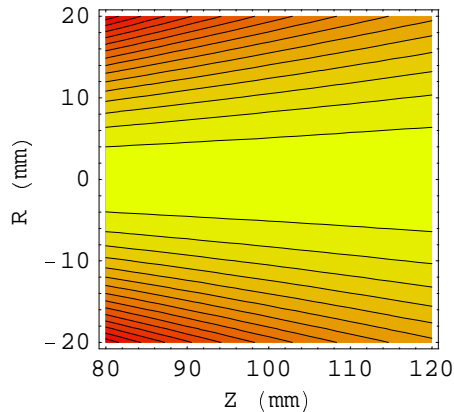
■ 5-4 Application to a MRI superconducting solenoids

■ 5-4-0 Introduction

As indicated in the first section (5-1-2), MRI magnet design is a very demanding task for which the use of spherical harmonics is particularly well adapted. The example is a magnetic system of 6 superconducting solenoids set at the isocenter.

■ 5-4-1 Entering Data

Magnetic system of 6 superconducting solenoids set at the center as 3 pairs set symmetrically around the isocenter.

```
superconducMRI=
AXSMagneticSystem[
{AXSSolenoid[SuperNbTi4, 1240, 1280, 300, 180.05374474797077, 635, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1280, 300, -180.05374474797077, 635, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1280, 250, 550.1194910091477, 598, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1280, 250, -550.1194910091477, 598, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1320, 250, 874.8480174899299, 1015, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1320, 250, -874.8480174899299, 1015, 735.2202]},
SystemLabel -> "6 coils MRI 1.5 T superconducting magnet"];
```

■ 5-4-2 Spherical Limits Data and Drawing

1. Using the function **VerboseSHLimits** to print out the limits as edited and formatted output

```
VerboseSHLimits[superconducMRI]
```

	Min	Max
Convergence Region	620.728	1198.04
Limits for using SH	577.277	1365.76
Limits for Automatic	453.131	1641.31

2. Drawing the limits for the MRI magnet using the function **DrawSHLimits.** The drawing shows clearly the 3 spherical regions. They are centered at the system isocenter

```
DrawSHLimits[superconducMRI];
```

> System name: **superconducMRI**
> *6 coils MRI 1.5 T superconducting*
> *magnet*
> 2D view ZR
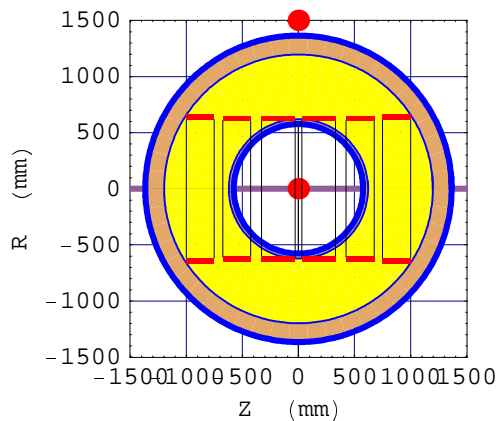
■ 5-4-3 Legendre Coefficients Table for MRI Superconducting Solenoids

This output has the print-out Legendre radius adjusted to representative values: ROI for the central region and room dimensions for the remote region.

- The system Legendre radius automatically calculated is 909.383 mm.
- The output Legendre radius is respectively 250 and 2500 mm.
- The central coefficients of order 2 and 4 are very small which is specific to a magnet of good homogeneity.
- The remote coefficients of the low order (2 to 6) are large which means an important stray field.

1. Display of array of source coefficients

```
LegendreCoefficients[superconducMRI]
```

$\{\{15000., 1.36424 \times 10^{-12}, -0.000833933, -9.09495 \times 10^{-13}, -0.014836,$
$2.6148 \times 10^{-12}, 4842.9, 3.18323 \times 10^{-12}, -49591.6, -2.16005 \times 10^{-12},$
$143440., -2.95586 \times 10^{-12}, -267536., 0., 417936., -1.97815 \times 10^{-11},$
$-772603., -5.37739 \times 10^{-11}, 1.87705 \times 10^{6}, 4.32578 \times 10^{-11}, -4.6495 \times 10^{6}\},$
$\{0., 0., 11121.9, 0., 27841.7, 0., 13450.2, 0., -55676.4, 0., -107595.,$
$7.27596 \times 10^{-12}, 5175.1, 0., 285436., 0., 356796., 0., -275331., 0., -1.29054 \times 10^{6}\}\}$

2. Sorting out all the central coefficients with the irrelevant values chopped out using the *Mathematica* function Chop

```
Chop[LegendreCoefficients[superconducMRI]][[1]]
```

$\{15000., 0, -0.000833933, 0, -0.014836, 0, 4842.9, 0, -49591.6, 0, 143440.,$
$0, -267536., 0, 417936., 0, -772603., 0, 1.87705 \times 10^{6}, 0, -4.6495 \times 10^{6}\}$

3. Editing the Legendre coefficients with central sphere Legendre radius set at 250 mm (ROI radius)
   and 2500 mm for remote coefficients

   VerboseLegendreCoefficients[superconducMRI, LegendreRadius -> {250, 2500}]

   ```
   System Name:                        superconducMRI
   System Legendre radius                909.383 mm
   Printout Legendre radius central      250.000 mm
   Printout Legendre radius remote      2500.000 mm
   System Legendre center position         0.000 mm
   Inner spherical radius limit          577.277 mm
   Outer spherical radius limit         1365.764 mm
   System magnetic parity                   True
   System magnetic constant field          0.000 G
   Mode: absolute Field (Gauss)


   Order           central coeff.          remote coeff.
   D0          15000.000067887            0.000000000
   D1              0.000000000            0.000000000
   D2             -0.000063026          535.302033980
   D3              0.000000000            0.000000000
   D4             -0.000084740          177.308722411
   D5              0.000000000            0.000000000
   D6              2.090564502           11.333837233
   D7              0.000000000            0.000000000
   D8             -1.617905108           -6.207741387
   D9              0.000000000            0.000000000
   D10             0.353671288           -1.587330253
   D11             0.000000000            0.000000000
   D12            -0.049853894            0.010102023
   D13             0.000000000            0.000000000
   D14             0.005885907            0.073724575
   D15             0.000000000            0.000000000
   D16            -0.000822329            0.012193769
   D17             0.000000000            0.000000000
   D18             0.000150991           -0.001245048
   D19             0.000000000            0.000000000
   D20            -0.000028266           -0.000772176
   ```

■ **5-4-4 Magnetic Field Calculation**

1. Target point in the central region (ROI): the difference between the SH and the VP mode is very
   small.

   rc = 200; zc = 150; precis = 12;
   MagneticaUserUnits[FieldUnit → Tesla];
   NumberForm[FieldPointSH[superconducMRI, {rc, zc}], precis]
   NumberForm[FieldPointVP[superconducMRI, {rc, zc}], precis]

   {0.0000871959865189, 1.49999340293}

   {0.000087195906579, 1.49999340298}

2. Target point away from the winding (stray field). Field unit in Milli Gauss, results using the verbose
   function:

```
rc = 3000; zc = 1000;
MagneticaUserUnits[FieldUnit → Gauss];
VerboseFieldPoint[superconducMRI, {rc, zc}, CalculationMode → SphericalHarmonics]
VerboseFieldPoint[superconducMRI, {rc, zc}, CalculationMode → VectorPotential]
```

```
System name: superconducMRI
The magnetic field (in Gauss) at the point {3000, 1000}(in Milli Meter)
Calculation mode: Automatic Mode
Br          96.176960 G
Bz         -89.661038 G
Bm         131.488058 G


System name: superconducMRI
The magnetic field (in Gauss) at the point {3000, 1000}(in Milli Meter)
Calculation mode: Automatic Mode
Br          96.176960 G
Bz         -89.661038 G
Bm         131.488058 G
```

3. Bz 2D Field Map in central region for an axial line at r=0 from z =-200 0 to z = 200 mm. Field unit is in Tesla (for this plot only)

```
FieldMap[superconducMRI,{Bz},Line[{0,-200},{0,200}],
CalculationMode->SphericalHarmonics,
FieldUnit-> Tesla];
```

System name: **superconducMRI**

6 coils MRI 1.5 T superconducting
magnet
Bz (T) (Continuous, Spherical Harmonics)
between {0., -200.} and {0., 200.}
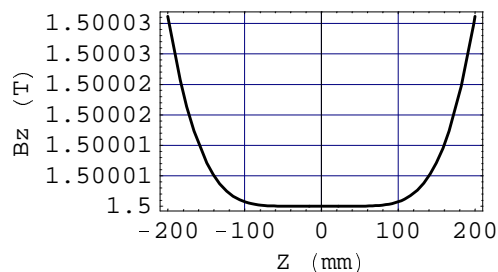
4. Bz 3D Field Map on a plane through the axis in the central region (ROI).

```
FieldMap[superconducMRI,{Bz},Plane[{-250,-250},{250,250}],
CalculationMode->SphericalHarmonics,
FieldUnit->Milli Tesla];
```

System name: **superconducMRI**

6 coils MRI 1.5 T superconducting
magnet

Bz (mT) (Continuous, Spherical
Harmonics)

between {-250., -250.} and {250., 250.}
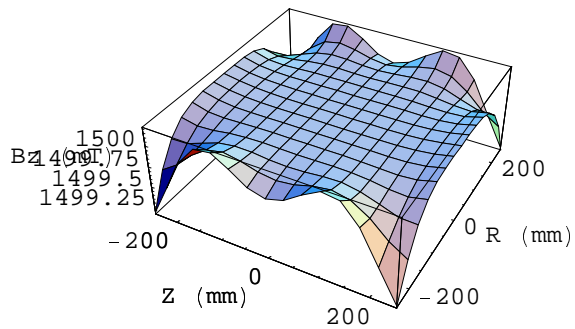
5. Bz 3D Contour Field Map on a plane through the magnetic system axis:

```
ContourFieldMap[superconducMRI,{Bz},Plane[{-250,-250},{250,250}],
CalculationMode->SphericalHarmonics];
```

System name: **superconducMRI**

6 coils MRI 1.5 T superconducting
magnet

Bz (G) (Continuous, Spherical Harmonics)

between {-250., -250.} and {250., 250.}
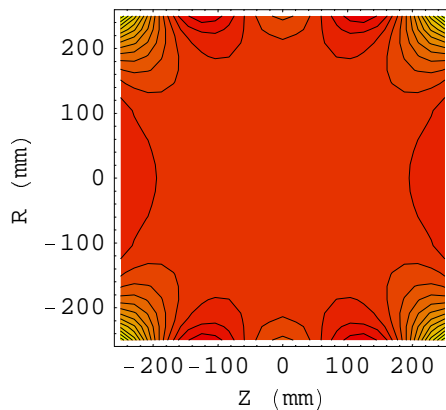
■ Chapter 6: Fundamentals on Ferromagnetism

■ 6-1 Introduction

A magnetic field can be generated by a device without any external electric current but taking advantage of the magnetic properties of the matter from which the device is made. As we will see in Section 6.3, this magnetic field can also be related to electric currents which are not visible being microscopic whereas, for current-carrying devices, they are measurable being macroscopic.

The magnetic field generated depends simultaneously on 4 factors:

- the intrinsic property of the materials (i.e. its magnetization)

- the geometric shape of the device

- the magnetic environment (i.e. the intensity of the magnetic field into which the device is plunged).

- the history of the interaction between the materials and the magnetic environment.

The relationship between these 4 factors is very intricate. The magnetic properties of the matter are the most important factor. A detailed account of these properties is developed in the Appendix B.1 "Physical and Magnetic Properties of Materials". But a first glance at the situation shows that all the materials can be classified in the following categories:

- Common materials. They have a very weak response to the magnetic field environment. This response is expressed by the magnetic susceptibility (χ). The value of the relative susceptibility is in the range of 10^{-6}. Most of the known substances, whether natural, artificial, synthetic, or biological, etc. belong to this category. We can distinguish 2 types of materials:

- paramagnetic materials. They have a positive response ($\chi > 0$)

- diamagnetic materials. They have a negative response ($\chi < 0$)

- Ferromagnetic materials. They have a very stong response to the magnetic environment. The relative susceptibility can reach values ranging for few hundred up to a million. These properties are limited to materials involving only a very few atomic elements, essentially iron, but also nickel, cobalt and some rare earth elements as Samarium or Neodymium. The magnetic properties depend not only on the composition on the substance, but also on its physical and chemical states as well as on the history of the magnetization process.

In terms of magnetostatics, we can distinguish the following ferromagnetic materials:

- hard ferromagnetic materials. These materials permanently keep a trace of the magnetic field to which they were initially subjected. This property is specific to a certain direction of the substance in relation to its crystal structure. This category of ferromagnetic material includes all permanent magnet devices. The most commonly used hard ferromagnetic materials at present are the ferrites (Iron oxide added to special chemical compounds) and Neodymium, Iron and Boron (NIB) compounds. Alnico steel (alloy of Iron, Nickel and Cobalt with a certain amount of Aluminum), is still used in some applications such as the magnetic compass needle.

- soft ferromagnetic materials. One can consider these materials as having a reversible response to the magnetic field environment; i.e. when the magnetic field is reduced to nothing, their

magnetization returns to nothing regardless of the magnetic field to which they have been submitted. This category can subdivide in 2 subsets:

- isotropic soft ferromagnetic (called grains not oriented). Their susceptibility is the same for all directions.

- anisotropic soft ferromagnetic (called grains oriented). Their susceptibility presents different values with respect to the crystal structure orientation.

The calculation of the magnetic field of a device made of magnetic material proceeds with a complex methodology. We have to take into account all the factors and, moreover, their relations are most often very unlinear. The basic methodology is developed further in this chapter.

The next 3 chapters are devoted to the implementation of the method:

In Chapter 7, we will start with the simplest situation: the case of a bar of cylindrical form made of permanent magnetization material and without any magnetic environment.

In Chapter 8, we will go to more complex situations involving materials with variable magnetization of much more complex geometric shape.

In Chapter 9, we will deal with more advanced situations.

■ 6-2 The Bases: Definitions and Relations

The origin of magnetization of matter is directly related to the electrons of the atomic elements of which the substance in question is made. For common materials, electron orbital motion, and for ferromagnetic materials electron spin are taken into account. Each electron can be considered as a tiny current loop (i.e. a dipole) producing a very small magnetic field. But in any device, even a device of small dimensions, the electrons are very numerous and their magnetic effect can become important due to a collective behavior. When the magnetic field of the environment is null, the motion of the electrons is randomly distributed. The resulting field generated by the device is also null. But if an external field is applied, the dipole may have 2 types of reactions:

1. a classical electromagnetic reaction according to Lenz's law (flux conservation)

2. an alignment of the elementary dipoles on the external magnetic field.

For diamagnetic materials, the first effect is predominant whereas for paramagnetic and all the more for ferromagnetic materials, the second effect is the most important. In the case of ferromagnetic materials, the crystal structure of the substances plays a very important role. The tiny dipoles interact with each other. This process initiates a collective behavior for a small volume of substance called the magnetic domain. In any particular domain, the tiny dipoles are all aligned in such a way that, *de facto*, their collective behavior generates a large dipole of the size of the domain. The various domains of the substance are separated by a thin wall which depends on the internal state of the matter. Chemical composition, atomic impurities, internal mechanical stress, dislocations are the key factors determining for domain size and domain wall behavior.

We will now use this basic physical concepts for magnetic field calculation starting with the definition of the various terms employed and continuing with the basic formulas.

- **Magnetization** or **Intensity of Magnetization** (acronym **M**). This is the volume density of the magnetic dipoles

- **Magnetic domain**: Actual elementary volume of the device with its own specific

Magnetization. Each domain can have different magnetization. The dimensions of the physical domain

can be a fraction of micrometer, as in film used for magnetic recording, up to several microns in high magnetic permeability materials.

- **Magnetic cell**. This is an abstract notion that extends the notion of domain for calculating purposes and represents the elementary volume of the device (i.e. its own specific Magnetization). The cell concept is paramount in the algorithm of field calculation.

- **Magnetizing field** or **Magnetic field intensity** (acronym **Hm** or **H**): The external magnetic field applied to a particular domain (and, by extension, to a cell). It must be pointed out that this proper definition implies that any field external to the domain (or cell) must be accounted for. Consequently, from the viewpoint of calculation, we have to distinguish the following 2 cases:

1. The device is made of only 1 cell. In that case, Hm is the external field to the device.

2. The device is made of several cells. In that case, Hm includes, for each cell, both the external field to the device and the magnetic field contribution of all other cells.

The first case is mathematically quite simple but physically very approximate. Most books on electro-magnetism deal only with this approach.

The second case is much more difficult to manipulate, but physically more accurate.

- **Discretization** or **meshing**: This is the measurement of the division of a device into cells. The higher the meshing (and the number of cells), the better is the simulation of the magnetic behavior of the device (and the field calculation). In *Magnetica*, the discretization is automatic but is parametrized with 4 levels of meshing.

  - **level 1** very minimum meshing → crude approximation, for preliminary tests,

  - **level 2** minimum meshing → first order approximation for fast calculation, (default case)

  - **level 3** moderate meshing → second order approximation, good calculation time,

  - **level 4** extensive meshing → fair order approximation, but major memory requirements and long calculation time.

- **Induction** or **Magnetic flux density** or **global Magnetic field (**acronym **B**). This is simply the sum of the magnetization M and the Magnetizing field Hm. Assuming a coherent system of magnetic units, we have :

    $B = M + \text{Hm} \quad (1)$

- **Law of magnetization**: This relationship between the magnetization M and the magnetizing field Hm is part of solid state physics. It is specific for each substance. Practically, the law is illustrated graphically with a plot of the curve of magnetization that has been measured experimentally. This relationship can be expressed in different ways:

  - M = f (Hm)

    $M = \chi.\text{Hm} \quad (2)$

    where χ is the **relative volume susceptibility.**

  - B = f (Hm)

    $B = \mu.\text{Hm} \quad (3)$

    where μ is the **relative volume permeability**

Both μ and χ are dimensionless parameters when B, M and Hm are given in a coherent system of magnetic units.

Considering equations (1, 2, 3), μ and χ are then very simply related. We have:

$$\mu = 1 + \chi \qquad (4)$$

In scientific books, commercial notices or handbooks, several other definitions are used for μ and χ. They can be expressed in different units such as Henry/m or they also can be referred to the mass of the matter and not to the volume. The transformation of these data into the relative volume values is only a problem of proper numerical conversion. The factor μ_v, the permeability of the vacuum is involved in the first case in association with the specific weight of the matter in the second case.

- **Geometry factor**: This concept although simplistic is the most difficult to formalize. It will be fully developed in the paragraph 6.4. The effect of the geometry of the cell is materialized by a numerical term called the **Demagnetization coefficient** $\mathfrak{D}$. This coefficient can be calculated directly using the Maxwell equations. $\mathfrak{D}$ represents the relative reduction of the magnetic flux density B due the size of the cell.

$$\mathfrak{D} = 1 - \frac{B}{M} \qquad (5)$$

If we consider a very long bar, B tends towards M and, therefore, the demagnetization coefficient tends towards zero. Reciprocally, for a very short bar, a flat pancake or a disk, B tends towards 0 and, consequently, D tends towards 1.

Considering the cell as a volume entity, the notion of demagnetization coefficient must be extended to all directions in space. As the result, the demagnetization coefficient is a vector. It can be shown that sum of the vector components is the unity. This is an asymptotic relation that is perfectly applicable to cuboid 3D cells and to cylindrical cells only when the cells are distant from the axis.

$$\sum \mathfrak{D}i \text{ (i space component)} \rightarrow 1 \qquad (6)$$

The demagnetization coefficient concept is a very important factor in the methodology of field calculation for magnetic devices.

■ 6-3 Magnetic Dipole, Magnetization and Surface Currents

The elementary magnetic dipole can be simulated by a current loop of electrons rotating on a circular orbit of a very tiny radius Rd as shown in next figure.

the elementary magnetic DIPOLE
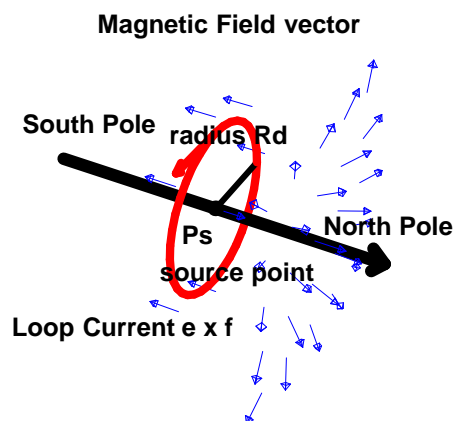
The cylindrical bar can be seen as a set of magnetic domains (or cells) made of a huge number of elementary dipoles. In the case of axial rigid magnetization, each of the domains (or cells) is identical and aligned on the axial direction as shown in the next figure. In other types of magnetization, the domains (or cells) are not identical, neither in direction nor in magnetization intensity.

cylindrical BAR MAGNET
uniform axial MAGNETIZATION
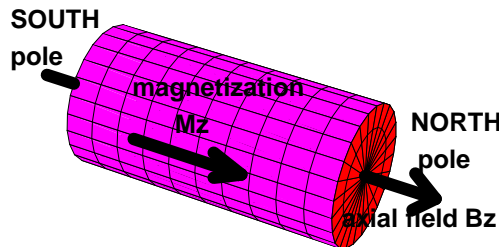
The representation of this physical model is illustrated in the next figure showing the bar cutaway.

Axial magnetization bar magnet – cutaway
elementary domaines (cells)
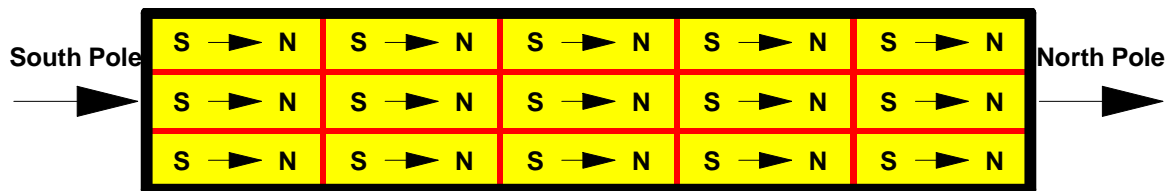
Assuming uniform axial magnetization, the dipole volume density is constant for any cross section of the bar. The global outcome of the elementary dipoles is a surface current of axial current density J_φ. This linear current density J_φ is directly related to the Magnetization M through the Maxwell equations and then the equivalent number of Amperes x Turns can be calculated from the length bf of the bar:

$$J_\varphi = \frac{M}{\mu_v} \tag{7}$$

$$\mathcal{N}at = J_\varphi \text{ x bf } = \frac{M}{\mu_v} \text{ x bf} \tag{8}$$

The next 2 figures allow visualizing the concept of surface current.

    - cross-section of the cylindrical bar

Axial magnetization bar magnet cross section
elementary dipoles and gobal surface current

Jϕ global   surface current

elementery dipoles
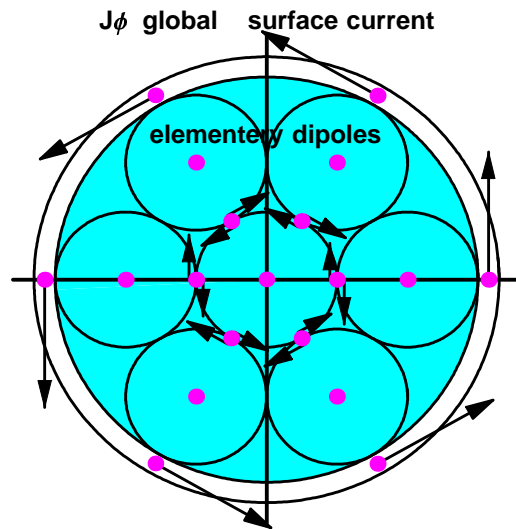
- axial cutaway of the cylindrical bar. The surface currents are a theoretical current sheet of null
    thickness.

Axial magnetization bar magnet – cutaway
schematic of surface currents

Jϕ azimuthal surface currents (forward)

Mz axial magnetization

Jϕ azimuthal surface currents (backward)
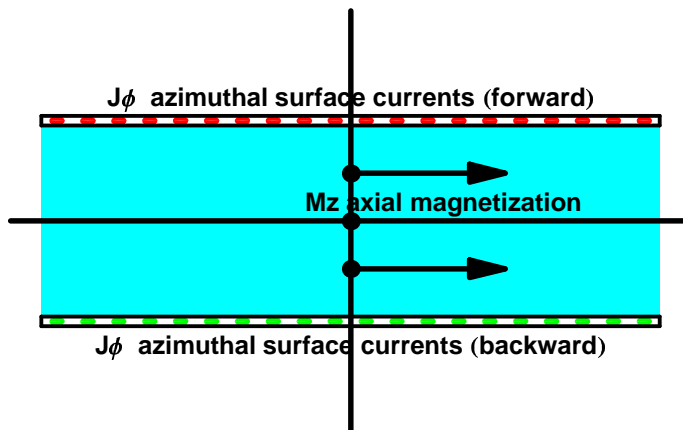
The result of this physical model of the cylindrical bar is a solenoid of the same dimensions made of a very thin winding
(current sheet) with the same linear current density J_φ (and amperes x turns). as shown in the next figure.

thin SOLENOID equivalent to magnetic bar
  (of rigid axial magnetization)
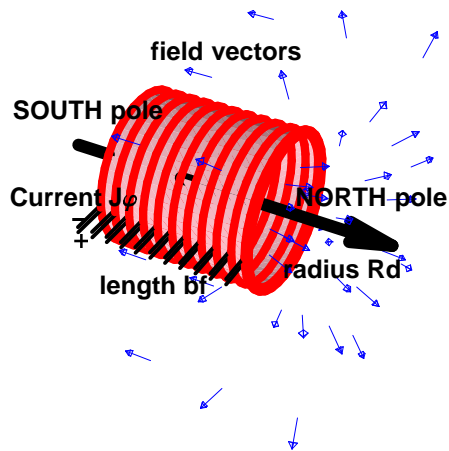
■ 6-4 Cell Geometry and Demagnetization Coefficient

The concept of Magnetic Flux induction B at the center of a cylindrical bar of a permanent magnet material, equation 1, is a pure hypothetical concept. It is similar to the famous Richard Feyman Brick problem. I quote:
"What is the middle plane of concrete brick? Very simple answer! Just cut the brick into 2 parts. But you still do not know what is inside the brick since what you observe is again the surface of the bar. Simply a new surface.", end of quote.

It is the same thing with the magnetic field of a bar. What we can observe and measure, is the field at the surface of a pole. From the solution of the Maxwell equations and the equivalent current sheet solenoid model, it can be shown that the field at the center of the surface is just half the strength of the magnetic flux at the center of the magnet. Then, if the 2 bar magnets are set very close to eachother, the north pole of the first faces the south of the second, the total field inside the gap, being the summation of the contributions, is the magnetic flux at the center of each magnet. The next figure shows the practical realization of the situation with a single magnet bent on a circular path with a small gap between the 2 poles.

**C shape bar magnet
field in the gap**
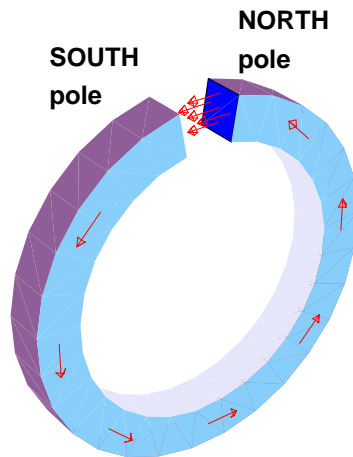
The next figure shows the magnetic flux line at the center of the cylindrical bar.

**Bar magnet
Induction Field line
in the central region**
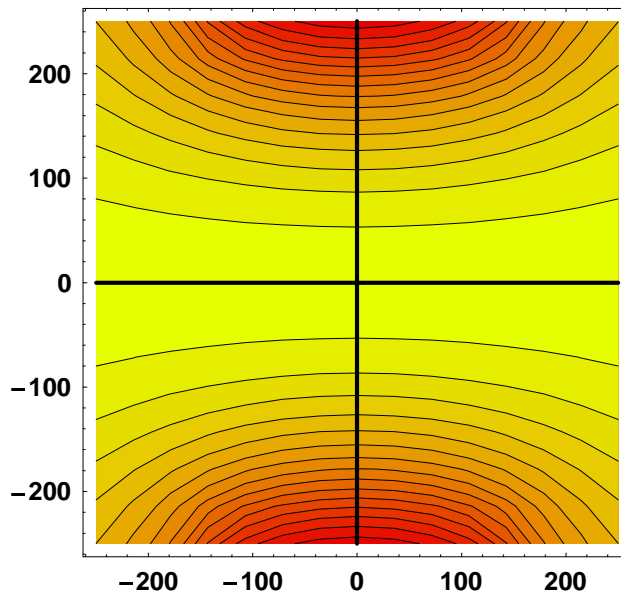
The figure shows clearly that at the center of an infinite bar, the field lines are strictly parallel to the axis. This implies that the only real field component is the axial component Bz and, therefore, the radial component Br is null. Consequently, the application of equation 5 gives with Bz = M:

$$\mathcal{D}z \; = \; 0 \qquad\qquad (9)$$

$$\mathcal{D}r \; = \; 1 \qquad\qquad (10)$$

When the cell is not on the axis, Br is not null and the 2 demagnetization components have numerical values between 0 and 1.

$$\mathcal{D}z = 1 - \frac{Bz}{M} \qquad\qquad (11)$$

$$\mathcal{D}r = 1 - \frac{Br}{M} \qquad\qquad (12)$$

■ 6-5 Functional Magnetization

The solution of Magnetostatics for magnetic systems with magnetic components requires solving simultaneously the mathematical formulas related to the 2 following independent physical concepts:

1. The Maxwell equations which can be seen as a geometrical relation between the sources of the field and the cell target point where the magnetization and the proper magnetizing field must be calculated. The term "sources" means all magnetic field generator components or sub-components such as the cells external to the target cell. The law of magnetic field composition implies that the global magnetic field B of the target cell is simply the linear vector summation of all magnetic source contributions.

   Bg = $\sum$ Be (elementary source) (13)

2. The law of magnetization of the materials, equation (2), applies to the specific material of the cells (see appendix B.1).

Further, since Magnetostatics implies time-independent situations, we assume that the material is in perfect equilibrium with its local environment at the cell level. This hypothesis is very acceptable even at higher frequencies if the magnetic system and the electrical circuits are constructed in such a way as to minimize the eddy currents and the conductor surface currents.

The type of solution depends on both the geometrical magnetic interactions and the law of magnetization.
The simplest solution occurs for a magnetic system simulated with a single cell. Then the geometrical field relation is simply dependent on the magnetic field produced, in the cell, by all the external generators (called **excitation field**). It should be pointed out that most education textbooks use implicitly this elementary model.
When the magnetic system is simulated with multiple cells, it is necessary for each cell, to complement the external field by the interaction effect of every other cell. This leads to an interaction matrix source / target cells.

The law of magnetization can be developed with the following models:

- **rigid magnetization:** The model applies only to the hard ferromagnetic materials. Then functional magnetization is just the intrinsic remanent magnetization of the substance.

- **linear magnetization:** The model implies a constant susceptibility, the magnetization being proportional to the magnetizing field. This model applies to all common materials whether diamagnetic or paramagnetic, and also to hard feromagnetics assuming they are operated within their safe range (see Appendix B.1). It is also possible to use it for soft ferromagnetic materials within a restricted range of operation.

- **non linear magnetization:** This model applies essentially to soft ferromagnetics for all ranges of application without restriction.

The simultaneous solution of the 2 equations produces for each cell the functional magnetization as well as the corresponding magnetizing field. Due to the effect of cell interaction, this resulting magnetizing field (called **functional magnetization**) can be very different from the **initial magnetization** resulting directly from the external **excitation field** (through equation 3).

■ Chapter 7: Magnetic Field Calculation for Permanent Magnet

■ 7-1 General Concepts

Hard ferromagnetic materials are characterized by 3 main features:

- Intrinsic **remanent magnetization** (acronym **Mr**) This is the permanent magnetic field generated by the substance when no external (excitation) field is applied. The field is specific to a direction called the **principal direction.** The directions orthogonal to this direction are called **transversal directions**.

- **Coercive field** (acronym **Hcoer**) The coercive field sets the limits of application of the substance as a magnetic field generator. If the magnetizing field, within any point of the substance, becomes lower than Hcoer, then a catastrophic demagnetization process occurs and the substance may partially or even totally lose its initial magnetization. Therefore, it is compulsory that the operating range of a permanent magnet stay above and away from the coercive field.

- **relative susceptibility** The relative susceptibilty of hard ferromagnetic materials is around or smaller than 1 and they appears very constant as long as the applied magnetizing field stays in the normal operating range. It is necessary to distinguish the susceptibility in the principal direction whose meaning is dM/dHm (usually called dynamic susceptibility) and susceptibility in the transverse direction. The transversal susecptibility is greater than the principal susceptibility; their ratio is around 2 for most materials.

The fact that the relative susceptibility of hard ferromagnetic materials is small and sometimes even not well known suggests the use of 2 different approaches for the field calculation. This is realized in *Magnetica* by using two different types of magnetic components:

- neglecting completely the susceptibility: the intrinsic remanent magnetization is an invariant. This model of component is : **AXSRigidMagneticComponent**

- taking into account the susceptibility both principal and transversal. This model of component is : **AXSHardMagneticComponent**

The next 2 sections treat each of these 2 models of components.

■ 7-2 Cylindrical Bar Magnet of Rigid Magnetization Material

■ 7-2-1 Introduction

The figure below shows a typical bar magnet. This is a cylinder of a hard ferromagnetic magnetic material such as any type of commercial ferrite.

cylindrical BAR MAGNET
uniform axial MAGNETIZATION
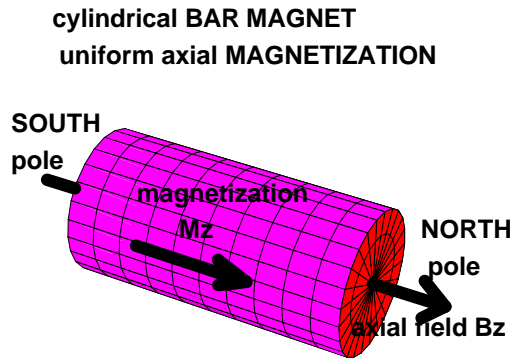
The magnet is characterized by its magnetic poles: the South pole on one side and the North pole on the other side. A magnetic pole is accurately defined as the point at the surface of the substance where the direction of the field is strictly orthogonal to the surface. In any single magnet, there are only 2 points that meet this situation. When the field enters the surface, this is the South pole and when it leaves it, this is the North pole. North and South poles cannot be separated. If the bar is cut into 2 pieces, then we obtain again 2 magnets, each with a North and a South pole. The next figure shows the field line in front of the North pole (for the further examples, see Case 1).

Bar magnet
Induction Field line
at the pole surface
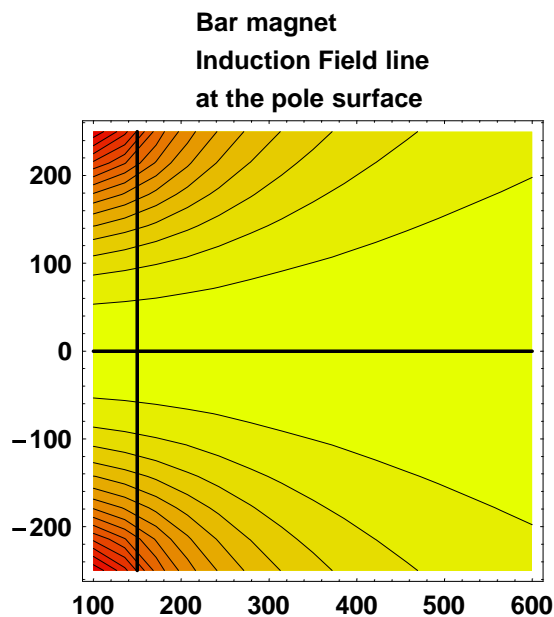
■ 7-2-2 Immediate Help

The *Magnetica* symbolic name for the cylindrical bar made of rigid magnetization material set in a 2D axisymmetrical space is referred to with the function **AXSRigidMagneticComponent**. The data of this component are :

- the material name (chosen among the table of *Magnetica* materials, see B.1).

- the inner diameter : for a bar centered on the axis this is 0.

- the outer diameter

- the magnet length

- the axial position of the center of the bar

- the filling factor of the active hard ferromagnetic substance.

- the angular direction of the magnetization with respect to the axial direction.

The intrinsic remanent magnetization (i.e the rigid value of the magnetization) of the material is implicit according to its name. However, it is possible (optionally) to modify the magnetization value by using the option: **RemanentMagnetization** → value.

> **? AXSRigidMagneticComponent**
>
> ```
> AXSRigidMagneticComponent[material,innerDiameter,outerDiameter,length,position,options]
> builds a Magnetica object which represents a cylindrical magnet (bar, ring),made of material
>    with rigid (permanent and invariant) magnetization in 2D  cylindrical axis symmetric space
>
> Options: RemanentMagnetization, FillingFactor, MagnetizationAngle, Temperature Details…
> ```
>
> **? RemanentMagnetization**
>
> ```
> Option for AXSRigidMagneticComponent and AXSHardMagneticComponent,allows setting the
>    Magnetization strength (if different from the material data) in gauss Details…
> ```
>
> **Options[AXSRigidMagneticComponent]**
>
> ```
> {RemanentMagnetization→ Automatic, FillingFactor→ 1,
>  MagnetizationAngle → 0, Temperature→ Automatic}
> ```

The properties of *Magnetica* materials can be accessed using the command **Properties** with the name of the material as argument. (For detailed account of *Magnetica* materials see Appendix B.1).

> **?Properties**
>
> ```
> Properties[system,options] gives the technical details of a
>    magnetic system. With the option ComponentsList->list or ComponentsList->
>    All, it also gives the properties of the specified list of components of s.
> See ?ComponentsList Details…
> Properties[material,options] displays the properties of the given material.
>    With Temperature->value it gives the information at the specified temperature
> ```

As we have indicated in the previous chapter (see 4.2) data and results can be expressed in any relevant units. The global command **MagneticaUserUnits[...]** sets units with immediate effect and for any further field calculation. It can be overridden for any selective calculation using the same specifications as additional option arguments for the concerned function. At any moment, the units that are taken into account can be accessed with **MagneticaCurrentUnits[]**

> **MagneticaCurrentUnits[]**
>
> ```
> Length                  Milli Meter
> Magnetic field          Gauss
> Magnetic potential      Micro Weber/Meter
> Magnetic flux           Micro Weber
> Magnetic field gradient Gauss/Meter
> Inductance              Micro Henry
> Magnetic energy         Joule
> Magnetic moment         Micro Ampere Meter²
> Current intensity       Ampere
> Power                   Watt
> Voltage                 Volt
> Electrical Resistance   Ohm
> ```

The technical specification of a system can be accessed with the command `Properties` this time using the name of the magnetic system as argument.

```
?Properties
```

```
Properties[system,options] gives the technical details of a
  magnetic system. With the option ComponentsList->list or ComponentsList->
  All, it also gives the properties of the specified list of components of s.
See ?ComponentsList Details…
Properties[material,options] displays the properties of the given material.
  With Temperature->value it gives the information at the specified temperature
```

```
?ComponentsList
```

```
ComponentsList is an option used to define a sublist of components of a given
  system. One can gives only a list of numbers, like ComponentsList->{1,3,6},
  or a list of numbers and lists representing intervals. Example ComponentsList->
  {1,2,{7,10},12} is equivalent to ComponentsList->{1,2,7,8,9,10,12} Details…
```

■ 7-2-3 Material Properties

We will use a typical ferrite as the hard ferromagnetic material for the example cases. The physical and magnetic data of this ferrite (*Magnetica* ref : **Ferrit2**) are obtained using the command **Properties** with the name as argument. The main features are :

- intrinsic remanent magnetization   **Mr**   : 4200 Gauss (0.42 Tesla, 334200 Amp/m)

- coercive field   **Hcoer**   :-3200 Gauss (0.32 Tesla, 254600 Amp/m)

N.B. these are nominal values. In practice, for industrial products a dispersion of ±3.5 per cent is standard.

- relative susceptibility   principal χ**pr** : 0.10 and transverse χ**tr :** 0.20

```
Properties[Ferrit2]
```

Material properties
```
material name                           Ferrit2
material category                       Permanent magnet
designation                             Ferrite O2 (42/32)
```
General Physical Properties
```
specific weight (g/cm3)                        4.900000
dilation coefficient                           0.000013
    (relative linear change)
resistivity (Ohm x cm)                         0.000250
Coefficient temperature resistivity            0.001000
```
Magnetic Physical Properties
```
intrinsic remanent magnetic field (G)       4200.000000
intrinsic coercive field (G)               -3200.000000
principal dynamic relative susceptibility      0.100000
secondary  relative susceptibility             0.200000
temperature coefficient susceptibility         0.000000
temperature coefficient remanent field        -0.002000
temperature coefficient coercive field         0.004000
```

The magnetization curves can be shown using the command **MaterialMagnetizationCurves** followed by the name as argument.

```
MaterialMagnetizationCurves[Ferrit2,{Mprincipal,Mtransversal}]
```

```
Material Magnetization Curve
Ferrit2  temperature  20°celsius
Ferrite O2 (42/32)
M Magnetization Principal  f(Hm)
```



```
Material Magnetization Curve
Ferrit2  temperature  20°celsius
Ferrite O2 (42/32)
M Magnetization Transversal  f(Hm)
```
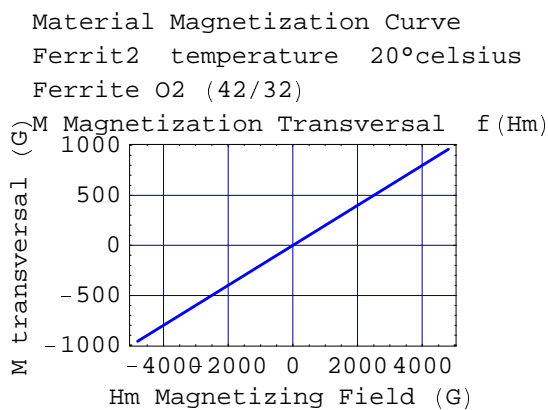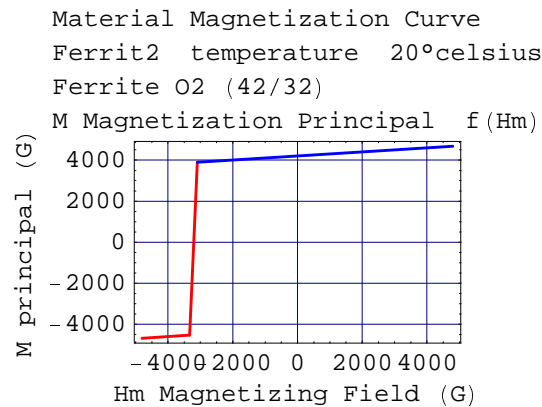

■ 7-2-4 Entering Data

The examples are based on rigid model of a cylindrical bar of 20 mm in diameter made of ferrite (reference Ferrit2), the magnetization being oriented along the axis.

We will illustrate the examples with the 4 following cases :

   1. a long bar (80 mm) :

```
barmagnet1 = AXSMagneticSystem[
   {AXSRigidMagneticComponent[Ferrit2, 0, 20, 80, 0]},
   SystemLabel -> "Ferrit2 long bar magnet, (rigid magnetization)"];
```

   2. a short bar, pellet like (10 mm) :

```
barmagnet2 = AXSMagneticSystem[
   {AXSRigidMagneticComponent[Ferrit2, 0, 20, 10, 0]},
   SystemLabel -> "Ferrit short bar magnet, (rigid magnetization)"];
```

   3. the same long bar (80 mm), but setting the intrinsic remanent magnetization at the arbitrary value of
      4000 Gauss:

```
barmagnet3 = AXSMagneticSystem[
   {AXSRigidMagneticComponent[Ferrit2, 0, 20, 80, 0,
     RemanentMagnetization -> 4000]},
   SystemLabel -> "Ferrit long bar magnet, (rigid magnetization at 4000 g)"];
```

   4. the current sheet solenoid equivalent to the long cylindrical bar of case 1. The application of
      equation (6- 8) gives a number of amper*turns of 26738.030 which can be made with 1000 turns of
      0.01 mm wire carrying 26.73803 amperes.

N.B. these values are not realistic but just to illustrate the relations:

```
ampbar1 = 26.7380304394;
solequivbar1 = AXSMagneticSystem[
    {AXSSolenoid[CopperOFHC, 19.99, 20.01, 80, 0, 1000, ampbar1, FillingFactor → 0.7]},
    SystemLabel -> "solenoid equivalent long bar case 1"];
```

■ 7-2-5 System Drawing and Technical Specifications

Drawing of the long bar; case 1

```
DrawSystem2D[barmagnet1, ShowArrows → True];
```

System name: **barmagnet1**
Ferrit2 long bar magnet, (rigid magnetization)
2D view ZR
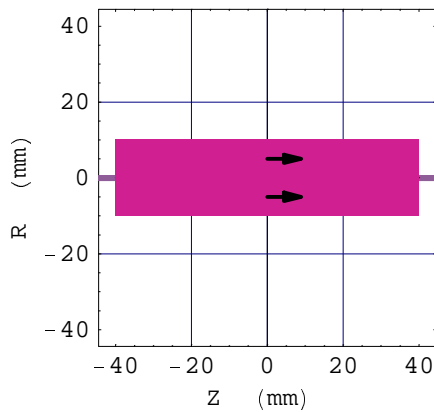
Drawing of the solenoid equivalent to the long bar; case 4

```
DrawSystem2D[solequivbar1];
```

System name: **solequivbar1**
solenoid equivalent long bar case 1
2D view ZR
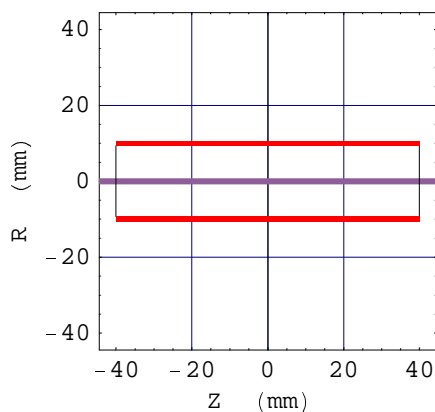
Drawing of the short bar; Case 2

DrawSystem2D[barmagnet2, ShowArrows → True];

System name: **barmagnet2**
Ferrit short bar magnet, (rigid
magnetization)
2D view ZR
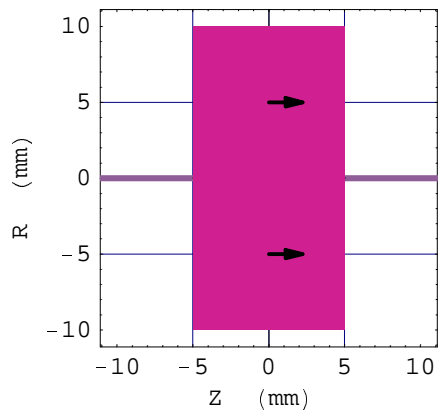
Technical specifications of the long bar Case 1:

```
Properties[barmagnet1,ComponentsList->All]
```

General properties

System name:	**barmagnet1**
System magnetic parity	True
System Legendre radius	20.6155
Magnetic field at the origin (G)	{0, 4074.6}
Field magnitude at the origin(G)	4074.6
System mass (Kg)	0.12315
System nominal power (W)	0
System effective power(W)	0
Number of components	1
Number of components with rigid magnetization	1

Component number 1

Type of component	cylindrical rigid magnet

Geometry of the cylindrical
magnetic structure

Nominal inner diameter	(mm)	0.00000
Effective inner diameter	(mm)	0.00000
Nominal outer diameter	(mm)	20.00000
Effective outer diameter	(mm)	20.00000
Nominal mean diameter	(mm)	10.00000
Nominal axial length	(mm)	80.00000
Effective axial length	(mm)	80.00000

Magnetic specifications
of the structure

Material		Ferrite O2 (42/32)
Material temperature (°Celsius)		20.00000

Intrinsic and effective
magnetic properties

Remanent magnetic field (G)	4200.00000
Coercive magnetic field (G)	-3200.00000
Dynamic principal susceptibility	0.10000
Secondary susceptibility	0.20000
Angle of magnetization(degrees)	0.00000
Axial magnetization (G)	4200.00000
Radial magnetization (G)	0.00000
Effective remanent magnet. field(G)	4200.00000
Effective coercive field(G)	-3200.00000

Technical specifications
of the structure

Nominal resistivity (Ω·cm)	0.00025
Material mass density (g/cm^3)	4.90000
Filling factor of winding (%)	100.00000
Mass of the material (kg)	0.12315
Nominal magnet resistance (Ω)	0.00010

Position of the structure
(in the referential space system)

Structure center coord.Z (mm)	0.00000
Effective inner radius (mm)	0.00000
Effective outer radius (mm)	10.00000
Effective lower Z extremum (mm)	-40.00000
Effective higher Z extremum (mm)	40.00000

Technical specifications of the solenoid equivalent to the long bar case 4:

`Properties[solequivbar1,ComponentsList->All]`

General properties

System name:	**solequivbar1**
System magnetic parity	True
System Legendre radius	25.6136
Magnetic field at the origin (G)	{0, 4074.6}
Field magnitude at the origin(G)	4074.6
System mass (Kg)	0.0000392322
System nominal power (W)	1.10631×10^7
System effective power(W)	1.10631×10^7
Number of components	1
Number of current carrying component	1

Component number 1

Type of component:	cylindrical solenoid

Geometry of the cylindrical current solenoid

Nominal inner diameter (mm)	19.99000
Effective inner diameter (mm)	19.99000
Nominal outer diameter (mm)	20.01000
Effective outer diameter (mm)	20.01000
Nominal mean diameter (mm)	20.00000
Nominal axial length (mm)	80.00000
Effective axial length (mm)	80.00000

Winding specifications

Material:high conductivity OFHC Copper

Winding temperature (°C)	20.00000
Nominal resistivity (Ω·cm)	1.72400e-6
Effective resistivity	1.72400e-6
Nominal volume relative susceptibility	-6.10000e-7
Effective susceptibility	-6.10000e-7
Material mass density (g/cm^3)	8.92000
Filling factor of winding (%)	70.00000
Mass of the material (kg)	0.00004

Position of the solenoid (in the referential space system)

Solenoid center coord.Z (mm)	0.00000
Effective inner radius (mm)	9.99500
Effective outer radius (mm)	10.00500
Effective lower Z extremum (mm)	-40.00000
Effective higher Z extremum (mm)	40.00000

Electrical specifications of the solenoid

Number of turns (in series)	1000.00000
Direction according the right hand rule :	counterclockwise
Direction of the turns	+
Type of curent density	constant
Current (per turns) (A)	26.73803
Total winding current (A·turns)	26738.03044
Conductor current density (A/mm^2)	381971.86342
Nominal solenoid resistance (Ω)	15474.58781
Effective solenoid resist. (Ω)	15474.58781
Nominal voltage drop (V)	413760.00000
Effective voltage drop (V)	413760.00000
Nominal solenoid power (W)	1.10631e7

```
Effective solenoid power (W)                           1.10631e7
Conductor dimensions
Conductor geometry                          circular
Effective conductor diameter (mm)                      0.00944
Nominal conductor length (mm)                      62831.85307
Effective conductor length (mm)                    62831.85307
```

■ 7-2-6 Punctual Field Calculation

The magnetic field calculation proceeds exactly as for current carrying devices. All the field calculation functions can be applied without any restriction (see Chapter 4.4, 4.5 and 4.6). We will emphasize some special features of the cylindrical bar magnet.

  1. Field on the axis at the isocenter point (R=0,Z=0) of the magnet.

  There is a question of denomination of this physical data. In term of the common practice, it should be called magnetic induction. We have seen in Chapter 6 on the fundamentals of ferromagnetism, the various denominations and the related concepts. In any case, whatever the denomination, the magnetic field at the isocenter refers to a physical datum which may be considered, for all materials, as the global magnetic field at the center at bar the center. It can be measured by cutting the magnet into 2 pieces leaving a very small gap in between and taking field measurement in the gap. We can notice that the isocenter induction field (4074.6 Gauss) is smaller than the intrinsic remanent magnetization (4200 gauss) in agreement with the concept of demagnetization coefficient (see paragraph 6.4) .

  FieldPoint[barmagnet1, {0, 0}]

  {0, 4074.6}

The isocenter field of the equivalent current sheet solenoid, case 4, appears exactly the same as the isocenter induction field for the long bar, Case 1.

  FieldPoint[solequivbar1, {0, 0}]

  {0, 4074.6}

The induction field isocenter is proportional to the intrinsic remanent magnetization as shown by the comparison between Cases 1 and 3 for the long bar respectively 4200 and 4000 for M and 4074.6 and 3880.57 for B:

  FieldPoint[barmagnet3, {0, 0}]

  {0, 3880.57}

The induction field isocenter depends strongly on the bar length (all the other factors being identical): respectively 4074.6 and 1878.3 gauss for the long and the short magnet, Cases 1 and 2.

  FieldPoint[barmagnet2, {0, 0}]

  {0, 1878.3}

The difference of the induction field isocenter with the intrinsic remanent field respectively 125.6 and 2327.7 gauss, expresses the adverse effect of reducing the length of the bar. The **Dz demagnetization coefficients** calculated by equation 6.8 for the long bar and the short bar are respectively: 0.0299 and 0.5542.

  2. Magnetic field at the surface of the pole.

  The following values show that the field of a single pole is always smaller than half the intrinsic

remanent magnetization (i.e. 2100 Gauss), and that the discrepancy increases the shorter the device is.

For the long bar Case 1, it is 2084 Gauss:

```
precis = 12;
NumberForm[FieldPoint[barmagnet1, {0, 40}], precis]
```

```
{0, 2083.7835411}
```

For the short bar Case 2, it is:

```
precis = 12;
NumberForm[FieldPoint[barmagnet2, {0, 5}], precis]
```

```
{0, 1484.92424049}
```

■ 7-2-7 Legendre Source Coefficients

The calculation of the Legendre Source Coefficients proceeds the same way as for current carrying components.

1. The drawing of the region's limits for the long bar, case 1, shows the non existence of the central region since the Legendre point is inside the bar.

```
DrawSHLimits[barmagnet1]
```

```
System name: barmagnet1
Ferrit2 long bar magnet, (rigid
magnetization)
2D view ZR
```
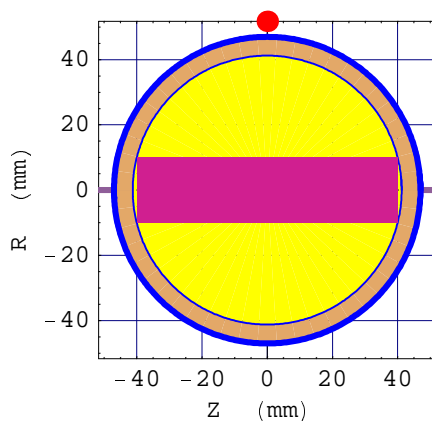


```
- Graphics -
```

2. The following table shows indeed that the central source coefficients are null whereas the remote coefficients extending up to infinity are real numbers. Remote coefficients are always calculable.

```
VerboseLegendreCoefficients[barmagnet1, LegendreRadius -> {100,100}]

System Name:                          barmagnet1
System Legendre radius                 20.616 mm
Printout Legendre radius central      100.000 mm
Printout Legendre radius remote       100.000 mm
System Legendre center position         0.000 mm
Inner spherical radius limit             0.000 mm
Outer spherical radius limit            47.003 mm
System magnetic parity                    True
System magnetic constant field           0.000 G
Mode: absolute Field (Gauss)


Order           central coeff.         remote coeff.
D0                0.000000000           0.000000000
D1                0.000000000           0.000000000
D2                0.000000000          16.800000000
D3                0.000000000           0.000000000
D4                0.000000000           5.124000000
D5                0.000000000           0.000000000
D6                0.000000000           1.091790000
D7                0.000000000           0.000000000
D8                0.000000000           0.189601650
D9                0.000000000           0.000000000
D10               0.000000000           0.027383573
D11               0.000000000           0.000000000
D12               0.000000000           0.003097785
D13               0.000000000           0.000000000
D14               0.000000000           0.000194743
D15               0.000000000           0.000000000
D16               0.000000000          -0.000023209
D17               0.000000000           0.000000000
D18               0.000000000          -0.000011651
D19               0.000000000           0.000000000
D20               0.000000000          -2.738973900e-6
```

■ 7-2-8 Field Map Graphics

The magnetic field mapping can be also executed without restriction just as for current carrying devices (see Chapter 4).

1. Field map of the long bar, case 1, on a radius line in front of the North pole for the magnetic field vector Bz (axial), Br (radial) and Bm. The Bz vector stays practically constant in front of the pole but falls off drastically after. The Br vector is zero on the axis and at a maximun at the edge of the pole.

N.B. This command implies a so-called continuous plot, with the default number of points, i.e. 25. But this is a minimum and for the field Br, which varies quickly at the corner of the cylinder, the number of calculated points is much larger.

```
FieldMap[barmagnet1, {Bz, Br, Bm}, Line[{-30, 40}, {30, 40}]];
```

System name: **barmagnet1**

Ferrit2 long bar magnet, (*rigid magnetization*)

Bz (G) (Continuous, Automatic Mode) between {-30., 40.} and {30., 40.}
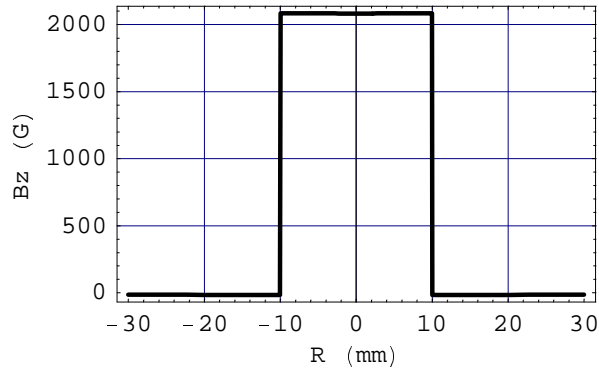
System name: **barmagnet1**

Ferrit2 long bar magnet, (*rigid magnetization*)

Br (G) (Continuous, Automatic Mode) between {-30., 40.} and {30., 40.}
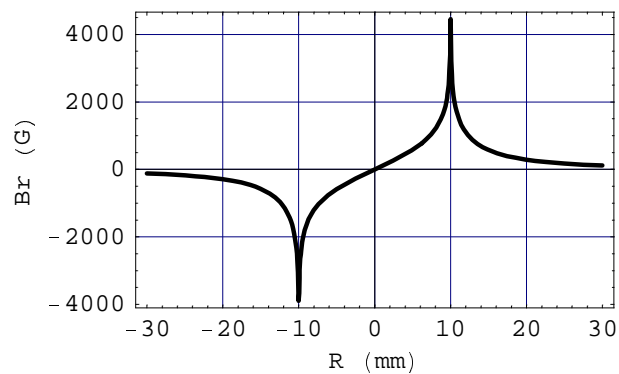
System name: **barmagnet1**

Ferrit2 long bar magnet, (*rigid magnetization*)

Bm (G) (Continuous, Automatic Mode) between {-30., 40.} and {30., 40.}
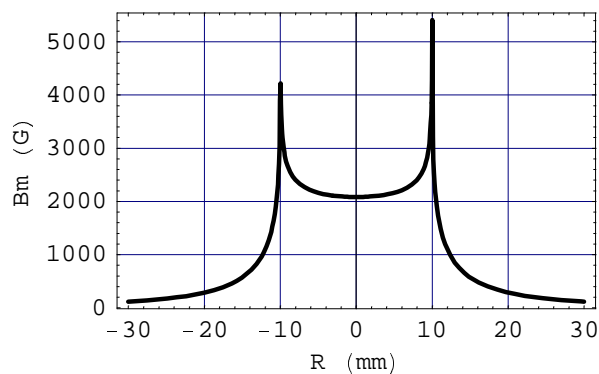
2. Field map plot exactly as in the previous plot but using the **FieldMapPoints** command.

N.B. The next plot implies using the default number of points, i.e. 25. This carries out the calculation at exactly 25 points equally distributed. This way of plotting cuts the highest peak values at the corner of the cylinder, giving a graphic representation with a very different look.

```
FieldMapPoints[barmagnet1, {Bz, Br, Bm}, Line[{-30, 40}, {30, 40}]];
```

System name: **barmagnet1**
Ferrit2 long bar magnet, (rigid magnetization)
Bz (G) (Discrete, Automatic Mode)
between {-30., 40.} and {30., 40.}
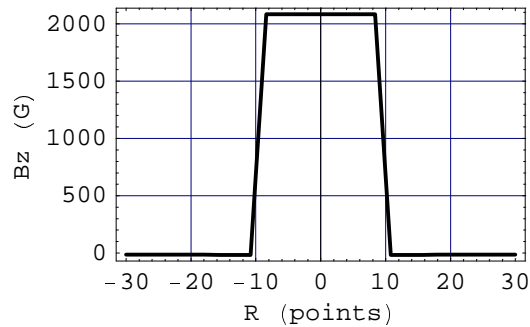
System name: **barmagnet1**
Ferrit2 long bar magnet, (rigid magnetization)
Br (G) (Discrete, Automatic Mode)
between {-30., 40.} and {30., 40.}
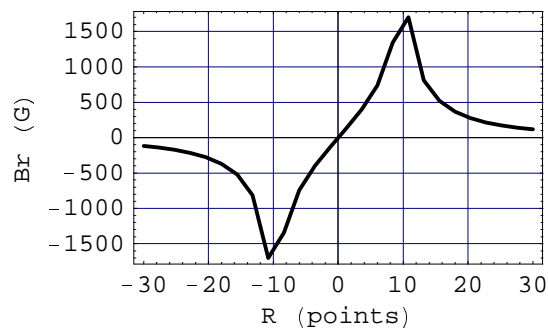
System name: **barmagnet1**
Ferrit2 long bar magnet, (rigid magnetization)
Bm (G) (Discrete, Automatic Mode)
between {-30., 40.} and {30., 40.}
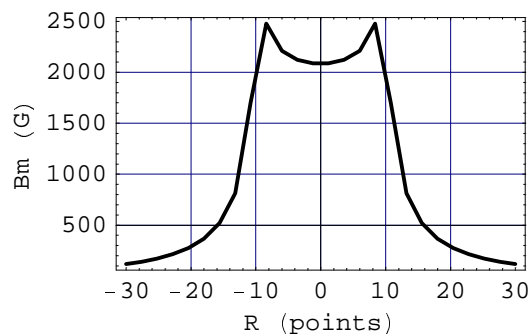
3. Field line map of the long bar, Case 1, on a plane through the axis, using extended graphics drawing parameter definition (25 x 25 points).

```
ContourFieldMap[barmagnet1, {Flux}, Plane[{-30, -50}, {30, 50}], NumberOfPoints → 25];
```

System name: **barmagnet1**

Ferrit2 long bar magnet, (*rigid magnetization*)

Flux (μWb) (Continuous, Automatic Mode)

between {-30., -50.} and {30., 50.}
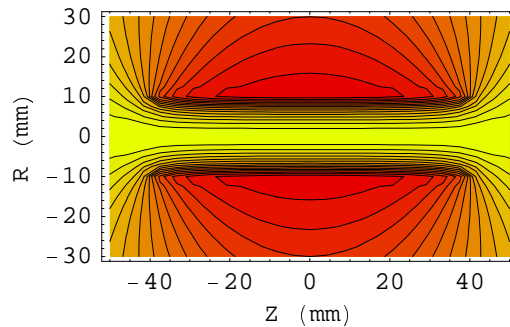
4. Same field line map but for the short bar, Case 2, on a plane through the axis.

```
ContourFieldMap[barmagnet2, {Flux}, Plane[{-30, -50}, {30, 50}], NumberOfPoints → 25];
```

System name: **barmagnet2**

Ferrit short bar magnet, (*rigid magnetization*)

Flux (μWb) (Continuous, Automatic Mode)

between {-30., -50.} and {30., 50.}
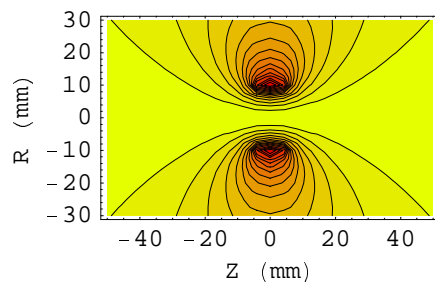
5. Stray magnetic field line map of the long bar, Case 1, on a plane through the axis.

```
DrawIsoFieldLines[barmagnet1, {30, 20, 10, 5, 1},
  AngularRange → {0, 360}, NumberOfPoints → 40, FieldUnit → Gauss];
```

System name: **barmagnet1**

Ferrit2 long bar magnet, (*rigid magnetization*)

Magnetic iso Field lines

Field magnitude (G) {30., 20., 10., 5., 1.}

2D view ZR
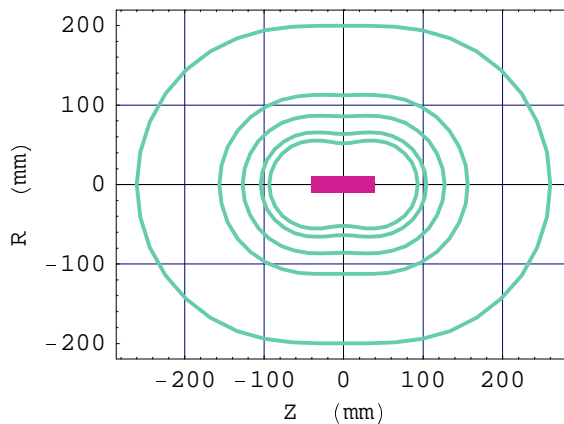
■ 7-3 Cylindrical Bar of Linear Magnetization Material

■ 7-3-1 Introduction

In order to take into account the linear variation of the magnetization with the applied magnetizing field, it is necessary, as indicated in the previous section, to subdivide the components into several small cells each with a uniform magnetization. Even if no excitation field is applied, when the magnet is set alone, each cell will be influenced by the interaction of all the other cells. This implies the use of a new axisymmetric component: **AXSHardMagnetic-Component.** The arguments are exactly the same as for the rigid case. However, the process of validation is very different since the component is divided into cells.

■ 7-3-2 Immediate Help

```
?MeshingLevel
```

MeshingLevel is an option for AXSMagneticSystem that
  specifies the granularity of the meshing. (1 -> Sparse, 2 -> intermediate (
  default), 3 -> High density, 4 -> Very high density) Details…

```
? AXSHardMagneticComponent
```

AXSHardMagneticComponent[material,innerDiameter,
  outerDiameter,length,position,fillingFactor,magnetizationAngle,options]
builds a Magnetica object which represents a cylindrical magnet (bar, ring),made of material
  with permanent (rigid or linear) magnetization in 2D cylindrical axis symmetric space

Options: see RemanentMagnetization, Temperature, Rigid, ComponentMeshing Details…

```
? Rigid
```

```
Option for AXSHardMagneticComponent. With Rigid->
  True the magnetization is constant (instead of linear) Details…
```

```
? ComponentMeshing
```

```
ComponentMeshing is an option for AXSHardMagneticComponent or
  AXSSoftMagneticComponent that specifies the meshing of the component.
ComponentMeshing -> Automatic or ComponentMeshing ->
  {nbRadialCells, nbAxialCells}. Details…
```

Functional magnetization can be displayed by using either of the 2 following commands:

```
? MagnetizationTable
```

```
MagnetizationTable[system , opts]
gives the magnetization table of the magnetic
  components (ferro, magnets) ComponentsList -> {components} or {{1,3}}
CellsList -> {1,2,3} or {{1,3}}
FullDetails -> True|False
CondensedForm -> True|False
 Details…
```

```
? MagnetizationCurves
```

```
MagnetizationCurves[system , functions, opts]
 plots the magnetization curves of the magnetic components (ferro, magnets) of the system.
  The available functions are Mr, Mz, Mm, Mp, Mt, Hr, Hz, Hm, Hp, Ht, Br, Bz, Bm, Bp,
  Bt, Susceptibility, SusceptibilityT, Permeability, PermeabilityT. The options are
PlotColumns -> {columns number}
 PlotRows -> {rows number}
ComponentsList -> {components} or {{1,3}}
ShowComponentLimits -> (True|False) Details…
```

■ 7-3-3 Entering Data

The examples are based on the same long cylindrical bar 20 mm in diameter, 80 mm long, made of the same ferrite (reference Ferrit2), the magnetization being oriented along the axis. We will illustrate the examples with the 4 following cases, each set at a different meshing level.

Case 5: meshing level 1 for crude approximation and quick answer

```
barmagnetmesh1 = AXSMagneticSystem[
   {AXSHardMagneticComponent[Ferrit2, 0, 20, 80, 0]},
   SystemLabel → "Ferrit long bar magnet, (meshing level 1)",
   MeshingLevel → 1];
```

Case 6: meshing level 2 for realistic approximation (default meshing)

```
barmagnetmesh2 = AXSMagneticSystem[
   {AXSHardMagneticComponent[Ferrit2, 0, 20, 80, 0]},
   SystemLabel → "Ferrit long bar magnet, (default meshing level)"];
```

Case 7: meshing level 3 for good approximation

```
barmagnetmesh3 = AXSMagneticSystem[
   {AXSHardMagneticComponent[Ferrit2, 0, 20, 80, 0]},
   SystemLabel → "Ferrit long bar magnet, (meshing level 3)", MeshingLevel → 3];
```

Case 8: meshing level 4 for excellent approximation but major memory requirements and long calculation time

```
barmagnetmesh4 = AXSMagneticSystem[
    {AXSHardMagneticComponent[Ferrit2, 0, 20, 80, 0]},
    SystemLabel → "Ferrit long bar magnet, (meshing level 4)",
    MeshingLevel → 4];
```

■ 7-3-4 Technical Specifications

System and component technical data and the permanent magnet component technical data for the case 7 (default meshing). Main feature: number of cells = 36

```
Properties[barmagnetmesh3,ComponentsList->All]
```

General properties

```
System name:                                        barmagnetmesh3
System magnetic parity                              False
System Legendre radius                              20.6155
Magnetic field at the origin (G)                    {0, 4060.33}
Field magnitude at the origin(G)                    4060.33
System mass  (Kg)                                   0.12315
System nominal power (W)                            0
System effective power(W)                           0
Number of components                                1
Number of ferro-magnetic (non rigid) components     1
Number of components with permanent  magnetization  1
```

n°	type	first cell	last cell	radial cells	axial cells
1	permanent	1	36	2	18

Component number 1

```
Type of component                        cylindrical permanent magnet
Automatic meshing
```
Geometry of the component
```
Nominal inner diameter    (mm)                  0.00000
Effective inner diameter  (mm)                  0.00000
Nominal outer diameter    (mm)                 20.00000
Effective outer diameter  (mm)                 20.00000
Nominal mean diameter     (mm)                 10.00000
Nominal axial length      (mm)                 80.00000
Effective axial length    (mm)                 80.00000
```
Magnetic specifications
of the component
```
Material                          Ferrite O2 (42/32)
Material temperature (°Celsius)         20.00000
```
Intrinsic and effective
magnetic properties
```
Magnet magnetization function            linear
Remanent magnetic field (G)            4200.00000
Coercive magnetic field  (G)          -3200.00000
Dynamic principal susceptibility          0.10000
Secondary susceptibility                  0.20000
Angle of magnetization(°C)                0.00000
Axial magnetization  (G)               4200.00000
Radial magnetization  (G)                 0.00000
Effective remanent magnet. field (G)   4200.00000
Effective coercive field(G)           -3200.00000
```
Technical specifications
of the component
```
Nominal resistivity (Ω·cm)                0.00025
Material mass density (g/cm^3)            4.90000
Filling factor of structure (%)         100.00000
Mass of the material (kg)                 0.12315
Nominal magnet resistance (Ω)             0.00010
```
Position of the component
(in the referential space system)
```
Structure  center coord.Z (mm)            0.00000
Effective inner radius (mm)               0.00000
Effective outer radius (mm)              10.00000
Effective lower Z extremum (mm)         -40.00000
Effective higher Z extremum (mm)         40.00000
```

■ 7-3-5 Punctual Field Calculation

1. Magnetic field at the isocenter center of the bar.

- Case 5, meshing level 1 : Bz = 4056.77 Gauss

    FieldPoint[barmagnetmesh1, {0, 0}]

    {0, 4056.77}

- Case 6, meshing level 2 (default): Bz = 4057.11 Gauss

    FieldPoint[barmagnetmesh2, {0, 0}]

    {0, 4057.11}

- Case 7, meshing level 3: Bz = 4060.33 Gauss

    FieldPoint[barmagnetmesh3, {0, 0}]

    {0, 4060.33}

- Case 8, meshing level 4 : Bz = 4060.94 Gauss

    FieldPoint[barmagnetmesh4, {0, 0}]

    {0, 4060.94}

2. Magnetic field in front of the North pole; meshing level 3 : 2000.25 Gauss.

    FieldPoint[barmagnetmesh3, {0, 40}]

    {0, 2000.25}

■ 7-3-6 Legendre Source Coefficients

The calculation of the Legendre Source coefficients proceeds the same way as for the rigid case.

- Case 7: The region's limits are identical with those of the rigid model (see section 6-2-5). The source coefficients are as follows:

```
VerboseLegendreCoefficients[barmagnetmesh3, LegendreRadius → {100, 100}]

System Name:                          barmagnetmesh3
System Legendre radius                  20.616 mm
Printout Legendre radius central       100.000 mm
Printout Legendre radius remote        100.000 mm
System Legendre center position          0.000 mm
Inner spherical radius limit             0.000 mm
Outer spherical radius limit            47.003 mm
System magnetic parity                    False
System magnetic constant field           0.000 G
Mode: absolute Field (Gauss)


Order            central coeff.          remote coeff.
D0                0.000000000             0.000000000
D1                0.000000000             0.000000000
D2                0.000000000            16.636205723
D3                0.000000000             0.000000000
D4                0.000000000             5.017886078
D5                0.000000000             0.000000000
D6                0.000000000             1.058261930
D7                0.000000000             0.000000000
D8                0.000000000             0.181714748
D9                0.000000000             0.000000000
D10               0.000000000             0.025854087
D11               0.000000000             0.000000000
D12               0.000000000             0.002846702
D13               0.000000000             0.000000000
D14               0.000000000             0.000160543
D15               0.000000000             0.000000000
D16               0.000000000            -0.000026667
D17               0.000000000             0.000000000
D18               0.000000000            -0.000011762
D19               0.000000000             0.000000000
D20               0.000000000            -2.679633100e-6
```

■ 7-3-7 Field Map Graphics

1. Field map of the ferrite bar on an axial line crossing the North pole. There is perfect continuity of Bz when the surface of the bar is crossed (at Z = 40 mm).

```
FieldMap[barmagnetmesh3, {Bz}, Line[{0, 30}, {0, 50}]];
```

System name: **barmagnetmesh3**
Ferrit long bar magnet, (*meshing
level 3*)
Bz (G) (Continuous, Automatic Mode)
between {0., 30.} and {0., 50.}
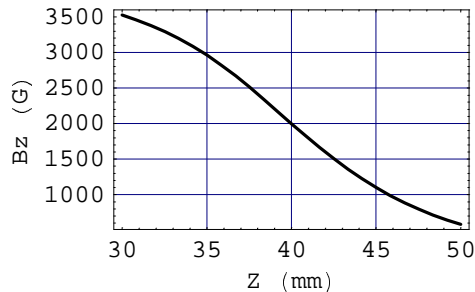
2. Field line map of the ferrite bar on a plane through the axis, using extended graphics parameter definition (25 x 25 points).

```
ContourFieldMap[barmagnetmesh3, {Flux},
   Plane[{-30, -50}, {30, 50}], NumberOfPoints → 25];
```

System name: **barmagnetmesh3**
Ferrit long bar magnet, (*meshing
level 3*)
Flux (μWb) (Continuous, Automatic
Mode)
between {-30., -50.} and {30., 50.}
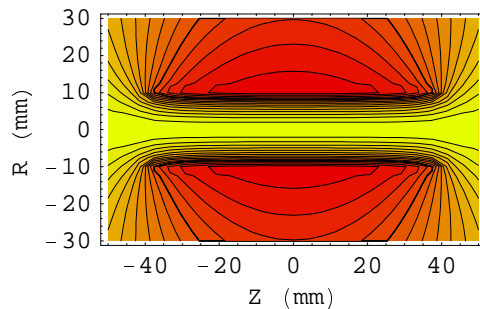
■ 7-3-8 Functional Magnetization (table and curves)

The functional magnetization for a magnetic system is the result of the solution of simultaneous equations coming from Maxwell geometrical relations and the law of magnetization of the substances. Consequently, the functional magnetization of each cell is in perfect magnetic equilibrium with its environment.

The relevant data can be displayed using the commands **MagnetizationTable** for the table of values or **Magnetization-Curves** for graph plots.

MagnetizationTable shows various data for a selected set of the system's cells, general output (and default command) :

- type, dimensions and center cell coordinates
- functional Magnetization intensity M

and when the editing option **FullDetails** is set to True:

- functional Magnetizing field Hm

- induction field B (i.e. global field M + Hm)

  these data are vectors given for both the system coordinates and the material principal magnetization direction

- χ and μ relative dynamic susceptibility and permeability in the principal and in the transversal directions

- the degree of margin between the functional magnetizing field and the coercive field as well as the warning flag (safe / unsafe) if the margin becomes negative.

The option command **CondensedForm** controls the display of the table.

1. Cells 1 to 3 for the system Case 7 (default meshing)

```
MagnetizationTable[barmagnetmesh3,CellsList->{1,2,3},
CondensedForm->True]
```

```
n°      type     first cell   last cell   radial cells   axial cells
1    permanent       1           36             2             18
```

```
Length unit: mm
Functional Magnetization Intensity unit (M):G
Functional Magnetizing field unit      (H):G
Induction field unit                   (B):G
```

	type	radial dim. axial dim.	radial pos. axial pos.	Mr Mz	magnitude
Cell 1 **Comp. 1**	permanent	5. 4.44444	0 −37.7778	0. 4041.87	4041.87
Cell 2 **Comp. 1**	permanent	5. 4.44444	0 −33.3333	0. 4108.75	4108.75
Cell 3 **Comp. 1**	permanent	5. 4.44444	0 −28.8889	0. 4145.79	4145.79

2. Full details data for Cells 26 to 29 for the same system Case 7 (default meshing)

The M, B, H vectors are given in relation both to system coordinates and to the material direction of magnetization.

```
MagnetizationTable[barmagnetmesh3,FullDetails->True,CellsList->{{26,29}},
CondensedForm->True]
```

```
n°      type      first cell   last cell   radial cells   axial cells
1    permanent        1           36           2             18
```

```
Length unit: mm
Functional Magnetization Intensity unit (M):G
Functional Magnetizing field unit       (H):G
Induction field unit                    (B):G
```

	type	radial dim. axial dim.	radial pos. axial pos.		
Cell 26 **Comp. 1**	permanent	5. 4.44444	7.5 −6.66667		
Hm_r −9.517939	Hm_z −129.778075	Hm_{mod} 130.126630	Hm_p −129.778075	Hm_t	−9.5
M_r −1.903588	M_z 4187.022190	M_{mod} 4187.022630	M_p 4187.022190	M_t	−1.9(
B_r −11.421527	B_z 4057.244120	B_m 4057.260190	B_p 4057.244120	B_t	−11.4:
χ_p 0.100000	χ_t 0.200000	μ_p 1.100000	μ_t 1.200000		
angle 0.	M_{rm} 4200.0000	H_{ci} −3200.0000	marging 3070.2000	rel. marg.	
Cell 27 **Comp. 1**	permanent	5. 4.44444	7.5 −2.22222		
Hm_r −2.981529	Hm_z −122.341433	Hm_{mod} 122.377758	Hm_p −122.341433	Hm_t	−2.9
M_r −0.596306	M_z 4187.765860	M_{mod} 4187.765900	M_p 4187.765860	M_t	−0.5$
B_r −3.577835	B_z 4065.424420	B_m 4065.426000	B_p 4065.424420	B_t	−3.5;
χ_p 0.100000	χ_t 0.200000	μ_p 1.100000	μ_t 1.200000		
angle 0.	M_{rm} 4200.0000	H_{ci} −3200.0000	marging 3077.7000	rel. marg.	
Cell 28 **Comp. 1**	permanent	5. 4.44444	7.5 2.22222		
Hm_r 2.981529	Hm_z −122.341433	Hm_{mod} 122.377758	Hm_p −122.341433	Hm_t	2.9
M_r 0.596306	M_z 4187.765860	M_{mod} 4187.765900	M_p 4187.765860	M_t	0.5$
B_r 3.577835	B_z 4065.424420	B_m 4065.426000	B_p 4065.424420	B_t	3.5;
χ_p 0.100000	χ_t 0.200000	μ_p 1.100000	μ_t 1.200000		
angle 0.	M_{rm} 4200.0000	H_{ci} −3200.0000	marging 3077.7000	rel. marg.	
Cell 29 **Comp. 1**	permanent	5. 4.44444	7.5 6.66667		
Hm_r 9.517939	Hm_z −129.778075	Hm_{mod} 130.126630	Hm_p −129.778075	Hm_t	9.5
M_r 1.903588	M_z 4187.022190	M_{mod} 4187.022630	M_p 4187.022190	M_t	1.9(
B_r 11.421527	B_z 4057.244120	B_m 4057.260190	B_p 4057.244120	B_t	11.4:
χ_p 0.100000	χ_t 0.200000	μ_p 1.100000	μ_t 1.200000		
angle 0.	M_{rm} 4200.0000	H_{ci} −3200.0000	marging 3070.2000	rel. marg.	

3. Plot of the functional magnetization Mz and magnetizing field Hmz along an axial line for the same Case 7.

```
MagnetizationCurves[barmagnetmesh3, {Mz, Hz}, ComponentsList → {1}, PlotRows → {1}]
```

System name: **barmagnetmesh3**
Ferrit long bar magnet, (*meshing level 3*)
Functional Magnetization Mz map on axial direction
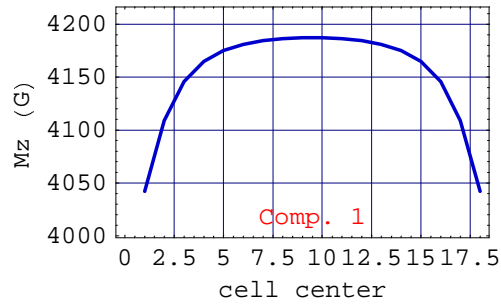
System name: **barmagnetmesh3**
Ferrit long bar magnet, (*meshing level 3*)
Functional magnetizing field Hz map on axial direction
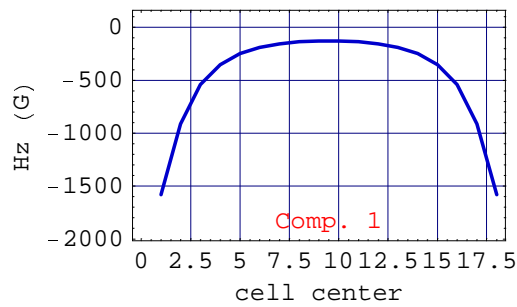
4. Plot of Mz and Hz field along a radial line for the same Case 7.

MagnetizationCurves[barmagnetmesh3, {Mz, Hz}, ComponentsList → {1}, PlotColumns → {1}]

System name: **barmagnetmesh3**
Ferrit long bar magnet, (*meshing level 3*)
Functional Magnetization Mz map on radial direction
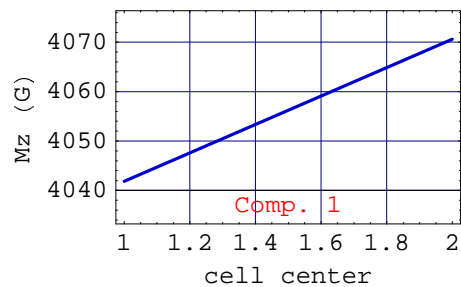
System name: **barmagnetmesh3**

Ferrit long bar magnet, (*meshing level 3*)

Functional magnetizing field Hz map on radial direction
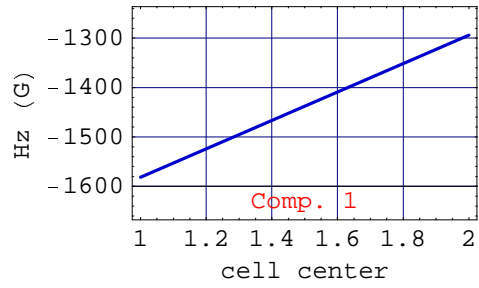
■ Chapter 8: Systems with Soft Magnetic Materials Components

■ 8-1 Introduction

This chapter deals with soft ferromagnetic materials. These materials are intensively used in most electrical devices such as: motors, alternators, transformers, inductance coils, magnetic shielding. The key to their magnetic behavior is both a very large magnetic response to even a small magnetic field excitation and secondly, as opposed to hard ferromagnetics, a reversible response.

Soft ferromagnetic materials are characterized by 2 basic features (see Appendix B.1 for a detailed explanation):

- a saturation of the magnetization when the applied magnetizing field becomes very large. This corresponds to a complete alignment of the elementary dipoles on the applied field. The relevant parameter is the **saturation magnetization Ms**.

- a non linear variation of the permeability (or the susceptibility) with the applied field. The permeability goes to a **maximum (μpeak** or χ**peak**) for a given magnetizing field (**Hpeak**). These last parameters heve a very large number of applications since in many cases soft magnetic materials are employed as amplifiers of the magnetizing field. This objective is fully met when the excitation field is below the peak. *Magnetica* offers the possibility of using both a linear or a non linear model of the magnetization curve. The linear model implies a constant permeability (and susceptibility) based on the approximation of the actual magnetization curve below the peak. It should be noted that many engineering models of magnet circuit calculation assumes implicitly a linear model.

The magnetic field calculation for this type of component implies discretization into several cells. It requires using the component function: **AXSSoftMagneticComponent**.

Increasing the number of cells improves the accuracy of the results. The meshing can be entirely automatic with 4 possible levels, level 2 being the default value. The choice between a linear and a non linear permeability model is optional, the non linear model being the default case.

■ 8-2 Immediate Help

```
? AXSMagneticSystem

AXSMagneticSystem[{comp1,comp2,...},options]  defines
   a new axis symmetric system made with the specified components.


Options: SystemLabel, MeshingLevel,
   LegendrePointPosition, LegendreRadius, LegendreOrder Details…


? MeshingLevel

MeshingLevel is  an option for AXSMagneticSystem that
   specifies the granularity of the meshing. (1 -> Sparse, 2 -> intermediate (
   default), 3 -> High density, 4 -> Very high density) Details…
```

? `AXSSoftMagneticComponent`

```
AXSSoftMagneticComponent[material,
   innerDiameter,outerDiameter,length,position,fillingFactor,options]
```

builds a Magnetica object which represents a cylindrical
  component (bar, ring), made of material with isotropic reversible (linear
  or non-linear) magnetization in 2D cylindrical axis symmetric space

Options: see LinearPermeability, Temperature, ComponentMeshing Details…

Options[`AXSSoftMagneticComponent`]

{LinearPermeability → False, FillingFactor → 1,
 ComponentMeshing → Automatic, Temperature → Automatic}

? `LinearPermeability`

Option for AXSSoftMagneticComponent.With LinearPermeability->
  True the magnetization function is assumed to be linear Details…

?`Properties`

```
Properties[system,options] gives the technical details of a
   magnetic system. With the option ComponentsList->list or ComponentsList->
   All, it also gives the properties of the specified list of components of s.
See ?ComponentsList Details…
Properties[material,options] displays the properties of the given material.
   With Temperature->value it gives the information at the specified temperature
```

? `FieldPointOrigin`

FieldPointOrigin[s] is a shortcut for VerboseFieldPoint[s,{0,0,0}] Details…

?`MagnetizationTable`

```
MagnetizationTable[system , opts]
gives the magnetization table of the magnetic
   components (ferro, magnets) ComponentsList -> {components} or {{1,3}}
CellsList -> {1,2,3} or {{1,3}}
FullDetails -> True|False
CondensedForm -> True|False
 Details…
```

?`MaterialMagnetizationCurves`

MaterialMagnetizationCurves[material,magnetizationFunctions,options] plots for the given
  material, the magnetization curve of each of function of magnetizationFunctions.

Options: MagnetizingFieldRange, Temperature, NumberOfPoints, PlotScale, CurveType

Units: MagnetizingUnit, MaterialMagnetizationUnit, InductionUnit, FieldUnit Details…

?`MagnetizingFieldRange`

MagnetizingFieldRange is an option for MaterialMagnetizationCurves or
  MaterialMagnetizationTable that determines the range for the magnetizing field.
  For Permanent magnet material, possible values are {value1,value2}, Operational,
  Extended or Automatic, and for other material {value1,value2} or Automatic Details…

?`PlotScale`

PlotScale is an option for FieldMap and other field drawing functions.
   Possible values: LinearMap, LinearLogMap, LogLinearMap LogLogMap Details…

```
?MagnetizingUnit
```

MagnetizingUnit is an option used to specify the unit of the magnetizing field (H) Details…

■ 8-3 Cylindrical Bar of Reversible Magnetization Material

■ 8-3-1 Introduction

The first example is a bar of the same dimensions as the long cylindrical bar magnet developed in Chapter 7 but set in the center of a solenoid that generates the excitation field for soft ferromagnetic material. All the field calculation functions (see Chapter 4) can be applied without restriction. We will just emphasize below some special features.

■ 8-3-2 Common Steel Properties

The physical and magnetic data of common steel (*Magnetica* reference : **SteelCommon1**) are obtained using the command **Properties** and the Magnetization and the permeability curves through the command **MaterialMagnetization-Curves**. Their main features are :

▪ The saturation magnetization is   **Ms**   : 20500 Gauss (2.05 Tesla)

▪ The peak of permeability occurring at **Hm**   : 3.15 Gauss (251 Amp/m)

▪ with the relative permeability   *μ***peak** : 2334

```
Properties[SteelCommon1]

Material properties
material name                               SteelCommon1
material category                           Ferromagnetic reversible non line
designation                                 Common steel annealed (H1)
General Physical Properties
specific weight (g/cm3)                          7.860000
dilation coefficient                             0.000012
    (relative linear change)
resistivity (Ohm x cm)                           0.000010
Coefficient temperature resistivity              0.001000
Magnetic Physical Properties
relative volume magnetic                      2100.238863
    (linear) susceptibility
temperature coefficient susceptibility          -0.000200
saturation magnetization (G)                 20500.000000
coercive field                                   1.080000
 (continuous reversibility limit,G)
temperature coefficient saturation mag          -0.000200
Specifications of the maximum of permeability
peak magnetizing field (G)                       3.152921
peak magnetization (G)                        7357.652663
peak susceptibility (relative to vacuum)      2333.598736
coefficient of linearization for susceptibility  0.900000
```

The following curves show the comparative plots of the experimental curve (Blue) of the non linear model and the linear model curve. It is obvious that the linear model loses all precision above the Magnetizing field peak. The comparison with permeability is significant. It should be noted that many engineering models of magnet circuit calculation assume implicitly a linear model.

```
MagneticaUserUnits[FieldUnit→Gauss];
MaterialMagnetizationCurves[SteelCommon1,{M,Permeability},
CurveType->Comparison]
```

```
Material Magnetization Curve
SteelCommon1  temperature  20°celsius
Common steel annealed (H1)
M Magnetization f(Hm)
Type of plot: Linear/Linear
```
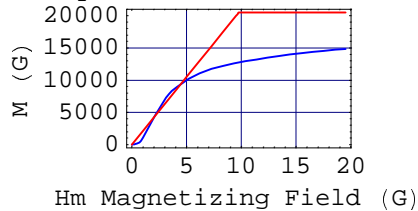
```
Material Magnetization Curve
SteelCommon1  temperature  20°celsius
Common steel annealed (H1)
μ Permeability f(Hm)
Type of plot: Linear/Linear
```
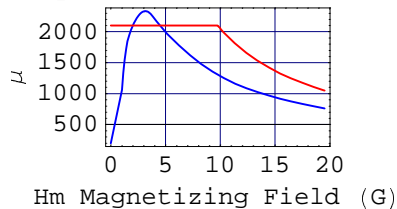
■ 8-3-3 Entering Data

The data of the magnetic system include both the solenoid (**AXSSolenoid**) and the soft ferromagnetic material bar (**AXSSoftMagneticComponent**) with component data such as described in the introduction of Chapter 6. The meshing level has the default value (2) which gives 30 cells for the 2 cases of the example.

Case 1: the material magnetizing model is non-linear (experimental curve)

```
barsteel1 = AXSMagneticSystem[
    {AXSSolenoid[CopperOFHC, 100, 200, 100, 0, 200, 20, FillingFactor -> 0.6],
     AXSSoftMagneticComponent[SteelCommon1, 0, 20, 80, 0]},
    SystemLabel → "Bar of Common Steel in a solenoid"];
```

Case 2: the material magnetizing model is linear (approximation for low fields)

```
barsteel2 = AXSMagneticSystem[
   {AXSSolenoid[CopperOFHC, 100, 200, 100, 0, 200, 20, FillingFactor -> 0.6],
    AXSSoftMagneticComponent[SteelCommon1, 0, 20, 80, 0,
      LinearPermeability -> True]}, SystemLabel →
    "Bar of Common Steel (linear)in a solenoid"]

- AXSMagneticSystem :{-solenoid-, -soft-} -
```

■ 8-3-4 System Drawing and Technical Specifications

Drawing Case 1 (same for 2)

DrawSystem2D[barsteel1]

    System name: **barsteel1**
    Bar of Common Steel in a solenoid
    2D view ZR
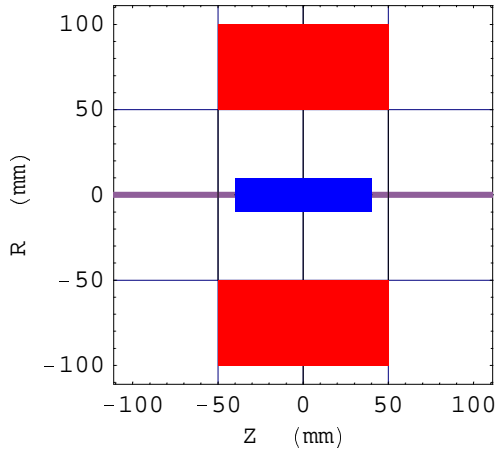
- Graphics -

Technical data Case 1

Properties[barsteel1]

General properties

System name:	**barsteel1**
System magnetic parity	True
System Legendre radius	55.9017
Magnetic field at the origin (G)	{0, 2679.74}
Field magnitude at the origin(G)	2679.74
System mass (Kg)	12.8079
System nominal power (W)	43.3288
System effective power(W)	43.3288
Number of components	2
Number of current carrying component	1
Number of ferro-magnetic (non rigid) components	1
Number of components with reversible magnetization	1

n°	type	first cell	last cell	radial cells	axial cells
2	soft	1	30	2	15

■ **8-3-5 Field Calculation and Functional Magnetization**

1. Induction field Bz (i.e the global field) at the bar isocenter (Case 1, non linear model).

    FieldPointOrigin[barsteel1]

    System name: **barsteel1**
    The magnetic field (in Gauss) at the origin
    Calculation mode: Automatic Mode
    Br   0.000000 G
    Bz   2679.735408 G
    Bm   2679.735408 G

2. Induction field Bz (i.e the global field) at the bar isocenter (Case 2, linear model).

```
FieldPointOrigin[barsteel2]
```

```
System name: barsteel2
The magnetic field (in Gauss) at the origin
Calculation mode: Automatic Mode
Br            0.000000 G
Bz         3747.284100 G
Bm         3747.284100 G
```

The 2 following calculations illustrate the magnetization for the inner central cell, radial coordinate r=0, i.e. on the axis and axial coordinate z=1.333mm from central radial plane. We have for the Case 1 (non linear):

- 282.559 gauss: excitation field created by the solenoid.

- 1.497 gauss: functional magnetizing field.

But the resulting magnetic field is the sum of the excitation field and the interaction of the 89 other cells:

- 2732.305 gauss: magnetization intensity

- 2733.802 gauss: induction field

- 1825.500   : relative susceptibility

- 1826.500   : relative permeability

3. Induction field Bz (i.e. the global field) at cell 16 center (Case 1)

```
NumberForm[FieldPoint[barsteel1, {0, 1.33333333}], 9]
```

```
{0, 2678.33607}
```

```
VerboseFieldPoint[barsteel1, {0, 1.33333333}, ComponentsList → All]
```

```
System name: barsteel1
The magnetic field (in Gauss) at the point {0, 1.33333}(in Milli Meter)
Calculation mode: Automatic Mode

Component n° 1
Component type: cylindrical solenoid
Br            0.000000 G
Bz          282.558958 G
Bm          282.558958 G


Component n° 2
Component type: cylindrical, reversible magnetization
Br            0.000000 G
Bz         2395.777108 G
Bm         2395.777108 G


Total
Br            0.000000 G
Bz         2678.336067 G
Bm         2678.336067 G
```

4. Magnetization data at cell 16 center (Case1)

```
MagnetizationTable[barsteel1,ComponentsList->{2},CellsList->{{16,16}},CondensedForm
->True,FullDetails->True]
```

```
n°   type    first cell   last cell    radial cells    axial cells
 2   soft           1          30               2              15
```

```
Length unit: mm
Functional Magnetization Intensity unit (M):G
Functional Magnetizing field unit       (H):G
Induction field unit                    (B):G
```

	type	radial dim.	radial pos.
		axial dim.	axial pos.
Cell 16	soft	5.	7.5
Comp. 2		5.33333	-37.3333

Hm_r -0.453405 Hm_z 1.309603 Hm_{mod} 1.385870
M_r -776.918214 M_z 2244.031130 M_{mod} 2374.716330
B_r -777.371619 B_z 2245.340730 B_m 2376.102200
χ 1713.519860 μ 1714.519860

The 2 following calculations show the field in front of the bar, (bar pseudo pole), coordinate r=0 and z=40 mm. We have:

- 941.538 gauss: total field coming from

- 227.849 gauss: contribution of the solenoid.

- 713.689 gauss: contribution of the material of the bar

5. Magnetic field Bz (i.e. the global field) at the bar axial surface (Case 1)

```
FieldPoint[barsteel1, {0, 40}]
```

```
{0, 916.57}
```

```
VerboseFieldPoint[barsteel1,{0,40},ComponentsList->All]
```

```
System name: barsteel1
The magnetic field (in Gauss) at the point {0, 40}(in Milli Meter)
Calculation mode: Automatic Mode

Component n° 1
Component type: cylindrical solenoid
Br          0.000000 G
Bz        227.848670 G
Bm        227.848670 G


Component n° 2
Component type: cylindrical, reversible magnetization
Br          0.000000 G
Bz        688.721605 G
Bm        688.721605 G


Total
Br          0.000000 G
Bz        916.570276 G
Bm        916.570276 G
```

6. Magnetic field Bz (i.e. the global field) at the bar axial surface (Case 2)

```
FieldPoint[barsteel2, {0, 40}]
```

{0, 1287.87}

```
VerboseFieldPoint[barsteel2,{0,40},ComponentsList->All]
```

```
System name: barsteel2
The magnetic field (in Gauss) at the point {0, 40}(in Milli Meter)
Calculation mode: Automatic Mode


Component n° 1
Component type: cylindrical solenoid
Br            0.000000 G
Bz          227.848670 G
Bm          227.848670 G


Component n° 2
Component type: cylindrical, reversible magnetization
Br            0.000000 G
Bz         1060.020380 G
Bm         1060.020380 G


Total
Br            0.000000 G
Bz         1287.869050 G
Bm         1287.869050 G
```

7. Field plot on the axis around the bar axial surface (Case 1)

```
FieldMap[barsteel1, Bz, Line[{0, 30}, {0, 50}]];
```

```
           System name: barsteel1
           Bar of Common Steel in a solenoid
           Bz (G) (Continuous, Automatic Mode)
           between {0., 30.} and {0., 50.}
```
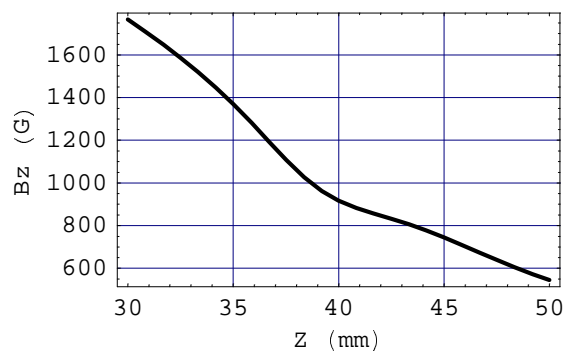

8. Field line inside and around the bar (Case 1).

```
ContourFieldMap[barsteel1, Flux, Plane[{-30, -50}, {30, 50}], NumberOfPoints → 25];
```

System name: **barsteel1**
Bar of Common Steel in a solenoid
Flux (μWb) (Continuous, Automatic
Mode)
between {-30., -50.} and {30., 50.}
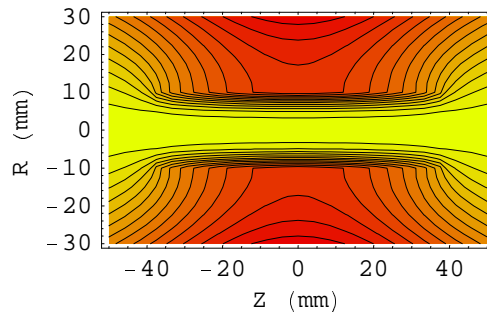
■ 8-4 Cylindrical Shielding of a Superconducting Solenoid

■ 8-4-1 Introduction

The example is a cylindrical shell made of silicium steel used for the shielding of the stray field of a strong MRI magnet. This example shows the application of the method to standard high permeability industrial steel used in small thicknesses. Shielding efficiently means using the minimum amount of material for the greatest reduction of the stray field. This implies working with a high excitation magnetization field. Consequently, the use of the non linear magnetization model appears very suitable.

■ 8-4-2 Silicium Steel Properties

Silicon steels (commercially called electrical steels) are the most used materials in all the electrical industry (motor, alternator, transformer manufacturing). Their production amounts to millions of tons. Their magnetic properties are strictly controlled. They have a much higher permeability than common steel so that they are excellent as magnetic field amplifiers. These materials are highly sophisticated substances based on iron but including small amounts of silicium as well as some other elements such as aluminum. The example is based on a high silicium content steel (ref : **SiSteel3**) with the following properties and magnetization curves:

- The saturation magnetization is **Ms** : 19800 Gauss (1.98 Tesla)

- The peak of permeability occurs at **Hm** : 0.86 Gauss (68 Amp/m)

- where the relative permeability is μ**peak** : 9062

Properties[SiSteel3]

Material properties

material name	SiSteel3
material category	Ferromagnetic reversible non line
designation	Silicon steel sheet, grains NO, h

General Physical Properties

specific weight (g/cm3)	7.600000
dilation coefficient	0.000012
(relative linear change)	
resistivity (Ohm x cm)	0.000010
Coefficient temperature resistivity	0.001000

Magnetic Physical Properties

relative volume magnetic	8155.441381
(linear) susceptibility	
temperature coefficient susceptibility	-0.000300
saturation magnetization (G)	19800.000000
coercive field	0.240000
(continuous reversibility limit,G)	
temperature coefficient saturation mag	-0.000300

Specifications of the maximum of permeability

peak magnetizing field (G)	0.859414
peak magnetization (G)	7787.668008
peak susceptibility (relative to vacuum)	9061.601534
coefficient of linearization for susceptibility	0.900000

Magnetization and permeability curves

MaterialMagnetizationCurves[SiSteel3,{M,Permeability}]

```
Material Magnetization Curve
SiSteel3  temperature  20°celsius
Silicon steel sheet, grains NO, high Silicon content, (S3)
M Magnetization f(Hm)
Type of plot: Linear/Linear
```
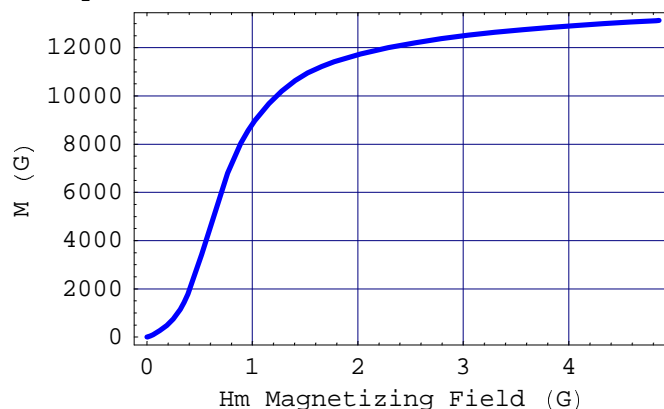
```
Material Magnetization Curve
SiSteel3  temperature  20°celsius
Silicon steel sheet, grains NO, high Silicon content, (S3)
μ Permeability f(Hm)
Type of plot: Linear/Linear
```
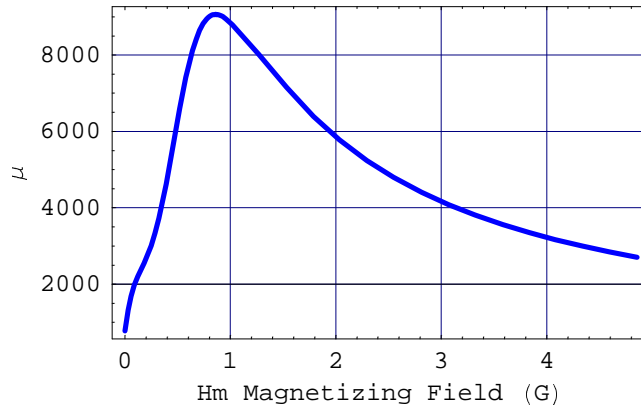

■ 8-4-3 Entering Data

The example starts with an unshielded magnet (Case 1) and continues with a shielded magnet of thicknesses of: 3 mm (Case 2), 10 mm (Case 3) and 30 mm (Case 4). The meshing level is at the default value (i.e. 2 -> 30 cells).

Case 1: MRI magnet without shield

```
unshieldMRI=
AXSMagneticSystem[
{AXSSolenoid[SuperNbTi4, 1240, 1280, 300, 180.05374474797077, 635, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1280, 300, -180.05374474797077, 635, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1280, 250, 550.1194910091477, 598, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1280, 250, -550.1194910091477, 598, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1320, 250, 874.8480174899299, 1015, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1320, 250, -874.8480174899299, 1015, 735.2202]},
SystemLabel -> "unshielded MRI 1.5 T superconducting magnet"];
```

Case 2: same MRI magnet with a cylindrical shell 3 mm thick

```
thick=3;
dshellin=3000;
dshellout=dshellin+2*thick;
shieldedMRI1=
AXSMagneticSystem[
{AXSSolenoid[SuperNbTi4, 1240, 1280, 300, 180.05374474797077, 635, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1280, 300, -180.05374474797077, 635, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1280, 250, 550.1194910091477, 598, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1280, 250, -550.1194910091477, 598, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1320, 250, 874.8480174899299, 1015, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1320, 250, -874.8480174899299, 1015, 735.2202],
AXSSoftMagneticComponent[SiSteel3,dshellin, dshellout,5000,0]},
SystemLabel -> "MRI 1.5 T superconducting magnet with 10 mm Si steel shield"];
```

Case 3: same MRI magnet with a cylindrical shell 10mm thick

```
thick=10;
dshellin=3000;
dshellout=dshellin+2*thick;
shieldedMRI2=
AXSMagneticSystem[
{AXSSolenoid[SuperNbTi4, 1240, 1280, 300, 180.05374474797077, 635, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1280, 300, -180.05374474797077, 635, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1280, 250, 550.1194910091477, 598, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1280, 250, -550.1194910091477, 598, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1320, 250, 874.8480174899299, 1015, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1320, 250, -874.8480174899299, 1015, 735.2202],
AXSSoftMagneticComponent[SiSteel3,dshellin, dshellout,5000,0]},
SystemLabel -> "MRI 1.5 T superconducting magnet with 10 mm Si steel shield"];
```

Case 4: same MRI magnet with a cylindrical shell 30 mm thick

```
thick=30;
dshellin=3000;
dshellout=dshellin+2*thick;
shieldedMRI3=
AXSMagneticSystem[
{AXSSolenoid[SuperNbTi4, 1240, 1280, 300, 180.05374474797077, 635, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1280, 300, -180.05374474797077, 635, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1280, 250, 550.1194910091477, 598, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1280, 250, -550.1194910091477, 598, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1320, 250, 874.8480174899299, 1015, 735.2202],
AXSSolenoid[SuperNbTi4, 1240, 1320, 250, -874.8480174899299, 1015, 735.2202],
AXSSoftMagneticComponent[SiSteel3,dshellin, dshellout,5000,0]},
SystemLabel -> "MRI 1.5 T superconducting magnet with 10 mm Si steel shield"];
```

■ 8-4-4 System Drawing and Technical Specifications

Drawing of Case 2: MRI coils and cylindrical shell

```
DrawSystem2D[shieldedMRI1];
```

```
System name: shieldedMRI1
MRI 1.5 T superconducting magnet
with 10 mm Si steel shield
2D view ZR
```
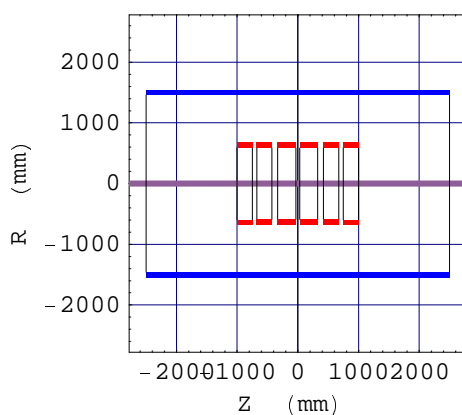

Technical data of Case 2: shielded MRI, 3 mm thick

```
Properties[shieldedMRI1]
```

General properties

```
System name:                                           shieldedMRI1
System magnetic parity                                 False
System Legendre radius                                 1768.87
Magnetic field at the origin (G)                       {0, 15019.4}
Field magnitude at the origin(G)                       15019.4
System mass  (Kg)                                      1249.23
System nominal power (W)                               0.
System effective power(W)                              0.
Number of components                                   7
Number of current carrying component                   6
Number of ferro-magnetic (non rigid) components        1
Number of components with reversible magnetization     1
n°  type   first cell   last cell   radial cells   axial cells
7   soft       1           30            1             30
```

■ **8-4-5 Field Calculation and Functional Magnetization**

1. Isocenter magnetic field

We can see that the shielding increases somewhat the isocenter field

Case 1: unshieldedMRI: 15000 Gauss (1.5 Tesla)

```
FieldPoint[unshieldMRI, {0, 0}]
```

```
{0, 15000.}
```

Case 2 : shielded MRI: 15019 Gauss (1.5019 Tesla). The shield generates an extra 19 Gauss at the isocenter

```
FieldPoint[shieldedMRI1, {0, 0}]
```

```
{0, 15019.4}
```

Case 3: shielded MRI: 15059 Gauss (1.5059 Tesla). The shield generates an extra 59 Gauss at the isocenter

```
FieldPoint[shieldedMRI2, {0, 0}]
```

```
{0, 15059.5}
```

Case 4: shielded MRI : 15145 Gauss (1.5145 Tesla). The shield generates an extra 145 Gauss at the isocenter

```
FieldPoint[shieldedMRI3, {0, 0}]
```

```
{0, 15145.7}
```

2. Shielding effect calculated at a point 3000 mm off axis on the central radial plane

   Case 1: unshielded MRI : stray field 129.4 Gauss

   Case 2: shielded 3 mm : stray field 124.4 Gauss, shielding effect 5 Gauss

   Case 3: shielded 10 mm : stray field 113.4 Gauss, shielding effect 16 Gauss

   Case 4: shielded 30 mm : stray field   85.4 Gauss, shielding effect 45 Gauss

```
rc = 3000; zc = 0;
VerboseFieldPoint[unshieldMRI, {rc, zc}]
VerboseFieldPoint[shieldedMRI1, {rc, zc}]
VerboseFieldPoint[shieldedMRI2, {rc, zc}]
VerboseFieldPoint[shieldedMRI3, {rc, zc}]

System name: unshieldMRI
The magnetic field (in Gauss) at the point {3000, 0}(in Milli Meter)
Calculation mode: Automatic Mode
Br           0.000000 G
Bz        -129.435019 G
Bm         129.435019 G

System name: shieldedMRI1
The magnetic field (in Gauss) at the point {3000, 0}(in Milli Meter)
Calculation mode: Automatic Mode
Br           0.000000 G
Bz        -124.421536 G
Bm         124.421536 G

System name: shieldedMRI2
The magnetic field (in Gauss) at the point {3000, 0}(in Milli Meter)
Calculation mode: Automatic Mode
Br           0.000000 G
Bz        -113.344577 G
Bm         113.344577 G

System name: shieldedMRI3
The magnetic field (in Gauss) at the point {3000, 0}(in Milli Meter)
Calculation mode: Automatic Mode
Br           0.000000 G
Bz         -85.377007 G
Bm          85.377007 G
```

3. System Legendre source coefficients for the Case 4 shielded MRI 30mm

VerboseLegendreCoefficients[shieldedMRI3, LegendreRadius -> {250, 2000}]

```
System Name:                          shieldedMRI3
System Legendre radius                1775.876 mm
Printout Legendre radius central       250.000 mm
Printout Legendre radius remote       2000.000 mm
System Legendre center position         0.000 mm
Inner spherical radius limit           577.277 mm
Outer spherical radius limit          3341.367 mm
System magnetic parity                   False
System magnetic constant field          0.000 G
Mode: absolute Field (Gauss)
```

```
Order            central coeff.          remote coeff.
D0           15145.716210802            0.000000000
D1               0.000000000            0.000000000
D2              -3.284322716          755.186244641
D3               0.000000000            0.000000000
D4              -0.036579248          746.361577872
D5               0.000000000            0.000000000
D6               2.091823076          493.368897819
D7               0.000000000            0.000000000
D8              -1.617913155         -778.101567208
D9               0.000000000            0.000000000
D10              0.353671179        -1559.623810719
D11              0.000000000            0.000000000
D12             -0.049853891         2556.195811151
D13              0.000000000            0.000000000
D14              0.005885907        11408.028684260
D15              0.000000000            0.000000000
D16             -0.000822329         5565.292211713
D17              0.000000000           -4.090000000e-10
D18              0.000150991       -48576.674149276
D19              0.000000000           -8.343000000e-10
D20             -0.000028266      -120586.713055288
```

4. Variation of the stray field with the radial distance for the Case 4 shielded MRI 30mm

FieldMap[shieldedMRI3, Bm, Line[{2000, 0}, {7000, 0}]]

System name: **shieldedMRI3**
MRI 1.5 T superconducting magnet
with 10 mm Si steel shield
Bm (G) (Continuous, Automatic Mode)
between {2000., 0.} and {7000., 0.}
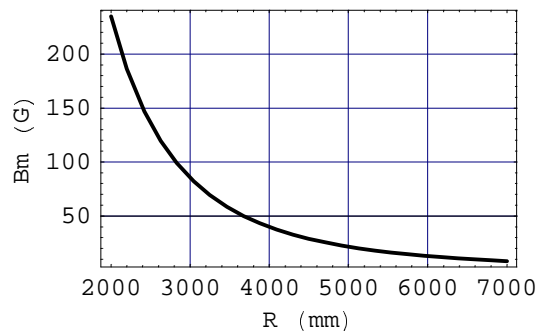
{- Graphics -}

5. Isofield lines of 30, 20, 10, 5, 1 gauss for the Case 4 shielded MRI 30mm

```
DrawIsoFieldLines[shieldedMRI3, {30, 20, 10, 5, 1},
   AngularRange → {0, 360}, NumberOfPoints → 40, FieldUnit → Gauss];
```

```
System name: shieldedMRI3
MRI 1.5 T superconducting magnet
with 10 mm Si steel shield
Magnetic iso Field lines
Field magnitude (G) {30., 20., 10., 5., 1.}
2D view ZR
```
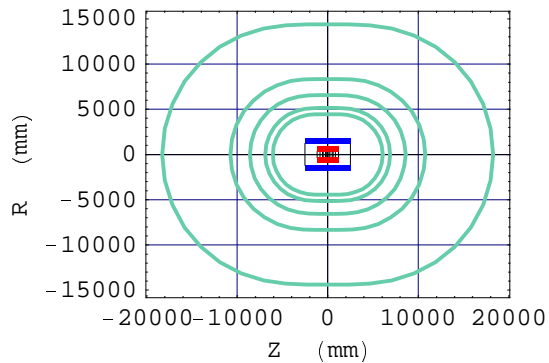

6. Functional magnetization Mz and magnetizing field Hz, along an axial line for the Case 4 (30 mm shell)

```
MagnetizationCurves[shieldedMRI3, {Mz, Hz}, ComponentsList → {7}, PlotRows → {1}]
```

```
System name: shieldedMRI3
MRI 1.5 T superconducting magnet
with 10 mm Si steel shield
Functional Magnetization Mz
map on axial direction
```
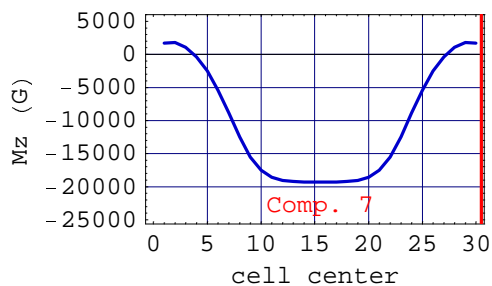
```
System name: shieldedMRI3
MRI 1.5 T superconducting magnet
with 10 mm Si steel shield
Functional magnetizing field Hz
map on axial direction
```
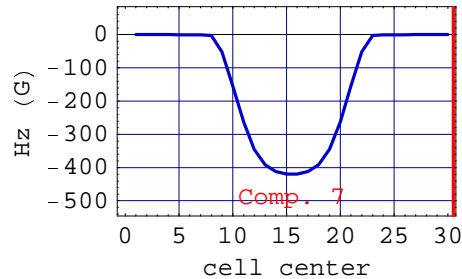

7. Magnetization at the extremity of the shell, Case 4 shell 30 mm.

The cell's functional magnetization and the magnetizing field are small, respectively around 1100 Gauss and 0.4 Gauss

```
MagnetizationTable[shieldedMRI3,ComponentsList->{7},CellsList->{{1,4}},CondensedFor
m->True,FullDetails->True]

n°   type   first cell   last cell   radial cells   axial cells
7    soft        1           30            1              30


Length unit: mm
Functional Magnetization Intensity unit (M):G
Functional Magnetizing field unit      (H):G
Induction field unit                   (B):G
```

		type	radial dim. axial dim.	radial pos. axial pos.
Cell 1 **Comp. 7**		soft	30. 166.667	1515. -2416.67

Hm_r −0.061134 Hm_z 0.378757 Hm_{mod} 0.383659
M_r −276.551431 M_z 1713.385120 M_{mod} 1735.560220
B_r −276.612565 B_z 1713.763880 B_m 1735.943880
χ 4523.710340 μ 4524.710340

		type	radial dim. axial dim.	radial pos. axial pos.
Cell 2 **Comp. 7**		soft	30. 166.667	1515. -2250.

Hm_r −0.067562 Hm_z 0.386961 Hm_{mod} 0.392815
M_r −314.424747 M_z 1800.861610 M_{mod} 1828.104330
B_r −314.492309 B_z 1801.248570 B_m 1828.497150
χ 4653.857790 μ 4654.857790

		type	radial dim. axial dim.	radial pos. axial pos.
Cell 3 **Comp. 7**		soft	30. 166.667	1515. -2083.33

Hm_r −0.101596 Hm_z 0.298058 Hm_{mod} 0.314897
M_r −370.127569 M_z 1085.867640 M_{mod} 1147.215310
B_r −370.229165 B_z 1086.165700 B_m 1147.530200
χ 3643.143410 μ 3644.143410

		type	radial dim. axial dim.	radial pos. axial pos.
Cell 4 **Comp. 7**		soft	30. 166.667	1515. -1916.67

Hm_r −0.161199 Hm_z −0.131122 Hm_{mod} 0.207794
M_r −436.327733 M_z −354.917809 M_{mod} 562.448701
B_r −436.488932 B_z −355.048931 B_m 562.656495
χ 2706.765790 μ 2707.765790

8. Magnetization at the middle of the shell, Case 4 shell 30 mm

The cell's functional magnetization appear very close to the saturation: 19300 gauss versus 19800 gauss

```
MagnetizationTable[shieldedMRI3,ComponentsList->{7},CellsList->{{44,47}},CondensedF
orm->True,FullDetails->True]

n°   type   first cell   last cell   radial cells   axial cells
7    soft        1           30            1             30


Length unit: mm
Functional Magnetization Intensity unit (M):G
Functional Magnetizing field unit      (H):G
Induction field unit                   (B):G
```

■ 8-5 Cylindrical Sheet of Mu Metal (Earth Magnetic Shielding)

■ 8-5-1 Introduction

The example is a cylindrical shell made of Mu metal used to obtain an amagnetic box, i.e. a volume with the Earth's magnetic field reduced as much as possible. This box was designed for biological research: growing bacteria outside of the Earth's magnetic field.

The example shows the application of the software for very high permeability materials used in very thin sheets. It shows also the use of the special magnetic component feature **AXSLargeMagneticSource** to deal with the Earth's magnetic field.

■ 8-5-2 Mu Metal Properties

The Mu metal is a very high permeability material ($\mu = 100\ 000$) which is made of an alloy of Iron and Nickel. Mu metal is intensively used for magnetic field shielding purposes but exclusively when the field is already low. For instance Mu metal is coated inside the case of high definition CRT monitors or TVs to reduce the adverse effect of the Earth's magnetic field. It should be noted that industry produces a wide range of high permeability materials other than Mu metal. The example is based on a standard Mu metal (ref: **MuMetal1**). Pertinent data are the magnetization at saturation and the permeability. The information and the curves are obtained using the command **Properties** and **MaterialMagnetizationCurves**. Its main features are:

- The saturation magnetization is   **Ms**   : 7800 Gauss (0.78 Tesla)

- The peak of permeability occurs at **Hm**   : 0.0.32 Gauss (2.5 Amp/m)

- where the relative permeability μ**peak** is: 105320

Properties[MuMetal1]

Material properties

material name	MuMetal1
material category	Ferromagnetic reversible non line
designation	Mu metal standard annealed, 0.1 m

General Physical Properties

specific weight (g/cm3)	8.700000
dilation coefficient	0.000013
(relative linear change)	
resistivity (Ohm x cm)	0.000058
Coefficient temperature resistivity	0.001000

Magnetic Physical Properties

relative volume magnetic	94788.251479
(linear) susceptibility	
temperature coefficient susceptibility	-0.000300
saturation magnetization (G)	7800.000000
coercive field	0.015000
(continuous reversibility limit,G)	
temperature coefficient saturation mag	-0.000300

Specifications of the maximum of permeability

peak magnetizing field (G)	0.032237
peak magnetization (G)	3395.227999
peak susceptibility (relative to vacuum)	105320.279421
coefficient of linearization for susceptibility	0.900000

MaterialMagnetizationCurves[MuMetal1,{M,Permeability},
MagnetizingUnit->Milli Gauss]

```
Material Magnetization Curve
MuMetal1  temperature  20°celsius
Mu metal standard annealed, 0.1 mm sheet (I2)
M Magnetization f(Hm)
Type of plot: Linear/Linear
```
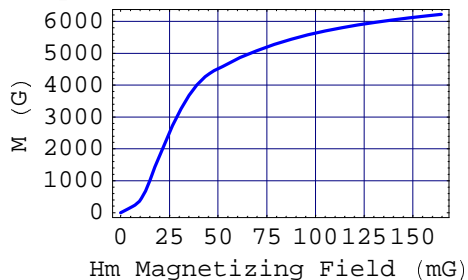


```
Material Magnetization Curve
MuMetal1  temperature  20°celsius
Mu metal standard annealed, 0.1 mm sheet (I2)
μ Permeability f(Hm)
Type of plot: Linear/Linear
```
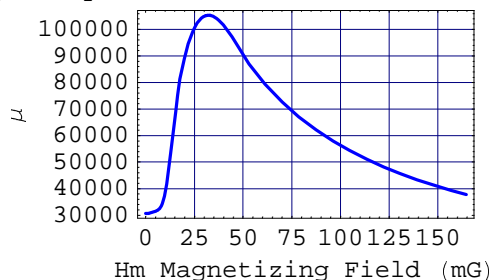
■ **8-5-3 Entering Data**

The examples consist of 2 cases: a single sheet box (Case 1) and a double sheets box (Case 2). It is assumed that the Earth's magnetic field is oriented along the axis of the box. The thickness of the Mu metal sheet is 0.1 mm and the meshing level is set at 2.

Case 1 : single sheet box

```
innerDiam = 200;
demiLength = 300;
thick = .1;
outerDiam = innerDiam + 2 * thick;
diskPosition = N[ (demiLength + thick) / 2];
amagbox1 =
  AXSMagneticSystem[{
    AXSLargeMagneticSource[0.4589],
    AXSSoftMagneticComponent[MuMetal1,
      innerDiam, outerDiam, demiLength / 2, -demiLength / 4.],
    AXSSoftMagneticComponent[MuMetal1, innerDiam, outerDiam,
      demiLength / 2, demiLength / 4.],
    AXSSoftMagneticComponent[MuMetal1, 0, outerDiam, thick, -diskPosition],
    AXSSoftMagneticComponent[MuMetal1, 0, outerDiam, thick, diskPosition]},
    SystemLabel -> "Amagnetic Box Mu metal 0.1 mm thick", MeshingLevel → 2];
```

Case 2 : double sheet box

```
innerDiam = 200;
demiLength1 = 300;
thick = .1;
outerDiam = innerDiam + 2 * thick;
diskPosition = N[ (demiLength1 + thick) / 2];
gap = 30;
demiLength2 = demiLength1 + gap;
amagbox2 =
  AXSMagneticSystem[{
    AXSLargeMagneticSource[0.4589],
    AXSSoftMagneticComponent[MuMetal1,
      innerDiam, outerDiam, demiLength1 / 2, -demiLength1 / 4.],
    AXSSoftMagneticComponent[MuMetal1, innerDiam, outerDiam,
      demiLength1 / 2, demiLength1 / 4.],
    AXSSoftMagneticComponent[MuMetal1, 0, outerDiam, thick, -diskPosition],
    AXSSoftMagneticComponent[MuMetal1, 0, outerDiam, thick, diskPosition],
    AXSSoftMagneticComponent[MuMetal1, innerDiam + gap,
      outerDiam + gap, demiLength2 / 2, -demiLength2 / 4.],
    AXSSoftMagneticComponent[MuMetal1, innerDiam + gap,
      outerDiam + gap, demiLength2 / 2, demiLength2 / 4.],
    AXSSoftMagneticComponent[MuMetal1, 0, outerDiam + gap, thick,
      -diskPosition - gap / 2], AXSSoftMagneticComponent[
      MuMetal1, 0, outerDiam + gap, thick, diskPosition + gap / 2]},
    SystemLabel -> "Amagnetic Box Mu metal 0.1 mm thick", MeshingLevel → 2];
```

■ **8-5-4 System Drawing and Technical Specifications**

Drawing the single sheet box

```
DrawSystem2DXZ[amagbox1];
```

```
System name: amagbox1
Amagnetic Box Mu metal 0.1 mm thick
2D view XZ
```
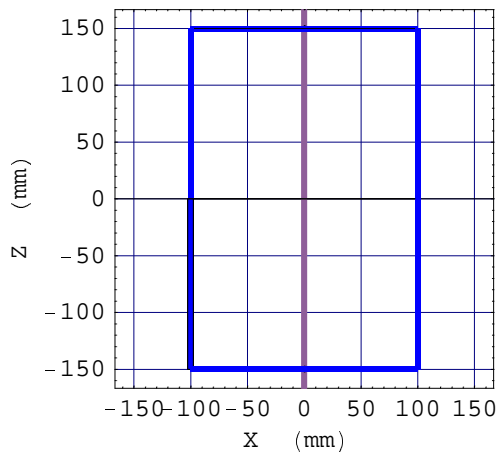

Drawing the double sheet box

```
DrawSystem2DXZ[amagbox2];
```

```
System name: amagbox2
Amagnetic Box Mu metal 0.1 mm thick
2D view XZ
```
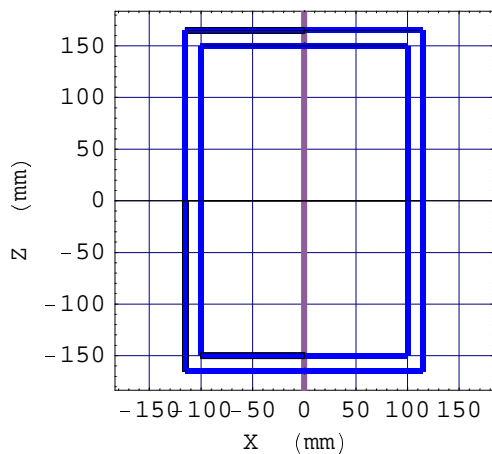

Technical data of the double sheet box

```
Properties[amagbox2]
```

General properties

```
System name:                                           amagbox2
System magnetic parity                                 True
System Legendre radius                                 150.631
Magnetic field at the origin (G)                       {0, 0.00453201}
Field magnitude at the origin(G)                       0.00453201
System mass  (Kg)                                      0.498804
System nominal power (W)                               0
System effective power(W)                              0
Number of components                                   9
Number of ferro-magnetic (non rigid) components        8
Number of components with reversible magnetization      8
Number of large sources                                1
```

n°	type	first cell	last cell	radial cells	axial cells
2	soft	91	120	1	30
3	soft	121	150	1	30
4	soft	31	60	30	1
5	soft	181	210	30	1
6	soft	61	90	1	30
7	soft	151	180	1	30
8	soft	1	30	30	1
9	soft	211	240	30	1

■ 8-5-5 Field Calculation and Functional Magnetization

1. Magnetic field at the origin, units in Nano tesla. The Earth's magnetic field is 45 890 nT

   Case 1: single sheet box: The field is 1568 nT, reduction of 44321 nT. The remaining field is 3.4 % of the initial field.

```
MagneticaUserUnits[FieldUnit → Nano Tesla];
FieldPoint[amagbox1, {0, 0}]
```

```
{0, 1567.99}
```

   Case 2: double sheet box: The field is 453 nT, reduction of 45437 nT. The remaining field is 0.9% of the initial field

```
FieldPoint[amagbox2, {0, 0}]
```

```
{0, 453.201}
```

2. Variation of the Bz field on an axial line for the Case 2 double sheet box

```
FieldMap[amagbox2, Bz, Line[{0, -400}, {0, 400}]];
```

> System name: **amagbox2**
> *Amagnetic Box Mu metal 0.1 mm thick*
> Bz (nT) (Continuous, Automatic
> Mode)
> between {0., -400.} and {0., 400.}
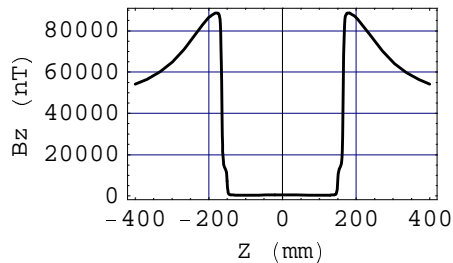
3. Contour plot of the Bz field on an axial line for the Case 2 double sheet box

```
ContourFieldMap[amagbox2, Bz,
  Plane[{-50, -60}, {50, 60}], NumberOfPoints → 25, ViewType → View3DXZ];
```

> System name: **amagbox2**
> *Amagnetic Box Mu metal 0.1 mm thick*
> Bz (nT) (Continuous, Automatic
> Mode)
> between {-50., -60.} and {50., 60.}
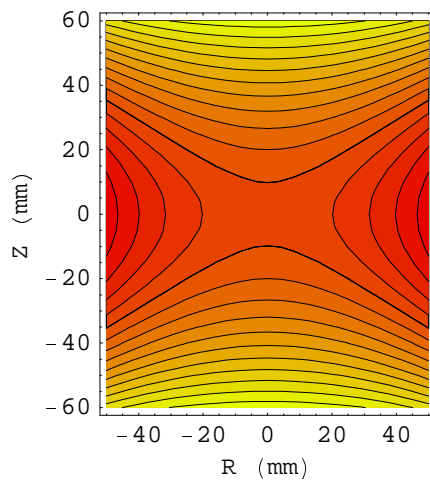
4. Variation of the functional magnetization Mz in the external sheet for the Case 2 double sheet box. Field units in Gauss

```
MagneticaUserUnits[FieldUnit → Gauss];
MagnetizationCurves[amagbox2, {Mz, Mr, Mm}, ComponentsList → {6, 7}, PlotRows -> {1}]
```

System name: **amagbox2**
Amagnetic Box Mu metal 0.1 mm thick
Functional Magnetization Mz
map on axial direction
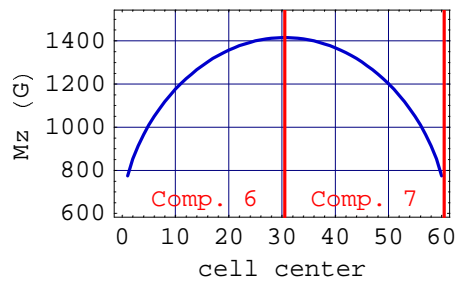
System name: **amagbox2**
Amagnetic Box Mu metal 0.1 mm thick
Functional Magnetization Mr
map on axial direction
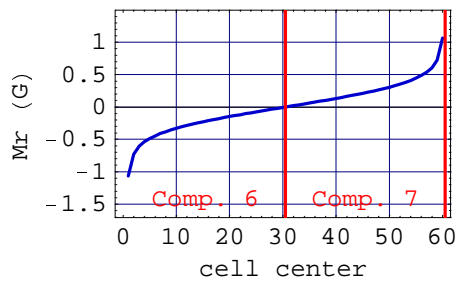
System name: **amagbox2**
Amagnetic Box Mu metal 0.1 mm thick
Functional Magnetization Mm
map on axial direction
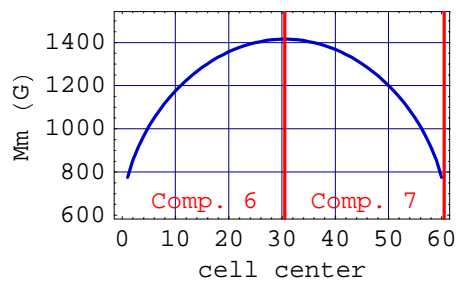
5. Magnetization at the extremity of the outer sheet, Case 2. The cell's functional magnetizations is 1410 Gauss with a permeability of 80000 which is a very effective shielding condition.

```
MagnetizationTable[amagbox2,ComponentsList->{7},CellsList->{{151,154}},CondensedFor
m->True,FullDetails->True]
```

n°	type	first cell	last cell	radial cells	axial cells
2	soft	91	120	1	30
3	soft	121	150	1	30
4	soft	31	60	30	1
5	soft	181	210	30	1
6	soft	61	90	1	30
7	soft	151	180	1	30
8	soft	1	30	30	1
9	soft	211	240	30	1

```
Length unit: mm
Functional Magnetization Intensity unit  (M):G
Functional Magnetizing field unit        (H):G
Induction field unit                     (B):G
```

	type	radial dim. axial dim.	radial pos. axial pos.
Cell 151 **Comp. 7**	soft	0.1 5.5	115.05 2.75

Hm_r 8.393280e-8 Hm_z 0.017508 Hm_{mod} 0.017508
M_r 0.006783 M_z 1415.011810 M_{mod} 1415.011810
B_r 0.006783 B_z 1415.029310 B_m 1415.029310
χ 80818.794800 μ 80819.794800

	type	radial dim. axial dim.	radial pos. axial pos.
Cell 152 **Comp. 7**	soft	0.1 5.5	115.05 8.25

Hm_r 2.520324e-7 Hm_z 0.017502 Hm_{mod} 0.017502
M_r 0.020362 M_z 1413.985950 M_{mod} 1413.985950
B_r 0.020362 B_z 1414.003460 B_m 1414.003460
χ 80789.733800 μ 80790.733800

	type	radial dim. axial dim.	radial pos. axial pos.
Cell 153 **Comp. 7**	soft	0.1 5.5	115.05 13.75

Hm_r 4.208376e-7 Hm_z 0.017489 Hm_{mod} 0.017489
M_r 0.033975 M_z 1411.931760 M_{mod} 1411.931760
B_r 0.033975 B_z 1411.949250 B_m 1411.949250
χ 80731.420200 μ 80732.420200

	type	radial dim. axial dim.	radial pos. axial pos.
Cell 154 **Comp. 7**	soft	0.1 5.5	115.05 19.25

Hm_r 5.908323e-7 Hm_z 0.017470 Hm_{mod} 0.017470
M_r 0.047647 M_z 1408.844200 M_{mod} 1408.844200
B_r 0.047647 B_z 1408.861670 B_m 1408.861670
χ 80643.467800 μ 80644.467800

■ Chapter 9: Magnetic System Applications

■ 9-1 Introduction

This chapter deals with extended applications of the **AXSSoftMagneticComponent.**

The first example proceeds with a complex magnetic system made of only hard and soft ferromagnetic materials.

The second example is the application of the method to solve the magnetization of either diamagnetic or paramagnetic common materials submitted to a very high magnetic field.

■ 9-2 Immediate Help

?AXSMagneticSystem

```
AXSMagneticSystem[{comp1,comp2,...},options]  defines
   a new axis symmetric system made with the specified components.

Options: SystemLabel, MeshingLevel,
   LegendrePointPosition, LegendreRadius, LegendreOrder Details…
```

?MeshingLevel

```
MeshingLevel is  an option for AXSMagneticSystem that
   specifies the granularity of the meshing. (1 -> Sparse, 2 -> intermediate (
   default), 3 -> High density, 4 -> Very high density) Details…
```

? AXSSoftMagneticComponent

```
AXSSoftMagneticComponent[material,
   innerDiameter,outerDiameter,length,position,fillingFactor,options]

builds a Magnetica object which represents a cylindrical
   component (bar, ring), made of material with isotropic reversible (linear
   or non-linear) magnetization in 2D  cylindrical axis symmetric space

Options: see LinearPermeability, Temperature, ComponentMeshing Details…
```

Options[AXSSoftMagneticComponent]

```
{LinearPermeability → False, FillingFactor → 1,
 ComponentMeshing → Automatic, Temperature → Automatic}
```

? LinearPermeability

```
Option for AXSSoftMagneticComponent.With LinearPermeability->
   True the magnetization function is assumed to be linear Details…
```

?Properties

```
Properties[system,options] gives the technical details of a
   magnetic system. With the option ComponentsList->list or ComponentsList->
   All, it also gives the properties of the specified list of components of s.
See ?ComponentsList Details…
Properties[material,options] displays the properties of the given material.
   With Temperature->value it gives the information at the specified temperature
```

?FieldPointOrigin

FieldPointOrigin[s] is a shortcut for VerboseFieldPoint[s,{0,0,0}] Details…

?MagnetizationTable

```
MagnetizationTable[system , opts]
gives the magnetization table of the magnetic
  components (ferro, magnets) ComponentsList -> {components} or {{1,3}}
CellsList -> {1,2,3} or {{1,3}}
FullDetails -> True|False
CondensedForm -> True|False
 Details…
```

?MaterialMagnetizationCurves

MaterialMagnetizationCurves[material,magnetizationFunctions,options] plots for the given
  material, the magnetization curve of each of function of magnetizationFunctions.

Options: MagnetizingFieldRange, Temperature, NumberOfPoints, PlotScale, CurveType

Units: MagnetizingUnit, MaterialMagnetizationUnit, InductionUnit, FieldUnit Details…

?MagnetizingFieldRange

MagnetizingFieldRange is an option for MaterialMagnetizationCurves or
  MaterialMagnetizationTable that determines the range for the magnetizing field.
  For Permanent magnet material, possible values are {value1,value2}, Operational,
  Extended or Automatic, and for other material {value1,value2} or Automatic Details…

?CurveType

CurveType is an option for MaterialMagnetizationCurves
  that specifies, for non-linear magnetization material, the type
  of curve (Experimental, LinearModel or Comparison) Details…

?PlotScale

PlotScale is an option for FieldMap and other field drawing functions.
   Possible values: LinearMap, LinearLogMap, LogLinearMap LogLogMap Details…

?MagnetizingUnit

MagnetizingUnit is an option used to specify the unit of the magnetizing field (H) Details…

■ 9-3 NIB (Neodymium Iron Boron) Magnet

■ 9-3-1 Introduction

The example is an H shaped permanent magnet made of 2 polar pieces of high performance permanent magnet material (NIB: Neodymium Iron Boron compound) combined with a yoke of standard pure iron steel acting as a magnetic flux return path.

The example shows the application of the software for a complex magnetic system made only of hard and soft ferromagnetic materials.

■ 9-3-2 Materials Properties: NIB Magnet and Soft Industrial Iron

NIB compounds are the most recent and the most powerful hard ferromagnetics ever found. Their use is spreading in many sectors of electrical machinery and magnet technology including large MRI magnets. Several specific products have been marketed. The example is based on a high coercive field material (*Magnetica* ref: **NIBcomp4**). Full physical and magnetic properties and magnetization curves are obtained using the command **Properties** and **Material-MagnetizationCurves** followed by the name as argument (see paragraph 7.1.2 or appendix B.1 for more complete explanations).

The main features of this NIB compounds are:

- Intrinsic remanent magnetization **Mr**   : 12000 Gauss (1.2 Tesla, 954930 Amp/m)

- Coercive field **Hcoer**   : -20000 Gauss (-2 Tesla, 1591550 Amp/m)

```
Properties[NIBcomp4]

Material properties
material name                          NIBcomp4
material category                      Permanent magnet
designation                            NdFeB compound U4 (120/200)
General Physical Properties
specific weight (g/cm3)                       7.500000
dilation coefficient                          4.000000e-6
    (relative linear change)
resistivity (Ohm x cm)                        0.000150
Coefficient temperature resistivity           0.001000
Magnetic Physical Properties
intrinsic remanent magnetic field (G)      12000.000000
intrinsic coercive field (G)              -20000.000000
principal dynamic relative susceptibility      0.100000
secondary  relative susceptibility             0.200000
temperature coefficient susceptibility         0.000000
temperature coefficient remanent field        -0.001200
temperature coefficient coercive field        -0.006000
```

```
MaterialMagnetizationCurves[NIBcomp4, {Mprincipal},
 MagnetizingUnit → Tesla,
 MaterialMagnetizationUnit → Tesla]
```
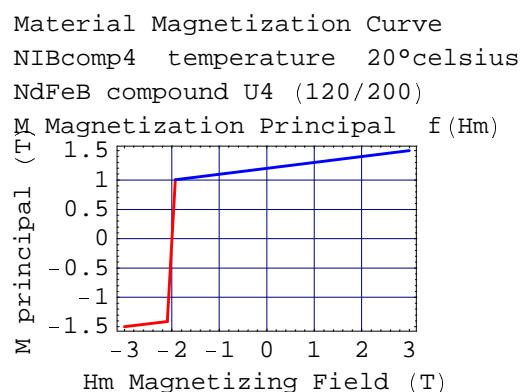

The yoke is made of a standard magnetic steel. Industry produces for magnetic applications a range of steels which *de facto*, are very pure iron metals marketed in a well-annealed state and with well-specified magnetic properties. The example is based on a typical material (ref: **IronInd3**). Pertinent physical and magnetic data and magnetization curves

are obtained using the commands **Properties** and **MaterialMagnetizationCurves**. Its
main features are:

- Saturation magnetization   **Ms**   : 21550 Gauss (2.155 Tesla)

- Peak of permeability at   **Hm**   : 2.09 Gauss (166 Amp/m)

- With relative permeability μ**peak**   : 4785

```
Properties[IronInd3]
```

Material properties
material name	IronInd3
material category	Ferromagnetic reversible non line
designation	Industrial Iron middle annealed

General Physical Properties
specific weight (g/cm3)	7.860000
dilation coefficient	0.000011
(relative linear change)	
resistivity (Ohm x cm)	0.000010
Coefficient temperature resistivity	0.001000

Magnetic Physical Properties
relative volume magnetic	4305.818592
(linear) susceptibility	
temperature coefficient susceptibility	-0.000300
saturation magnetization (G)	21550.000000
coercive field	0.400000
(continuous reversibility limit,G)	
temperature coefficient saturation mag	-0.003000

Specifications of the maximum of permeability
peak magnetizing field (G)	2.087626
peak magnetization (G)	9987.709726
peak susceptibility (relative to vacuum)	4784.242880
coefficient of linearization for susceptibility	0.900000

```
MaterialMagnetizationCurves[IronInd3,{M,Permeability}]
```

```
        Material Magnetization Curve
        IronInd3   temperature   20°celsius
        Industrial Iron middle annealed (A1)
        M Magnetization f(Hm)
        Type of plot: Linear/Linear
```

```
Material Magnetization Curve
IronInd3  temperature  20°celsius
Industrial Iron middle annealed (A1)
μ Permeability f(Hm)
Type of plot: Linear/Linear
```



```
Hm Magnetizing Field (G)
```
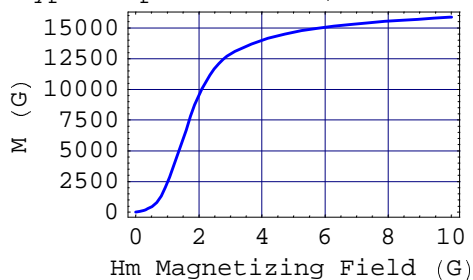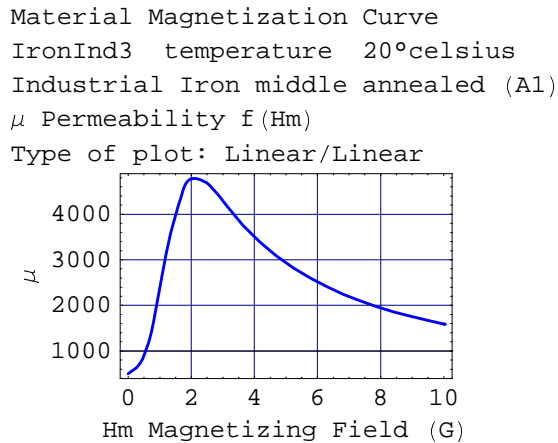
■ 9-3-3 Entering Data

The examples show 2 cases that are identical except for the length air gap. Meshing level is set to 1 for quick evaluation.

Case 1: small air gap (high field)

```
airgapLength = 20;
poleDiameter = 80;
poleLength = 20;
polePosition = (poleLength + airgapLength) / 2;
yokeDiam = 120;
yokepoleLength = 8;
ringLength = 6;
ringPosition = polePosition + (poleLength - ringLength) / 2;
yokepolePosition = polePosition + (poleLength + yokepoleLength) / 2;
yokeshellThick = 5;
yokeshellDiam = yokeDiam + 2 * yokeshellThick;
yokeshellLength = 2 * yokepolePosition + yokepoleLength;

NIBairgapmagnet1 = AXSMagneticSystem[{
    AXSHardMagneticComponent[NIBcomp4, 0, poleDiameter, poleLength, -polePosition],
    AXSHardMagneticComponent[NIBcomp4, 0, poleDiameter, poleLength, polePosition],
    AXSSoftMagneticComponent[IronInd3,
     poleDiameter, yokeDiam, ringLength, -ringPosition],
    AXSSoftMagneticComponent[IronInd3, poleDiameter,
     yokeDiam, ringLength, ringPosition],
    AXSSoftMagneticComponent[IronInd3, 0, yokeDiam,
     yokepoleLength, -yokepolePosition],
    AXSSoftMagneticComponent[IronInd3, 0, yokeDiam,
     yokepoleLength, yokepolePosition],
    AXSSoftMagneticComponent[IronInd3, yokeDiam,
     yokeshellDiam, yokeshellLength, 0]},
   SystemLabel -> "NIB air gap magnet with iron yoke",
   MeshingLevel → 1];
```

Case 2: large air gap (reduced field)

```
airgapLength = 60;
poleDiameter = 80;
poleLength = 20;
polePosition = (poleLength + airgapLength) / 2;
yokeDiam = 120;
yokepoleLength = 8;
ringLength = 6;
ringPosition = polePosition + (poleLength - ringLength) / 2;
yokepolePosition = polePosition + (poleLength + yokepoleLength) / 2;
yokeshellThick = 5;
yokeshellDiam = yokeDiam + 2 * yokeshellThick;
yokeshellLength = 2 * yokepolePosition + yokepoleLength;

NIBairgapmagnet2 = AXSMagneticSystem[{
    AXSHardMagneticComponent[NIBcomp4, 0, poleDiameter, poleLength, -polePosition],
    AXSHardMagneticComponent[NIBcomp4, 0, poleDiameter, poleLength, polePosition],
    AXSSoftMagneticComponent[IronInd3,
     poleDiameter, yokeDiam, ringLength, -ringPosition],
    AXSSoftMagneticComponent[IronInd3, poleDiameter,
     yokeDiam, ringLength, ringPosition],
    AXSSoftMagneticComponent[IronInd3, 0, yokeDiam,
     yokepoleLength, -yokepolePosition],
    AXSSoftMagneticComponent[IronInd3, 0, yokeDiam,
     yokepoleLength, yokepolePosition],
    AXSSoftMagneticComponent[IronInd3, yokeDiam,
     yokeshellDiam, yokeshellLength, 0]},
   SystemLabel -> "NIB air gap magnet with iron yoke",
   MeshingLevel -> 1];
```

■ 9-3-4 System Drawing

Drawing of Case 1, small air gap

```
DrawSystem2DXZ[NIBairgapmagnet1, ShowArrows → True];
```

System name: **NIBairgapmagnet1**
NIB air gap magnet with iron yoke
2D view XZ
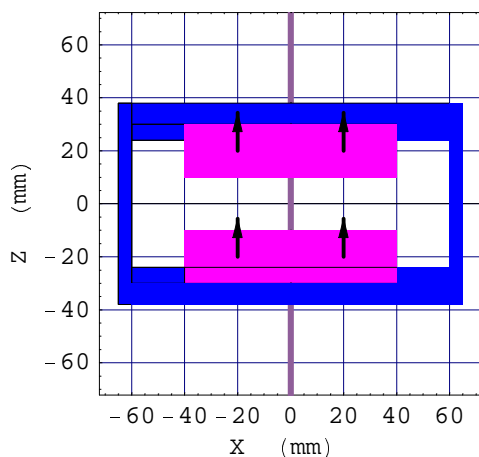
Drawing of Case 2, large air gap

```
DrawSystem2DXZ[NIBairgapmagnet2, ShowArrows → True];
```

System name: **NIBairgapmagnet2**
NIB air gap magnet with iron yoke
2D view XZ
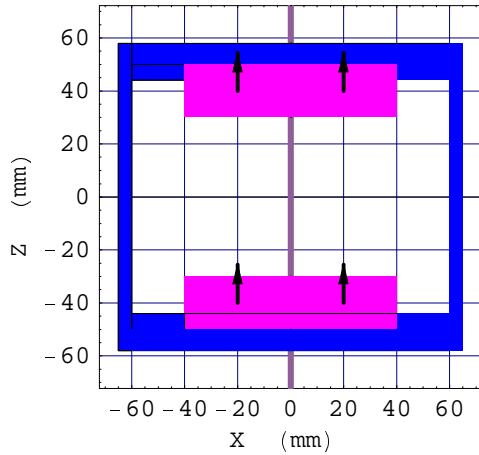
System technical data of Case 1, small air gap

```
Properties[NIBairgapmagnet1]
```

General properties

System name:	**NIBairgapmagnet1**
System magnetic parity	False
System Legendre radius	42.6464
Magnetic field at the origin (G)	{0, 7001.13}
Field magnitude at the origin(G)	7001.13
System mass (Kg)	4.69582
System nominal power (W)	0
System effective power(W)	0
Number of components	7
Number of ferro-magnetic (non rigid) components	7
Number of components with reversible magnetization	5
Number of components with permanent magnetization	2

n°	type	first cell	last cell	radial cells	axial cells
1	permanent	11	14	2	2
2	permanent	20	23	2	2
3	soft	6	10	5	1
4	soft	24	28	5	1
5	soft	1	5	5	1
6	soft	29	33	5	1
7	soft	15	19	1	5

■ 9-3-5 Field and Magnetization Map

1. Isocenter magnetic field of Case 1, small air gap

  • 7001.1 Gauss total field This is made of the following contributions:

  • 4076.1 Gauss from the NIB polar pieces

  • 2925.0 Gauss from the iron steel yoke

FieldPoint[NIBairgapmagnet1, {0, 0}]

{0, 7001.13}

FieldPointOrigin[NIBairgapmagnet1, ComponentsList → All]

System name: **NIBairgapmagnet1**
The magnetic field (in Gauss) at the origin
Calculation mode: Automatic Mode

Component n° 1
Component type: cylindrical permanent magnet
Br   0.000000 G
Bz   2038.073749 G
Bm   2038.073749 G

Component n° 2
Component type: cylindrical permanent magnet
Br   0.000000 G
Bz   2038.073749 G
Bm   2038.073749 G

Component n° 3
Component type: cylindrical, reversible magnetization
Br   0.000000 G
Bz   16.651020 G
Bm   16.651020 G

Component n° 4
Component type: cylindrical, reversible magnetization
Br   0.000000 G
Bz   16.651020 G
Bm   16.651020 G

Component n° 5
Component type: cylindrical, reversible magnetization
Br   0.000000 G
Bz   1139.094153 G
Bm   1139.094153 G

Component n° 6
Component type: cylindrical, reversible magnetization
Br   0.000000 G
Bz   1139.094153 G
Bm   1139.094153 G

Component n° 7
Component type: cylindrical, reversible magnetization
Br   0.000000 G
Bz   613.496486 G
Bm   613.496486 G

Total
Br   0.000000 G
Bz   7001.134330 G
Bm   7001.134330 G

2. Isocenter magnetic field of Case 2, large air gap

- 3122.8 Gauss total field This is made of the following contributions:

- 2032.0 Gauss from the NIB polar pieces

- 1090.8 Gauss from the iron steel yoke

  FieldPoint[NIBairgapmagnet2, {0, 0}]

  {0, 3122.8}

```
FieldPointOrigin[NIBairgapmagnet2, ComponentsList → All]

System name: NIBairgapmagnet2
The magnetic field (in Gauss) at the origin
Calculation mode: Automatic Mode


Component n° 1
Component type: cylindrical permanent magnet
Br           0.000000 G
Bz        1016.013280 G
Bm        1016.013280 G


Component n° 2
Component type: cylindrical permanent magnet
Br           0.000000 G
Bz        1016.013280 G
Bm        1016.013280 G


Component n° 3
Component type: cylindrical, reversible magnetization
Br           0.000000 G
Bz         -20.679556 G
Bm          20.679556 G


Component n° 4
Component type: cylindrical, reversible magnetization
Br           0.000000 G
Bz         -20.679556 G
Bm          20.679556 G


Component n° 5
Component type: cylindrical, reversible magnetization
Br           0.000000 G
Bz         412.156476 G
Bm         412.156476 G


Component n° 6
Component type: cylindrical, reversible magnetization
Br           0.000000 G
Bz         412.156476 G
Bm         412.156476 G


Component n° 7
Component type: cylindrical, reversible magnetization
Br           0.000000 G
Bz         307.817525 G
Bm         307.817525 G


Total
Br           0.000000 G
Bz        3122.797926 G
Bm        3122.797926 G
```

3. Bz field plot on an axial line Case 1, small gap

FieldMap[NIBairgapmagnet1, Bz, Line[{-50, 0}, {50, 0}]];

System name: **NIBairgapmagnet1**
NIB air gap magnet with iron yoke
Bz (T) (Continuous, Automatic Mode)
between {-50., 0.} and {50., 0.}
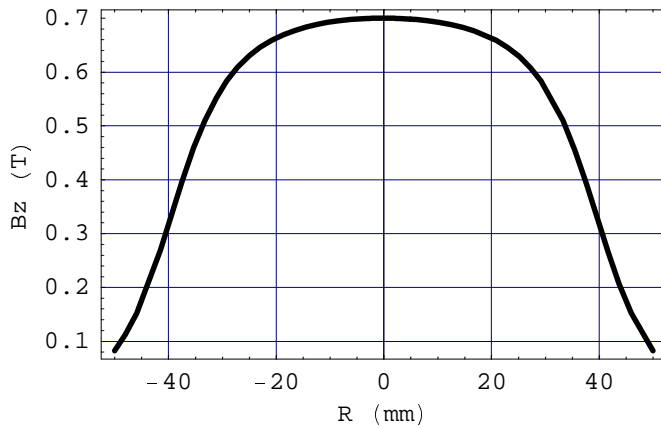
4. Plot of functional magnetization and magnetizing field in the NIB polar piece along a radial line (starting from the axis), Case 1 small gap

MagnetizationCurves[NIBairgapmagnet1,
  {Mz, Hz}, ComponentsList → {1}, PlotColumns → {1}]

System name: **NIBairgapmagnet1**
NIB air gap magnet with iron yoke
Functional Magnetization Mz
map on radial direction
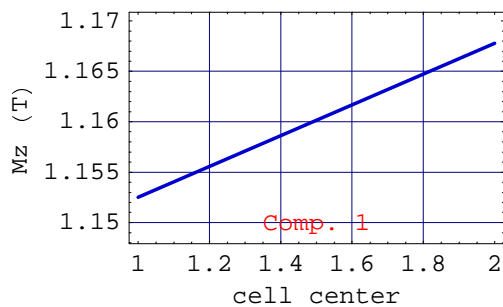
System name: **NIBairgapmagnet1**
NIB air gap magnet with iron yoke
Functional magnetizing field Hz
map on radial direction
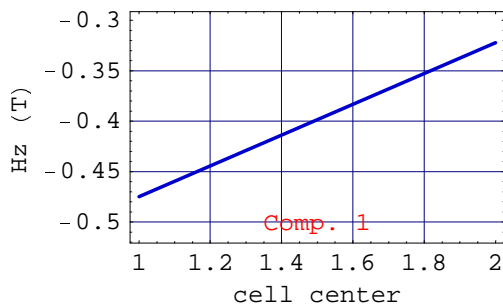
5. Table of functional magnetization and magnetizing field in the NIB polar piece, Case 1 small gap

```
MagnetizationTable[NIBairgapmagnet1, CellsList → {{11, 14}}, FullDetails → True,
 CondensedForm → True]
```

```
n°      type     first cell   last cell   radial cells   axial cells
1    permanent       11           14            2              2
2    permanent       20           23            2              2
3      soft           6           10            5              1
4      soft          24           28            5              1
5      soft           1            5            5              1
6      soft          29           33            5              1
7      soft          15           19            1              5
```

```
Length unit: mm
Functional Magnetization Intensity unit  (M):G
Functional Magnetizing field unit         (H):G
Induction field unit                      (B):G
```

	type	radial dim. axial dim.	radial pos. axial pos.		
Cell 11 **Comp. 1**	permanent	20. 10.	0 −25.		
Hm_r 0.000000	Hm_z −4749.782650	Hm_{mod} 4749.782650	Hm_p −4749.782650	Hm_t	
M_r 0.000000	M_z 11525.021700	M_{mod} 11525.021700	M_p 11525.021700	M_t	
B_r 0.000000	B_z 6775.239090	B_m 6775.239090	B_p 6775.239090	B_t	
χ_p 0.100000	χ_t 0.200000	μ_p 1.100000	μ_t 1.200000		
angle 0.	M_{rm} 12000.0000	H_{ci} −20000.0000	marging 15250.0000	rel. mar	
Cell 12 **Comp. 1**	permanent	20. 10.	0 −15.		
Hm_r 0.000000	Hm_z −4375.191310	Hm_{mod} 4375.191310	Hm_p −4375.191310	Hm_t	
M_r 0.000000	M_z 11562.480900	M_{mod} 11562.480900	M_p 11562.480900	M_t	
B_r 0.000000	B_z 7187.289560	B_m 7187.289560	B_p 7187.289560	B_t	
χ_p 0.100000	χ_t 0.200000	μ_p 1.100000	μ_t 1.200000		
angle 0.	M_{rm} 12000.0000	H_{ci} −20000.0000	marging 15625.0000	rel. mar	
Cell 13 **Comp. 1**	permanent	20. 10.	30. −25.		
Hm_r 347.242569	Hm_z −3220.159450	Hm_{mod} 3238.827610	Hm_p −3220.159450	Hm_t	34
M_r 69.448514	M_z 11677.984100	M_{mod} 11678.190600	M_p 11677.984100	M_t	6
B_r 416.691083	B_z 8457.824610	B_m 8468.082930	B_p 8457.824610	B_t	41
χ_p 0.100000	χ_t 0.200000	μ_p 1.100000	μ_t 1.200000		
angle 0.	M_{rm} 12000.0000	H_{ci} −20000.0000	marging 16780.0000	rel. mar	
Cell 14 **Comp. 1**	permanent	20. 10.	30. −15.		
Hm_r 1071.829610	Hm_z −3939.325360	Hm_{mod} 4082.536340	Hm_p −3939.325360	Hm_t	10
M_r 214.365921	M_z 11606.067500	M_{mod} 11608.047000	M_p 11606.067500	M_t	21
B_r 1286.195530	B_z 7666.742100	B_m 7773.881490	B_p 7666.742100	B_t	128
χ_p 0.100000	χ_t 0.200000	μ_p 1.100000	μ_t 1.200000		
angle 0.	M_{rm} 12000.0000	H_{ci} −20000.0000	marging 16061.0000	rel. mar	

6. Plot of functional magnetization Mz, Mr, and M modulus in the yoke iron polar piece along a radial line (starting from the axis), Case 1, small gap

```
MagnetizationCurves[NIBairgapmagnet1,
 {Mr, Mz, Mm}, ComponentsList → {5}, PlotColumns → {1}]
```

System name: **NIBairgapmagnet1**
NIB air gap magnet with iron yoke
Functional Magnetization Mr
map on radial direction
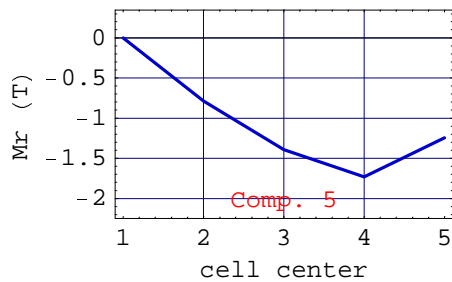
System name: **NIBairgapmagnet1**
NIB air gap magnet with iron yoke
Functional Magnetization Mz
map on radial direction
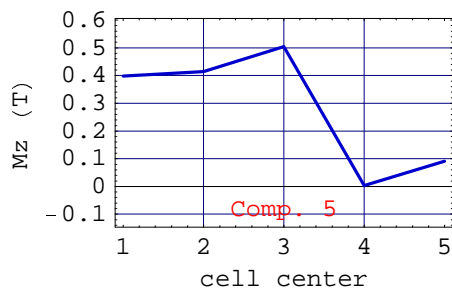
System name: **NIBairgapmagnet1**
NIB air gap magnet with iron yoke
Functional Magnetization Mm
map on radial direction
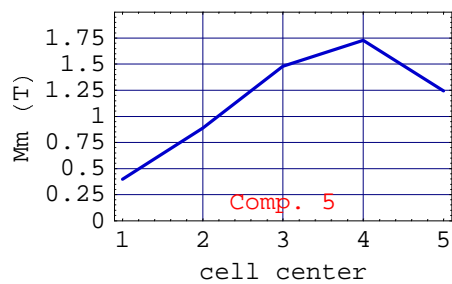
7. Table of Mz, Mr, and M modulus (functional magnetization) in the yoke iron polar piece, Case 1, small gap

MagnetizationTable[NIBairgapmagnet1, CellsList → {{1, 5}}, FullDetails → True,
 CondensedForm → True]

n°	type	first cell	last cell	radial cells	axial cells
1	permanent	11	14	2	2
2	permanent	20	23	2	2
3	soft	6	10	5	1
4	soft	24	28	5	1
5	soft	1	5	5	1
6	soft	29	33	5	1
7	soft	15	19	1	5

Length unit: mm
Functional Magnetization Intensity unit (M):G
Functional Magnetizing field unit   (H):G
Induction field unit   (B):G

	type	radial dim. axial dim.	radial pos. axial pos.
Cell 1 **Comp. 5**	soft	12. 8.	0 -34.

Hm_r 1.000000e-13 Hm_z 1.228543 Hm_{mod} 1.228543
M_r 4.346000e-10 M_z 3984.150780 M_{mod} 3984.150780
B_r 4.347000e-10 B_z 3985.379320 B_m 3985.379320
χ 3242.989440 μ 3243.989440

	type	radial dim. axial dim.	radial pos. axial pos.
Cell 2 **Comp. 5**	soft	12. 8.	18. -34.

Hm_r -1.665358 Hm_z 0.876429 Hm_{mod} 1.881899
M_r -7866.131590 M_z 4139.712900 M_{mod} 8888.939710
B_r -7867.796950 B_z 4140.589330 B_m 8890.821610
χ 4723.387620 μ 4724.387620

	type	radial dim. axial dim.	radial pos. axial pos.
Cell 3 **Comp. 5**	soft	12. 8.	30. -34.

Hm_r -5.008301 Hm_z 1.816382 Hm_{mod} 5.327506
M_r -13921.504500 M_z 5048.972610 M_{mod} 14808.795100
B_r -13926.512800 B_z 5050.789000 B_m 14814.122600
χ 2779.686100 μ 2780.686100

	type	radial dim. axial dim.	radial pos. axial pos.
Cell 4 **Comp. 5**	soft	12. 8.	42. -34.

Hm_r -23.334260 Hm_z 0.047429 Hm_{mod} 23.334308
M_r -17302.693500 M_z 35.169030 M_{mod} 17302.729300
B_r -17326.027800 B_z 35.216459 B_m 17326.063600
χ 741.514563 μ 742.514563

	type	radial dim. axial dim.	radial pos. axial pos.
Cell 5 **Comp. 5**	soft	12. 8.	54. -34.

Hm_r -2.762674 Hm_z 0.202081 Hm_{mod} 2.770055
M_r -12418.586900 M_z 908.379570 M_{mod} 12451.765100
B_r -12421.349600 B_z 908.581651 B_m 12454.535200
χ 4495.133060 μ 4496.133060

■ 9-4 Para and Diamagnetic Materials under a Very High Field

■ 9-3-1 Introduction

The example is an extension of the method applied to common diamagnetic or paramagnetic materials. Usually their magnetic behavior can be neglected. However, under very high magnetic fields such as used in magnetic resonance spectroscopy (up to 20 Tesla) the magnetization of the materials that constitute the coil sensor interferes with the resonance signal. The example shows the application of the software for this very special magnetic problem. 3 cases are developed each with different materials.

■ 9-3-2 Materials Properties

1. Material : copper OFHC Volume relative diamagnetic susceptibility: - 0.000 000 61

```
Properties[CopperOFHC]
```

Material properties
```
material name                        CopperOFHC
material category                    Resistive current carrying
designation                          high conductivity OFHC Copper
```
General Physical Properties
```
specific weight (g/cm3)                    8.920000
dilation coefficient                       0.000016
    (relative linear change)
resistivity (Ohm x cm)                     1.724000e-6
Coefficient temperature resistivity        0.003930
```
Magnetic Physical Properties
```
relative volume magnetic                  -6.100000e-7
    (linear) susceptibility
temperature coefficient susceptibility     0.000010
```

2. Material: electrical quality aluminum Volume relative paramagnetic susceptibility: 0.000 001 40

```
Properties[AlumElec]
```

Material properties
```
material name                        AlumElec
material category                    Resistive current carrying
designation                          high conductivity Aluminum
```
General Physical Properties
```
specific weight (g/cm3)                    2.700000
dilation coefficient                       0.000022
    (relative linear change)
resistivity (Ohm x cm)                     2.820000e-6
Coefficient temperature resistivity        0.004200
```
Magnetic Physical Properties
```
relative volume magnetic                   1.400000e-6
    (linear) susceptibility
temperature coefficient susceptibility    -0.000010
```

3. Material: copper aluminum composite Volume relative paramagnetic susceptibility: 0.000 000 029

```
Properties[Composit1]
```

```
Material properties
material name                           Composit1
material category                       Resistive current carrying
designation                             Copper (.682) and Aluminum (.318) Composit
General Physical Properties
specific weight (g/cm3)                        7.540000
dilation coefficient                           10.000000e-7
    (relative linear change)
resistivity (Ohm x cm)                         1.970000e-6
Coefficient temperature resistivity            0.000000
Magnetic Physical Properties
relative volume magnetic                       2.920000e-8
    (linear) susceptibility
temperature coefficient susceptibility         0.000000
```

■ 9-3-2 Entering the Data

The examples show 3 cases for a tiny sensor coil submitted to an excitation field of 10 teslas using a meshing level of 1 for quick evaluation :

Case 1: copper coil sensor (copper is a diamagnetic material)

Case 2: aluminum coil sensor (aluminum is a paramagnetic material)

Case 3: copper-aluminum composite coil sensor (the composite is quasi amagnetic; susceptibility very low)

Case 1: copper coil sensor (diamagnetic)

```
innerDiam = 0.1;
length = 0.01;
thick = 0.001;
outerDiam = innerDiam + 2 * thick;
position1 = length * 3 / 2 + 15 * thick;
position2 = length / 2 + 5 * thick;
bApplied = 100000;
sensor1 = AXSMagneticSystem[{
    AXSLargeMagneticSource[bApplied],
    AXSSoftMagneticComponent[
     CopperOFHC, innerDiam, outerDiam, length, -position1],
    AXSSoftMagneticComponent[CopperOFHC, innerDiam, outerDiam, length, -position2],
    AXSSoftMagneticComponent[CopperOFHC, innerDiam, outerDiam, length, position2],
    AXSSoftMagneticComponent[CopperOFHC, innerDiam, outerDiam, length, position1]},
   SystemLabel -> "double Copper flat conductor" , MeshingLevel → 1];
```

Case 2: aluminum coil sensor (paramagnetic)

```
innerDiam = 0.1;
length = 0.01;
thick = 0.001;
outerDiam = innerDiam + 2 * thick;
position1 = length * 3 / 2 + 15 * thick;
position2 = length / 2 + 5 * thick;
bApplied = 100000;
sensor2 = AXSMagneticSystem[{
    AXSLargeMagneticSource[bApplied],
    AXSSoftMagneticComponent[AlumElec, innerDiam, outerDiam, length, -position1],
    AXSSoftMagneticComponent[AlumElec, innerDiam, outerDiam, length, -position2],
    AXSSoftMagneticComponent[AlumElec, innerDiam, outerDiam, length, position2],
    AXSSoftMagneticComponent[AlumElec, innerDiam, outerDiam, length, position1]},
   SystemLabel -> "double Aluminium flat conductor" , MeshingLevel → 1];
```

Case 3: copper-aluminium composite coil sensor (quasi amagnetic)

```
innerDiam = 0.1;
length = 0.01;
thick = 0.001;
outerDiam = innerDiam + 2 * thick;
position1 = length * 3 / 2 + 15 * thick;
position2 = length / 2 + 5 * thick;
bApplied = 100000;
sensor3 = AXSMagneticSystem[
   {AXSLargeMagneticSource[bApplied],
    AXSSoftMagneticComponent[Composit1, innerDiam, outerDiam, length, -position1],
    AXSSoftMagneticComponent[Composit1, innerDiam, outerDiam, length, -position2],
    AXSSoftMagneticComponent[Composit1, innerDiam, outerDiam, length, position2],
    AXSSoftMagneticComponent[Composit1, innerDiam, outerDiam, length, position1]},
   SystemLabel -> "double composite Aluminium/Copper flat conductor" ,
   MeshingLevel → 1];
```

■ 9-3-3 System Drawing

Drawing Case 1. (the geometry is identical for all cases)

```
DrawSystem2D[sensor1];
```

```
System name: sensor1
double Copper flat conductor
2D view ZR
```
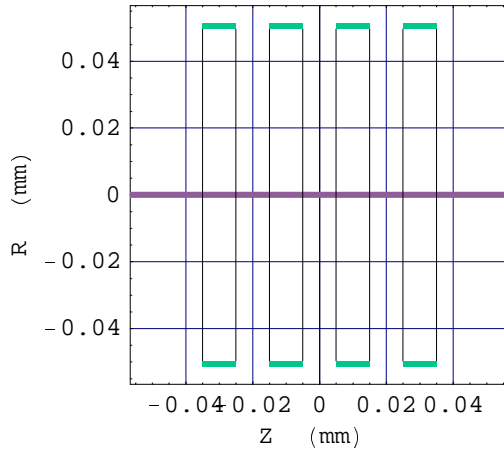

System technical data Case 1

```
Properties[sensor1]
```

General properties

```
System name:                                      sensor1
System magnetic parity                            True
System Legendre radius                            0.056052
Magnetic field at the origin (G)                  {0, 100000.}
Field magnitude at the origin(G)                  100000.
System mass  (Kg)                                 1.13213 × 10⁻¹⁰
System nominal power (W)                           0
System effective power(W)                          0
Number of components                               5
Number of ferro-magnetic (non rigid) components    4
Number of components with reversible magnetization 4
Number of large sources                            1
```

n°	type	first cell	last cell	radial cells	axial cells
2	soft	1	5	1	5
3	soft	6	10	1	5
4	soft	11	15	1	5
5	soft	16	20	1	5

■ **9-3-4 Field and Magnetization Map**

1. Case 1, copper coil. The magnetic perturbation of the coil winding at the isocenter is 23.22 Nano Tesla

```
MagneticaUserUnits[FieldUnit → Nano Tesla];
FieldPointOrigin[sensor1, ComponentsList → {2, 3, 4, 5}]

System name: sensor1
The magnetic field (in Nano Tesla) at the origin
Calculation mode: Automatic Mode


Component n° 2
Component type: cylindrical, reversible magnetization
Br          0.000000 nT
Bz          1.718580 nT
Bm          1.718580 nT


Component n° 3
Component type: cylindrical, reversible magnetization
Br          0.000000 nT
Bz          9.910264 nT
Bm          9.910264 nT


Component n° 4
Component type: cylindrical, reversible magnetization
Br          0.000000 nT
Bz          9.910264 nT
Bm          9.910264 nT


Component n° 5
Component type: cylindrical, reversible magnetization
Br          0.000000 nT
Bz          1.718580 nT
Bm          1.718580 nT


Total
Br          0.000000 nT
Bz         23.257688 nT
Bm         23.257688 nT
```

2. Case 2: Aluminum coil. The magnetic perturbation of the coil winding at the isocenter is -53.36 Nano Tesla

```
MagneticaUserUnits[FieldUnit → Nano Tesla];
FieldPointOrigin[sensor1, ComponentsList → {2, 3, 4, 5}]
```

```
FieldPointOrigin[sensor2,ComponentsList→{{2,5}}]

System name: sensor2
The magnetic field (in Nano Tesla) at the origin
Calculation mode: Automatic Mode


Component n° 2
Component type: cylindrical, reversible magnetization
Br          0.000000 nT
Bz         -3.944282 nT
Bm          3.944282 nT


Component n° 3
Component type: cylindrical, reversible magnetization
Br          0.000000 nT
Bz        -22.744863 nT
Bm         22.744863 nT


Component n° 4
Component type: cylindrical, reversible magnetization
Br          0.000000 nT
Bz        -22.744863 nT
Bm         22.744863 nT


Component n° 5
Component type: cylindrical, reversible magnetization
Br          0.000000 nT
Bz         -3.944282 nT
Bm          3.944282 nT


Total
Br          0.000000 nT
Bz        -53.378290 nT
Bm         53.378290 nT
```

3. Case 3, copper-aluminum coil. The magnetic perturbation of the coil winding at the isocenter is -1.10 Nano Tesla

```
FieldPointOrigin[sensor3,ComponentsList→{{2,5}}]
```

```
System name: sensor3
The magnetic field (in Nano Tesla) at the origin
Calculation mode: Automatic Mode


Component n° 2
Component type: cylindrical, reversible magnetization
Br            0.000000 nT
Bz           -0.082266 nT
Bm            0.082266 nT


Component n° 3
Component type: cylindrical, reversible magnetization
Br            0.000000 nT
Bz           -0.474393 nT
Bm            0.474393 nT


Component n° 4
Component type: cylindrical, reversible magnetization
Br            0.000000 nT
Bz           -0.474393 nT
Bm            0.474393 nT


Component n° 5
Component type: cylindrical, reversible magnetization
Br            0.000000 nT
Bz           -0.082266 nT
Bm            0.082266 nT


Total
Br            0.000000 nT
Bz           -1.113319 nT
Bm            1.113319 nT
```

4. Plot of the Bz field on an axial line for Case 3 copper-aluminum composite coil.

```
MagneticaUserUnits[FieldUnit → Tesla];
FieldMap[sensor3, Bz, Line[{0.03, -0.03}, {0.03, 0.03}]];
```

> System name: **sensor3**
> *double composite Aluminium/Copper*
> *flat conductor*
> Bz (T) (Continuous, Automatic Mode)
> between {0.03, -0.03} and {0.03, 0.03}
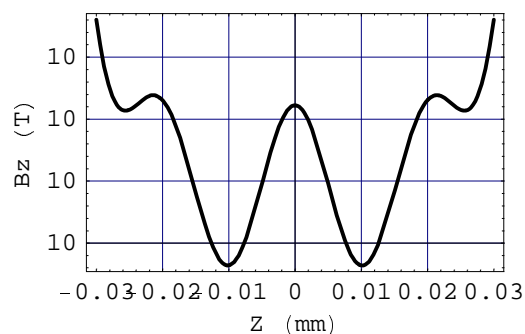
5. Contour plot of the magnitude Bm of the magnetic field on a rectangle of 0.06mm×0.06 mm dimensions, for the Case 3 copper-aluminum composite coil.

```
ContourFieldMap[sensor3, {Bm},
  Plane[{-0.03, -0.03}, {0.03, 0.03}], NumberOfPoints → 30];
```

System name: **sensor3**
double composite Aluminium/Copper
flat conductor
Bm (T) (Continuous, Automatic Mode)
between {-0.03, -0.03} and {0.03, 0.03}
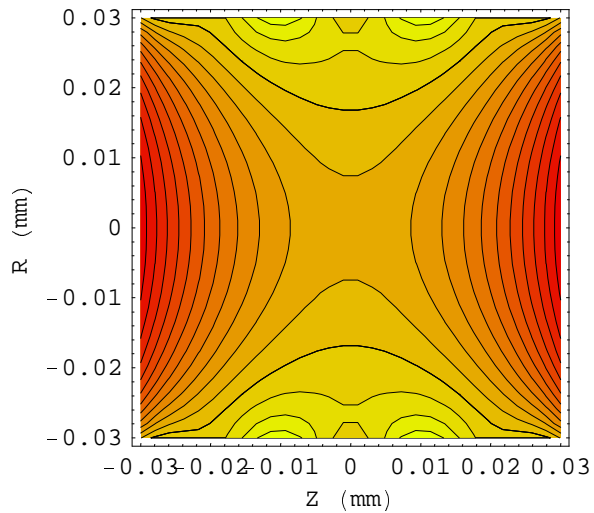
■ Chapter 10: Fundamentals of Inductance

■ 10-1 Fundamentals of Inductance

■ 10-1-1 The Basic Concepts of Inductance

The concept of inductance appears generally as something connected with the time varying applications of electrical circuits. This approach suggests, mistakenly, that inductance is not within the scope of magnetostatics. On the other hand, if we look at the physical definition of inductance, it is simply the ratio of the magnetic flux in a circuit divided by a pertinent current intensity, both basic data of magnetostatics for current carrying components:

$$\mathcal{I} = \frac{\phi}{i} \tag{1}$$

More specifically, the flux ϕ is the flux generated on the target circuit (t) from the source circuit (s). The definition of source and target appears essential for a good understanding of inductance calculation methodology.

The following inductance concepts must be distinguished:

- **Self-inductance.** The source and target are identical but the methodology of calculation remains unchanged. The self-inductance $\mathcal{L}$ (t) in the circuit (t) is the inductance resulting from the flux ϕ (t) generated by its own current i(s):

$$\mathcal{L}(t) = \frac{\phi(t)}{i(s)} \tag{2}$$

    Self-inductance is a positive number whatever the direction of rotation of the turns, clockwise or counterclockwise, and the direction of current. Indeed $\mathcal{L}$ is proportional to the square of the number of turns.

- **Mutual inductance** $\mathcal{M}$ between 2 current carrying circuits. In that case, the source and target are distinct. The mutual inductance resulting from the magnetic flux generated by the source circuit (s) acting on the target circuit (t) is:

$$\mathcal{M}(s{\rightarrow}t) = \frac{\phi(t \text{ from } s)}{i(s)} \tag{3}$$

    It can be shown, from the application of Lenz's law for the conservation of flux, that the reciprocal mutual inductance (t→s) is identical to the direct mutual inductance (s→t):

$$\mathcal{M}(t{\rightarrow}s) = \frac{\phi(s \text{ from } t)}{i(t)} = \mathcal{M}(s{\rightarrow}t) \tag{4}$$

    It is now necessary to define more precisely the sign to be used for $\mathcal{M}$. The sign depends not only on the direction of the winding, either clockwise or counterclockwise, but also on how the 2 circuits are connected. If the 2 circuits are not connected, the sign depends only on the direction of the winding If the winding are identical: the sign is positive; if they are different, then the sign is negative. If 2 circuits are electrically series connected with identical winding directions, then $\mathcal{M}$ is positive. Otherwise it is negative. The next 2 figures show these cases.

    Figure 1: the 2 coil systems are identically wound and electrically series connected, then the sign of $\mathcal{M}$ is positive.

**Schematic of inductance coupling
between 2 current carrying coils
magnetically in series**

current +

current ·

mutual M12

coil 1

self L11

coil 2

self L22
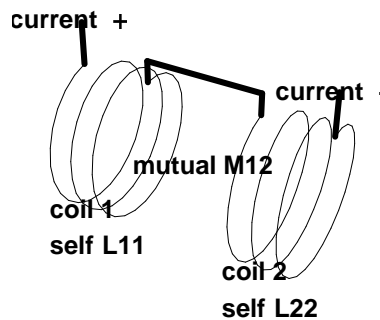
Figure 2: the 2 coils are wound in opposite directions and electrically series connected, then the sign
of $\mathcal{M}$ is negative.

**Schematic of inductance coupling
between 2 current carrying coils
magnetically in opposition**

current +

current ·

mutual M12

coil 1

self L11

coil 2

self L22
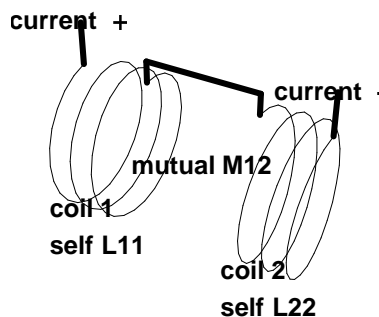
It must be pointed out, that *Magnetica* automatically takes care of the sign of $\mathcal{M}$ from the data
defining the magnetic system's current-carrying.

▪ **The rule of association** of self and mutual inductance. The question is, what is the total inductance of
a magnetic system made of several current carrying coils.

To understand this concept, let's take a magnetic system made of 2 identically wound coils and
electrically series connected (and therefore magnetically series connected as shown in the
previous Figure 1) then the total inductance of the system is:

$\mathcal{S}$total = 2 x $\mathcal{L}$ + 2 x $\mathcal{M}$ (5)

It can be shown that the self-inductance of a coil is proportional to the square of the number of turns
Ns, the factor of proportionality **Fself** depending on the size of the coil and on its shape. (see for
instance the Grover-Nagaoka formula for approximate calculation of $\mathcal{L}$) .

$\mathcal{L}$ = **Fself x Ns^2**   (6)

Similarly, it can be shown that mutual inductance is proportional to the product of the number of
turns of the source **Ns** and target **Nt**, the factor of proportionality **Fmutual** depending on the size
and shape of the 2 coils and also on their relative positions.

$$\mathcal{M} = \text{Fmutual x Ns} \times \text{Nt} \tag{7}$$

The 2 coils being identical with therefore the same number of turns, the mutual inductance equation becomes:

$$\mathcal{M} = \text{Fmutual x Ns}^2 \tag{8}$$

Now let's imagine that the 2 coils are identical coils and that they are merged such that they become a single coil. Then it can be shown that **Fmutual** → **Fself** since the size and shape are identical and that the relative position dependence becomes maximal, which is the case for self-inductance. Consequently, the mutual inductance $\mathcal{M}$ becomes identical to $\mathcal{L}$. This is the maximum possible magnitude for $\mathcal{M}$.

$$\mathcal{M} = \mathcal{L} \tag{9}$$

Therefore, the total inductance of 2 identical merged coils becomes :

$$\mathcal{L}2 = \mathcal{S}\text{total} = 4 \times \mathcal{L}1 \tag{10}$$

If the 2 coils are not identical and if the same rationale is pursued for the merging process, it can be shown that the maximum of the mutual inductance is given by the following relation:

$$\mathcal{M}\text{ maximum} = \sqrt{\mathcal{L}s \times \mathcal{L}t} \tag{11}$$

where $\mathcal{L}s$ and $\mathcal{L}t$ are the respective self-inductance of the source and target.

- the **inductance coupling**. This concept encompasses the measurement of the efficiency of the magnetic coupling between 2 coils. It is of the utmost importance for all problems involving coupled inductive devices. The following factors are used:

- the **coupling coefficient** : this is the ratio between the effective mutual inductance $\mathcal{M}$st and the maximum (theoretical) mutual inductance:

$$\mathcal{C}c(s \rightarrow t) = \frac{\mathcal{M}st}{\mathcal{M}max} = \frac{\mathcal{M}st}{\sqrt{\mathcal{L}s \times \mathcal{L}t}} \tag{12}$$

- the **coupling efficiency**: this measures the efficiency in terms of magnetic power transmitted (energy per unit of time):

$$\mathcal{C}e(s \rightarrow t) = \mathcal{C}c^2 = \frac{\mathcal{M}st^2}{\mathcal{L}s \times \mathcal{L}t} \tag{13}$$

- the **leakage coefficient**: this measures the non efficiency (i.e. the leakage) in terms of electromagnetic power:

$$\mathcal{C}k(s \rightarrow t) = 1 - \mathcal{C}e = \frac{\mathcal{L}s \times \mathcal{L}t - \mathcal{M}st^2}{\mathcal{L}s \times \mathcal{L}t} \tag{14}$$

- the **transformation coefficient**: this gives, for a transformer, i.e. a device made of 2 independent coils, the effective ratio between the secondary voltage and the primary voltage. It can be shown that this ratio is the quotient of the respective flux. In the case of good magnetic coupling, this coefficient approaches asymptotically the theoretical transformation ratio which is simply the ratio of the number of turns.

$$\mathcal{C}tr(s \rightarrow t) = \frac{\phi ct + \phi cst}{\phi cs + \phi cts} = \frac{Vt}{Vs} \rightarrow \frac{Nt}{Ns} \tag{15}$$

■ 10-1-2 The Concepts of Ferro Inductance

This is an extension of the concept of inductance for magnetic systems including ferromagnetic structures. Under the excitation of the magnetic field generated by the source coil (s) the ferromagnetic structures become magnetized and that induces a new flux $\phi\mathbf{f}$ on the target coil creating a complementary self-inductance $\mathcal{F}$s(t): the ferro inductance.

This concept is displayed in the next figure.

Figure 3 represents a single current-carrying coil with a bar of soft ferromagnetic material at the center.

Schematic of ferro inductance coupling between a current carrying coil and a ferromagnetic core
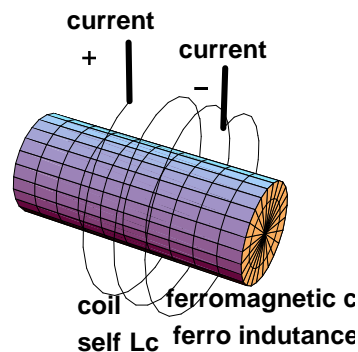
The magnetization $\mathcal{M}\mathbf{c(s)}$ of the ferromagnetic structures depends both on the structures shape, size and magnetic properties of the material as well as the coil current intensity and position with respect to the magnetic structure. On the one hand, it is obvious that materials with high magnetic permeability are naturally much more efficient than low permeability materials and also that the effect will be improved if the ferromagnetic structures are set very close to the winding.

We have explained in Chapter 6 the basic principle for the calculation of magnetization and shown in Chapter 8 the result of the calculation for a cylindrical bar of common steel excited by a single solenoid (section 8.3). The solution provides the magnetization M for all volumes of the ferromagnetic structure via the characteristics of the magnetic cells. This is equivalent to having the full definition of the "magnetic circuit", such as described in most textbooks on electrical engineering (but without providing any solutions). As we have shown in these chapters, each magnetic cell can be processed as a field generator allowing for the precise calculation of the magnetic flux on the target coil.

- **Ferro Self- Inductance $\mathcal{F}$s.** As for direct self-inductance, the source and the target are confounded and the excitation of the ferromagnetic is considered to be due only to the source coil.

$$\mathcal{F}\mathbf{s(t)} = \frac{\phi\mathbf{f}(t)}{i(s)} \tag{16}$$

- **Ferro Mutual Inductance $\mathcal{F}$m**. This concept arises in the same way as for the direct mutual inductance in the case of 2 (or more) coils with the same defintion of source and target. Equation (3), becomes:

$$\mathcal{F}\mathbf{m(s \to t)} = \frac{\phi\mathbf{f}(s\,to\,t)}{i(s)} \tag{17}$$

Magnetica uses a specific and proprietary algorithm (patent pending) to calculate both the ferroself and the ferromutual inductance.

■ 10-1-3 The Concept of Total Inductance

- **Total Self Inductance** $\mathcal{G}$. This is simply the summation of the 2 effects, the direct (current wise) inductance from the electrical coil and the inductance from the ferromagnetic components:

$$\mathcal{G}(t) = \mathcal{L}(t) + \mathcal{F}s(t) \qquad\qquad (18)$$

- **Total Mutual Inductance** $\mathcal{M}$g. This is applicable in a way similar to self-inductance:

$$\mathcal{M}g(t) = \mathcal{M}c(t) + \mathcal{F}m(t) \qquad\qquad (19)$$

The concept of total inductance, including both the direct inductance and ferro inductance, is fundamental in the physics of many electromagnetic devices such as the transformers.

- **Inductance Coupling** (in the case of a ferromagnetic core): The formulas for the inductance coupling in the case of a ferromagnetic core must now incorporate the contribution of both the air core coupling and the iron core coupling. The formulas for the various coefficients become the following:

- **Coupling Coefficient**: this is the ratio between the total actual mutual inductance Mg and the maximum total mutual inductance:

$$Cfc(s{\to}t) = \frac{\mathcal{M}gst}{\mathcal{M}max} = \frac{\mathcal{M}st + \mathcal{F}m(st)}{\sqrt{\mathcal{L}s \times \mathcal{L}t} + \mathcal{F}m(st)} \qquad\qquad (20)$$

The direct consequence of using a ferromagnetic core around the current-carrying coils is to increase substantially the coupling coefficient rendering the AC voltage transformer an efficient device.

- **Coupling Efficiency**: this measures the efficiency in terms of electromagnetic power (energy per unit of time)

$$Cfe(s{\to}t) = Cfc^2 \qquad\qquad (21)$$

- **Leakage Coefficient**: this measures the non efficiency (i.e. the leakage) in terms of electromagnetic power

$$Cfk(s{\to}t) = 1 - Cfe \qquad\qquad (22)$$

- **Transformation Coefficient**: the definition is the same as for the air core transformer, but takes into account the contribution of the iron core, the formula becomes:

$$Cftr(s{\to}t) = \frac{\phi gt}{\phi g\ s} = \frac{\phi c(t)+\phi f(t)+\phi c(st)+\phi f(st)}{\phi c(s)+\phi f(s)+\phi c(ts)+\phi f(ts)} = \frac{Vt}{Vs} \to \frac{Nt}{Ns} \qquad (23)$$

■ 10-1-4 Magnetic Energy

Magnetic energy is associated with a magnetic field. In some devices, magnetic energy is of utmost importance and its calculation becomes an essential feature. Magnetic energy calculation is directly related to inductance calculation. The basic general equation for magnetic energy is:

$$Wmag = \frac{1}{2 \times \mu v} \int B^2\, dv \qquad\qquad (24)$$

where the integration must be carried out for all space up to infinity . It can be shown that the magnetic energy of an electrical circuit or ferromagnetic core can be calculated using the relevant inductance for specific single inductance elements:

$$Wself = \frac{1}{2} \mathcal{L} \times i(t)^2 \qquad\qquad (25)$$

$$\textbf{Wmutual} \quad = \quad \tfrac{1}{2} \quad \mathcal{M}\text{c(s or t)} \quad \text{x} \quad i\,(s) \quad \text{x} \quad i\,(t) \qquad\qquad (26)$$

$$\textbf{WFerroSelf} \quad = \quad \tfrac{1}{2} \quad \mathcal{F}\text{s} \quad \text{x} \quad i(t)^{\,2} \qquad\qquad\qquad (27)$$

$$\textbf{WFerromutual} = \quad \tfrac{1}{2} \quad \mathcal{F}\text{m(s or t)} \quad \text{x} \quad i\,(s) \quad \text{x} \quad i\,(t) \qquad\qquad (28)$$

The 4 above equations give only the part of the global magnetic energy which is available in the current carrying coils due to the magnetic coupling. However, the total magnetic energy of the ferromagnetic components can be calculated using Equation (24) where B means the total induction field (M + Hm) and where the integration is carried out for the full volume of the magnetic components.

■ 10-2 Inductance Calculation Using *Magnetica*

■ 10-2-1 Inductance Calculation: Basic Principles

As we have seen in the previous section, the inductance calculation is fundamentally related to magnetostatics. Nevertheless, its use concerns a large number of time-dependent electromagnetic problems and the corresponding devices. It is not wihin the scope of *Magnetica* to solve these problems, but only to develop results useful for their solution.

The results of inductance calculation can be presented in various ways depending on the complexity of the magnetic system and on the objectives of the calculations.

The simplest case is a magnetic system with a single current carrying circuit. Then its inductance is the self-inductance $\mathcal{L}$ which is given by Equation (2) or (18), where *t* refers to the target coil.

$$\pounds = \mathcal{L}(t) \ \text{ or } \pounds = \mathcal{G}(t) \qquad\qquad\qquad\qquad (29)$$

For multi-component magnetic systems, the most convenient and significant expression to visualize all the relevant elements is the inductance matrix (shown here for a 2 circuit system):

$$\mathbb{m}\textbf{c} \; = \; \begin{pmatrix} \textbf{L11} & \textbf{M12} \\ \textbf{M21} & \textbf{L22} \end{pmatrix} \qquad\qquad\qquad (30)$$

where the matrix rows are seen as the magnetic source components and the matrix columns as the target components. The question of the sign of each element now becomes an intrinsic feature since the magnetic behavior of the circuits depends on the way the coils are both wound and connected (series, opposition, independent).

The implementation of the concept of ferro inductance is carried out through rules similar to those of air core inductance so that the association of air core inductances and ferro core inductances is straightforward. The ferro self-inductance $\mathcal{F}$s of a current-carrying component is calculated by assuming it is the only source, all other sources being ignored.

Similarly ferro mutual inductance $\mathcal{F}$m is calculated assuming that only the relevant pair of current carrying components is active and taking into account the correct association rule. This method provides satisfactory values for the pair $\mathcal{F}$m(1→2) and $\mathcal{F}$m(2→1), the small differences being accountable to the non linearity of the magnetization curves and the approximation on the current data, particularly the meshing level.

Consequently the ferromagnetic inductance matrix is:

$$\mathbb{m}\textbf{f} \; = \; \begin{pmatrix} \mathcal{F}\textbf{s11} & \mathcal{F}\textbf{m12} \\ \mathcal{F}\textbf{m21} & \mathcal{F}\textbf{s22} \end{pmatrix} \qquad\qquad\qquad (31)$$

Then the total inductance matrix is simply the summation of the 2 contributing matrices:

$$\mathbb{M}\boldsymbol{t} = \mathbb{M}\boldsymbol{c} + \mathbb{M}\boldsymbol{f} = \begin{pmatrix} \mathtt{Mt11} & \mathtt{Mt12} \\ \mathtt{Mt21} & \mathtt{Mt22} \end{pmatrix} \qquad\qquad (32)$$

For the complete magnetic system, the total inductance is simply the summation of all matrix elements (which entails taking into account all their respective signs due to electric connections and direction of the windings). The total inductance corresponds to the self-inductance of the system assuming that all the coils are electrically series connected:

$$\textbf{Ltotal} = \sum (\mathbb{M}\mathtt{t11} + \mathbb{M}\mathtt{t12} + \mathbb{M}\mathtt{t21} + \mathbb{M}\mathtt{t22}) \qquad\qquad \textbf{(33)}$$

The matrix representation applies to all inductance-related physical data such as magnetic flux and magnetic energy. The total flux matrix $\mathbb{M}\phi t$ is given by the following equations where i1 and i2 are the current intensities in the respective coils.

$$\mathbb{M}\phi\boldsymbol{t} = \begin{pmatrix} \mathtt{Mt11} & \mathtt{Mt12} \\ \mathtt{Mt21} & \mathtt{Mt22} \end{pmatrix} \boldsymbol{x} \,(\,\mathtt{i1} \quad \mathtt{i2}\,) \qquad\qquad (34)$$

The total magnetic energy matrix $\mathbb{M}\mathcal{E}t$ is similarly.

$$\mathbb{M}\mathcal{E}\boldsymbol{t} = \frac{1}{2} \, \boldsymbol{x} \,(\,\mathtt{i1} \quad \mathtt{i2}\,) \, \boldsymbol{x} \begin{pmatrix} \mathtt{Mt11} & \mathtt{Mt12} \\ \mathtt{Mt21} & \mathtt{Mt22} \end{pmatrix} \boldsymbol{x} \,(\,\mathtt{i1} \quad \mathtt{i2}\,) \qquad (35)$$

■ 10-2-2 Inductance Calculation : *Magnetica* Definitions and Syntax

Inductance calculations can be done many ways depending on the objective pursued. The basic *Magnetica* functions and options are: (see the next chapter on Immediate Help for details on the function syntax).

- **SelfInductance**: basic inductance of a single current carrying component with option for the direct (air core) contribution and the ferro contribution.

- **MutualInductance**: basic cross-inductance of a pair of current carrying components with the same basic option.

- **SystemInductance**: complete inductance calculation taking into account all the components of a magnetic system.

- **InductanceMatrix**: inductance elements for magnetic systems made of single or multi current-carrying components and shown as a square matrix source / target

- **FluxInductanceMatrix**: flux inductance elements of the flux system matrix similar to the Inductance Matrix.

- **MagneticEnergyMatrix**: inductance magnetic energy elements of the magnetic energy system matrix similar to the Inductance Matrix.

- **InductanceCoupling** : calculation of the magnetic coupling parameters between 2 current carrying coils of a magnetic system. The method is based on the traditional formulas found in Electrical Engineering textbooks. This can be applied to any pair of coils and *Magnetica* takes into account the ferromagnetic core. But, it must be remembered that these formulas assume a very crude approximation for the magnetic circuit and that they do not take into account the effect of the frequency.

- **TransformerAnalysis**: calculation of the relevant parameters of a monophase transformer for a given frequency and load. The method takes into account the complete geometry of the current-carrying coils as well as of the ferromagnetic structure using the *Magnetica* standard for entering data. The frequency of the current is also a parameter of the TransformerAnalysis algorithm. The result is a

much better approximation of the transformer's behavior providing many useful data such as active and reactive currents, phase angle and power losses.

- **TransformerMagneticLosses**: calculation of the magnetic losses in the ferromagnetic structure of a transformer made of materials with known curves of losses versus the Induction and for a given frequency. (N.B. for Silicon Steel and MuMetal exclusively)

- **Option Parameters**

  - **ContributionType:** This option parameter specifies the scope of applications of the basic functions according to the following choice:

    - **Proper**: the calculation is restricted to the current carrying component specified (this is the default option)

    - **Ferro**: the calculation applies to the contribution of all Ferromagnetic components (in case of absence of such components the null value is returned).

    - **Total**: this works out the 2 previous calculations and gives the global inductance

  - **PrecisionLevel:** This option parameter specifies the precision of the calculation:

    - **Level 1**: to provides, **at least**, a precision in the range of 0.1 percent, i.e. 3 decimal places. This is the default case. This precision is sufficient for most of the applications.

    - **Level 2**: the precision is in the range of 10 ppm, i.e. 5 decimal places. The calculation is much longer.

    - **Level 3**: the precision is in the range of 0.1 ppm, i.e. 7 decimal places. The calculation is much, much longer.

  - **ComponentsList:** This option parameter specifies the list of the components of the magnetic system on which the inductance calculation will be done.

- **VerboseSelfInductance[....]** Prefixing with the word "Verbose" any of the previous functions provides a detailed and/or formatted output for self inductance.

The *Magnetica* default units for inductance and magnetic energy are respectively Micro Henry and Micro Joules. However, any other units can be used using **MagneticaUserUnits**[InductanceUnit→Henry] function for instance.

The complete set of actual *Magnetica* units is displayed with:

```
MagneticaCurrentUnits[]
```

```
Length                      Milli Meter
Magnetic field              Tesla
Magnetic potential          Micro Weber/Meter
Magnetic flux               Micro Weber
Magnetic field gradient     Gauss/Meter
Inductance                  Micro Henry
Magnetic energy             Joule
Magnetic moment             Micro Ampere Meter²
Current intensity           Ampere
Power                       Watt
Voltage                     Volt
Electrical Resistance       Ohm
```

■ **10-2-3 Basic Inductance Parameters and Approximate Formulas**

The precise calculation of inductance is an intricate process implying many parameters, some related to the shape of the winding, and for mutual or ferro inductance, to the distance between the components. However, 2 parameters are prominent:

  1. the dimension of the coils materialized by the mean coil diameter

  2. the number of turns of conductors:

   ▪ for self-inductance and ferro inductance the effect depends on the square of the number of turns Nt, the field being proportional to Nt and the flux in circuit being proportional to the field and again to the number of turns.

   ▪ for mutual inductance this is the product Ns x Nt

The dependence on coil shape and relative distance is very unlinear and cannot be given by simple formulas.

Many publications and books have been devoted to inductance calculation and all Electrical Engineering handbooks give approximate formulas, tables and curves. The most used of theses formulas are known as Nagaoka's, Grover's or Terman's formulas from the names of their authors. A good account is given in the book by Grover: (Grover, Inductances Calculation Working Formulas and Tables, Dover, 1946). But the precision of these formulas is rather limited, usually in the percent range or less, and in the best cases to a few thousands. This requires interpolation on numerical tables and the process is quite laborious. Moreover their ranges of applications are also limited.

The Nagaoka-Grover formula applies a to thin solenoid. The coefficients Fnagaoka and Fsolenoid are given from tables as functions of the ratios depending on both the length *b* and the height *h* of the winding and the mean diameter *dm*. For dimensions in mm, the result is in μHenry, and we have:

$$\mathcal{L}\text{ng} = 0.0009865 \text{ x } \frac{\text{dm}^2}{b} \text{ x Nturns}^2 \text{ x (Fnagaoka-Fsolenoid)} \qquad (36)$$

The Grover formula applies to a thick solenoid including a solenoid of square cross section. The coefficient Fgrover is a function of the ratio h / dm and b /dm as defined above. We have then, using the same units:

$$\mathcal{L}\text{gr} = 0.00005 \text{ x dm x Nturns}^2 \text{ x Fgrover} \qquad (37)$$

Brook's configuration is a special case of square solenoid for which the inner radius is identical to the side of the winding (see next figure). This configuration is often presented as generating the highest inductance for a given amount of winding material but this appears valid only under certain constraints.

$$\mathcal{L}\text{br} = 0.0008497 \text{ x dm x Nturns}^2 \qquad (38)$$

Magnetic System : solenoid2
square solenoid type BROOK
2D View ZR
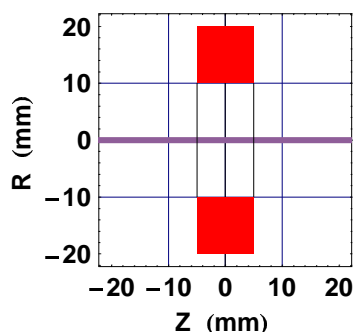
■ 10-3 Inductance Applications

■ 10-3-1 Introduction

Although the basic concept and the basic methodology of the calculation of inductance are related to the Magnetostatics, the applications of inductance deal essentially with time varying devices. Then, we can question if the inductance calculated from the static case can be used for such time varying circuits. A first answer is obtained from the fundamental Physics involved in these devices. That implies that the **magnetic coupling** is the dominant phenomenon and consequently that the **electromagnetic radiation coupling** is a minor phenomenon. If d is the dimension of the device (mm) and f the frequency (Hertz), the condition is given by the formula (related to the speed of the light):

$$d \ \text{x} \ f \ \ll 5 \times 10^{10} \qquad\qquad\qquad (39)$$

For the domestic AC power device with a frequency in the range of 50 / 60 Hertz and dimensions of a meter or less, the condition is straightforward: the coupling is totally magnetic. For electronic devices the answer is sensitive. In the gigahertz range, the dimensions are of the order of a millimeter.

Assuming that this basic condition is fulfilled, then inductance appears as a physical concept independent of frequency. However, various secondary phenomena connected to the properties of the materials may alter the fundamental behavior of the device. Time changing voltage generates "eddy currents" particularly if the substance of the coils is used in a uniform bulk state. Suitable division of the substance into thin electrically isolated foils, reduces drastically the deleterious eddy current effect. At very high frequencies, a new phenomenon occurs: the "skin effect". The distribution of the current inside the conductors is modified so that the resulting flux and therefore the static inductance calculation becomes altered.

Nevertheless, these secondary deleterious effects can be drastically reduced if the materials are properly adapted. Exhaustive experimental studies made a few decades ago, on inductance devices show a decrease of only 5% in the high frequency range with the theoretical, zero frequency, inductance. (Terman, Radio Engineers Handbook , 1943).

On the other hand, since these effects are specific to substance and not related to the fundamental methodology of calculation, it is always possible to append to the basic *Magnetica* software a special function in order to take them into consideration in the case where these substances are used.

The applications of inductance to practical device analysis or design can be classified into 3 main categories :

   ▪ devices related to resonant circuits,

   ▪ devices related to magnetic coupling circuits such as transformers,

- special applications devices.

■ 10-3-2 Resonant Circuits Analysis

A resonant circuit implies the use of a capacitor associated with an inductance. The basic formula for resonant circuit is:

$$\mathbf{fr} \ = \ \frac{1}{2\pi \times \sqrt{L \times C}} \qquad\qquad (40)$$

where **fr** is in Hertz, when **L** in Henry and **C** in Farad. In the case of a device with a ferromagnetic core, the total inductance must be used.

The topics related to resonant circuits are numerous, such as:

- measurement of an inductance, knowing **fr** and C

- generating a specific resonant frequency **fr** for a given coil by adjusting C

- designing an inductance L with special constraints to obtain a specific resonant frequency **fr** using a given capacitor C.

Designing is the most exacting task and the constraints must be clearly defined before starting the design. For instance, if the constraint is simply the minimization of the dimensions, then using Brooks's approximate formula is a first guess. It is, then possible to proceed with better data to improve the design. See Chaper 13 on inductance Application, Cases 1 and 2.

■ 10-3-3 Transformer Analysis

A transformer can be looked upon as a magnetic coupling device actuated by an electrical network with a constant alternating sine voltage at a given frequency on the primary circuit to provide power on a given external load on the secondary circuit.

The data resulting from the methodology calculation developed in section 10-1 lead to a valuable analysis of the characteristics and efficiency of the transformer using the proper hypothesis. Indeed, the inductance coupling relations shown (Equations 12/15 for air core or 20/23 for iron core) imply knowledge of the current intensities for all windings involved. On the other hand, a transformer is defined as a device actuated by an electrical sinusoidal constant rms voltage in order to produce a different voltage with little distortion in phase. Consequently, *a priori*, the current intensities are not known. It is therefore necessary to assume initial values for these data in order to start the calculation and then, after a first trial, an iterative procedure must be used to complete the calculation with results of greater accuracy.

As an example, let's consider a single phase transformer as being a 2 current-carrying coil device with a soft iron core loaded from a constant primary voltage **V1** of frequency **f** set to generate a secondary voltage **V2** in an electrical circuit such that the load current intensity is **I2.** Then we have the following :

- Transformer power. This very crude definition contains a fundamental specification: the nominal power:

$$\mathbf{Pwn} \ = \ \mathbf{V2} \times \mathbf{I2} \qquad\qquad (41)$$

- Nominal primary current intensity **I1**. Assuming, as a first guess, perfect efficiency in the electrical power transfer between the primary and the secondary, then the current intensity in the primary is calculated as:

$$\mathbf{I1} \ = \ \mathbf{I2} \ \ \mathbf{x} \ \frac{V2}{V1} \qquad\qquad (42)$$

- Number of turns N1 and N2. If the purpose of the calculation is the analysis of a given transformer, then the number of turns is known. But, on the other hand, if the objective is the design of a new transformer, then these data must be guessed. However, the ratio of the number of turns must obey the nominal transformation ratio:

$$\frac{N2}{N1} = \frac{V2}{V1} \tag{43}$$

- Transformer dimensions and miscellaneous data. Here again, if the objective is the analysis of a fully described transformer, all the data are known. That includes: dimensions of the coils (primary and secondary), iron core dimensions as well as the nature of the ferromagnetic materials. If not, the unknown data must be guessed using useful technical information that can be deduced from common engineering practice. The simplest way is probably to start with the data of a similar transformer and progressively to refine the data. *Magnetica* offers in the case examples a certain number of different transformers from which the new design can be started.

The analysis requires that all the data of the magnetic system are settled. That includes for a single phase transformer :

- the data of the primary coil

- the data of the secondary coil

- the data of the iron core; it can be divided in several sub-components such as the central core, the 2 side armatures and the return yoke so that the "magnetic circuit" presents a good closed path.

The validation of this magnetic system is followed by the calculation of **InductanceMatrix** which provides all the necessary data for a thorough analysis of transformer behavior using the special functions **TransformerAnalysis** and **TransformerMagneticLosses**.

- Magnetic specifications of the iron core. This includes **M** the functional magnetization, **B** the induction and **Hm** the magnetizing field (see Chapter 6 and 8). The complete set of magnetic data of the cells of the ferromagnetic core is available. This allows for a very precise calculation of the effects that are dependent on **M**, **B** or **Hm** such as flux coupling and magnetic losses.

- Inductance specifications of the magnetic system. The total inductance matrix $\mathfrak{M}t$ (Equation 32) and total flux inductance matrix $\mathfrak{M}\phi t$ (Equation 34) provide all data necessary for the inductance coupling and transformer analysis.

- Coils resistance R1 and R2. These data are calculated in the validation process

- Angular frequency ω and reactive coil impedance Z1 and Z2 are calculated from the frequency f and the total self-inductance $\mathcal{G}11$ and $\mathcal{G}22$:

$$\omega = 2\pi \text{ x f} \tag{44}$$

$$Z1 = \omega \times \mathcal{G}11 \tag{45}$$

$$Z2 = \omega \times \mathcal{G}22 \tag{46}$$

The **TransformerAnalysis** function processes various data and generates the following results:

- actual secondary voltage: V2e

- active, reactive and modulus for primary and secondary current intensities

- magnetic losses in the iron core, (restricted to ferromagnetic materials for which the curve of losses is known and published: Silicon Steels and MuMetal)

- Joule effect losses in the electrical conductors of the primary and secondary winding

- phase angle φ and Power factor $\cos(\varphi)$

- transformer power efficiency

The precision of the results depends on the validity of the initial guess for the current intensities **I1** and **I2.** If the final total (modulus) intensities **i1t** and **i2t** differ substantially from **I1** and **I2,** it is necessary to undertake a new calculation using **i1t** and **i2t** as initial values. The method appears to converge very rapidly (with 3 iterations the results are better than 1%).

Further insight into transformer behavior can be obtained through a point by point analysis in a single period, the electrical current intensities and the voltage being parameterized with time. The generated secondary voltage and current appear as a distortion of the primary sine voltage due to the non linearity of the law of magnetization of the ferromagnetic core.

Section 13-5 displays data and results for the case of a domestic network step-down transformer 15000/220 volts of 50KVA power and shows the curves of various parameters as a function of the load. Further examples are given in the demonstration cases Appendix.

■ 10-3-4 Special Applications Devices

This is the case of IRM superconducting magnets. These devices of large dimensions are made of several electrically connected solenoids. The permanent electrical current passed in the coils generates a constant magnetic field of high strength (currently from 1 to 4 Tesla). Consequently, the stored magnetic energy is very great, in the range of several Megajoules. It is of the utmost importance to evaluate the behavior of the coils in the case of an accident such as the breakdown of a superconducting wire. This implies knowing the magnetic energy as well as the system inductance matrix (See Chapter 13 on Inductance Application, Case 5).

■ Chapter 11: Inductance Calculations

■ 11-1 Introduction

This chapter deals with "direct" inductance calculations where the inductance is strictly restricted to current-carrying components. In comparison with the ferro inductance as presented in the previous chapter (see 10-1-2), the present inductance could be called Proper inductance.

The chapter shows inductance calculations for the following cases :

1. Self-inductance of a circular loop.

2. Self-inductance of a square solenoid.

3. Self-inductance of a long solenoid.

4. Mutual inductance of a pair of loops.

5. Mutual inductance of a pair of square solenoids, set side by side.

6. Mutual inductance of a pair of radially superposed long solenoids.

■ 11-2 Self-inductance Calculation of Circular Loop

■ 11-2-1 Introduction

The circular loop inductance calculation is performed using the 2 different types of modelization of the circular loop, either the **filament** mode or the **tubular** mode (see `Chapter 4`, Section 4).

For the filament mode, the calculation is theoretically impossible since the theoretical winding dimension is null. However, the exact dimension of the winding is used to calculate the self-inductance through an approximation of the flux inside the winding.

For the tubular mode, the assumption of constant current density enables a much more precise calculation but the calculation time is considerably increased.

The next examples of inductance calculations are based on the same loop as in Section 4-4. Self-inductance is carried out according to both modes.

■ 11-2-2 Entering and Drawing the Data

The relevant data are:

- Mean loop diameter: 100 mm,

- Winding diameter: 30 mm,

- Axial position of the loop: 0, (origin centered),

- Number of turns (right hand rule, i.e. counterclockwise): 20,

- Current intensity: 0.5 Ampere.

```
loopFilament = AXSMagneticSystem[
   {AXSCircularLoop[CopperOFHC, 100, 30, 0, 20, 0.5, FillingFactor -> 0.7]}]
loopTubular = AXSMagneticSystem[{AXSCircularLoop[CopperOFHC,
     100, 30, 0, 20, 0.5, FillingFactor -> 0.7, FilamentMode -> False]}]
```

– AXSMagneticSystem :{-loop-} –

– AXSMagneticSystem :{-loop-} –

Coil 2D drawing

```
DrawSystem2D[loopFilament]
```

System name: **loopFilament**

2D view ZR
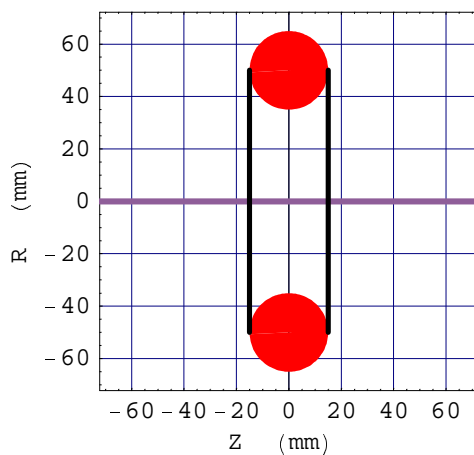
- Graphics -

■ 11-2-3 Self-inductance Calculation

Basic inductance calculations at default precision level (level 1); inductance is in Micro Henry.

```
SelfInductance[loopFilament, 1]
SelfInductance[loopTubular, 1]
```

41.125

39.7611

Same calculations but using the NumberForm function to obtain screen output with 12 decimal digits. The internal accuracy is to 16 decimal digits.

```
NumberForm[SelfInductance[loopFilament, 1], 12]
NumberForm[SelfInductance[loopTubular, 1], 12]
```

41.1249901127

39.7611108556

Setting the inductance unit in Henry:

```
MagneticaUserUnits[InductanceUnit → Henry]
SelfInductance[loopFilament, 1]
SelfInductance[loopTubular, 1]
```

{InductanceUnit → Henry, FieldUnit → Tesla}

0.000041125

0.0000397611

Basic magnetic flux calculation, unit in Micro Weber:

```
SelfFluxInductance[loopFilament, 1]
SelfFluxInductance[loopTubular, 1]
```

20.5625

19.8806

Basic magnetic energy calculation, unit in Micro Joule:

```
SelfMagneticEnergy[loopFilament, 1]
SelfMagneticEnergy[loopTubular, 1]
```

5.14062×10^{-6}

4.97014×10^{-6}

Setting the MagneticEnergy unit into Joule:

```
MagneticaUserUnits[MagneticEnergyUnit → Joule]
SelfMagneticEnergy[loopFilament, 1]
SelfMagneticEnergy[loopTubular, 1]
```

{MagneticEnergyUnit → Joule, InductanceUnit → Henry, FieldUnit → Tesla}

5.14062×10^{-6}

4.97014×10^{-6}

Restoring the default units. Then displaying self-inductance calculations with the "Verbose" version of the functions.

```
MagneticaUserUnits[];
VerboseSelfInductance[loopTubular, 1]
VerboseSelfFluxInductance[loopTubular, 1]
VerboseSelfMagneticEnergy[loopTubular, 1]

System name: loopTubular
Self inductance
Precision Level : 1


Component n° 1
Component type: circular loop winding
Number of turns:        20.
Current:        0.500000 A
Proper self inductance:                39.761111 μH
Ferromagnetic contribution:             0.000000 μH
Total self inductance:                 39.761111 μH

System name: loopTubular
Self flux inductance
Precision Level : 1


Component n° 1
Component type: circular loop winding
Number of turns:        20.
Current:        0.500000 A
Proper self inductance:                19.880555 μWb
Ferromagnetic contribution:             0.000000 μWb
Total self inductance:                 19.880555 μWb

System name: loopTubular
Self magnetic energy
Precision Level : 1


Component n° 1
Component type: circular loop winding
Number of turns:        20.
Current:        0.500000 A
Proper self inductance:                4.970139e-6 J
Ferromagnetic contribution:            0.000000 J
Total self inductance:                 4.970139e-6 J
```

Self-inductance with precision level 2 (at least 5 decimal digits of precision).

```
NumberForm[SelfInductance[loopFilament, 1, PrecisionLevel→2], 12]
NumberForm[SelfInductance[loopTubular, 1, PrecisionLevel→2], 12]

41.1252393014

39.73912736
```

Self-inductance calculation with precision level 3 (at least 7 decimal digits of precision).

```
NumberForm[SelfInductance[loopFilament, 1, PrecisionLevel→3], 12]
NumberForm[SelfInductance[loopTubular, 1, PrecisionLevel→3], 12]

41.1252772737

39.7346849847
```

■ 11-3 Self-inductance Calculation of Square Solenoids

■ 11-3-1 Introduction

The self-inductance of a solenoid depends on both

- the shape of the rectangular winding cross section, the simplest being a rectangle with equal sides: a square.

- the distance between this rectangle and the coordinate axis: the inner diameter.

The next examples of self-inductance calculations are based on square cross section solenoids with variable radial positions:

- Case 1: winding close to the axis.

- Case 2: winding at the so called Brook configuration. The inner diameter is equal to the side of the square.

- Case 3: winding far from the axis.

■ 11-3-2 Entering and Drawing the Data

- Case 1: winding close to the axis, relevant data:

  - inner diameter: 10 mm,

  - outer diameter: 50 mm, (resulting winding height 20 mm and mean diameter 30 mm),

  - solenoid length: 20 mm,

  - axial position of the solenoid: 0, (i.e. origin centered),

  - number of turns (right hand rule, i.e. counterclockwise): 400

  - current intensity: 0.2 Ampere,

  - conductor diameter: 0.2 mm.

```
solenoid1 = AXSMagneticSystem[
  {AXSSolenoid[CopperOFHC, 10, 50, 20, 0, 400, 0.2, ConductorDiameter -> 0.2]},
  SystemLabel -> "square solenoid set close to the axis"]
```

  - AXSMagneticSystem :{-solenoid-} -

Solenoid 2D drawing:

DrawSystem2D[solenoid1]

> System name: **solenoid1**
> *square solenoid set close to the*
> *axis*
> 2D view ZR
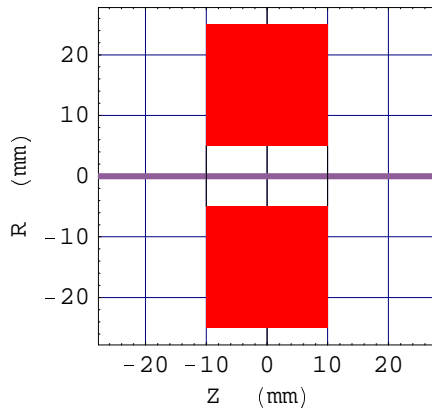
▪ Graphics ▪

Solenoid technical data (general data only):

Properties[solenoid1]

General properties

System name:	**solenoid1**
System magnetic parity	True
System Legendre radius	15.9629
Magnetic field at the origin (G)	{0, 29.3053}
Field magnitude at the origin(G)	29.3053
System mass (Kg)	0.0105644
System nominal power (W)	0.82752
System effective power(W)	0.82752
Number of components	1
Number of current carrying component	1

▪ Case 2: winding at the so called Brook configuration. The data are identical to those of Case 1, with the exception of the coil dimensions. New relevant data are:

- inner diameter: 20 mm,

- outer diameter: 40 mm, (resulting winding height 10 mm and mean diameter 30 mm),

- winding length: 10 mm.

```
solenoid2 = AXSMagneticSystem[
  {AXSSolenoid[CopperOFHC, 20, 40, 10, 0, 400, 0.2, ConductorDiameter -> 0.2]},
  SystemLabel → "square solenoid type BROOK"]
```

– AXSMagneticSystem :{-solenoid-} –

Coil 2D drawing:

`DrawSystem2D[solenoid2]`

System name: **solenoid2**
square solenoid type BROOK
2D view ZR
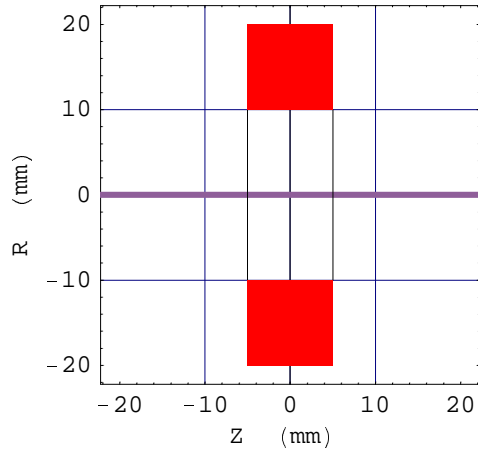
▪ Graphics ▪

Coil technical data (general data only):

`Properties[solenoid2]`

General properties

System name:	**solenoid2**
System magnetic parity	True
System Legendre radius	15.3078
Magnetic field at the origin (G)	{0, 32.7267}
Field magnitude at the origin(G)	32.7267
System mass (Kg)	0.0105644
System nominal power (W)	0.82752
System effective power(W)	0.82752
Number of components	1
Number of current carrying component	1

▪ Case 3: winding far from the axis. The data are identical to those of Case 1, with the exception of coil dimensions. The new relevant data are:

- inner diameter: 25 mm,

- outer diameter: 35 mm, (resulting winding height 5 mm and mean diameter is 30mm),

- winding length: 5 mm.

`solenoid3 = AXSMagneticSystem[`
`{AXSSolenoid[CopperOFHC, 25, 35, 5, 0, 400, 0.2, ConductorDiameter → 0.2]},`
`SystemLabel → "square solenoid set far"]`

– AXSMagneticSystem :{-solenoid-} –

Coil 2D drawing:

DrawSystem2D[solenoid3]

System name: **solenoid3**
square solenoid set far
2D view ZR
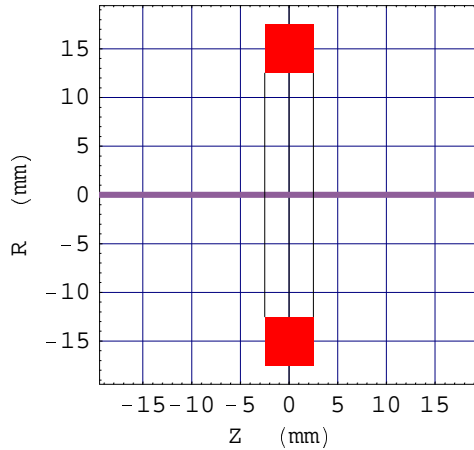
- Graphics -

Coil technical data (general data and component data):

Properties[solenoid3, ComponentsList → {1}]

General properties

System name:	**solenoid3**
System magnetic parity	True
System Legendre radius	15.0888
Magnetic field at the origin (G)	{0, 33.3444}
Field magnitude at the origin(G)	33.3444
System mass (Kg)	0.0105644
System nominal power (W)	0.82752
System effective power(W)	0.82752
Number of components	1
Number of current carrying component	1

Component number 1

Type of component:	cylindrical solenoid

Geometry of the cylindrical current solenoid

Nominal inner diameter (mm)	25.00000
Effective inner diameter (mm)	25.00000
Nominal outer diameter (mm)	35.00000
Effective outer diameter (mm)	35.00000
Nominal mean diameter (mm)	30.00000
Nominal axial length (mm)	5.00000
Effective axial length (mm)	5.00000

Winding specifications

Material:high conductivity OFHC Copper	
Winding temperature (°C)	20.00000
Nominal resistivity (Ω·cm)	1.72400e-6
Effective resistivity	1.72400e-6
Nominal volume relative susceptibility	-6.10000e-7
Effective susceptibility	-6.10000e-7
Material mass density (g/cm^3)	8.92000

```
Material mass density (g/cm^3)                        0.92000
Filling factor of winding (%)                        50.26548
Mass of the material (kg)                             0.01056
```
Position of the solenoid
  (in the referential space system)
```
Solenoid center coord.Z (mm)                          0.00000
Effective inner radius (mm)                          12.50000
Effective outer radius (mm)                          17.50000
Effective lower Z extremum (mm)                      -2.50000
Effective higher Z extremum (mm)                      2.50000
```
Electrical specifications of
 the solenoid
```
Number of turns  (in series)                        400.00000
Direction according the                      counterclockwise
right hand rule :
Direction of the turns                       +
Type of curent density                       constant
Current (per turns)  (A)                              0.20000
Total winding current (A·turns)                      80.00000
Conductor current density (A/mm^2)                    6.36620
Nominal solenoid resistance (Ω)                      20.68800
Effective solenoid resist. (Ω)                       20.68800
Nominal voltage drop (V)                              4.13760
Effective voltage drop (V)                            4.13760
Nominal solenoid power  (W)                           0.82752
Effective solenoid power (W)                          0.82752
```
Conductor dimensions
```
Conductor geometry                           circular
Effective conductor diameter (mm)                     0.20000
Nominal conductor length (mm)                     37699.11184
Effective conductor length (mm)                   37699.11184
```

■ 11-3-3 Self-inductance Calculations

Only appropriate results are given for this example, see Section 10-2 for the complete set of self-inductance functions.

Basic inductance calculation at default precision level (level 1), inductance is in Micro Henry.

```
SelfInductance[solenoid1, 1]
SelfInductance[solenoid2, 1]
SelfInductance[solenoid3, 1]

2396.83

4079.01

6038.14
```

The approximate formulas, Equation 36 in Chapter 10, using the tables from the Grover book gives respectively for the Grover coefficient and the inductance (in Micro Henry):

- Case 1 (winding close to the axis): CGrover = 9.98509, Self = 2396.42

- Case 2 (winding Brook configuration): CGrover = 16.9951, Self = 4078.82

- Case 3 (winding far form the axis): CGrover = 25.1618, Self = 6038.83

The agreement between the 2 calculations is very good, the accuracy is in the range $5 * 10^{-5}$. However, as opposed to the statement in Grover's book, the optimum of the self-inductance with respect to the amount of materials (37.7 meters of wire in all cases) occurs for the Case 3 and not for the Brook configuration Case 2.

Self-inductance calculation with precision level 2 (at least 5 decimal digits of precision), unit set in Milli Henry:

```
MagneticaUserUnits[InductanceUnit → Milli Henry];
NumberForm[SelfInductance[solenoid1, 1, PrecisionLevel → 2], 12]
NumberForm[SelfInductance[solenoid2, 1, PrecisionLevel → 2], 12]
NumberForm[SelfInductance[solenoid3, 1, PrecisionLevel → 2], 12]

2.39619823503

4.07850996365

6.03766491547
```

Self-inductance calculation with level 3 (at least 7 decimal digits of precision), unit remains in Milli Henry:

```
NumberForm[SelfInductance[solenoid1, 1, PrecisionLevel → 3], 12]
NumberForm[SelfInductance[solenoid2, 1, PrecisionLevel → 3], 12]
NumberForm[SelfInductance[solenoid3, 1, PrecisionLevel → 3], 12]

2.39619613226

4.07850786832

6.03766282301
```

∎ 11-4 Self-inductance Calculations of Rectangular Section Solenoids

∎ 11-4-1 Introduction

The next examples of self-inductance calculations are based on 2 radically different solenoids:

- Case 1: thin solenoid, current sheet like,
- Case 2: thick solenoid, pancake like.

∎ 11-4-2 Entering and Drawing the Data

- Case 1: thin, current sheet type, solenoid. Relevant data: (N.B. this solenoid has been built for use as self-inductance reference)

  - inner diameter: 541.090mm,

  - outer diameter: 542.358mm, (made of a single layer of round enamel copper wire)

  - winding length: 305.51mm,

  - axial position of the loop: 0 (i.e. origin centered),

  - number of turns (right hand rule, i.e. counterclockwise): 440,

  - current intensity: 0.35 Ampere,

  - conductor diameter: 0.632mm.

```
solenoid4 = AXSMagneticSystem[{AXSSolenoid[CopperOFHC,
    541.090, 542.358, 305.51, 0, 440, 0.35, ConductorDiameter -> 0.634]},
   SystemLabel → "thin solenoid single layer of small wire"]
```

– AXSMagneticSystem :{-solenoid-} –

Coil 2D drawing:

```
DrawSystem2D[solenoid4];
```

System name: **solenoid4**
thin solenoid single layer of small
wire
2D view ZR
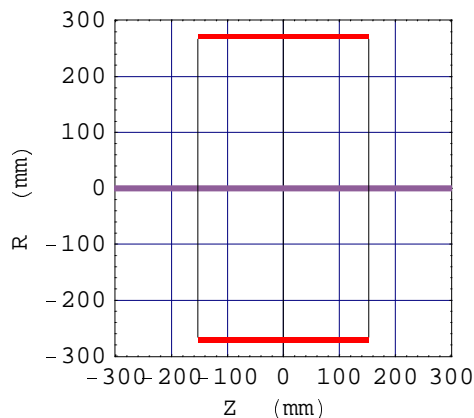
Coil technical data:

```
Properties[solenoid4]
```

General properties

System name:	**solenoid4**
System magnetic parity	True
System Legendre radius	290.894
Magnetic field at the origin (G)	{0, 3.11162}
Field magnitude at the origin(G)	3.11162
System mass (Kg)	2.1087
System nominal power (W)	5.0094
System effective power(W)	5.0094
Number of components	1
Number of current carrying component	1

▪ Case 2: thick solenoid, pancake like. Relevant data: (N.B.: this solenoid has been built for use in MRI magnet research)

   - inner diameter: 856 mm,

   - outer diameter: 1530 mm, (made of a single disk of wound rectangular copper ribbon)

   - winding length: 51 mm,

   - axial position: 0 (origin centered),

   - number of turn (right hand rule, i.e. counterclockwise): 254

   - current intensity: 150 Amperes,

- condutor dimensions: 2.5 mm x 20 mm.

```
solenoid5 = AXSMagneticSystem[{AXSSolenoid[CopperOFHC, 856, 1530, 51, 0, 254, 150,
    ConductorHeightWidth → {2.5, 20}]}, SystemLabel → "pancake resistive coil"]
```

– AXSMagneticSystem :{-solenoid-} –

Coil 2D drawing:

```
DrawSystem2D[solenoid5]
```

System name: **solenoid5**
pancake resistive coil
2D view ZR
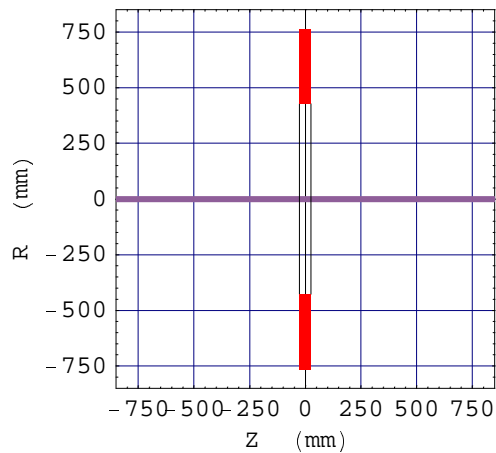
■ Graphics ■

Coil technical data:

```
Properties[solenoid5]
```

General properties

System name:	**solenoid5**
System magnetic parity	True
System Legendre radius	596.712
Magnetic field at the origin (G)	{0, 412.108}
Field magnitude at the origin(G)	412.108
System mass (Kg)	424.579
System nominal power (W)	7385.4
System effective power(W)	7385.4
Number of components	1
Number of current carrying component	1

■ **11-4-3 Self-inductance Calculations**

▪ Case 1: thin solenoid, current sheet type.

Basic inductance calculations at default precision level (level 1), inductance is in Milli Henry.

```
MagneticaUserUnits[InductanceUnit → Milli Henry]
SelfInductance[solenoid4, 1]
```

$\{\text{InductanceUnit} \rightarrow \text{Henry Milli}\}$

```
101.667
```

```
VerboseSelfInductance[solenoid4, 1]
```

```
System name: solenoid4
Self inductance
Precision Level : 1


Component n° 1
Component type: cylindrical solenoid
Number of turns:       440.
Current:       0.350000 A
Proper self inductance:           101.667378 mH
Ferromagnetic contribution:         0.000000 mH
Total self inductance:            101.667378 mH
```

Self-inductance calculations with precision level 2 (5 decimal digits of precision), unit in Milli Henry.

```
NumberForm[SelfInductance[solenoid4, 1, PrecisionLevel → 2], 12]
```

```
101.66734766
```

the approximate formula, Equation 37 of Chapter10, using the table from Grover's book gives the following results:

Self = 101.69 (±0.1) Milli Henry

The agreement between the 2 calculations is very good, the accuracy is in the range 10^{-4}.

The stored magnetic energy at current intensity of 0.35 Ampere is:

```
SelfMagneticEnergy[solenoid4, 1]
VerboseSelfMagneticEnergy[solenoid4, 1]
```

```
0.00622713
```

```
System name: solenoid4
Self magnetic energy
Precision Level : 1


Component n° 1
Component type: cylindrical solenoid
Number of turns:       440.
Current:       0.350000 A
Proper self inductance:           0.006227 J
Ferromagnetic contribution:       0.000000 J
Total self inductance:            0.006227 J
```

▪ Case 2: thick solenoid, pancake type.

Basic inductance calculations at default precision level (level 1), inductance is in Milli Henry.

```
MagneticaUserUnits[InductanceUnit → Milli Henry]
SelfInductance[solenoid5, 1]
```

{InductanceUnit → Henry Milli}

98.1242

```
VerboseSelfInductance[solenoid5, 1]
```

System name: **solenoid5**
Self inductance
Precision Level : 1

Component n° 1
Component type: cylindrical solenoid
Number of turns: 254.
Current: 150.000000 A
Proper self inductance: 98.124209 mH
Ferromagnetic contribution: 0.000000 mH
Total self inductance: 98.124209 mH

Self-inductance calculation with precision level 2 (5 decimal digits of precision), unit in Milli Henry.

```
NumberForm[SelfInductance[solenoid5, 1, PrecisionLevel → 2], 12]
```

98.110533218

the approximate formula, Equation 36 Chapter 10, using the table from Grover's book gives the following results:

Self = 97.811 Milli Henry

The agreement between the 2 calculations is good, the accuracy is in range $3\ 10^{-3}$.

The stored magnetic energy at current intensity of 150 Ampere is:

```
SelfMagneticEnergy[solenoid5, 1]
VerboseSelfMagneticEnergy[solenoid5, 1]
```

1103.9

System name: **solenoid5**
Self magnetic energy
Precision Level : 1

Component n° 1
Component type: cylindrical solenoid
Number of turns: 254.
Current: 150.000000 A
Proper self inductance: 1103.897352 J
Ferromagnetic contribution: 0.000000 J
Total self inductance: 1103.897352 J

◾ 11-5 Mutual Inductance of Pair of Circular Loops

◾ 11-5-1 Entering and Drawing the Data

▪ Case 1: system with 2 identical loops. The relevant data are:

- mean loop diameter: 200mm,

- winding diameter: 50mm,

- axial position: -50 for one loop and 50 for the other,

- number of turns (right hand rule, i.e. counterclockwise): 100,

- current intensity: 2 amperes.

```
pairloopidentic = AXSMagneticSystem[
   {AXSCircularLoop[CopperOFHC, 200, 50, -50, 100, 2, FillingFactor -> 0.6],
    AXSCircularLoop[CopperOFHC, 200, 50, 50, 100, 2, FillingFactor -> 0.6]},
   SystemLabel -> "pair of identical filament loops"]
```

– AXSMagneticSystem :{-loop-, -loop-} –

```
DrawSystem2D[pairloopidentic]
```

    System name: **pairloopidentic**
    pair of identical filament loops
    2D view ZR
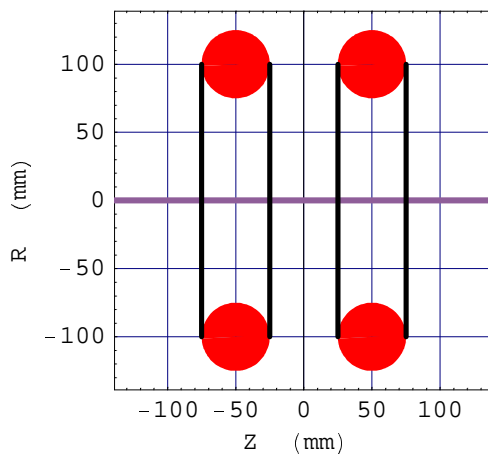
- Graphics -

- Case 2: system with 2 dissimilar loops. The relevant data are for the two loops:

  - mean diameter: 400 mm and 700 mm,

  - winding diameter: 150 mm and 60 mm,

  - axial position of the loops: -150 mm and 50 mm,

  - number of turns: 60 and 100

  - current intensity: 2 amperes and 1.2 amperes.

```
pairloopdissim = AXSMagneticSystem[
   {AXSCircularLoop[CopperOFHC, 400, 150, -150, 60, 2, FillingFactor -> 0.6],
    AXSCircularLoop[CopperOFHC, 700, 60, 50, 100, 1.2, FillingFactor -> 0.6]},
   SystemLabel -> "pair of dissimilar filament loops"]
```

– AXSMagneticSystem :{-loop-, -loop-} –

DrawSystem2D[pairloopdissim]

System name: **pairloopdissim**
pair of dissimilar filament loops
2D view ZR
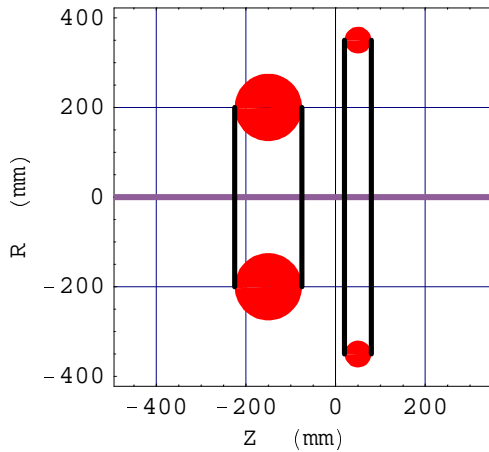
- Graphics -

▪ 11-5-2   Mutual Inductance of Identical Loops

MagneticaUserUnits[InductanceUnit -> Milli Henry]

{InductanceUnit → Henry Milli}

MutualInductance[pairloopidentic, {1, 2}]

0.497252

VerboseMutualInductance[pairloopidentic, {1, 2}]

System name: **pairloopidentic**
Mutual inductance
Precision Level : 1

	Primary coil	Secondary coil
Component number:	1	2
Component category:	circular loop winding	circular loop winding
Circuits current	2.000000 A	2.000000 A
Number of turns:	100.000000	100.000000
Mutual inductance	0.497252 mH	0.497252 mH

InductanceMatrix[pairloopidentic]

{{{2.27842, 0.497252}, {0.497252, 2.27842}}, {1, 2}}

InductanceMatrix[pairloopidentic][[1]] // MatrixForm

$\begin{pmatrix} 2.27842 & 0.497252 \\ 0.497252 & 2.27842 \end{pmatrix}$

VerboseInductanceMatrix[pairloopidentic]

```
System name: pairloopidentic
Inductance matrix (in Milli Henry)
Precision Level : 1
   1          2
1    2.2784      0.4973
2    0.4973      2.2784


Involved components
components     type    nb turns        current
      1        loop    100.            2.000000 A
      2        loop    100.            2.000000 A
```

To obtain the summary of Flux information

VerboseFluxInductanceMatrix[pairloopidentic]

```
System name: pairloopidentic
Flux inductance matrix (in Micro Weber)
Precision Level : 1
   1          2
1   4556.8355    994.5042
2    994.5042   4556.8355


Involved components
components     type    nb turns        current
      1        loop    100.            2.000000 A
      2        loop    100.            2.000000 A
```

InductanceCoupling[pairloopidentic, {1, 2}]

```
{{0.218244, Irrelevant}, {0.0476307, Irrelevant},
 {0.952369, Irrelevant}, {1., Irrelevant}}
```

NB : the term Irrelevant is related to the ferromagnetic contribution which is irrelevant in the absence of ferromagnetic components.

```
VerboseInductanceCoupling[pairloopidentic, {1, 2}]
```

```
System name: pairloopidentic
Inductance Coupling
Precision Level :                           1
Number of components                   2
Number of current carrying component    2
                          primary coil          secondary coil
Component number:          1.0000                   2.0000
Component category:    loop                      loop
Circuits current (A):      2.0000                   2.0000
Number of turns:         100.0000                 100.0000
Transfo turn ratio                  1.000
air core data
Self inductance (mH)       2.2784                   2.2784
Mutual induc. 1->2 (mH)              0.4973
Self flux (μWb)         4556.8355               4556.8355
Mutual flux 1->2 (μWb)            994.5042
                          air core
Coupling coefficient            0.218244
Efficiency coefficient          0.047631
Leakage coefficient             0.952369
Transformation coeff.      1.0000
```

■ 11-5-3 Mutual Inductance of Dissimilar Loops

```
MutualInductance[pairloopdissim, {1, 2}]
```

```
0.850428
```

```
VerboseMutualInductance[pairloopdissim, {1, 2}]
```

```
System name: pairloopdissim
Mutual inductance
Precision Level : 1
                          Primary coil          Secondary coil
Component number:        1                      2
Component category:    circular loop winding   circular loop winding
Circuits current          2.000000 A            1.200000 A
Number of turns:         60.000000             100.000000
Mutual inductance         0.850428 mH           0.850428 mH
```

```
InductanceMatrix[pairloopdissim]
```

```
{{{1.28745, 0.850428}, {0.850428, 12.6294}}, {1, 2}}
```

```
InductanceMatrix[pairloopdissim][[1]] // MatrixForm
```

$$\begin{pmatrix} 1.28745 & 0.850428 \\ 0.850428 & 12.6294 \end{pmatrix}$$

VerboseInductanceMatrix[pairloopdissim]

```
System name: pairloopdissim
Inductance matrix (in Milli Henry)
Precision Level : 1
   1          2
1     1.2874      0.8504
2     0.8504     12.6294


Involved components
components     type     nb turns          current
      1        loop      60.                2.000000 A
      2        loop     100.                1.200000 A
```

To obtain the summary of Flux informations

VerboseFluxInductanceMatrix[pairloopdissim]

```
System name: pairloopdissim
Flux inductance matrix (in Micro Weber)
Precision Level : 1
   1          2
1   2574.8974   1700.8554
2   1020.5133   15155.3210


Involved components
components     type     nb turns          current
      1        loop      60.                2.000000 A
      2        loop     100.                1.200000 A
```

InductanceMatrix[pairloopdissim, InductanceUnit → Milli Henry]

```
{{{1.28745, 0.850428}, {0.850428, 12.6294}}, {1, 2}}
```

InductanceCoupling[pairloopdissim, {1, 2}]

```
{{0.210902, Irrelevant}, {0.0444796, Irrelevant},
 {0.95552, Irrelevant}, {4.68825, Irrelevant}}
```

```
VerboseInductanceCoupling[pairloopdissim, {1, 2}]

System name: pairloopdissim
Inductance Coupling
Precision Level :                          1
Number of components                       2
Number of current carrying component       2
                       primary coil            secondary coil
Component number:          1.0000                   2.0000
Component category:    loop                     loop
Circuits current (A):      2.0000                   1.2000
Number of turns:          60.0000                 100.0000
Transfo turn ratio                  1.667
air core data
Self inductance (mH)       1.2874                  12.6294
Mutual induc. 1->2 (mH)                0.8504
Self flux (μWb)        2574.8974                15155.3210
Mutual flux 1->2 (μWb)              1700.8554
                            air core
Coupling coefficient              0.210902
Efficiency coefficient            0.044480
Leakage coefficient               0.955520
Transformation coeff.         4.6882
```

■ 11-6 Mutual Inductance of Solenoids Set Side by Side

■ 11-6-1 Entering and Drawing the Data

System with 2 identical square cross section solenoids set side by side

```
pairsolidentic = AXSMagneticSystem[
  {AXSSolenoid[CopperOFHC, 175, 225, 25, -30, 200, 2, FillingFactor -> 0.6],
   AXSSolenoid[CopperOFHC, 175, 225, 25, 30, 200, 2, FillingFactor -> 0.6]},
  SystemLabel → "pair of identical solenoids"]

- AXSMagneticSystem :{-solenoid-, -solenoid-} -
```

DrawSystem2D[pairsolidentic]

> System name: **pairsolidentic**
> *pair of identical solenoids*
> 2D view ZR
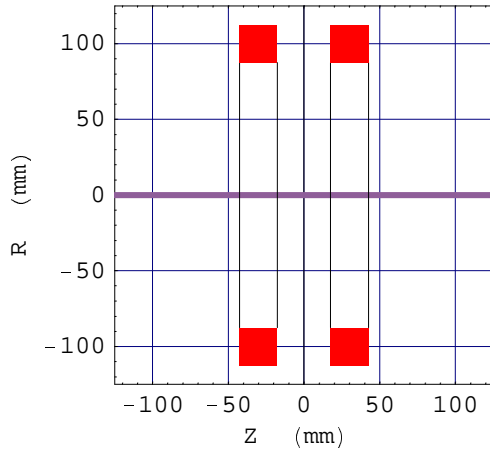
- Graphics -

■ 11-6-2 Mutual Inductance of Similar Solenoids

MutualInductance[pairsolidentic, {1, 2}]

3.73709

InductanceMatrix[pairsolidentic]

{{{11.4721, 3.73709}, {3.73709, 11.4721}}, {1, 2}}

InductanceMatrix[pairsolidentic, PrecisionLevel → 2][[1]] // MatrixForm

$$\begin{pmatrix} 11.4713 & 3.73728 \\ 3.73728 & 11.4713 \end{pmatrix}$$

VerboseInductanceMatrix[pairsolidentic, PrecisionLevel → 3]

System name: **pairsolidentic**
Inductance matrix (in Milli Henry)
Precision Level : 3
```
   1           2
1    11.4713      3.7373
2     3.7373     11.4713
```

Involved components
components	type	nb turns	current
1	solenoid	200.	2.000000 A
2	solenoid	200.	2.000000 A

VerboseFluxInductanceMatrix[pairsolidentic]

System name: **pairsolidentic**
Flux inductance matrix (in Micro Weber)
Precision Level : 1
```
   1          2
1  22944.1113   7474.1768
2   7474.1768  22944.1113
```

Involved components
```
components      type      nb turns      current
    1         solenoid    200.          2.000000 A
    2         solenoid    200.          2.000000 A
```

VerboseMagneticEnergyMatrix[pairsolidentic, MagneticEnergyUnit → Milli Joule]

System name: **pairsolidentic**
Magnetic energy matrix (in Milli Joule)
Precision Level : 1
```
   1          2
1    22.9441     7.4742
2     7.4742    22.9441
```

Involved components
```
components      type      nb turns      current
    1         solenoid    200.          2.000000 A
    2         solenoid    200.          2.000000 A
```

VerboseSelfInductance[pairsolidentic, All]

System name: **pairsolidentic**
Self inductance
Precision Level : 1

Component n° 1
Component type: cylindrical solenoid
Number of turns: 200.
Current: 2.000000 A
Proper self inductance: 11.472056 mH
Ferromagnetic contribution: 0.000000 mH
Total self inductance: 11.472056 mH

Component n° 2
Component type: cylindrical solenoid
Number of turns: 200.
Current: 2.000000 A
Proper self inductance: 11.472056 mH
Ferromagnetic contribution: 0.000000 mH
Total self inductance: 11.472056 mH

Total (in series)
Proper self inductance: 22.944111 mH
Ferromagnetic contribution: 0.000000 mH
Total self inductance: 22.944111 mH

InductanceCoupling[pairsolidentic, {1, 2}]

{{0.325756, Irrelevant}, {0.106117, Irrelevant},
 {0.893883, Irrelevant}, {1., Irrelevant}}

■ 11-7 Mutual inductance of Superimposed Solenoids

■ 11-7-1 Entering and Drawing the Data

System with 2 dissimilar superposed square cross section solenoids.

```
pairsoldissim = AXSMagneticSystem[
  {AXSSolenoid[CopperOFHC, 260, 320, 150, 0, 500, 2, FillingFactor -> 0.6],
   AXSSolenoid[CopperOFHC, 150, 250, 100, 0, 80, 15, FillingFactor -> 0.6]},
  SystemLabel → "pair of dissimilar superposed solenoids"]
```

- AXSMagneticSystem :{-solenoid-, -solenoid-} -

```
DrawSystem2D[pairsoldissim]
```

System name: **pairsoldissim**
pair of dissimilar superposed solenoids
2D view ZR
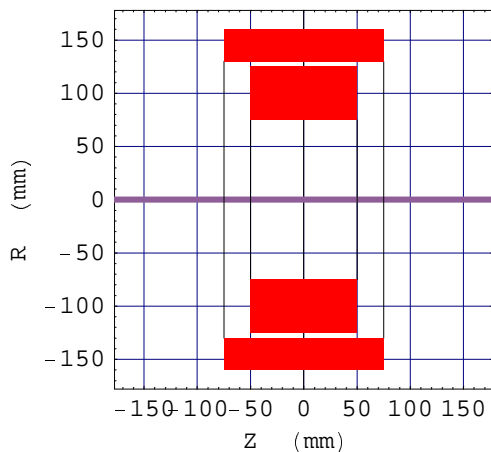
- Graphics -

■ 11-7-2 Mutual Inductance of Dissimilar Solenoids

```
MutualInductance[pairsoldissim, {1, 2}]
```

5.3942

```
InductanceMatrix[pairsoldissim]
```

{{{65.5243, 5.3942}, {5.3942, 1.01098}}, {1, 2}}

```
InductanceMatrix[pairsoldissim, PrecisionLevel → 2][[1]] // MatrixForm
```

$$\begin{pmatrix} 65.52 & 5.39474 \\ 5.39474 & 1.01059 \end{pmatrix}$$

VerboseInductanceMatrix[pairsoldissim, PrecisionLevel → 3]

System name: **pairsoldissim**
Inductance matrix (in Milli Henry)
Precision Level : 3
```
   1          2
1    65.5200      5.3947
2     5.3947      1.0106
```

Involved components
```
components       type       nb turns        current
    1         solenoid     500.          2.000000 A
    2         solenoid      80.         15.000000 A
```

VerboseFluxInductanceMatrix[pairsoldissim]

System name: **pairsoldissim**
Flux inductance matrix (in Micro Weber)
Precision Level : 1
```
   1            2
1  131048.5910  10788.4000
2  80913.0002   15164.7168
```

Involved components
```
components       type       nb turns        current
    1         solenoid     500.          2.000000 A
    2         solenoid      80.         15.000000 A
```

VerboseMagneticEnergyMatrix[pairsoldissim, MagneticEnergyUnit → Milli Joule]

System name: **pairsoldissim**
Magnetic energy matrix (in Milli Joule)
Precision Level : 1
```
   1          2
1   131.0486     80.9130
2    80.9130    113.7354
```

Involved components
```
components       type       nb turns        current
    1         solenoid     500.          2.000000 A
    2         solenoid      80.         15.000000 A
```

VerboseSelfInductance[pairsoldissim, All]

System name: **pairsoldissim**
Self inductance
Precision Level : 1

Component n° 1
Component type: cylindrical solenoid
Number of turns: 500.
Current: 2.000000 A
Proper self inductance: 65.524295 mH
Ferromagnetic contribution: 0.000000 mH
Total self inductance: 65.524295 mH

Component n° 2
Component type: cylindrical solenoid
Number of turns: 80.
Current: 15.000000 A
Proper self inductance: 1.010981 mH
Ferromagnetic contribution: 0.000000 mH
Total self inductance: 1.010981 mH

Total (in series)
Proper self inductance: 66.535277 mH
Ferromagnetic contribution: 0.000000 mH
Total self inductance: 66.535277 mH

InductanceCoupling[pairsoldissim, {1, 2}]

{{0.662757, Irrelevant}, {0.439247, Irrelevant},
 {0.560753, Irrelevant}, {0.122443, Irrelevant}}

VerboseInductanceCoupling[pairsoldissim, {1, 2}]

System name: **pairsoldissim**
Inductance Coupling
Precision Level :   1
Number of components   2
Number of current carrying component   2
                       primary coil   secondary coil
Component number: 1.0000   2.0000
Component category: solenoid   solenoid
Circuits current (A):   2.0000   15.0000
Number of turns: 500.0000   80.0000
Transfo turn ratio   1 / 6.250
air core data
Self inductance (mH)   65.5243   1.0110
Mutual induc. 1->2 (mH)   5.3942
Self flux (μWb)   131048.5910   15164.7168
Mutual flux 1->2 (μWb)   10788.4000
                        air core
Coupling coefficient   0.662757
Efficiency coefficient   0.439247
Leakage coefficient   0.560753
Transformation coeff.   1 / 8.1671

■ Chapter 12 : Ferro Inductance Calculations

■ 12-1 Introduction

This chapter deals with "ferro" inductance calculation, i.e. the induced inductance brought upon a current-carrying coil by the ferromagnetic core components excited by the source coil. The ferro inductance is a contribution to the coil total inductance in addition to the Proper inductance.

This chapter shows Ferro inductance calculations for the following cases :

1. Ferro Self-inductance of a circular loop.

2. Ferro Self-inductance of a square solenoid.

3. Ferro Self-inductance of a long solenoid.

4. Ferro mutual inductance of a pair of loops.

5. Ferro mutual inductance of a pair of square solenoids, set side by side.

6. Ferro mutual inductance of a pair of radially superposed long solenoids.

The ferromagnetic core can have any axisymmetric shape but in all cases shown the iron core is an axial soft iron bar set at the center of the system and made of different non linear soft ferromagnetic materials.

■ 12-2 Ferro Self-inductance of a Circular Loop

■ 12-2-1 Entering and Drawing the Data

Single circular loop and an axial soft ferromagnetic cylindrical bar.

```
loopferro1 = AXSMagneticSystem[
  {AXSCircularLoop[CopperOFHC, 275, 80, 0, 120, 150, FillingFactor -> 0.6],
   AXSSoftMagneticComponent[SteelCommon1, 0, 160, 350, 0]},
  SystemLabel → "single loop and ferromagnetic core"]

- AXSMagneticSystem :{-loop-, -soft-} -
```

`DrawSystem[loopferro1]`

System name: **loopferro1**
single loop and ferromagnetic core
3D view ZYX
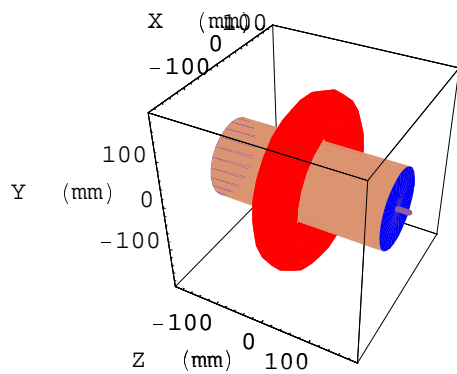
- Graphics3D -

`DrawSystem2D[loopferro1]`

System name: **loopferro1**
single loop and ferromagnetic core
2D view ZR
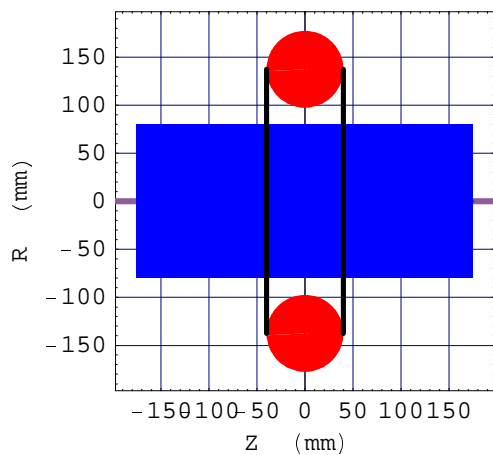
- Graphics -

■ 12-2-2 Ferro Self-inductance of Circular Loop

Induction field Bz (i.e. total magnetic field or flux density) at the center of the bar

```
FieldPointOrigin[loopferro1]
```

```
System name: loopferro1
The magnetic field (in Gauss) at the origin
Calculation mode: Automatic Mode
Br          0.000000 G
Bz       3452.188421 G
Bm       3452.188421 G
```

SelfInductance for the first component of the system, using the ContributionType option, that allows selecting either the Proper Selfinductance, or the Ferro SelfInductance or the Total. The default unit is Micro Henry:

```
SelfInductance[loopferro1, 1, ContributionType → Proper]
SelfInductance[loopferro1, 1, ContributionType → Ferro]
SelfInductance[loopferro1, 1, ContributionType → Total]
```

```
4.14529
```

```
6.56634
```

```
10.7116
```

SelfFluxInductance: total magnetic Flux for the first component of the system. Unit is set in Milli Weber.

```
MagneticaUserUnits[FluxUnit → Milli Weber];
SelfFluxInductance[loopferro1, 1, ContributionType → Proper]
SelfFluxInductance[loopferro1, 1, ContributionType → Ferro]
SelfFluxInductance[loopferro1, 1, ContributionType → Total]
MagneticaUserUnits[];
```

```
621.793
```

```
984.951
```

```
1606.74
```

SelfMagneticEnergy: magnetic energy for the first component of the system. Unit is set in Joule

```
MagneticaUserUnits[MagneticEnergyUnit → Joule]
SelfMagneticEnergy[loopferro1, 1, ContributionType → Proper]
SelfMagneticEnergy[loopferro1, 1, ContributionType → Ferro]
SelfMagneticEnergy[loopferro1, 1, ContributionType → Total]
MagneticaUserUnits[];
```

```
{MagneticEnergyUnit → Joule}
```

```
46.6345
```

```
73.8713
```

```
120.506
```

VerboseSelfInductance: formatted data of the self-inductance for the first component of the system. Default unit (Micro Henry).

```
VerboseSelfInductance[loopferro1, 1]
```

```
System name: loopferro1
Self inductance
Precision Level : 1


Component n° 1
Component type: circular loop winding
Number of turns:        120.
Current:      150.000000 A
Proper self inductance:              4145.288709 μH
Ferromagnetic contribution:          6566.338185 μH
Total self inductance:              10711.626894 μH
```

VerboseSelfFluxInductance: formatted data of the self flux inductance for the first component of the system. Unit is set in Milli Weber.

```
VerboseSelfFluxInductance[loopferro1, 1, FluxUnit → Milli Weber]
```

```
System name: loopferro1
Self flux inductance
Precision Level : 1


Component n° 1
Component type: circular loop winding
Number of turns:        120.
Current:      150.000000 A
Proper self inductance:               621.793306 mWb
Ferromagnetic contribution:           984.950728 mWb
Total self inductance:               1606.744034 mWb
```

VerboseSelfMagneticEnergy: formatted data of the magnetic energy for the first component of the system. Unit is set in Joule.

```
VerboseSelfMagneticEnergy[loopferro1, 1, MagneticEnergyUnit → Joule]
```

```
System name: loopferro1
Self magnetic energy
Precision Level : 1


Component n° 1
Component type: circular loop winding
Number of turns:        120.
Current:      150.000000 A
Proper self inductance:                46.634498 J
Ferromagnetic contribution:            73.871305 J
Total self inductance:                120.505803 J
```

■ 12-3 Ferro Self-inductance of a Square Section Solenoid

■ 12-3-1 Entering and Drawing the Data

Single square section solenoid and an axial soft ferromagnetic cylindrical bar

```
solferro1 = AXSMagneticSystem[
  {AXSSolenoid[CopperOFHC, 215, 375, 80, 0, 120, 150, FillingFactor -> 0.6],
   AXSSoftMagneticComponent[IronInd3, 0, 160, 350, 0]},
  SystemLabel -> "single square solenoid with ferromagnetic core"]
```

– AXSMagneticSystem :{-solenoid-, -soft-} –

DrawSystem[solferro1]

System name: **solferro1**
single square solenoid with ferromagnetic
core
3D view ZYX
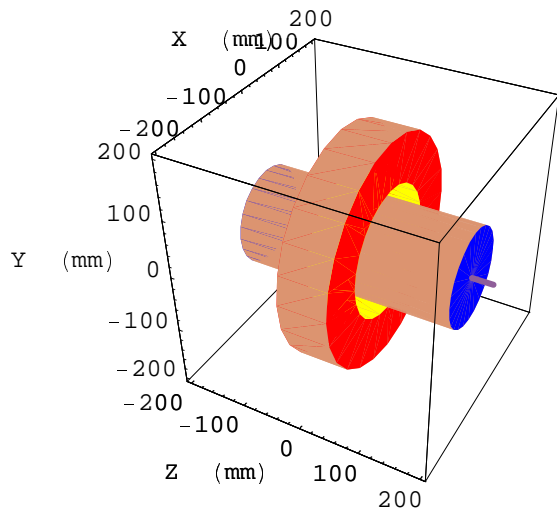
- Graphics3D -

DrawSystem2D[solferro1]

System name: **solferro1**
single square solenoid with ferromagnetic
core
2D view ZR
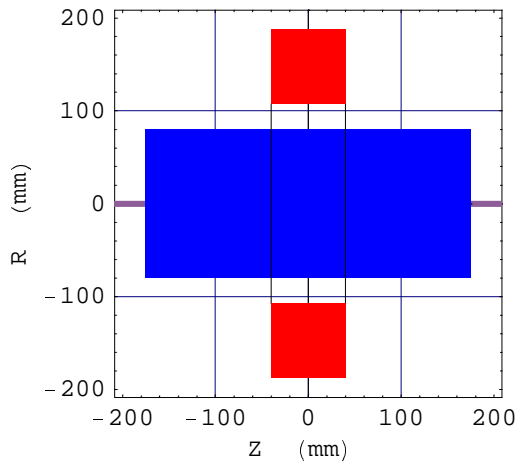
■ Graphics ■

■ 12-3-2 Ferro Self-inductance of a Square Section Solenoid

Induction field Bz (i.e. total magnetic field or flux density) at the center of the bar

FieldPointOrigin[solferro1]

System name: **solferro1**
The magnetic field (in Gauss) at the origin
Calculation mode: Automatic Mode
Br   0.000000 G
Bz   3212.042336 G
Bm   3212.042336 G

SelfInductance for the first component of the system, using the ContributionType option, that allows selecting either the Proper SelfInductance, or the Ferro SelfInductance or the Total. The default unit is Micro Henry:

SelfInductance[solferro1, 1, ContributionType → Proper]
SelfInductance[solferro1, 1, ContributionType → Ferro]
SelfInductance[solferro1, 1, ContributionType → Total]

4111.13

5989.21

10100.3

SelfFluxInductance: total magnetic Flux for the first component of the system. Unit is set in Milli Weber

```
MagneticaUserUnits[FluxUnit → Milli Weber];
SelfFluxInductance[solferro1, 1, ContributionType → Proper]
SelfFluxInductance[solferro1, 1, ContributionType → Ferro]
SelfFluxInductance[solferro1, 1, ContributionType → Total]
MagneticaUserUnits[];
```

616.67

898.382

1515.05

SelfMagneticEnergy: magnetic energy for the first component of the system. Unit is set in Joule.

```
MagneticaUserUnits[MagneticEnergyUnit → Joule]
SelfMagneticEnergy[solferro1, 1, ContributionType → Proper]
SelfMagneticEnergy[solferro1, 1, ContributionType → Ferro]
SelfMagneticEnergy[solferro1, 1, ContributionType → Total]
MagneticaUserUnits[];
```

{MagneticEnergyUnit → Joule}

46.2503

67.3787

113.629

VerboseSelfInductance: formatted data of the self-inductance for the first component of the system. Default unit (Micro Henry).

```
VerboseSelfInductance[solferro1, 1]
```

```
System name: solferro1
Self inductance
Precision Level : 1

Component n° 1
Component type: cylindrical solenoid
Number of turns:         120.
Current:       150.000000 A
Proper self inductance:             4111.134187 μH
Ferromagnetic contribution:         5989.213516 μH
Total self inductance:             10100.347702 μH
```

VerboseSelfFluxInductance: formatted data of the self flux inductance for the first component of the system. Unit is set in Milli Weber.

```
VerboseSelfFluxInductance[solferro1, 1, FluxUnit → Milli Weber]

System name: solferro1
Self flux inductance
Precision Level : 1


Component n° 1
Component type: cylindrical solenoid
Number of turns:        120.
Current:        150.000000 A
Proper self inductance:                616.670128 mWb
Ferromagnetic contribution:            898.382027 mWb
Total self inductance:                1515.052155 mWb
```

VerboseSelfMagneticEnergy: formatted data of the magnetic energy for the first component of the system. Unit is set in Joule.

```
VerboseSelfMagneticEnergy[solferro1, 1, MagneticEnergyUnit → Joule]

System name: solferro1
Self magnetic energy
Precision Level : 1


Component n° 1
Component type: cylindrical solenoid
Number of turns:        120.
Current:        150.000000 A
Proper self inductance:                46.250260 J
Ferromagnetic contribution:            67.378652 J
Total self inductance:                113.628912 J
```

## ■ 12-4 Ferro Self-inductance of a Long Solenoid

■ 12-4-1 Entering and Drawing the Data

Single long solenoid and an axial soft ferromagnetic cylindrical bar.

```
solferro2 = AXSMagneticSystem[
  {AXSSolenoid[CopperOFHC, 215, 265, 348, 0, 120, 150, FillingFactor -> 0.6],
   AXSSoftMagneticComponent[SiSteel2, 0, 160, 350, 0]},
  SystemLabel → "long solenoid with ferromagnetic core"]

- AXSMagneticSystem :{-solenoid-, -soft-} -
```

DrawSystem[solferro2]

System name: **solferro2**
long solenoid with ferromagnetic
core
3D view ZYX
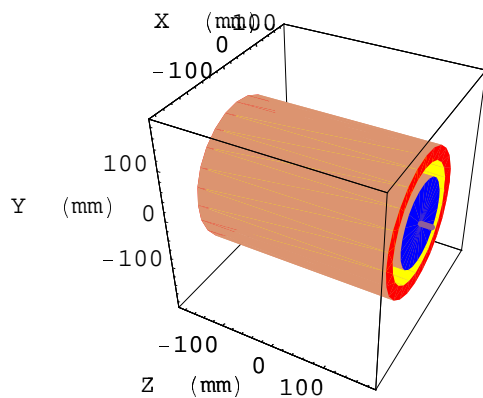
▪ Graphics3D ▪

DrawSystem2D[solferro2]

System name: **solferro2**
long solenoid with ferromagnetic
core
2D view ZR
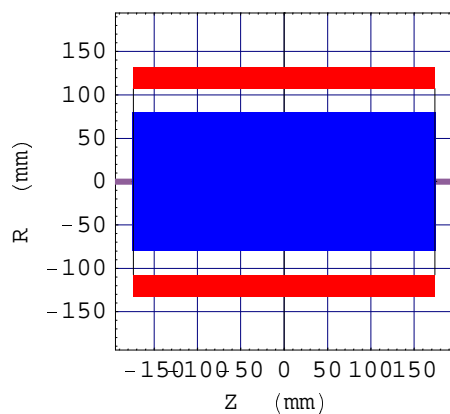
▪ Graphics ▪

▪ 12-4-2 Ferro Self-inductance of a Long Solenoid

Induction field Bz (i.e. total magnetic field or flux density) at the center of the bar

```
FieldPointOrigin[solferro2]
```

```
System name: solferro2
The magnetic field (in Gauss) at the origin
Calculation mode: Automatic Mode
Br          0.000000 G
Bz       2581.031978 G
Bm       2581.031978 G
```

SelfInductance for the first component of the system, using the ContributionType option, that allows selecting either the Proper SelfInductance, or the Ferro SelfInductance or the Total. The default unit is Micro Henry:

```
SelfInductance[solferro2, 1, ContributionType → Proper]
SelfInductance[solferro2, 1, ContributionType → Ferro]
SelfInductance[solferro2, 1, ContributionType → Total]
```

```
1645.83
```

```
4616.04
```

```
6261.87
```

SelfFluxInductance: total magnetic Flux for the first component of the system. Unit is set in Milli Weber.

```
MagneticaUserUnits[FluxUnit → Milli Weber];
SelfFluxInductance[solferro2, 1, ContributionType → Proper]
SelfFluxInductance[solferro2, 1, ContributionType → Ferro]
SelfFluxInductance[solferro2, 1, ContributionType → Total]
MagneticaUserUnits[];
```

```
246.874
```

```
692.406
```

```
939.28
```

SelfMagneticEnergy: magnetic energy for the first component of the system. Unit is set in Joule.

```
MagneticaUserUnits[MagneticEnergyUnit → Joule]
SelfMagneticEnergy[solferro2, 1, ContributionType → Proper]
SelfMagneticEnergy[solferro2, 1, ContributionType → Ferro]
SelfMagneticEnergy[solferro2, 1, ContributionType → Total]
MagneticaUserUnits[];
```

```
{MagneticEnergyUnit → Joule}
```

```
18.5155
```

```
51.9304
```

```
70.446
```

VerboseSelfInductance: formatted data of the self-inductance for the first component of the system. Default unit (Micro Henry).

```
VerboseSelfInductance[solferro2, 1]

System name: solferro2
Self inductance
Precision Level : 1


Component n° 1
Component type: cylindrical solenoid
Number of turns:       120.
Current:      150.000000 A
Proper self inductance:            1645.825760 μH
Ferromagnetic contribution:        4616.039742 μH
Total self inductance:             6261.865501 μH
```

VerboseSelfFluxInductance: formatted data of the self flux inductance for the first component of the system. Unit is set in Milli Weber.

```
VerboseSelfFluxInductance[solferro2, 1, FluxUnit → Milli Weber]

System name: solferro2
Self flux inductance
Precision Level : 1


Component n° 1
Component type: cylindrical solenoid
Number of turns:       120.
Current:      150.000000 A
Proper self inductance:            246.873864 mWb
Ferromagnetic contribution:        692.405961 mWb
Total self inductance:             939.279825 mWb
```

VerboseSelfMagneticEnergy: formatted data of the magnetic energy for the first component of the system. Unit is set in Joule

```
VerboseSelfMagneticEnergy[solferro2, 1, MagneticEnergyUnit → Joule]

System name: solferro2
Self magnetic energy
Precision Level : 1


Component n° 1
Component type: cylindrical solenoid
Number of turns:        120.
Current:      150.000000 A
Proper self inductance:            18.515540 J
Ferromagnetic contribution:        51.930447 J
Total self inductance:             70.445987 J
```

■ 12-5 Ferro Mutual Inductance of Circular Loops

■ 12-5-1 Entering and Drawing the Data

A pair of circular loops, set symmetrically with respect to the middle plane and an axial soft ferromagnetic cylindrical bar.

```
looppairferro1 = AXSMagneticSystem[{
    AXSCircularLoop[CopperOFHC, 275, 80, -80, 120, 150, FillingFactor → 0.6],
    AXSCircularLoop[CopperOFHC, 275, 80, 80, 120, 150, FillingFactor → 0.6],
    AXSSoftMagneticComponent[SteelCommon1, 0, 160, 350, 0]},
  SystemLabel → "pair of loops and ferromagnetic core"]
```

- AXSMagneticSystem :{-loop-, -loop-, -soft-} -

DrawSystem[looppairferro1]

System name: **looppairferro1**
*pair of loops and ferromagnetic
core*
3D view ZYX
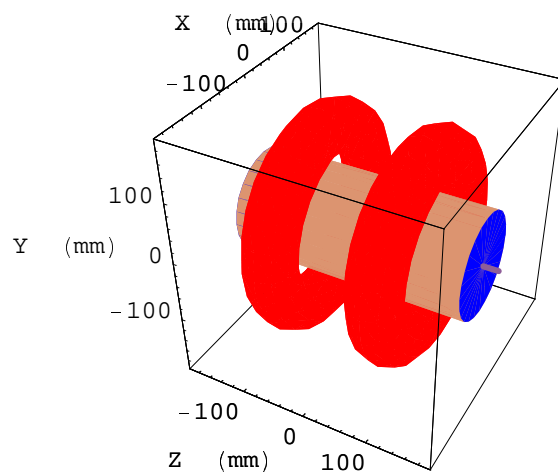
- Graphics3D -

DrawSystem2D[looppairferro1]

System name: **looppairferro1**
*pair of loops and ferromagnetic
core*
2D view ZR
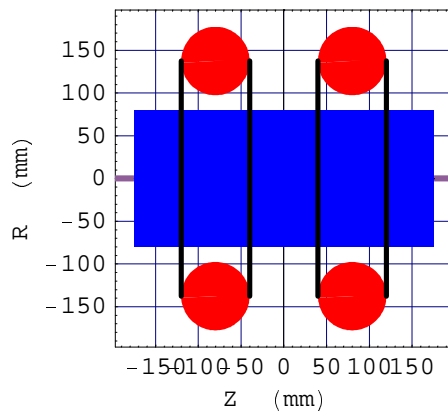
- Graphics -

■ **12-5-2 Ferro Mutual Inductance of Circular Loops**

Induction field Bz (i.e. total magnetic field or flux density) at the center of the bar

>   **FieldPointOrigin[looppairferro1]**
>
>   System name: **looppairferro1**
>   The magnetic field (in Gauss) at the origin
>   Calculation mode: Automatic Mode
>   Br   0.000000 G
>   Bz   5273.405637 G
>   Bm   5273.405637 G
>
>   **MutualInductance[looppairferro1, {1, 2}, ContributionType → Proper]**
>   **MutualInductance[looppairferro1, {1, 2}, ContributionType → Ferro]**
>   **MutualInductance[looppairferro1, {1, 2}, ContributionType → Total]**
>
>   783.422
>
>   4107.03
>
>   4890.45
>
>   **VerboseInductanceMatrix[looppairferro1]**
>
>   System name: **looppairferro1**
>   Inductance matrix (in Micro Henry)
>   Precision Level : 1
>   1   2
>   1   4145.2887   783.4223
>   2   783.4223   4145.2887
>
>   Involved components
>   components   type   nb turns   current
>   1   loop   120.   150.000000 A
>   2   loop   120.   150.000000 A
>
>   **VerboseInductanceMatrix[looppairferro1, ContributionType → Ferro]**
>
>   System name: **looppairferro1**
>   Inductance matrix (in Micro Henry)
>   Precision Level : 1
>   1   2
>   1   5567.8592   4107.0297
>   2   4107.0297   5567.8592
>
>   Involved components
>   components   type   nb turns   current
>   1   loop   120.   150.000000 A
>   2   loop   120.   150.000000 A

VerboseInductanceMatrix[looppairferro1, ContributionType → Total]

System name: **looppairferro1**
Inductance matrix (in Micro Henry)
Precision Level : 1
```
   1          2
1   9713.1479   4890.4520
2   4890.4520   9713.1479
```

Involved components
```
components      type     nb turns        current
     1          loop      120.          150.000000 A
     2          loop      120.          150.000000 A
```

InductanceMatrix[looppairferro1][[1]] // MatrixForm

$$\begin{pmatrix} 4145.29 & 783.422 \\ 783.422 & 4145.29 \end{pmatrix}$$

VerboseFluxInductanceMatrix[looppairferro1, FluxUnit → Milli Weber]

System name: **looppairferro1**
Flux inductance matrix (in Milli Weber)
Precision Level : 1
```
   1          2
1    621.7933    117.5133
2    117.5133    621.7933
```

Involved components
```
components      type     nb turns        current
     1          loop      120.          150.000000 A
     2          loop      120.          150.000000 A
```

VerboseFluxInductanceMatrix[looppairferro1,
 FluxUnit → Milli Weber, ContributionType → Ferro]

System name: **looppairferro1**
Flux inductance matrix (in Milli Weber)
Precision Level : 1
```
   1          2
1    835.1789    616.0545
2    616.0545    835.1789
```

Involved components
```
components      type     nb turns        current
     1          loop      120.          150.000000 A
     2          loop      120.          150.000000 A
```

VerboseFluxInductanceMatrix[looppairferro1,
 FluxUnit → Milli Weber, ContributionType → Total]

System name: **looppairferro1**
Flux inductance matrix (in Milli Weber)
Precision Level : 1
```
    1           2
1   1456.9722    733.5678
2    733.5678   1456.9722
```

```
Involved components
components      type    nb turns        current
    1           loop    120.        150.000000 A
    2           loop    120.        150.000000 A
```

FluxInductanceMatrix[looppairferro1, ContributionType → Total]

$\{\{\{1.45697 \times 10^6,\ 733568.\},\ \{733568.,\ 1.45697 \times 10^6\}\},\ \{1,\ 2\}\}$

VerboseMagneticEnergyMatrix[looppairferro1, MagneticEnergyUnit → Joule]

System name: **looppairferro1**
Magnetic energy matrix (in Joule)
Precision Level : 1
```
    1           2
1    46.6345      8.8135
2     8.8135     46.6345
```

```
Involved components
components      type    nb turns        current
    1           loop    120.        150.000000 A
    2           loop    120.        150.000000 A
```

VerboseMagneticEnergyMatrix[looppairferro1,
 ContributionType → Ferro, MagneticEnergyUnit → Joule]

System name: **looppairferro1**
Magnetic energy matrix (in Joule)
Precision Level : 1
```
    1           2
1    62.6384     46.2041
2    46.2041     62.6384
```

```
Involved components
components      type    nb turns        current
    1           loop    120.        150.000000 A
    2           loop    120.        150.000000 A
```

MagneticEnergyMatrix[looppairferro1,
 ContributionType → Ferro, MagneticEnergyUnit → Joule]

$\{\{\{62.6384,\ 46.2041\},\ \{46.2041,\ 62.6384\}\},\ \{1,\ 2\}\}$

```
VerboseMagneticEnergyMatrix[looppairferro1,
 ContributionType → Total, MagneticEnergyUnit → Joule]

System name: looppairferro1
Magnetic energy matrix (in Joule)
Precision Level : 1
   1          2
1  109.2729    55.0176
2   55.0176   109.2729


Involved components
components     type     nb turns        current
        1      loop     120.         150.000000 A
        2      loop     120.         150.000000 A


InductanceCoupling[looppairferro1, {1, 2}]

{{0.188991, 0.592616}, {0.0357176, 0.351193}, {0.964282, 0.648807}, {1., 1.}}


VerboseInductanceCoupling[looppairferro1, {1, 2}]

System name: looppairferro1
Inductance Coupling
Precision Level :                                      1
Number of components                                   3
Number of current carrying component                   2
Number of ferro-magnetic (non rigid) components        1
Number of components with reversible magnetization     1
                        primary coil           secondary coil
Component number:           1.0000                 2.0000
Component category:     loop                   loop
Circuits current (A):    150.0000                150.0000
Number of turns:         120.0000                120.0000
Transfo turn ratio               1.000
air core data
Self inductance (μH)    4145.2887               4145.2887
Mutual induc. 1->2 (μH)           783.4223
Self flux (μWb)        621793.3060             621793.3060
Mutual flux 1->2 (μWb)         117513.3430
ferro core data
Self inductance (μH)    5567.8592  5567.8592
Mutual induc. 1->2 (μH)           4107.0297
Self flux (μWb)        835178.8730             835178.8730
Mutual flux 1->2 (μWb)         616054.4600
                         air core        with ferro core
Coupling coefficient        0.188991           0.592616
Efficiency coefficient      0.035718           0.351193
Leakage coefficient         0.964282           0.648807
Transformation coeff.    1.0000              1.0000
```

■ 12-6 Ferro Mutual Inductance of Square Section Solenoids

■ 12-6-1 Entering and Drawing the Data

A pair of square cross section solenoids, set symmetrically with respect to the middle plane and an axial soft ferromagnetic cylindrical bar.

```
solpairferro1 = AXSMagneticSystem[
  {AXSSolenoid[CopperOFHC, 215, 375, 80, -80, 120, 150, FillingFactor → 0.6],
   AXSSolenoid[CopperOFHC, 215, 375, 80, 80, 120, 150, FillingFactor → 0.6],
   AXSSoftMagneticComponent[IronInd3, 0, 160, 350, 0]},
  SystemLabel → "pair square solenoids with ferromagnetic core"]
```

- AXSMagneticSystem :{-solenoid-, -solenoid-, -soft-} -

DrawSystem[solpairferro1]

System name: **solpairferro1**
pair square solenoids with ferromagnetic
core
3D view ZYX
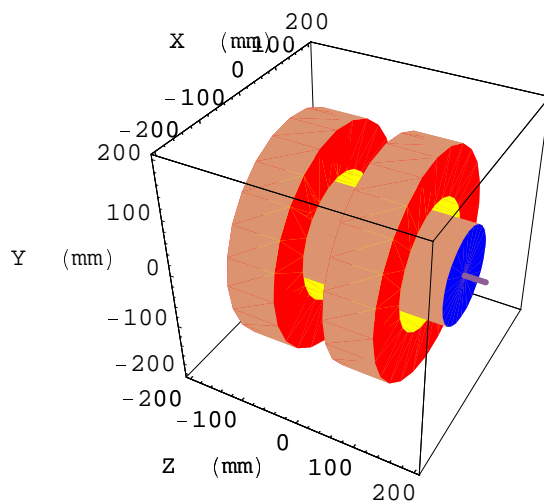
- Graphics3D -

DrawSystem2D[solpairferro1]

System name: **solpairferro1**
pair square solenoids with ferromagnetic
core
2D view ZR
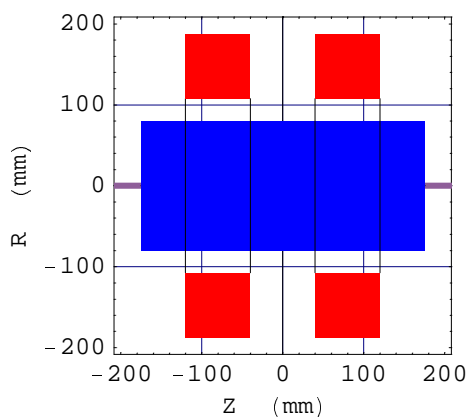
- Graphics -

■ 12-6-2 Ferro Mutual Inductance of Square Section Solenoids

Induction field Bz (i.e. total magnetic field or flux density) at the center of the bar

```
FieldPointOrigin[solpairferro1]
```

```
System name: solpairferro1
The magnetic field (in Gauss) at the origin
Calculation mode: Automatic Mode
Br            0.000000 G
Bz         5050.199689 G
Bm         5050.199689 G
```

```
MutualInductance[solpairferro1, {1, 2}, ContributionType → Proper]
MutualInductance[solpairferro1, {1, 2}, ContributionType → Ferro]
MutualInductance[solpairferro1, {1, 2}, ContributionType → Total]
```

```
957.298
```

```
3924.42
```

```
4881.72
```

```
VerboseInductanceMatrix[solpairferro1]
```

```
System name: solpairferro1
Inductance matrix (in Micro Henry)
Precision Level : 1
   1           2
1   4111.1342    957.2979
2    957.2979   4111.1342
```

```
Involved components
components       type      nb turns        current
     1         solenoid    120.         150.000000 A
     2         solenoid    120.         150.000000 A
```

```
VerboseInductanceMatrix[solpairferro1, ContributionType → Ferro]
```

```
System name: solpairferro1
Inductance matrix (in Micro Henry)
Precision Level : 1
   1           2
1   5054.3533    3924.4226
2   3924.4226    5054.3533
```

```
Involved components
components       type      nb turns        current
     1         solenoid    120.         150.000000 A
     2         solenoid    120.         150.000000 A
```

VerboseInductanceMatrix[solpairferro1, ContributionType → Total]

System name: **solpairferro1**
Inductance matrix (in Micro Henry)
Precision Level : 1
```
   1          2
1   9165.4874   4881.7205
2   4881.7205   9165.4874
```

```
Involved components
components        type        nb turns        current
     1          solenoid      120.          150.000000 A
     2          solenoid      120.          150.000000 A
```

InductanceMatrix[solpairferro1][[1]] // MatrixForm

$$\begin{pmatrix} 4111.13 & 957.298 \\ 957.298 & 4111.13 \end{pmatrix}$$

VerboseFluxInductanceMatrix[solpairferro1, FluxUnit → Milli Weber]

System name: **solpairferro1**
Flux inductance matrix (in Milli Weber)
Precision Level : 1
```
   1          2
1   616.6701    143.5947
2   143.5947    616.6701
```

```
Involved components
components        type        nb turns        current
     1          solenoid      120.          150.000000 A
     2          solenoid      120.          150.000000 A
```

VerboseFluxInductanceMatrix[solpairferro1,
 FluxUnit → Milli Weber, ContributionType → Ferro]

System name: **solpairferro1**
Flux inductance matrix (in Milli Weber)
Precision Level : 1
```
   1          2
1   758.1530    588.6634
2   588.6634    758.1530
```

```
Involved components
components        type        nb turns        current
     1          solenoid      120.          150.000000 A
     2          solenoid      120.          150.000000 A
```

VerboseFluxInductanceMatrix[solpairferro1,
 FluxUnit → Milli Weber, ContributionType → Total]

```
System name: solpairferro1
Flux inductance matrix (in Milli Weber)
Precision Level : 1
   1          2
1   1374.8231    732.2581
2    732.2581   1374.8231


Involved components
components       type        nb turns        current
     1         solenoid     120.          150.000000 A
     2         solenoid     120.          150.000000 A
```

FluxInductanceMatrix[solpairferro1, ContributionType → Total]

$\{\{\{1.37482 \times 10^6, 732258.\}, \{732258., 1.37482 \times 10^6\}\}, \{1, 2\}\}$

VerboseMagneticEnergyMatrix[solpairferro1, MagneticEnergyUnit → Joule]

```
System name: solpairferro1
Magnetic energy matrix (in Joule)
Precision Level : 1
   1          2
1    46.2503    10.7696
2    10.7696    46.2503


Involved components
components       type        nb turns        current
     1         solenoid     120.          150.000000 A
     2         solenoid     120.          150.000000 A
```

VerboseMagneticEnergyMatrix[solpairferro1,
 ContributionType → Ferro, MagneticEnergyUnit → Joule]

```
System name: solpairferro1
Magnetic energy matrix (in Joule)
Precision Level : 1
   1          2
1    56.8615    44.1498
2    44.1498    56.8615


Involved components
components       type        nb turns        current
     1         solenoid     120.          150.000000 A
     2         solenoid     120.          150.000000 A
```

```
VerboseMagneticEnergyMatrix[solpairferro1,
 ContributionType → Total, MagneticEnergyUnit → Joule]
```

```
System name: solpairferro1
Magnetic energy matrix (in Joule)
Precision Level : 1
   1          2
1   103.1117    54.9194
2    54.9194   103.1117


Involved components
components        type      nb turns         current
      1         solenoid    120.         150.000000 A
      2         solenoid    120.         150.000000 A
```

```
InductanceCoupling[solpairferro1, {1, 2}]
```

{{0.232855, 0.607515}, {0.0542214, 0.369074}, {0.945779, 0.630926}, {1., 1.}}

```
VerboseInductanceCoupling[solpairferro1, {1, 2}]
```

```
System name: solpairferro1
Inductance Coupling
Precision Level :                                        1
Number of components                                     3
Number of current carrying component                     2
Number of ferro-magnetic (non rigid) components          1
Number of components with reversible magnetization       1
                       primary coil           secondary coil
Component number:          1.0000                 2.0000
Component category:     solenoid                solenoid
Circuits current (A):   150.0000                150.0000
Number of turns:        120.0000                120.0000
Transfo turn ratio                 1.000
air core data
Self inductance (μH)    4111.1342               4111.1342
Mutual induc. 1->2 (μH)             957.2979
Self flux (μWb)         616670.1280             616670.1280
Mutual flux 1->2 (μWb)          143594.6810
ferro core data
Self inductance (μH)    5054.3533   5054.3533
Mutual induc. 1->2 (μH)             3924.4226
Self flux (μWb)         758152.9880             758152.9880
Mutual flux 1->2 (μWb)          588663.3960
                        air core            with ferro core
Coupling coefficient            0.232855            0.607515
Efficiency coefficient          0.054221            0.369074
Leakage coefficient             0.945779            0.630926
Transformation coeff.    1.0000              1.0000
```

■ 12-7 Ferro Mutual Inductance of Pair of Long Solenoids

■ 12-7-1 Entering and Drawing the Data

A pair of radially superposed solenoids of long rectangular cross section, and an axial soft ferromagnetic cylindrical bar.

```
solpairferro2 = AXSMagneticSystem[{
    AXSSolenoid[CopperOFHC, 215, 265, 348, 0, 120, 150, FillingFactor → 0.6],
    AXSSolenoid[CopperOFHC,  162, 212, 348, 0, 120, 150, FillingFactor → 0.6],
    AXSSoftMagneticComponent[SiSteel2, 0, 160, 350, 0]},
  SystemLabel → "pair of long superposed solenoids with ferromagnetic core"]
```

– AXSMagneticSystem :{-solenoid-, -solenoid-, -soft-} –

DrawSystem[solpairferro2]

System name: **solpairferro2**
pair of long superposed solenoids
with ferromagnetic core
3D view ZYX
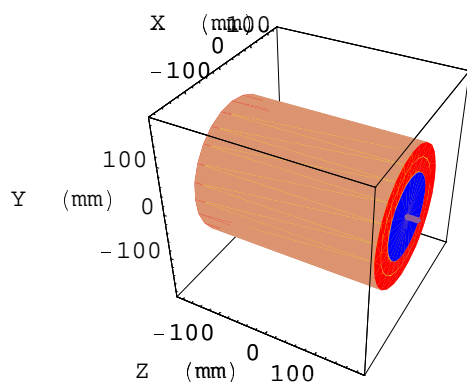
▪ Graphics3D ▪

DrawSystem2D[solpairferro2]

System name: **solpairferro2**
pair of long superposed solenoids
with ferromagnetic core
2D view ZR
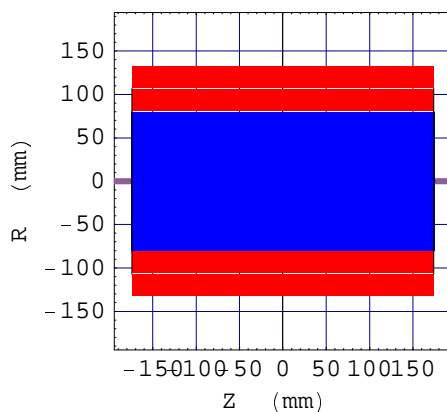
▪ Graphics ▪

■ 12-7-2 Ferro Mutual Inductance of Long Solenoids

Induction field Bz (i.e. total magnetic field or flux density) at the center of the bar

FieldPointOrigin[solpairferro2]

System name: **solpairferro2**
The magnetic field (in Gauss) at the origin
Calculation mode: Automatic Mode
Br   0.000000 G
Bz   5351.936535 G
Bm   5351.936535 G

MutualInductance[solpairferro2, {1, 2}, ContributionType → Proper]
MutualInductance[solpairferro2, {1, 2}, ContributionType → Ferro]
MutualInductance[solpairferro2, {1, 2}, ContributionType → Total]

1066.02

5008.05

6074.07

VerboseInductanceMatrix[solpairferro2]

System name: **solpairferro2**
Inductance matrix (in Micro Henry)
Precision Level : 1
   1   2
1   1645.8258   1066.0167
2   1066.0167   1036.3390

Involved components
components   type   nb turns   current
     1   solenoid   120.   150.000000 A
     2   solenoid   120.   150.000000 A

VerboseInductanceMatrix[solpairferro2, ContributionType → Ferro]

System name: **solpairferro2**
Inductance matrix (in Micro Henry)
Precision Level : 1
   1   2
1   4616.0397   5008.0506
2   5007.4878   5434.8157

Involved components
components   type   nb turns   current
     1   solenoid   120.   150.000000 A
     2   solenoid   120.   150.000000 A

VerboseInductanceMatrix[solpairferro2, ContributionType → Total]

System name: **solpairferro2**
Inductance matrix (in Micro Henry)
Precision Level : 1
```
   1          2
1   6261.8655   6074.0673
2   6073.5045   6471.1547
```

Involved components
```
components       type      nb turns         current
    1          solenoid    120.          150.000000 A
    2          solenoid    120.          150.000000 A
```

InductanceMatrix[solpairferro2][[1]] // MatrixForm

$$\begin{pmatrix} 1645.83 & 1066.02 \\ 1066.02 & 1036.34 \end{pmatrix}$$

VerboseFluxInductanceMatrix[solpairferro2, FluxUnit → Milli Weber]

System name: **solpairferro2**
Flux inductance matrix (in Milli Weber)
Precision Level : 1
```
   1          2
1   246.8739    159.9025
2   159.9025    155.4509
```

Involved components
```
components       type      nb turns         current
    1          solenoid    120.          150.000000 A
    2          solenoid    120.          150.000000 A
```

VerboseFluxInductanceMatrix[solpairferro2,
 ContributionType → Ferro, FluxUnit → Milli Weber]

System name: **solpairferro2**
Flux inductance matrix (in Milli Weber)
Precision Level : 1
```
   1          2
1   692.4060    751.2076
2   751.1232    815.2224
```

Involved components
```
components       type      nb turns         current
    1          solenoid    120.          150.000000 A
    2          solenoid    120.          150.000000 A
```

VerboseFluxInductanceMatrix[solpairferro2,
 ContributionType → Total, FluxUnit → Milli Weber]

```
System name: solpairferro2
Flux inductance matrix (in Milli Weber)
Precision Level : 1
   1          2
1   939.2798    911.1101
2   911.0257    970.6732


Involved components
components        type        nb turns        current
     1          solenoid     120.         150.000000 A
     2          solenoid     120.         150.000000 A
```

VerboseMagneticEnergyMatrix[solpairferro2, MagneticEnergyUnit → Joule]

```
System name: solpairferro2
Magnetic energy matrix (in Joule)
Precision Level : 1
   1          2
1    18.5155     11.9927
2    11.9927     11.6588


Involved components
components        type        nb turns        current
     1          solenoid     120.         150.000000 A
     2          solenoid     120.         150.000000 A
```

VerboseMagneticEnergyMatrix[solpairferro2,
 ContributionType → Ferro, MagneticEnergyUnit → Joule]

```
System name: solpairferro2
Magnetic energy matrix (in Joule)
Precision Level : 1
   1          2
1    51.9304     56.3406
2    56.3342     61.1417


Involved components
components        type        nb turns        current
     1          solenoid     120.         150.000000 A
     2          solenoid     120.         150.000000 A
```

VerboseMagneticEnergyMatrix[solpairferro2,
 ContributionType → Total, MagneticEnergyUnit → Joule]

```
System name: solpairferro2
Magnetic energy matrix (in Joule)
Precision Level : 1
   1          2
1    70.4460     68.3333
2    68.3269     72.8005


Involved components
components        type        nb turns        current
     1          solenoid     120.         150.000000 A
     2          solenoid     120.         150.000000 A
```

InductanceCoupling[solpairferro2, {1, 2}]

{{0.816246, 0.961992}, {0.666258, 0.925429},
 {0.333742, 0.0745708}, {0.77525, 1.01701}}

VerboseInductanceCoupling[solpairferro2, {1, 2}]

System name: **solpairferro2**
Inductance Coupling
Precision Level :   1
Number of components   3
Number of current carrying component   2
Number of ferro-magnetic (non rigid) components   1
Number of components with reversible magnetization   1

	primary coil	secondary coil
Component number:	1.0000	2.0000
Component category:	solenoid	solenoid
Circuits current (A):	150.0000	150.0000
Number of turns:	120.0000	120.0000
Transfo turn ratio	1.000	

air core data

Self inductance (μH)	1645.8258	1036.3390
Mutual induc. 1->2 (μH)	1066.0167	
Self flux (μWb)	246873.8640	155450.8510
Mutual flux 1->2 (μWb)	159902.5080	

ferro core data

Self inductance (μH)	4616.0397	5434.8157
Mutual induc. 1->2 (μH)	5008.0506	
Self flux (μWb)	692405.9610	815222.3560
Mutual flux 1->2 (μWb)	751207.5940	

	air core	with ferro core
Coupling coefficient	0.816246	0.961992
Efficiency coefficient	0.666258	0.925429
Leakage coefficient	0.333742	0.074571
Transformation coeff.	1 / 1.2899	1.0170

■ Chapter 13: Inductance Applications

■ 13-1 Introduction

The following cases are given as examples of *Magnetica* uses in electro-magnetic problems related to inductance.

1. Calculation of an inductance and LC resonance frequency.

2. Designing an air core inductance and effect of an iron core.

3. Inductance matrix and magnetic energy of an unshielded MRI superconducting magnet.

4. Inductance matrix and magnetic energy of a shielded MRI superconducting magnet.

5. Simulation of a single phase transformer.

■ 13-2 Inductance and LC Resonance Frequency

■ 13-2-1 Description

The problem: What is the inductance of an antenna made of 54 turns of copper wire on a 2 meter diameter circular frame, what is the resonant frequency when this antenna is coupled with a 1 Micro Farad capacitor and what capacitor will deliver a 1 Kilo Hertz resonance frequency ?

Results of the Analysis

1. inductance of the antenna: 17570 µHenry;

2. resonance frequency using a 1 µFarad capacitor: 1200 Hertz;

3. capacitor to obtain a 1000 Hertz resonance frequency.

■ 13-2-2 Antenna Coil: Data and Drawing

```
anten1 = AXSMagneticSystem[{
    AXSCircularLoop[CopperOFHC, 2000, 25, 0, 54, 2, ConductorDiameter → 1.67]},
   SystemLabel -> "2 meters diameter, 54 turns antenna"]

- AXSMagneticSystem :{-loop-} -
```

DrawSystem[anten1]

```
System name: anten1
2 meters diameter, 54 turns antenna
3D view ZYX
```
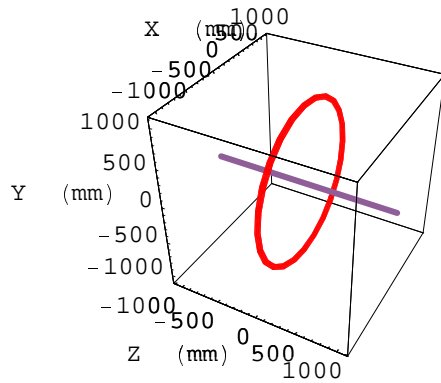


```
- Graphics3D -
```

■ 13-2-3 Antenna Coil: Inductance, Frequency and Capacitor Calculations

Inductance of the coil

selfant = SelfInductance[anten1, 1]

```
17570.
```

VerboseSelfInductance[anten1, 1]

```
System name: anten1
Self inductance
Precision Level : 1

Component n° 1
Component type: circular loop winding
Number of turns:        54.
Current:        2.000000 A
Proper self inductance:           17570.047895 μH
Ferromagnetic contribution:           0.000000 μH
Total self inductance:            17570.047895 μH
```

Resonant frequency antenna coupled with a 1 μFarad capacitor

ResonantFrequency[selfant, 1]

```
1200.7
```

Capacitor to obtain a 1 Kilo Hertz resonant frequency

ResonantCapacitor[selfant, 1000]

```
1.44167
```

■ 13-3 Designing an Air Core and Iron Core Inductance

■ 13-3-1 Description

Design objective: inductance specification 1 Milli Henry as small as possible using 0.2 mm copper wire both as

Case 1: an air core solenoid.

Case 2: an iron core system.

Results of the Design

1. Square solenoid of 280 turns using 13.7 meters of 0.2 mm diameter copper wire.

2. Air core inductance: 999.3 μHenry.

3. Soft iron core inductance 1436.1 μHenry.

■ 13-3-2 Preliminary Guess Calculation

Initial guess using Brook coil approximate formula: the Brook type coil shape provides an efficient way to design the coil in terms of the quantity of wire. Let's assume an average diameter of 15 mm.

```
? TurnsBrookCoil
```

```
TurnsBrookCoil Number of turns of a Brook's type coil of a given inductance Details…
```

```
TurnsBrookCoil[15, 1000]
```

```
280.883
```

■ 13-3-3 Design of the Air Core Solenoid

Air core system data using a 280 turn coil

```
milliHAirCore = AXSMagneticSystem[{
    AXSSolenoid[CopperOFHC, 10, 20, 5, 0, 280, 0.1, ConductorDiameter → 0.2]},
  SystemLabel -> "milli Henry coil, air core"]
```

```
- AXSMagneticSystem :{-solenoid-} -
```

DrawSystem[milliHAirCore]

> System name: **milliHAirCore**
> *milli Henry coil, air core*
> 3D view ZYX
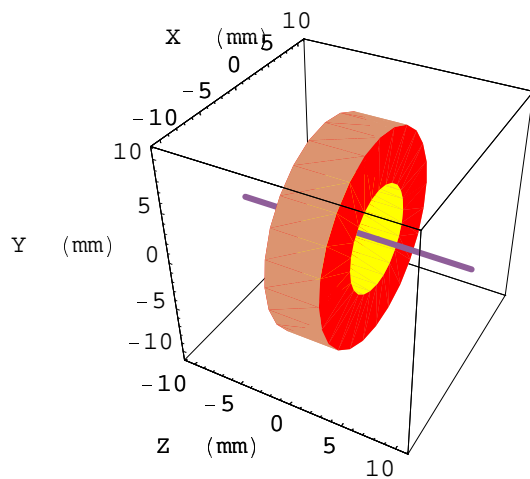
- Graphics3D -

DrawSystem2D[milliHAirCore]

> System name: **milliHAirCore**
> *milli Henry coil, air core*
> 2D view ZR
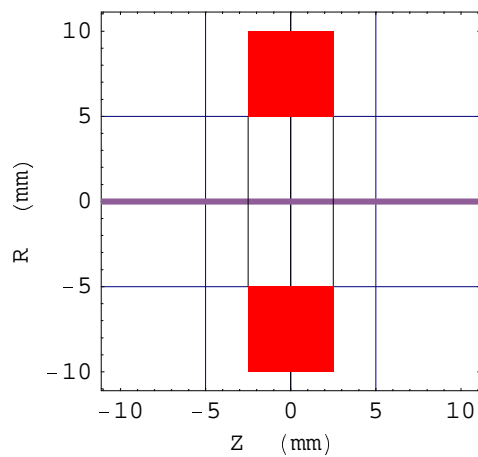
- Graphics -

Self-inductance of the air core coil

> **selfair = SelfInductance[milliHAirCore, 1]**

999.359

```
VerboseSelfInductance[milliHAirCore, 1]
```

```
System name: milliHAirCore
Self inductance
Precision Level : 1


Component n° 1
Component type: cylindrical solenoid
Number of turns:      280.
Current:      0.100000 A
Proper self inductance:           999.358670 µH
Ferromagnetic contribution:         0.000000 µH
Total self inductance:            999.358670 µH
```

■ 13-3-4 Design of the Ferromagnetic Core Solenoid

Adding a ferromagnetic pellet as iron core

```
milliHIronCore = AXSMagneticSystem[{
    AXSSolenoid[CopperOFHC, 10, 20, 5, 0, 280, 0.1, ConductorDiameter → 0.2],
    AXSSoftMagneticComponent[IronInd3, 0, 9.5, 5, 0]},
   SystemLabel -> "milli Henry coil, iron core"]
```

- AXSMagneticSystem :{-solenoid-, -soft-} -

```
DrawSystem[milliHIronCore]
```

System name: **milliHIronCore**
milli Henry coil, iron core
3D view ZYX
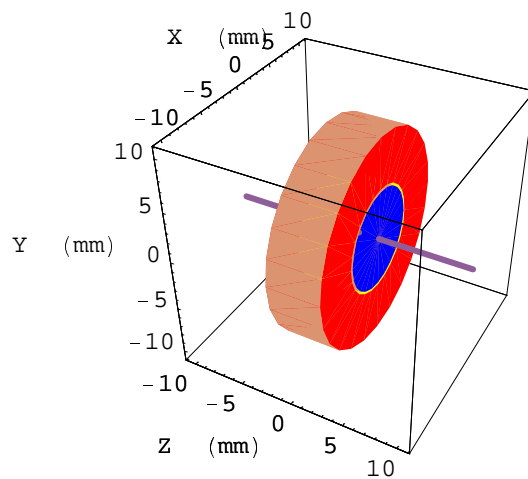
- Graphics3D -

```
DrawSystem2D[milliHIronCore]
```

> System name: **milliHIronCore**
> *milli Henry coil, iron core*
> 2D view ZR
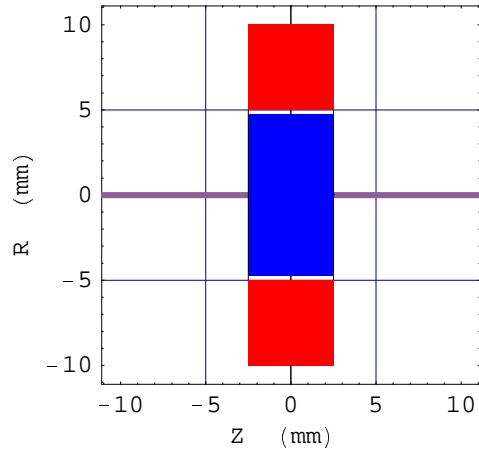
- Graphics -

Self-inductance of the iron core coil

```
selfiron = SelfInductance[milliHIronCore, 1]
```

```
999.359
```

```
VerboseSelfInductance[milliHIronCore, 1]
```

```
System name: milliHIronCore
Self inductance
Precision Level : 1

Component n° 1
Component type: cylindrical solenoid
Number of turns:        280.
Current:        0.100000 A
Proper self inductance:              999.358670 µH
Ferromagnetic contribution:          488.743590 µH
Total self inductance:              1488.102259 µH
```

■ 13-4 Magnetic Energy of an Unshielded MRI Superconducting Magnet

■ 13-4-1 Description

The MRI magnet is made of 3 pairs of superconducting coils and set to generate a 1.5 Tesla magnetic field. All the coils are in series.

Results of the Analysis

1. total inductance, assuming all coils in series: 11.93 Henry

2. stored magnetic energy: 3.22 Mega Joules.

▪ 13-4-2 Entering and Drawing Data

System data:

```
solsuper1 = AXSMagneticSystem[
  {AXSSolenoid[SuperNbTi4, 1240, 1320, 250, -874.8480174899299, 1015, 735.221],
   AXSSolenoid[SuperNbTi4, 1240, 1280, 250, -550.1194910091477, 598, 735.221],
   AXSSolenoid[SuperNbTi4, 1240, 1280, 300, -180.05374474797077, 635, 735.221],
   AXSSolenoid[SuperNbTi4, 1240, 1280, 300, 180.05374474797077, 635, 735.221],
   AXSSolenoid[SuperNbTi4, 1240, 1280, 250, 550.1194910091477, 598, 735.221],
   AXSSolenoid[SuperNbTi4, 1240, 1320, 250, 874.8480174899299, 1015, 735.221]},
  SystemLabel -> "unshielded MRI 1.5 T superconducting magnet"]
```

```
- AXSMagneticSystem :
 {-solenoid-, -solenoid-, -solenoid-, -solenoid-, -solenoid-, -solenoid-} -
```

Drawing

```
DrawSystem[solsuper1]
```

```
System name: solsuper1
unshielded MRI 1.5 T superconducting
magnet
3D view ZYX
```



```
- Graphics3D -
```

DrawSystem2D[solsuper1]

　System name: **solsuper1**
　unshielded MRI 1.5 T superconducting
　magnet
　2D view ZR

- Graphics -

■ 13-4-3 Inductance and Magnetic Energy Calculations

Field at the origin:

FieldPointOrigin[solsuper1]

System name: **solsuper1**
The magnetic field (in Gauss) at the origin
Calculation mode: Automatic Mode
Br　　　　　0.000000 G
Bz　　　15000.016390 G
Bm　　　15000.016390 G

Inductance system calculation

InductanceMatrix[solsuper1, InductanceUnit → Milli Henry][[1]] // MatrixForm

$$
\begin{pmatrix}
1975.66 & 448.55 & 183.729 & 87.4547 & 43.1059 & 46.1938 \\
448.55 & 691.051 & 244.349 & 98.0499 & 43.343 & 43.1059 \\
183.729 & 244.349 & 726.855 & 271.679 & 98.0499 & 87.4547 \\
87.4547 & 98.0499 & 271.679 & 726.855 & 244.349 & 183.729 \\
43.1059 & 43.343 & 98.0499 & 244.349 & 691.051 & 448.551 \\
46.1938 & 43.1059 & 87.4547 & 183.729 & 448.551 & 1975.66
\end{pmatrix}
$$

VerboseInductanceMatrix[solsuper1, InductanceUnit → Milli Henry]

```
System name: solsuper1
Inductance matrix (in Milli Henry)
Precision Level : 1
   1           2           3           4           5           6
1  1975.6628   448.5503    183.7292    87.4547     43.1059     46.1938
2  448.5503    691.0506    244.3493    98.0499     43.3430     43.1059
3  183.7292    244.3493    726.8548    271.6790    98.0499     87.4547
4  87.4547     98.0499     271.6790    726.8548    244.3493    183.7292
5  43.1059     43.3430     98.0499     244.3493    691.0506    448.5510
6  46.1938     43.1059     87.4547     183.7292    448.5510    1975.6628


Involved components
components       type       nb turns          current
        1        solenoid   1015.             735.221000 A
        2        solenoid   598.              735.221000 A
        3        solenoid   635.              735.221000 A
        4        solenoid   635.              735.221000 A
        5        solenoid   598.              735.221000 A
        6        solenoid   1015.             735.221000 A
```

? SystemInductance

```
SystemInductance[system] returns a list with the total
  Inductance, FluxInductane and Magnetic Energy (see MagneticEnergyMatrix).
  The options are ContributionType, PrecisionLevel, Componentlist Details…
```

SystemInductance[solsuper1]

$\{1.19305 \times 10^7, 8.77157 \times 10^9, 3.22452 \times 10^6\}$

VerboseSystemInductance[solsuper1, InductanceUnit → Henry,
 FluxUnit → Weber, MagneticEnergyUnit → Mega Joule]

```
System name: solsuper1
System Inductance
Precision Level :                       1
Number of components                    6
Number of current carrying component    6
                    Proper
Inductance (H)      11.9305
Flux (Wb)           8771.57
Energy (MJ)         3.22452
```

Magnetic energy stored in the system coils (it is the third element of the list returned by SystemInductance).

SystemInductance[solsuper1, MagneticEnergyUnit → Mega Joule][[3]]

```
3.22452
```

■ 13-5 Magnetic Energy of a Shielded MRI Superconducting Magnet

■ 13-5-1 Description

The MRI magnet is made of 3 pairs of superconducting coils identical to the previous case but with a thick steel shield surrounding the coils (mass 27 tons).

The coils are series connected and the current intensity is adjusted to generate a 1.5 Tesla magnetic field.

Results of the Analysis

1. total inductance, assuming all coils in series: 13.167 Henry

2. stored magnetic energy: 3.02 Mega Joules

■ 13-5-2 Entering and Drawing Data

System data:

```
iampf = 677.412;
solsuper2 = AXSMagneticSystem[
  {AXSSolenoid[SuperNbTi4, 1240, 1320, 250, -874.8480174899299, 1015, iampf],
   AXSSolenoid[SuperNbTi4, 1240, 1280, 250, -550.1194910091477, 598, iampf],
   AXSSolenoid[SuperNbTi4, 1240, 1280, 300, -180.05374474797077, 635, iampf],
   AXSSolenoid[SuperNbTi4, 1240, 1280, 300, 180.05374474797077, 635, iampf],
   AXSSolenoid[SuperNbTi4, 1240, 1280, 250, 550.1194910091477, 598, iampf],
   AXSSolenoid[SuperNbTi4, 1240, 1320, 250, 874.8480174899299, 1015, iampf],
   AXSSoftMagneticComponent[SteelCommon2, 1600, 2000, 2400, 0],
   AXSSoftMagneticComponent[SteelCommon2, 1300, 2000, 200, -1300],
   AXSSoftMagneticComponent[SteelCommon2, 1300, 2000, 200, 1300]},
  SystemLabel -> "shielded MRI 1.5 T superconducting magnet",
  MeshingLevel → 2]
```

```
- AXSMagneticSystem :{-solenoid-, -solenoid-, -solenoid-,
  -solenoid-, -solenoid-, -solenoid-, -soft-, -soft-, -soft-} -
```
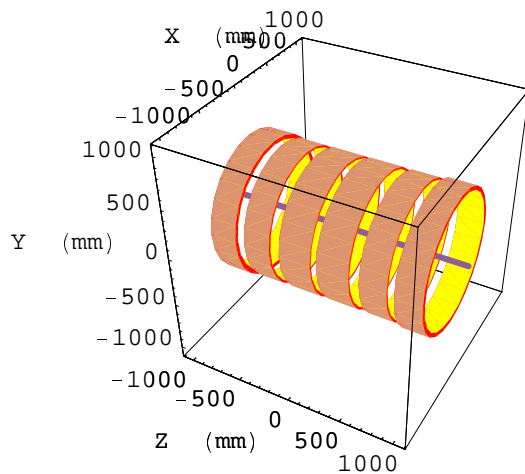
Drawing

```
DrawSystem2D[solsuper2]
```

```
System name: solsuper2
shielded MRI 1.5 T superconducting
magnet
2D view ZR
```
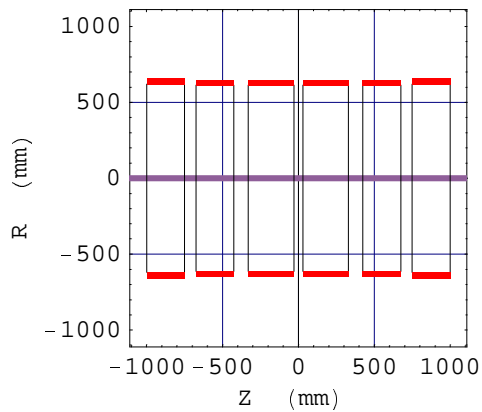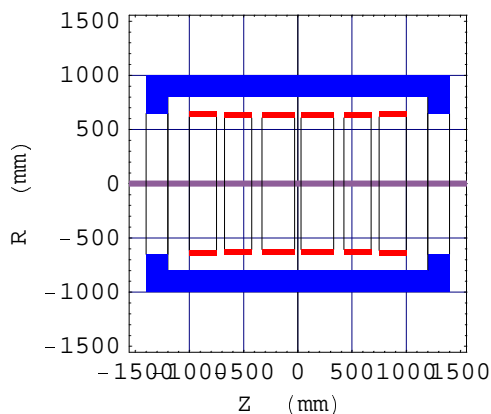


```
- Graphics -
```

■ **13-5-3 Inductance and Magnetic Energy Calculations**

Field at the origin (this could take time)

> **FieldPointOrigin[solsuper2]**
>
> System name: **solsuper2**
> The magnetic field (in Gauss) at the origin
> Calculation mode: Automatic Mode
> Br   0.000000 G
> Bz   15000.349488 G
> Bm   15000.349488 G

Inductance system calculation (this could take time)

> **InductanceMatrix[solsuper2,**
>   **InductanceUnit → Milli Henry, ContributionType → Total][[1]]**
>
> {{2293.42, 534.188, 183.882, 62.8324, 19.625, 13.402},
> {533.221, 772.423, 281.399, 93.748, 29.1166, 19.6238},
> {183.637, 281.401, 813.079, 312.08, 93.7486, 62.7596},
> {62.7596, 93.7486, 312.08, 813.079, 281.401, 183.637},
> {19.6238, 29.1166, 93.748, 281.399, 772.423, 533.221},
> {13.402, 19.625, 62.8324, 183.882, 534.189, 2293.42}}

> **VerboseInductanceMatrix[solsuper2, InductanceUnit → Henry, ContributionType → Total]**
>
> System name: **solsuper2**
> Inductance matrix (in Henry)
> Precision Level : 1

	1	2	3	4	5	6
1	2.2934	0.5342	0.1839	0.0628	0.0196	0.0134
2	0.5332	0.7724	0.2814	0.0937	0.0291	0.0196
3	0.1836	0.2814	0.8131	0.3121	0.0937	0.0628
4	0.0628	0.0937	0.3121	0.8131	0.2814	0.1836
5	0.0196	0.0291	0.0937	0.2814	0.7724	0.5332
6	0.0134	0.0196	0.0628	0.1839	0.5342	2.2934

> Involved components

components	type	nb turns	current
1	solenoid	1015.	677.412000 A
2	solenoid	598.	677.412000 A
3	solenoid	635.	677.412000 A
4	solenoid	635.	677.412000 A
5	solenoid	598.	677.412000 A
6	solenoid	1015.	677.412000 A

> **SystemInductance[solsuper2, ContributionType → Total,**
> **InductanceUnit → Henry, FluxUnit → Weber, MagneticEnergyUnit → Mega Joule]**
>
> {13.1672, 8919.6, 3.02112}

```
VerboseSystemInductance[solsuper2, ContributionType → Total,
  InductanceUnit → Henry, FluxUnit → Weber, MagneticEnergyUnit → Mega Joule]

System name: solsuper2
System Inductance
Precision Level :                                          1
Number of components                                       9
Number of current carrying component                       6
Number of ferro-magnetic (non rigid) components            3
Number of components with reversible magnetization         3
                    Proper        Ferro         Total
Inductance (H)      11.9305       1.23665       13.1672
Flux (Wb)           8081.88       837.721       8919.6
Energy (MJ)         2.73738       0.283741      3.02112
```

Magnetic energy stored in the system coils.

```
MagneticEnergyMatrix[solsuper2,
  MagneticEnergyUnit → Kilo Joule, ContributionType → Total][[1]]

{{526.211, 122.566, 42.1906, 14.4165, 4.50282, 3.075},
 {122.344, 177.227, 64.5652, 21.5099, 6.68061, 4.50255},
 {42.1343, 64.5656, 186.556, 71.6047, 21.51, 14.3998},
 {14.3998, 21.51, 71.6047, 186.556, 64.5656, 42.1343},
 {4.50255, 6.68061, 21.5099, 64.5652, 177.227, 122.344},
 {3.075, 4.50282, 14.4165, 42.1906, 122.566, 526.211}}

SystemInductance[solsuper2,
  ContributionType → Total, MagneticEnergyUnit → Mega Joule][[3]]

3.02112
```

▪ 13-6 Simulation of a Single Phase Transformer

▪ 13-6-1 Description

A single phase transformer can be viewed as a pair of independent magnetically coupled electric circuits. The common transformer has a soft iron core.

The first questions are: why an iron core and how to is the effect of the ferromagnetic core to be appreciated.

The use of the ferro inductance concept allows understanding this beneficial effect. We consider 3 cases of a monophase transformer:

    Case 1: without any iron core (i.e. air core).

    Case 2: with a limited iron armature.

    Case 3: with a complete iron armature.

The *magnetica* functionalities:

- `InductanceCoupling`,
- `TransformerAnalysis`,
- `TransformerMagneticLosses`.

provide a comprehensive analysis of transformer behavior. But there are significant differences between the various results as seen in Section10-2-2. **InductanceCoupling** works with usual formulas but the results are very approximate whereas **TransformerAnalysis** gives more accurate results as well many useful data. **TransformerMagneticLosses** is a subset of **TransformerAnalysis** limited to the calculation of the magnetic losses (hysteresis and eddy current) for specific materials.

The example concerns a 50KVA step-down transformer (15000/220 Volts) with an iron core made of Silicon Steel (mild Si content) used as an insulated sheet 0.5 mm thick (this is a standard industrial material for transformer construction).

■ 13-6-2 Summary of the Analysis

The results are the following for a nominal load of 200 Amperes (active current).

▪ Results from inductance coupling (crude approximation)

Cases	air core	partial armature	full iron core
Coupling efficiency %	81.6	96.0	97.1
Leakage coefficient %	33.4	7.9	5.8
Power efficiency %	66.6	92.1	94.2
Transformation coefficient	87.32	66.83	68.31

▪ Results from transformer analysis (first calculation)

	partial armature	full iron core
Voltage output effective (Volt)	224.4	219.6
Z impedance primary (Ohms at 50 Hertz)	1480	10917
Z impedance secondary (Ohms at 50 Hertz)	0.331	2.385
B inductance (cell mean value) (Gauss)	3912	13348
Primary phase angle	71.6	24.7
Power factor %	31.5	90.8
Magnetic Losses (Watt)	13	339
Joule heat losses (Watt)	5677	543
Transformer power input (Kilo Watt)	50.5	48.3
Power efficiency %	88.8	98.0

▪ Results from transformer analysis (after 2 iterations)

	partial armature	full iron core
Voltage output effective (Volt)	223.5	219.3
Z impedance primary (Ohms at 50 Hertz)	1480	10167
Z impedance secondary (Ohms at 50 Hertz)	0.331	2.209
B inductance (cell mean value) (Gauss)	14814	13596
Primary phase angle	71.6	26.3
Power factor %	31.5	89.6
Magnetic Losses (Watt)	113	353
Joule heat losses (Watt)	5645	557
Transformer power input (Kilo Watt)	50.4	49.9
Power efficiency %	88.6	98.0

This shows very clearly the very beneficial effect of a full iron core versus partial iron core or air core. The transformer power efficiencies are respectively 88.6% and 98.0% (values from **TransformerAnalysis**) and 66.6%, 92.1% and 94.2% (values from **InductanceCoupling**). The values calculated after 2 iterations using **Transformer**

Analysis differ only slightly from those of the initial calculation. The very large differences seen on the joule effect power losses between the partial core and the full core transformer are due to the reactive current intensities. These intensities are particularly important for the partial iron core as a consequence of a reduced ferro self-inductance in comparison with the full iron core transformer.

The following figures show the variation of transformer performance versus the load (active current)

Figure 1: losses in Watt
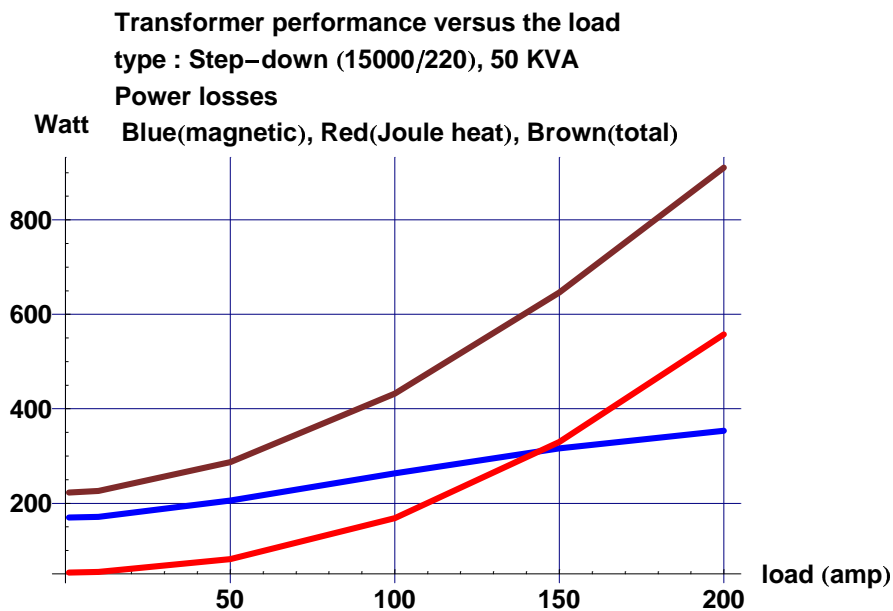
Figure 2: transformer power efficiency %
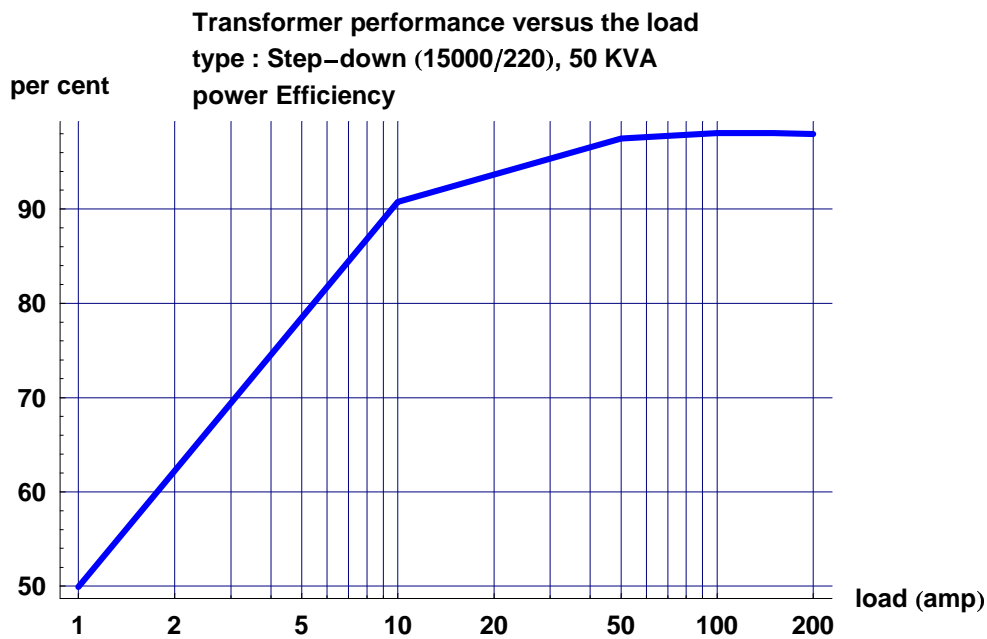
Figure 3: power factor (cos φ)

Transformer performance versus the load
type : Step–down (15000/220), 50 KVA
Power factor
Blue(primary), Red(secondary)

per cent

Figure 4: phase angle (φ)

Transformer performance versus the load
type : Step–down (15000/220), 50 KVA
Phase angle
Blue(primary), Red(secondary)
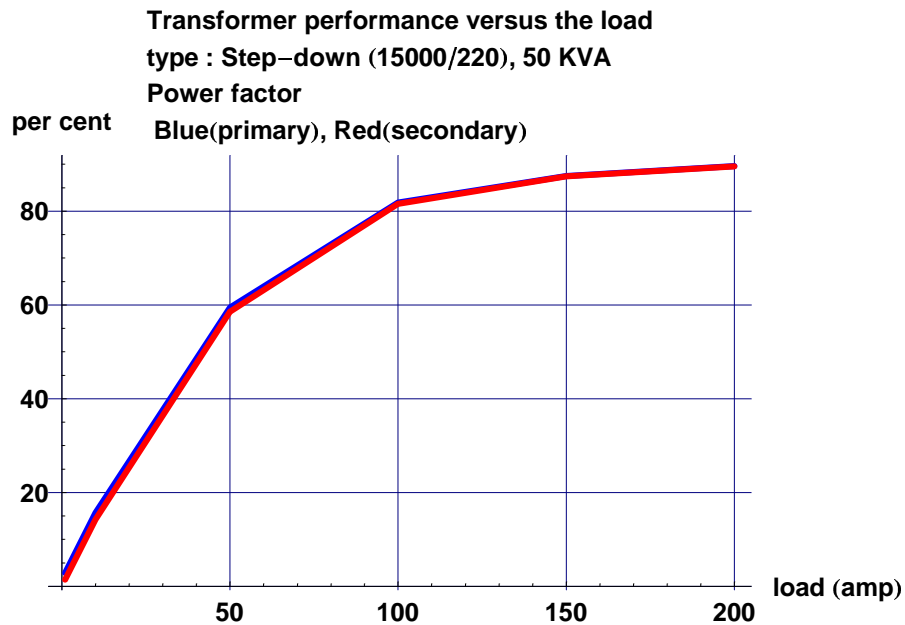
degree
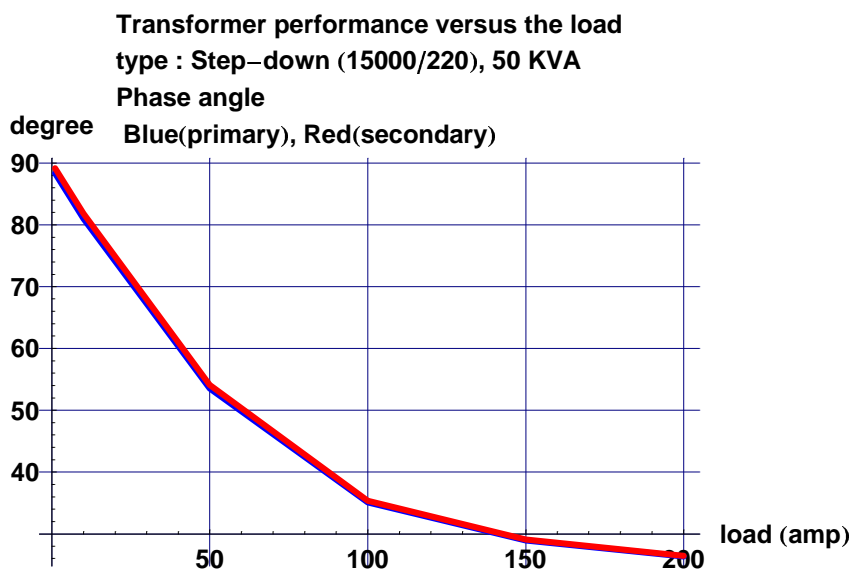
■ 13-6-3 Entering and Drawing Data

Case 2: system data of the transformer with limited ferromagnetic core. The information for Case 1 (without any iron core) is obtained by using the option ContributionType->Proper.

```
transfo1Partial=AXSMagneticSystem[
{AXSSolenoid[CopperOFHC,215,265,348,0,3380,3,FillingFactor->0.6],
AXSSolenoid[CopperOFHC,162,212,348,0,50,200,FillingFactor->0.6],
AXSSoftMagneticComponent[SiSteel2,0,150,350,0]},
SystemLabel->"monophase transformer (partial iron core)",
MeshingLevel->2]

- AXSMagneticSystem :{-solenoid-, -solenoid-, -soft-} -
```

DrawSystem2DXZ[transfo1Partial];

System name: **transfo1Partial**

monophase transformer (*partial iron core*)

2D view XZ
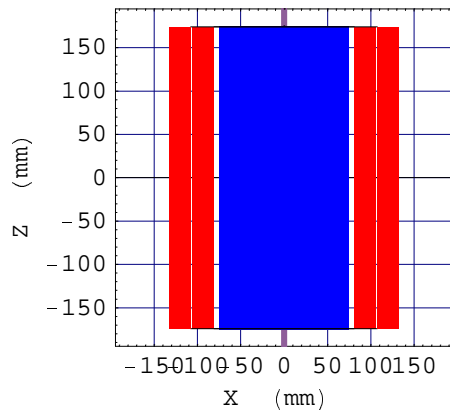
Case 3: system data of the transformer with full ferromagnetic core

```
transfo1Full=AXSMagneticSystem[
{AXSSolenoid[CopperOFHC,215,265,348,0,3380,3,FillingFactor->0.6],
AXSSolenoid[CopperOFHC,162,212,348,0,50,200,FillingFactor->0.6],
AXSSoftMagneticComponent[SiSteel2,0,150,350,0],
AXSSoftMagneticComponent[SiSteel2,0,267,50,-201],
AXSSoftMagneticComponent[SiSteel2,0,267,50,201],
AXSSoftMagneticComponent[SiSteel2,267,327,452,0]},
SystemLabel->"monophase transformer (full iron core)",
MeshingLevel->2]
```

- AXSMagneticSystem :{-solenoid-, -solenoid-, -soft-, -soft-, -soft-, -soft-} -

DrawSystem2DXZ[transfo1Full];

System name: **transfo1Full**

monophase transformer (*full iron core*)

2D view XZ
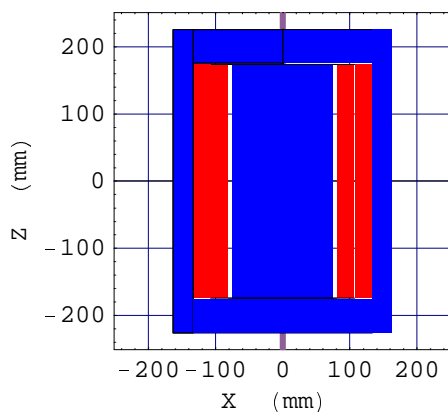
■ 13-6-4 Inductance Matrix

Inductance matrix system data of the transformer with air core

```
InductanceMatrix[transfo1Partial,
    InductanceUnit → Milli Henry, ContributionType → Proper][[1]] // MatrixForm
```

$$\begin{pmatrix} 1305.73 & 12.5109 \\ 12.5109 & 0.17992 \end{pmatrix}$$

Ferro inductance matrix system data of the transformer with partial iron core

```
InductanceMatrix[transfo1Partial,
    InductanceUnit → Milli Henry, ContributionType → Ferro][[1]] // MatrixForm
```

$$\begin{pmatrix} 3405.3 & 54.612 \\ 54.6426 & 0.874463 \end{pmatrix}$$

Total inductance matrix system data of the transformer with partial iron core. We can observe that, even for a partial iron core, the contribution of the ferro inductance is dominant in relation to the proper coil inductance.

```
InductanceMatrix[transfo1Partial,
    InductanceUnit → Milli Henry, ContributionType → Total][[1]] // MatrixForm
```

$$\begin{pmatrix} 4711.03 & 67.1229 \\ 67.1535 & 1.05438 \end{pmatrix}$$

Ferro inductance matrix system data of the transformer with full iron core

```
InductanceMatrix[transfo1Full,
    InductanceUnit → Milli Henry, ContributionType → Ferro][[1]] // MatrixForm
```

$$\begin{pmatrix} 33444.5 & 80.3325 \\ 79.8852 & 7.41124 \end{pmatrix}$$

Total inductance matrix system data of the transformer with full iron core. The contribution of ferro inductance becomes overwhelming.

```
InductanceMatrix[transfo1Full,
    InductanceUnit → Milli Henry, ContributionType → Total][[1]] // MatrixForm
```

$$\begin{pmatrix} 34750.2 & 92.8434 \\ 92.3961 & 7.59116 \end{pmatrix}$$

■ 13-6-5 Inductance Coupling and Transformer Analysis

Inductance coupling of the transformer with partial iron core

```
? InductanceCoupling
```

```
InductanceCoupling[system, {n1, n2}, options] returns the
  inductance coupling for the components n1 and n2 in the system. The results
  are {airCoupling, ferroCoupling}. If the system does not contains any
  ferromagnetic material, ferroCoupling is equal to Irrelevant. Details…
```

```
InductanceCoupling[transfo1Partial, {1, 2}]
```

```
{{0.816246, 0.95973}, {0.666258, 0.921082},
 {0.333742, 0.0789184}, {0.0114523, 0.014956}}
```

? VerboseInductanceCoupling

VerboseInductanceCoupling[system, {n1, n2}, options] display the
  inductance coupling for the components n1 and n2 in the system. Details…

VerboseInductanceCoupling[transfo1Partial,
 {1, 2}, InductanceUnit → Milli Henry, FluxUnit → Milli Weber]

System name: **transfo1Partial**
Inductance Coupling
Precision Level :   1
Number of components   3
Number of current carrying component   2
Number of ferro-magnetic (non rigid) components   1
Number of components with reversible magnetization   1

	primary coil	secondary coil
Component number:	1.0000	2.0000
Component category:	solenoid	solenoid
Circuits current (A):	3.0000	200.0000
Number of turns:	3380.0000	50.0000
Transfo turn ratio	1 / 67.600	

air core data

	primary coil	secondary coil
Self inductance (mH)	1305.7342	0.1799
Mutual induc. 1->2 (mH)	12.5109	
Self flux (mWb)	3917.2025	35.9840
Mutual flux 1->2 (mWb)	37.5327	

ferro core data

	primary coil	secondary coil
Self inductance (mH)	3405.2952	0.8745
Mutual induc. 1->2 (mH)	54.6120	
Self flux (mWb)	10215.8857	174.8925
Mutual flux 1->2 (mWb)	163.8360	

	air core	with ferro core
Coupling coefficient	0.816246	0.959730
Efficiency coefficient	0.666258	0.921082
Leakage coefficient	0.333742	0.078918
Transformation coeff.	1 / 87.3187	1 / 66.8626

Transformer analysis in the case of a partial iron core

? TransformerAnalysis

TransformerAnalysis[system, {n1, voltage1}, {n2, voltage2}, activeLoad, options] returns as
  a list the results of VerboseTransformerAnalysis (see VerboseTransformerAnalysis).
  Use the option InputFrequency to override the default frequency. Details…

Options[TransformerAnalysis]

{PrecisionLevel → 1, InputFrequency → 50, FullDetails → False}

VerboseTransformerAnalysis[transfo1Partial, {1, 15000}, {2, 220}, 200]

```
System name: transfo1Partial
Analysis of a monophase Transformer
     Transformer performance
Network frequency (Hz)                    50
Primary voltage (input) (V)               15000
Effective secondary voltage (V)           224.341
(Active) secondary load (A)               200
Total primary power (input) (W)           50545.3
Phase angle, φ (degree)                    71.6091
Power factor cos φ                         0.315498
B induction (effective average) (G)        2938.31
Magnetic losses power (W)                  12.9717
Joules losses total (W)                    5664.21
Total transformer losses (W)               5677.18
Secondary load power (output) (W)          44868.1
Relative transformer losses %              11.2319
Transformer power efficiency %             88.7681
```

Inductance coupling of the transformer with full iron core

InductanceCoupling[transfo1Full, {1, 2}]

```
{{0.816246, 0.970558}, {0.666258, 0.941982},
 {0.333742, 0.058018}, {0.0114523, 0.01464}}
```

VerboseInductanceCoupling[transfo1Full, {1, 2}]

```
System name: transfo1Full
Inductance Coupling
Precision Level :                                   1
Number of components                                6
Number of current carrying component                2
Number of ferro-magnetic (non rigid) components     4
Number of components with reversible magnetization  4
                    primary coil           secondary coil
Component number:         1.0000                  2.0000
Component category:    solenoid              solenoid
Circuits current (A):     3.0000              200.0000
Number of turns:       3380.0000               50.0000
Transfo turn ratio              1 / 67.600
air core data
Self inductance (μH)      1.3057e6              179.9200
Mutual induc. 1->2 (μH)           12510.8907
Self flux (μWb)           3.9172e6            35983.9932
Mutual flux 1->2 (μWb)            37532.6721
ferro core data
Self inductance (μH)      3.3444e7  7411.2400
Mutual induc. 1->2 (μH)           80332.5500
Self flux (μWb)           1.0033e8             1.4822e6
Mutual flux 1->2 (μWb)           240997.6500
                        air core          with ferro core
Coupling coefficient        0.816246            0.970558
Efficiency coefficient      0.666258            0.941982
Leakage coefficient         0.333742            0.058018
Transformation coeff.    1 / 87.3187        1 / 68.3061
```

Transformer analysis in the case of a full iron core

1. Under the 200 Ampere nominal load

VerboseTransformerAnalysis[transfo1Full, {1, 15000}, {2, 220}, 200]

System name: **transfo1Full**
Analysis of a monophase Transformer
    Transformer performance
Network frequency (Hz)   50
Primary voltage (input) (V)   15000
Effective secondary voltage (V)   219.6
(Active) secondary load (A)   200
Total primary power (input) (W)   44801.5
Phase angle, φ (degree)   24.7037
Power factor cos φ 0.908481
B induction (effective average) (G)   12944.1
Magnetic losses power (W)   338.95
Joules losses total (W)   542.645
Total transformer losses (W)   881.594
Secondary load power (output) (W)   43919.9
Relative transformer losses %   1.96778
Transformer power efficiency %   98.0322

TransformerMagneticLosses[transfo1Full]

{338.95, 13348.}

2. Analysis for load reduced to 10 Amperes

VerboseTransformerAnalysis[transfo1Full, {1, 15000}, {2, 220}, 10]

System name: **transfo1Full**
Analysis of a monophase Transformer
    Transformer performance
Network frequency (Hz)   50
Primary voltage (input) (V)   15000
Effective secondary voltage (V)   219.6
(Active) secondary load (A)   10
Total primary power (input) (W)   2631.14
Phase angle, φ (degree)   82.7247
Power factor cos φ 0.126636
B induction (effective average) (G)   12944.1
Magnetic losses power (W)   338.95
Joules losses total (W)   96.1959
Total transformer losses (W)   435.145
Secondary load power (output) (W)   2196.
Relative transformer losses %   16.5383
Transformer power efficiency %   83.4617

3. Analysis for a 200 Ampere nominal load but with a 60 Hz network frequency

```
VerboseTransformerAnalysis[transfo1Full,
 {1, 15000}, {2, 220}, 200, InputFrequency → 60]

System name: transfo1Full
Analysis of a monophase Transformer
      Transformer performance
Network frequency (Hz)                   60
Primary voltage (input) (V)              15000
Effective secondary voltage (V)          219.6
 (Active) secondary load (A)             200
Total primary power (input) (W)          44842.2
Phase angle, φ (degree)                  20.9572
Power factor cos φ                       0.933848
B induction (effective average) (G)      12944.1
Magnetic losses power (W)                408.163
Joules losses total (W)                  514.128
Total transformer losses (W)             922.291
Secondary load power (output) (W)        43919.9
Relative transformer losses %            2.05675
Transformer power efficiency %           97.9433
```

A more complete analysis can be obtained with the option FullDetails. For the full iron core transformer, 50 Hz at the 200 Ampere nominal load, we have:

```
VerboseTransformerAnalysis[transfo1Full,
 {1, 15000}, {2, 220}, 200, FullDetails → True]
```

System name: **transfo1Full**
Analysis of a monophase Transformer
      System general data

Precision Level:	1
Number of components	6
Number of current carrying component	2
Number of ferro-magnetic (non rigid) components	4
Number of components with reversible magnetization	4

      Transformer performance

Network frequency (Hz)	50
Primary voltage (input) (V)	15000
Effective secondary voltage (V)	219.6
(Active) secondary load (A)	200
Total primary power (input) (W)	44801.5
Phase angle, φ (degree)	24.7037
Power factor cos φ	0.908481
B induction (effective average) (G)	12944.1
Magnetic losses power (W)	338.95
Joules losses total (W)	542.645
Total transformer losses (W)	881.594
Secondary load power (output) (W)	43919.9
Relative transformer losses %	1.96778
Transformer power efficiency %	98.0322

      Transformer Technical Data

	Primary coil		Secondary coil
Component Number	1.000		2.000
Component category	solenoid		solenoid
Number of turns	3380.000		50.000
Component mass (Kg)	35.107		27.354
Soft ferromagnetic mass (Kg)		186.937	
Resistance (Ω)	28.449		0.005
Impedance (Ω)	10917.105		2.385
Nominal transfo ratio		1 / 67.600	
Effective transfo ratio		1 / 68.306	
Nominal voltage (V)	15000.000		220.000
Nominal current (A)	3.000		200.000
Effective voltage (V)	15000.000		219.600
B induction from flux (G)		12944.057	
B induction cell average (G)		13348.002	
Active current (A)	2.987		200.000
Reactive current (A)	1.374		92.082
Effective current (A)	3.288		220.180
Phase angle (degree)	24.704		24.722
Power factor	0.908		0.908
Active power (W)	44801.512		43919.918
Reactive power (W)	20609.860		20221.112
KVA power (W)	49314.722		48351.345
Joule loss power (W)	307.491		235.154
Total joule loss (W)		542.645	
Magnetic loss power (W)		338.950	
Total loss (W)		881.594	

Many fundamental questions arise from these data. Indeed, what are the meanings and applications for the various current intensities appearing in the calculations starting from the initial values given in the definition of the magnetic system up to the final values produced by TransformerAnalysis?

The initial values must be thought of as approximate data necessary for calculating the Induction B in the magnetic circuit of the transformer whereas the final values are specific to the transformer behavior under the proper active load. Naturally, the initial values should ideally correspond to the final values. A better approximation can then be obtained using these first values as input for a second calculation. It appears that this iterative process converges rapidaly: 2 or 3 iterations are sufficient to obtain highly precise results. Moreover, if we limit the outcome to the performance data, the first approximation already provides satisfactory data, much better than the conventional method as given in most professional books.

For instance, the results of the second and the third approximations for the above transformer are respectively:

```
transfo2Full=AXSMagneticSystem[
{AXSSolenoid[CopperOFHC,215,265,348,0,3380,3.28,FillingFactor->0.6],
AXSSolenoid[CopperOFHC,162,212,348,0,50,220,FillingFactor->0.6],
AXSSoftMagneticComponent[SiSteel2,0,150,350,0],
AXSSoftMagneticComponent[SiSteel2,0,267,50,-201],
AXSSoftMagneticComponent[SiSteel2,0,267,50,201],
AXSSoftMagneticComponent[SiSteel2,267,327,452,0]},
SystemLabel->"monophase transformer (full iron core)",
MeshingLevel->2]
```

 – AXSMagneticSystem :{–solenoid–, –solenoid–, –soft–, –soft–, –soft–, –soft–} –

VerboseTransformerAnalysis[transfo2Full,
 {1, 15000}, {2, 220}, 200, FullDetails→True]

System name: **transfo2Full**
Analysis of a monophase Transformer
     System general data

Precision Level:	1
Number of components	6
Number of current carrying component	2
Number of ferro-magnetic (non rigid) components	4
Number of components with reversible magnetization	4

     Transformer performance

Network frequency (Hz)	50
Primary voltage (input) (V)	15000
Effective secondary voltage (V)	219.315
(Active) secondary load (A)	200
Total primary power (input) (W)	44769.6
Phase angle, φ (degree)	26.09
Power factor cos φ	0.898105
B induction (effective average) (G)	13151.
Magnetic losses power (W)	351.39
Joules losses total (W)	555.141
Total transformer losses (W)	906.531
Secondary load power (output) (W)	43863.1
Relative transformer losses %	2.02488
Transformer power efficiency %	97.9751

     Transformer Technical Data

	Primary coil	Secondary coil
Component Number	1.000	2.000
Component category	solenoid	solenoid
Number of turns	3380.000	50.000
Component mass (Kg)	35.107	27.354
Soft ferromagnetic mass (Kg)	186.937	
Resistance (Ω)	28.449	0.005
Impedance (Ω)	10263.344	2.230
Nominal transfo ratio	1 / 67.600	
Effective transfo ratio	1 / 68.395	
Nominal voltage (V)	15000.000	220.000
Nominal current (A)	3.280	220.000
Effective voltage (V)	15000.000	219.315
B induction from flux (G)	13151.048	
B induction cell average(G)	13566.665	
Active current (A)	2.985	200.000
Reactive current (A)	1.462	98.358
Effective current (A)	3.323	222.877
Phase angle (degree)	26.090	26.188
Power factor	0.898	0.897
Active power (W)	44769.610	43863.079
Reactive power (W)	21922.679	21571.503
KVA power (W)	49848.991	48880.461
Joule loss power (W)	314.189	240.952
Total joule loss (W)	555.141	
Magnetic loss power (W)	351.390	
Total loss (W)	906.531	

```
transfo3Full=AXSMagneticSystem[
{AXSSolenoid[CopperOFHC,215,265,348,0,3380,3.323,FillingFactor->0.6],
AXSSolenoid[CopperOFHC,162,212,348,0,50,222.9,FillingFactor->0.6],
AXSSoftMagneticComponent[SiSteel2,0,150,350,0],
AXSSoftMagneticComponent[SiSteel2,0,267,50,-201],
AXSSoftMagneticComponent[SiSteel2,0,267,50,201],
AXSSoftMagneticComponent[SiSteel2,267,327,452,0]},
SystemLabel->"monophase transformer (full iron core)",
MeshingLevel->2]
```

– AXSMagneticSystem :{–solenoid–, –solenoid–, –soft–, –soft–, –soft–, –soft–} –

```
VerboseTransformerAnalysis[transfo3Full,
 {1, 15000}, {2, 220}, 200, FullDetails → True]

System name: transfo3Full
Analysis of a monophase Transformer
       System general data
Precision Level:                                      1
Number of components                                  6
Number of current carrying component                  2
Number of ferro-magnetic (non rigid) components       4
Number of components with reversible magnetization    4


       Transformer performance
Network frequency (Hz)                50
Primary voltage (input) (V)           15000
Effective secondary voltage (V)       219.272
(Active) secondary load (A)           200
Total primary power (input) (W)       44764.5
Phase angle, φ (degree)               26.3
Power factor cos φ                    0.896487
B induction (effective average) (G)   13179.1
Magnetic losses power (W)             353.089
Joules losses total (W)               557.075
Total transformer losses (W)          910.164
Secondary load power (output) (W)     43854.4
Relative transformer losses %         2.03323
Transformer power efficiency %        97.9668


       Transformer Technical Data
                          Primary coil          Secondary coil
Component Number                1.000                2.000
Component category        solenoid              solenoid
Number of turns             3380.000               50.000
Component mass (Kg)           35.107               27.354
Soft ferromagnetic mass (Kg)            186.937
Resistance (Ω)                28.449                0.005
Impedance (Ω)              10169.967                2.209
Nominal transfo ratio                1 / 67.600
Effective transfo ratio              1 / 68.408
Nominal voltage (V)        15000.000              220.000
Nominal current (A)            3.323              222.900
Effective voltage (V)      15000.000              219.272
B induction from flux (G)            13179.135
B induction cell average(G)          13596.327
Active current (A)             2.984              200.000
Reactive current (A)           1.475               99.266
Effective current (A)          3.329              223.280
Phase angle (degree)          26.300               26.397
Power factor                   0.896                0.896
Active power (W)           44764.540            43854.375
Reactive power (W)         22123.966            21766.257
KVA power (W)              49933.294            48958.923
Joule loss power (W)         315.253              241.822
Total joule loss (W)                   557.075
Magnetic loss power (W)                353.089
Total loss (W)                         910.164
```

The convergence of the process is demonstrated by the successive values of

   1. the average induction B in the iron core: 12944, 13151 and 13179 Gauss

   2. the transformer power efficiency: 98.03%, 97.98% and 97.97 %

Appendices

■ Appendix A1: *Magnetica* Built-in Cases

■ A1-1 Introduction

Magnetica provides a large set of ready to use examples called predefined *Magnetica* Cases or simply *Magnetica* Cases. We will describe next what exactly is meant by a *Magnetica* Case, how to use a *Magnetica* Case and how to search and find, among the "*Magnetica* Cases" Data Base, the ones that correspond to certain criteria.

The two last sections will present more precise information about the themes associated with the Keywords.

■ A1-2 How to Use a *Magnetica* Case ?

A *Magnetica* Case is a *Magnetica* expression corresponding to a predefined magnetic system stored in the "Magnetica-Cases" Data Base.

It is referred to by a number.

It is associated with thematic Keywords and textual descriptions that facilitate researching desired relevant Cases.

The materials that it involves can also be a basis for selecting Cases. (See next sections)

You can get the evaluated expression corresponding to a *Magnetica* Case in the following way:
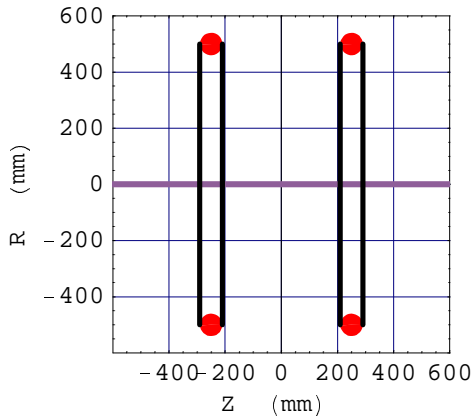
```
s = MagneticaCase[13]
```

```
- AXSMagneticSystem :{-loop-, -loop-} -
```

Now, the expression has been stored in the variable named "s", and is ready to be used for calculations, drawings, and maps as in the following inputs:

```
FieldPoint[s, {10, 100}]
```

```
{3.47406, 9982.32}
```

```
DrawSystem2D[s]
```

```
System name: s
superconducting 1 tesla tubular
loops Helmholtz
2D view ZR
```


- Graphics -

But you can also get the input form of the expression of a MagneticaCase by using the DefinitionCase function:

```
DefinitionCase[13]
```

```
AXSMagneticSystem[{AXSCircularLoop[SuperNbTi5, 1000, 80, -250, 700, 794.6322],
  AXSCircularLoop[SuperNbTi5, 1000, 80, 250, 700, 794.6322]},
 SystemLabel -> "superconducting 1 tesla tubular loops Helmholtz"]
```

and then modify some data, add or subtract some components, attribute a variable name and validate.

You can obtain a full description of a MagneticaCase with the DescribeCase function:

```
DescribeCase[13]
```

```
case number: 13
themes: {ElectricalMagnet}
superconducting 1 tesla, tubular loops Helmholtz
```

quasi second order simulation of a MRI magnet look
for the Legendre coefficients (second order) and the
field Map

If a description of many Cases is desired, the DescribeCases (with an "s") function is provided. It takes as arguments a list composed of numbers and intervals (list of two numbers) as in the following example:

```
DescribeCases[{1, 3, {5, 8}, 12}];
```

gives the description of Cases {1,3,5,6,7,8,12}. To see all the Cases, just type:

```
DescribeCases[All];
```

■ A1-3 The "Magnetica Cases" Data Base

Magnetica provides a large set of predefined *Magnetica* Cases, also called the "Magnetica Cases" Data Base.
The main tool, namely the SearchCases function, for searching (and finding) the Cases corresponding to given criteria is presented in the following section.

In the version of *Magnetica* you are using, the number of available *Magnetica* Cases is stored in the variable named:

```
$NumberMagneticaCases
```

```
131
```

These Magnetica Cases, can be used for different purposes such as:

- a demonstration tool to show, through concrete cases, the methodology of specifying data for any type of magnetic system as well as how to use *Magnetica*.

- a learning and training tool for the science of Magnetostatics. Through a series of well-selected cases, the various physics concepts are exposed, starting from the elementary circular loop up to the amperian formulation of magnetic materials. A special entry deals with the fundamentals of inductance calculations.

- an analysis and design tool for devices and apparatuses that are directly dependent on, or related to, magnetic field calculations.

■ A1-4 Searching Cases in the Data Base

■ A1-4-1 Introduction

In order to efficiently use the *Magnetica* Cases, Magnetica provides a searching tool, the SearchCases function, to navigate throughout the "Magnetica Cases" Data Base. The use of specific Keywords or any other words allows selecting and finding the Cases corresponding to a theme or a word.

■ A1-4-2 Thematic and Material Keywords

To identify some themes of application and material categories, some short words have been reserved. They are called Keywords, and the list of all of them, together with the corresponding meaning, is available by calling:

```
CasesKeywords[]

Thematic keywords
BasicHardMagnetic          Learning and Training supported through hard magnetic
BasicInductanceAir         Learning and Training related to inductance calculati
BasicInductanceIron        Learning and Training related to inductance calculati
BasicLoop                  Learning and Training supported through circular loop
BasicSoftMagnetic          Learning and Training supported through soft magnetic
BasicSolenoid              Learning and Training supported through circular sole
ActiveShielding            Analysis and Design of active shielding
ElectricalMagnet           Analysis and Design of electrical magnet
EMCAutoShielding           Analysis and Design of EMC auto shielding
FerromagneticShielding     Analysis and Design of ferromagnetic shielding
InductanceCoil             Analysis and Design of inductance coil
MagneticEnergy             Analysis and Design of magnetic energy in magnet
MRIMagnet                  Analysis and Design of MRI magnet
PermanentMagnet            Analysis and Design of permanent magnet
SensorCoil                 Analysis and Design of sensor coil
Transformer                Analysis and Design of transformer
Material related keywords
Resistor                   electrical resistive material
Superconductor             superconducting material
Diamagnetic                diamagnetic material
Paramagnetic               paramagnetic material
HardFerro                  hard Ferromagnetics material
SoftFerro                  soft Ferromagnetics material
Vacuum                     space vacuum propoperties (suppose the existence of L
```

As shown in the output, there are two kinds of Keywords, Thematic Keywords and Material Keywords.

Each *Magnetica* Case is internally associated to one or more Thematic Keywords, and the materials it involves also connect it to the corresponding Material Keywords.

For Thematic Keywords, some are related to basic use, learning and training, and they begin with "Basic". The others refer to applications, analysis and design. Each category of application will be more extensively developed in the two last sections.

Material Keywords refer to the electrical and magnetic properties of the components involved in the magnetic system. A special meaning is given to "Vacuum". This is not a concrete material but the properties of space in the absence of any materials. The magnetic field is transmitted perfectly well in the vacuum without any interference. The vacuum is magnetically neutral, neither diamagnetic nor paramagnetic (susceptibility = 0, relative permeability = -1). It is implicitly used in *Magnetica* calculations as the medium in between the concrete components of any magnetic system. This can be more explicitly visualized for the Large Source component which generates, by definition, a constant magnetic field at any target point in space.

It must be pointed out that, whatever the selection, all the *Magnetica* Cases can be used with all the functions of *Magnetica* for field calculation, mapping and graphic display. The thematic approach is just a tool that facilitates the selection of relevant and significant examples.

■ A1-4-3 Searching with Keywords

To obtain all the *Magnetica* Cases corresponding to a predefined Keyword (Theme or Material), just use this Keyword between double quotes as argument for the function SearchCases, as in the following example:

```
SearchCases["BasicLoop"]

search magnetica cases for: basicloop
10 exact matches 0 other matches
Page 1/1 exact match page:1/1
exact match cases:
case number: 1 Score (100.)
Basic current carrying circular Loop
case number: 2 Score (100.)
Basic single Loop, Legendre radius : 100 mm
case number: 3 Score (100.)
Basic single Loop, Legendre point off center : 100 mm
case number: 4 Score (100.)
Basic single Loop, asymmetric position
case number: 5 Score (100.)
Basic single Loop, temperature 27 ° Celsius
case number: 9 Score (100.)
Helmholtz Loop pair, Legendre radius : 150 mm
case number: 10 Score (100.)
Helmholtz Loop pair, Temperature 27 ° Celsius
case number: 18 Score (100.)
single filament loop in the Earth magnetic Field
case number: 8 Score (94.)
Helmholtz Loop pair (Basic data)
case number: 103 Score (94.)
Helmholtz Loops pair in the Earth magnetic Field
```

All the relevant Cases are displayed with their associated number, and a short description. The associated number is the way to handle the Magnetica Case for all other functions MagneticaCase, DefinitionCase, DescribeCase.

When the number of matching Cases is too big, only the first page is displayed. Then to visualize the other pages, use the PageNumber option

```
SearchCases["BasicSolenoid", PageNumber → 2]

search magnetica cases for: basicsolenoid
11 exact matches 0 other matches
Page 2/2 exact match page:2/2
exact match cases:
case number: 29 Score (93.)
MRI 1.0 T superconducting magnet unshielded
```

▪ A1-4-4 Free Searching

The SearchCases function also allows the use of any word (or part of a word) other than the predefined Keywords. Indeed, each Magnetica Case is also associated to a textual description, displayed by the DescribeCase function. Using words other than the Keywords launches the searching process on all the textual descriptions, to select the pertinent Cases. Here is an example:

```
SearchCases["spectro"]

search magnetica cases for: spectro
10 exact matches 0 other matches
Page 1/1 exact match page:1/1
exact match cases:
case number: 70 Score (90.)
NMR Spectroscopy sensor coil 6 turns Copper
case number: 71 Score (90.)
NMR Spectroscopy sensor coil 6 turns Gold
case number: 72 Score (90.)
NMR Spectroscopy sensor coil 6 turns Aluminum
case number: 73 Score (90.)
NMR Spectroscopy sensor coil 6 turns Composit Cu/Al
case number: 74 Score (90.)
NMR Spectroscopy  sensor coil 6 turns Copper (inductance)
case number: 122 Score (90.)
NMR Spectroscopy sensor coil 4 turns Copper
case number: 123 Score (90.)
NMR Spectroscopy sensor coil 4 turns Aluminum
case number: 124 Score (90.)
NMR Spectroscopy sensor coil 4 turns Composit Cu/Al
case number: 125 Score (90.)
NMR Spectroscopy sensor coil 4 turns Gold
case number: 126 Score (90.)
NMR Spectroscopy sensor coil 4 turns Copper (inductance)
```

■ A1-4-5 More Complex Searching

Finally, more complex searching can also be tried, by giving as an argument more than one word separated by blank spaces, with the entire sentence being placed between double quotes, as in the following:

```
SearchCases["MRImagnet super softferro"]

search magnetica cases for: mrimagnet softferro super
5 exact matches 68 other matches
Page 1/8 exact match page:1/1
exact match cases:
case number: 31 Score (92.3333)
MRI 1.5 T superconducting magnet, close  shield
case number: 32 Score (92.3333)
MRI 1.5 T superconducting magnet, room shield
case number: 39 Score (92.3333)
MRI open superconducting magnet, iron core, meshing 2
case number: 40 Score (92.3333)
MRI open superconducting magnet, iron core, meshing 3
case number: 38 Score (92.)
MRI open superconducting magnet, iron core, meshing 1
```

As shown in this example, it is possible to combine one or more Keywords with other free words. An internal algorithm sorts the answers by pertinence. It is not always perfect, but it is often helpful.

◾ A1-5 Basic Magnetostatics

◾ A1-5-1 Introduction

This section covers the use of the predefined Cases as a learning and training tool for the fundamentals of Magnetostatics.

◾ A1-5-2 Basic Magnetostatics: Magnetic Fields Generated by Electrical Current - Keywords: BasicLoop and BasicSolenoid.

The aim of the chosen cases is to facilitate the understanding of the primary formulas and relations that govern field calculation. The field generators are based on the electrical current that circulates in their winding and their geometric shape is simple and easy to visualize. This starts with the theoretical circular loop which is a filament of current with an infinitely small cross-section. The relevant field relations are thoroughly developed in Chapter 4 of the User Book on the Vector Potential (VP) method and in Chapter 5 on the Spherical Harmonics (SH) method. Theses 2 methods are applied in *Magnetica*. The VP method is the easiest to understand but the SH method, although more complex, presents many very special properties, particularly swiftness, for computerized calculation. Using the Cases shows, through punctual calculation or graphics maps, how the magnetic field and related functions (vector potential, flux and gradient) are calculated, how they vary and how they are employed.

Then we proceed with the extension of the initial formulation for the filament loop to more complex geometrical shapes such as the circular tubular loop or a solenoid whose winding of sizable dimension is made of wires equally distributed or even for Bitter type solenoids that have a current density that depends on the radius.

The required Keywords to get these Cases are BasicLoop and BasicSolenoid:

```
SearchCases["BasicLoop"];

SearchCases["BasicSolenoid"];
```

◾ A1-5-3 Basic Magnetostatics: Magnetic Field Generated by Magnetic Matter - Keywords: BasicHardMagnetic and BasicSoftMagnetic.

This applies to magnetic field generators based on the physical properties of the matter. The origin is the electrons of the atoms or even in case of ferromagnetics, the nucleus of the atoms. Though there is no visible macroscopic electrical current, the matter can nevertheless become the source of a magnetic field.

How to relate the intrinsic magnetic properties of the matter to magnetic field calculation is a primordial concept of magnetostatics. We use the Amperian approach that relates the magnetic properties directly to the microscopic surface currents. This approach, asopposed to the coulombian approach, has the advantage of being physically correct without restriction and also to allow using basic mathematical formulations similar to those of macroscopic current generators.

The prefixes Hard and Soft refer to the magnetic response of the matter to the excitation of an applied field: Hard is for matter that retains a certain state of magnetization once the excitation has disappeared and, by opposition, Soft is used for matter for which no magnetization remains.

Before doing any field calculations for permanent magnets or any structure made of magnetic materials, it is recommended that the user become familiar with the material properties materialized with the relevant formulas and their magnetization curves as shown in Chapter 6 and in Appendix B.1.

As for the previous section, the predefined Cases are chosen to facilitate understanding, starting from the simplest situation of a bar of a permanent magnet assuming a rigid law of magnetization (i.e. absolutely constant) as developed in

Chapter 7 up to more elaborate situations that concern materials with linear and non linear laws of magnetization, (as shown in Chapter 8 for soft materials) and finally, the complex situations used for some concrete applications.

Through the different predefined Cases, the various concepts and definitions can be progressively apprehended. They concern either the intrinsic properties of materials (such as linear and the non linear law of magnetization, coercivity, hysteresis curves) or the methodology of field calculation (such as structure discretization and cells, functional magnetization, field formulas).

The required Keywords to get these Cases are BasicHardMagnetic and BasicSoftMagnetic:

```
SearchCases["BasicHardMagnetic"];

SearchCases["BasicSoftMagnetic"];
```

■ A1-5-4 Inductance Calculation - Keywords: BasicInductanceAir and BasicInductanceIron.

Inductance defines the response of an electrical circuit to its magnetic environment. Inductance, as was shown in Chapter 10, can be seen as a primordial application of the magnetostatics based on the magnetic coupling between electrical circuits. Many practical devices such as inductance coils and electrical transformers are dependent on the concept of inductance. Other devices such as the MRI superconducting magnets also have behavior also that depends on inductance. The application of the concept of inductance to these devices will be treated in the next section. In the present section, the basic methodology underlying the calculation of inductance in *Magnetica*, is progressively explained.

There are 2 fundamental ideas to approach basic inductance calculations:

- either the magnetic system is made of only current-carrying components (in terms of inductor devices, they are designated as "air core" and the corresponding Keyword is BasicInductanceAir).
- or the magnetic system also has soft ferromagnetic components (devices are designated as "iron core" and the corresponding Keyword is BasicInductanceIron).

The air core situation is the simplest and the starting circuit is the one turn loop for which the inductance has only one element: self-inductance. In the case of 2 (or more) circuits, a new element appears that characterizes the magnetic interaction between the two circuits: mutual inductance. The inductance is then displayed as a matrix. From the concept of inductance, other physical concepts are developed, such as magnetic flux (coupling) and magnetic energy.

Iron core devices are very important for certain applications, particularly for electrical transformers. The *Magnetica* approach to field calculation has allowed developing a very innovative way of calculating the effect of the iron core. An original concept, ferro-inductance, has purposely been introduced. Specific calculation methods (patent pending) have been thoroughly developed.

The Keywords related to basic inductance calculations are BasicInductanceAir and BasicInductanceIron.

```
SearchCases["BasicInductanceAir"];

SearchCases["BasicInductanceIron"];
```

■ A1-5-5 *Magnetica* Materials properties

Magnetica provides a large set of electrical and magnetic materials. They are all visible on the *MagneticaMaterials* palette. However, the Magnetica Cases use only a reduced set of Magnetica materials. The list of these materials is given by

```
CasesMaterials[]
```

Materials	Material categories	Number of cases involved
Alnico1	{HardFerro}	1
AlumElec	{Resistor, Paramagnetic}	3
Composit1	{Resistor, Paramagnetic}	2
CopperFHard	{Resistor, Diamagnetic}	1
CopperOFHC	{Resistor, Diamagnetic}	79
Ferrit1	{HardFerro}	6
Ferrit2	{HardFerro}	8
Gold	{Resistor, Diamagnetic}	2
HighPerm1	{HardFerro}	1
IronInd2	{HardFerro}	3
IronInd3	{HardFerro}	18
MuMetal1	{HardFerro}	2
NIBcomp2	{HardFerro}	4
NIBcomp3	{HardFerro}	2
NIBcomp4	{HardFerro}	4
NIBcomp7	{HardFerro}	1
SiSteel2	{HardFerro}	9
SiSteel3	{HardFerro}	7
SteelCommon1	{HardFerro}	4
SteelCommon2	{HardFerro}	1
SuperNbTi4	{Superconductor, Diamagnetic}	24
SuperNbTi5	{Superconductor, Diamagnetic}	5
SuperNbTi8	{Superconductor, Diamagnetic}	1
Vacuum	{Vacuum}	16

■ A1-6 Applications

■ A1-6-1 Introduction

Many apparatuses and devices used every day are dependent on or related to magnetic phenomenon. We have selected a certain number of applications where *Magnetica* can be used as a tool for the analysis and/or design of relevant apparatuses or devices.

These applications cover the following area:

- magnets: devices designed to generate magnetic fields.

- sensor coils: devices that receive a magnetic field signal.

- shieldings: apparatuses to reduce the deleterious effects of magnetic fields.

The inductance-related devices are treated independently as:

- inductance coils: devices designed to have a given inductance.

- magnetic energy devices: apparatuses where the magnetic energy is an important feature.

- electrical transformers: devices to step up or down the voltage in AC circuits.

■ A1-6-2 How to Do Analysis and Design

Analysis is a relatively straight-forward process. It is simply necessary to set the data that represent the actual magnetic system.

- In case of doubt, it is advisable to use an example of the predefined Cases similar to the actual system and modify, add or subtract old and new components as desired

- Then it is suggested to look for drawings using, for 2D graphics, the proper plane projection and to finalize with 3D graphics.

- In case the actual system contains magnetic components, it is recommended to start validation with the lowest meshing level. Indeed, the calculation time increases very quickly with the number of cells (the dependence is a power between 2 and 3). In many situations, meshing level 2 is sufficient to give the proper answer with an acceptable degree of accuracy. Naturally, meshing level 3 provides better results. Meshing level 4 must be used for exceptional situations, calculation time may become prohibitive and memory requirements are considerable.

- Then analysis can proceed: it is recommended to select and use the proper magnetic field functions and all the features of field mapping, verbose output and numerical results. It is should be remembered that the default field calculation mode is Automatic which means that whenever possible, the Spherical Harmonics mode is used. However, it is always possible to switch to the Vector Potential mode without restriction, either momentarily or for a full session.

- If working with inductance calculation is required, it should be remembered that the default precision parameter is 1 providing a precision of at least 3 decimal digits. Greater precision is possible, but naturally at the expense of longer calculation time. In the case of a magnetic system with an iron core, it is necessary to become familiar with the concept of ferro inductance which is a unusual new concept brought by *Magnetica*.

Design proceeds much more heuristically.

- Before even starting, it is necessary to set clearly the objective and the irreducible constraints such as, for instance, field strength and useful dimensions for magnets or electrical power for transformers.

- From this point, the best way will be to select among the predefined Cases those that are the closest to the objective and then to start improving progressively the guess in order to reach the objective.

- Special situations arise for MRI high homogeneity magnets. The condition of high homogeneity cannot be approached by a simple trial and error process. Magneticasoft is able to propose optimized magnetic systems that comply with given constraints. (If interested please inquire).

■ A1-6-3 Magnet Analysis and Design

This area of application is divided into 3 themes, corresponding to 3 keywords:

- **Electrical** magnets: the generation of magnetic field proceeds from an electrical current fed into a winding.

- **MRI** magnets: this application is treated separately due to the actual importance and the special features of such magnets, such as field homogeneity which is an absolute requirement for obtaining a significant image.

- **Permanent** magnets: devices made with hard ferromagnetic materials.

The Cases proposed by using the keywords **Electrical**, **MRI** and **Permanent** are given by:

```
SearchCases["Electrical"]
```

```
SearchCases["MRI"]
```

```
SearchCases["Permanent"]
```

■ A1-6-4 Special Sensor Coil Analysis and Design

The Cases obtained by using the keyword **SensorCoil** treat some very special examples: special because of the very tiny dimensions (less than 1mm) and also special because of the magnetic materials since common diamagnetic and paramagnetic materials (copper, aluminum, gold) are involved.

```
SearchCases["SensorCoil"]
```

■ A1-6-5 Magnetic Shielding Analysis and Design

This area of applications is divided into 3 themes, corresponding to 3 keywords:

- **Ferromagnetic** shielding: the reduction of the disturbing magnetic field is obtained by adding soft iron type materials that are appropriately distributed around the magnet to achieve the desired results.

- **Active** shielding: the reduction of the distributing magnetic field is obtained by generating a side magnetic field (counter field) that opposes the main magnetic field, the effect of the side field being the greatest in the external region.

- **EMCAuto** shielding: this is a very special theme related to MRI superconducting magnets (the most common MRI magnets). In working conditions, these devices are electrical circuits of absolutely 0 resistance. This implies that the total magnetic flux stays constant notwithstanding any disturbance from the outside world (Lenz's law). Depending on the strength and shape of the disturbing signal, a small variation of current, and therefore of the magnetic field may occurs in the magnet. If this event occurs during the imaging process, it generates a noise which may be harmful to the final image or spectrum. It must be remembered that MRI imaging is a hyper sensitive process (the requirement ratio noise to signal may reach 10^{-9}). Assessing the rejection mode due to EMC Auto shielding allows for designing appropriate counter measures.

The Cases proposed by using the keywords **Ferromagnetic**, **Active** and **EMCAuto** are given by:

```
SearchCases["Ferromagnetic"]
```

```
SearchCases["Active"]
```

```
SearchCases["EMCAuto"]
```

■ A1-6-6 Inductance Coil Analysis and Design

Along with the resistor and the capacitor, the inductance coil, i.e. the inductor, is one of the key components of any electrical or electronic circuit. Inductors are necessary to make resonant circuits that have frequency characteristics. The calculation of inductance coils proceeds present by empirical methods and is very approximate. *Magnetica* offers a completely new method of calculation that is both precise and swift. With the use of the original concept of ferro inductance, the design of an iron core inductor can be treated in the same way. The method may be applied without restriction both for tiny inductors used in microelectronic circuits as well as for very large coils used in nuclear research laboratories. The proposed Cases show some crude examples, but any other Case example can be used for evaluation of the coil's inductance.

Using the keyword **InductanceCoil** gives the relevant Cases for inductance calculation:

```
SearchCases["InductanceCoil"]
```

■ A1-6-7 Magnetic Energy Analysis

Magnetic energy is stored by a magnetic field and the availability of this energy depends on the magnetic system's inductance characteristics.

In some applications such as the MRI magnet, the magnetic energy is a problem that has to be dealt with. Indeed, the magnetic energy stored in these devices while in normal operation is huge (of the order of several MegaJoules). In the case of a technical failure such as the loss of superconductivity, the energy is liberated catastrophically: an explosion may occur.

Consequently, in order to avoid any dangerous outcome, magnet design must take into account this special feature. Since MRI magnets, particularly magnets with active shielding, are made of a large number of coils, the calculation of the complete and detailed inductance is a necessary but difficult task. This can be done very precisely and quasi instantaneously with the Inductance functions of *Magnetica*. Designing circuits of optimal safety is thus greatly facilitated.

In other practical devices used in the mechanical industries, the magnetic energy of coils is the basis of magnetic forming. This working process makes use of the magnetic energy of the coil which is explosively transferred to the sheet set in front of a mold. This energy heats the sheet allowing for plastic deformation under the resulting electromagnetic forces that push the sheet strongly onto the mold. Magnetic inductance calculation methods can be used to optimize these formatting devices and to improve further the magnetic forming process.

Using the keyword **MagneticEnergy** in the SearchCases function shows examples of magnetic energy related to MRI magnets but any other example case can be used to evaluate its magnetic energy.

```
SearchCases["MagneticEnergy"]
```

■ A1-6-8 Transformer Analysis

The transformer is an essential device for the production and distribution of electrical power. Public electrical systems use time variable current, sinusoidal alternative current, or AC, with a constant frequency (50 or 60 Hertz). Transformers are very widely used in public networks to adjust (step up or down) the voltage to the most economical and practical conditions. But they are also very commonly used in electronic systems in a much large range of frequencies for impedance matching between the key elements of the system. These latter transformers are essential in audio systems but also in professional and military electronic systems. Designing transformers is, therefore, a common task of the research and development divisions of the electrical and electronics industries.

Basically, a transformer can be seen as a device with 2 circuits that are magnetically coupled. In order to have very good power efficiency, the magnetic coupling must be excellent. This implies the use of a ferromagnetic armature (i.e. iron core) made of the appropriate materials. In public electrical networks, silicon steels, namely electrical steels, are widely used. In electronic systems, many other ferromagnetic materials are used, including some high permeability materials. All these materials are called, according to our terminology, soft ferromagnetic materials. Their magnetization curves are very non-linear. In order to obtain high power efficiency, it is absolutely necessary to make the best use of the non-linear effect of the iron core.

So far, the calculation of transformers has been very primitive. It was not possible to take into account the non linear effect of the ferromagnetic materials. Moreover, the way of apprehending the magnetic circuit was also unrealistic. *Magnetica* allows, through the discretization of the magnetic circuit and the precise modeling of the magnetization curve, for a completely new approach to transformer calculation. The 2 specific functions **InductanceCoupling** and

TransformerAnalysis open up a new methodology for transformer analysis. This will be a valuable asset for all segments of the electrical and electronics industries.

The cases related to transformer analysis are obtained by using the keyword **Transformer**:

```
SearchCases["Transformer"]
```

■ Appendix B1: Physical and Magnetic Properties of *Magnetica* Materials

■ B1-1 Fundamentals on *Magnetica* Materials

■ B1-1-1 Introduction

In *Magnetica*, the use of specific material is implied for all the components of a magnetic system. This material must be chosen among the proposed list of *Magnetica* material. They are known and used by their calling name. The physical properties of these materials are completely set. That allows for precise calculation of the magnetic field under any temperature conditions. The physical and magnetic data provided come either from handbooks on materials or from published notices or tables of the industrial producers.

The choice of the relevant material is facilitated by using the specific palette "MagneticaMaterials" (just click on the material's name, see Appendix B3 on palettes).

■ B1-1-2 Typology of *Magnetica* Materials

The materials are divided into 6 types:

1. Resistive Current-carrying Materials: electrical conductors with a specific resistivity at the operating temperature. The conductor and the coil dimensions set the coil resistance and consequently the electrical power necessary to drive the coil and generate the magnetic field.

2. Superconductor Current-carrying Materials: a zero resistance electrical conductor, operated at cryogenic temperatures. The current-carrying capacity is a specific property of the wire used to make the coil and not a property of the bulk material. *Magnetica* data are based on the published specifications of a typical set of commercially available industrial products. This explains why, in *Magnetica*, wire diameter is an intrinsic property of superconductor materials/products.

3. Reversible Linear Soft Ferromagnetic Materials: this modeling of the law of magnetization assumes a linear dependence of magnetization with respect to the magnetizing field which is equivalent to a constant permeability or a constant susceptibility. The linear relation is valid only in the low range of the magnetizing field. In the high range, the magnetization becomes constant due to a saturation effect (all the elementary dipoles are aligned). The linear modeling is a first approximation for the calculation of the magnetic behavior of devices and apparatuses using soft ferromagnetic materials. It is currently used in the electrical industry. In addition, this allows carrying out approximate calculations for materials whose precise law of magnetization is not well known.

4. Reversible Non-Linear Soft Ferromagnetic Materials: the actual law of magnetization coming from experimental measurements is used through a proper mathematical modeling of the experimental curve. The permeability presents a maximum at a given magnetizing field: the crest point. The mathematical simulation method (patent pending) is a proprietary method of Magneticasoft. This representation allows for accurate calculation for electrical devices and apparatuses in any application range.

5. Hard Ferromagnetic, Anisotropic, Permanent Magnets: the dynamic susceptibility of the useful materials appears quasi constant when the magnetizing field stays within the limits of the current operating range of applications. The limit, called the coercitive magnetic field, is the minimum (negative) applied magnetizing field. Beyond this limit the material undergoes a catastrophic process of demagnetization. Then the material loses its capacity to generate a magnetic field in the initial direction. Consequently, within the operating limits, a linear law of magnetization is assumed.

6. General Materials: this means any material of interest in the design or the analysis of electromagnetic devices or apparatuses that is not dependent on the above types.

An overview of *Magnetica* materials is given by the palette **MagneticaMaterials**.

■ B1-1-3 Basic Properties

Magnetica materials general data, applicable to all types of materials, are a set of physical and magnetic properties which include:

- the specific weight

- the dilation coefficient: the relative variation of the linear dimension as a function of the temperature.

- the electrical resistivity at the nominal temperature (20° Celsius)

- the electrical resistivity coefficient: the relative first order variation of resistivity as a function of temperature,

- the relative volume magnetic susceptibility at the nominal temperature (20° Celsius). For non linear ferromagnetic materials, this is the average value for the low magnetization range.

- the relative volumic magnetic susceptibility coefficient: the first order variation of the susceptibility as a function of the temperature.

These properties are properties of the volume bulk substances. It should be pointed out that for the magnetic properties, we have to distinguish between intrinsic properties of the substance as opposed to the functional properties that depend on the shape and dimensions of the sample or the apparatus made with the substance.

The general physical and magnetic properties of any *Magnetica* material can be shown by calling the function `Properties` with the proper material name as argument. The material name can be either typed or "clicked" from the Palette. An optional argument allows setting the temperature (if different from the nominal temperature 20° C). For specific types of materials, some complementary data are given.

■ B1-1-4 Curves and Tables

The magnetic properties can also be displayed, either as tables or as graphic plots, as functions of the applied magnetizing field Hm. The 2 functions to use are respectively:

- `MaterialMagnetizationTable`,
- `MaterialMagnetizationCurves`.

For isotropic materials, the magnetic functions that can be displayed or plotted are obtained by using the following symbols:

- `B` (Induction),
- `M` (Magnetization, also called intensity of magnetic polarization),
- `Permeability` (relative permeability),

- Susceptibility (relative susceptibility),

- Reluctivity (relative reluctivity).

The data or the curves can be set in any relevant units: Tesla, Gauss, Ampere/Meter, etc.
Specific applications for every type of material are given in the following sections.

For permanent magnet materials, the magnetic functions that can be displayed or plotted are:

- Mprincipal (magnetization in the principal direction)

- Mtransversal (magnetization in the transversal direction)

- Bprincipal (induction field in the principal direction)

- Btransversal (induction field in the transversal direction)

■ B1-2 Resistive Conductors

■ B1-2-1 Resistive Conductors - Properties

This type of material includes 28 different items:

- 14 single atomic element metals: Aluminum (very pure annealed or standard hard drawn), Copper (very pure annealed or standard hard drawn), Gold, Graphite, Mercury, Niobium, Rhodium, Silver, Tin, Titanium, Tungsten, Zinc.

- 13 alloys: Constantan, Copper Cadmium, Copper Chromium, Hardened Beryllium Copper, Overaged Beryllium Copper, Titanium Beryllium Copper, Zirconium Copper, Yellow Brass, Red Brass, Silver (White) Brass, Duralumin, Magapyr, Bronze.

- 1 composite: Copper Aluminum.

The relevant properties of the resistive conductors are strictly the general properties set.

The following example: CopperOFHC (Oxygen Free High Conductivity) at the default temperature:

```
Properties[CopperOFHC]

Material properties
material name                        CopperOFHC
material category                    Resistive current carrying
designation                          high conductivity OFHC Copper
General Physical Properties
specific weight (g/cm3)                    8.920000
dilation coefficient                       0.000016
    (relative linear change)
resistivity (Ohm x cm)                     1.724000e-6
Coefficient temperature resistivity        0.003930
Magnetic Physical Properties
relative volume magnetic                  -6.100000e-7
    (linear) susceptibility
temperature coefficient susceptibility     0.000010
```

The same material properties at the temperature of 45° Celsius:

```
Properties[CopperOFHC, Temperature → 45]
```

Material properties
```
material name                          CopperOFHC
material category                      Resistive current carrying
designation                            high conductivity OFHC Copper
```
General Physical Properties
```
specific weight (g/cm3)                       8.920000
dilation coefficient                          0.000016
    (relative linear change)
resistivity (Ohm x cm)                        1.724000e-6
Coefficient temperature resistivity           0.003930
```
Magnetic Physical Properties
```
relative volume magnetic                     -6.100000e-7
    (linear) susceptibility
temperature coefficient susceptibility        0.000010
```
Properties at the temperature 45 °Celsius
```
linear dilation (variation %)                 0.040500
electrical resistivity (variation %)          9.825000
relative susceptibility (variation %)         0.025000
effective linear dilation  coefficient        1.000405
effective electrical resistivity (ohm-cm)     1.893383e-6
effective volume magnetic susceptibility     -6.101525e-7
```

■ B1-2-2 Resistive Conductors - Tables and Graphics

These materials present a very small magnetic dependence with respect to the applied magnetizing field Hm. However, in high field magnetic resonance spectroscopy, their magnetic behavior becomes of interest. These materials have a perfect linear magnetization: their susceptibility is very constant.

- Copper (diamagnetic): CopperOFHC (Oxygen Free High Conductivity) in the range 0 to 10 Telsas. Table with a step of 1 Tesla.

```
MaterialMagnetizationTable[CopperOFHC, FieldUnit → Tesla,
 NumberOfPoints → 11, MagnetizingFieldRange → {0, 10}]
```

Magnetization Table

CopperOFHC temperature 20 °Celsius
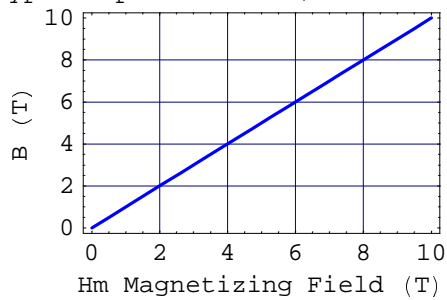
high conductivity OFHC Copper

Range for the magnetizing field: 0. -- 10. (T)

| Hm (T)	M (T)	B (T)	
0.000000	0.000000	0.000000	-6.100000e
1.000000	-6.100000e-7	0.999999	-6.100000e
2.000000	-1.220000e-6	1.999999	-6.100000e
3.000000	-1.830000e-6	2.999998	-6.100000e
4.000000	-2.440000e-6	3.999998	-6.100000e
5.000000	-3.050000e-6	4.999997	-6.100000e
6.000000	-3.660000e-6	5.999996	-6.100000e
7.000000	-4.270000e-6	6.999996	-6.100000e
8.000000	-4.880000e-6	7.999995	-6.100000e
9.000000	-5.490000e-6	8.999995	-6.100000e
10.000000	-6.100000e-6	9.999994	-6.100000e

- Plot for CopperOFHC, magnetization curves: B (induction), M (magnetization), Susceptibility:

```
MaterialMagnetizationCurves[CopperOFHC, {B, M, Susceptibility},
  FieldUnit → Tesla, MagnetizingFieldRange → {0, 10}]
```

```
Material Magnetization Curve
CopperOFHC  temperature  20°celsius
high conductivity OFHC Copper
B Induction f(Hm)
Type of plot: Linear/Linear
```

```
Material Magnetization Curve
CopperOFHC  temperature  20°celsius
high conductivity OFHC Copper
M Magnetization f(Hm)
Type of plot: Linear/Linear
```
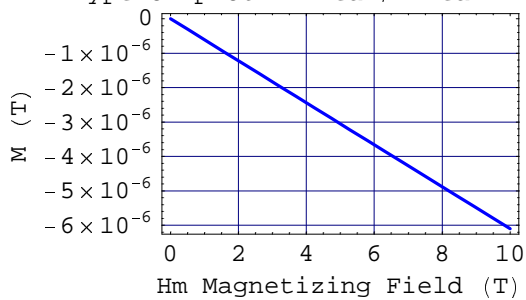
```
Material Magnetization Curve
CopperOFHC  temperature  20°celsius
high conductivity OFHC Copper
χ Susceptibility f(Hm)
Type of plot: Linear/Linear
```
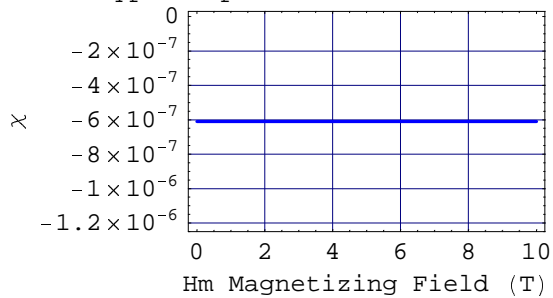
▪ B1-3 Superconductors

▪ B1-3-1 Superconductors - Properties

This type of material includes 10 items:

- ▪ 8 wires made of Niobium Titanium alloys imbedded in a copper matrix.

- 2 wires made of Niobium Tin composites imbedded in a special matrix.

The relevant properties include the general properties and some special specifications such as the wire diameter and the current-carrying capacity in a given magnetic field.

The example: Niobium Titanium wire 1.5 mm in diameter: `SuperNbTi5`.

```
Properties[SuperNbTi5]

Material properties
material name                       SuperNbTi5
material category                   Superconductor current carrying
designation                         Nb-TI superconductor wire (A36)
General Physical Properties
specific weight (g/cm3)                      8.350000
dilation coefficient                         0.000000
    (relative linear change)
resistivity (Ohm x cm)                       0.000000
Coefficient temperature resistivity          0.000000
Magnetic Physical Properties
relative volume magnetic                    -0.015600
    (linear) susceptibility
temperature coefficient susceptibility       0.000100
nominal diameter (mm)                        1.500000
ratio Copper/superconductor                  6.400000
critical intensity (Amp)                  1180.000000
magnetic field for I critical (Tesla)        4.000000
```

■ B1-3-2 Superconductors - Tables and Graphics

These materials are highly diamagnetic. The soft superconductor materials, at cryogenic temperature, such as Tin, Lead or Niobium, are even perfectly diamagnetic: χ (relative susceptibility) = -1. However, the hard superconductors used in magnet technology such as the Niobium-Titanium composite made of a bundle of very tiny filaments imbedded in a Copper matrix, or the Niobium-Tin compound are imperfectly diamagnetic. The values given in the table are very approximate and present little interest since these materials are used for their very high current-carrying capacity and not for their magnetic properties.

Example is given for a Niobium-Titanium wire: `SuperNbTi5` wire. The range: 0 to 10 Teslas, step of 1 Tesla.

```
MaterialMagnetizationTable[SuperNbTi5, FieldUnit → Tesla,
 MagnetizingFieldRange → {0, 10}, NumberOfPoints → 11]
```

Magnetization Table

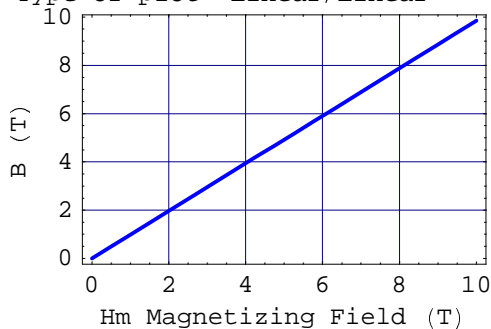**SuperNbTi5 temperature 20 °Celsius**

Nb-TI superconductor wire (A36)

Range for the magnetizing field: 0. -- 10. (T)

| Hm (T)	M (T)	B (T)	χ
0.000000	0.000000	0.000000	−0.015600
1.000000	−0.015600	0.984400	−0.015600
2.000000	−0.031200	1.968800	−0.015600
3.000000	−0.046800	2.953200	−0.015600
4.000000	−0.062400	3.937600	−0.015600
5.000000	−0.078000	4.922000	−0.015600
6.000000	−0.093600	5.906400	−0.015600
7.000000	−0.109200	6.890800	−0.015600
8.000000	−0.124800	7.875200	−0.015600
9.000000	−0.140400	8.859600	−0.015600
10.000000	−0.156000	9.844000	−0.015600

Plot for SuperNbTi5, of the magnetization curves: B (induction), M (magnetization), and Susceptibility (χ):

```
MaterialMagnetizationCurves[SuperNbTi5, {B, M, Susceptibility},
 FieldUnit → Tesla, MagnetizingFieldRange → {0, 10}]
```

```
Material Magnetization Curve
SuperNbTi5   temperature   20°celsius
Nb-TI superconductor wire (A36)
B Induction f(Hm)
Type of plot: Linear/Linear
```



```
Material Magnetization Curve
SuperNbTi5   temperature   20°celsius
Nb-TI superconductor wire (A36)
M Magnetization f(Hm)
Type of plot: Linear/Linear
```

```
Material Magnetization Curve
SuperNbTi5  temperature  20°celsius
Nb-TI superconductor wire (A36)
χ Susceptibility f(Hm)
Type of plot: Linear/Linear
```



```
     Hm Magnetizing Field (T)
```
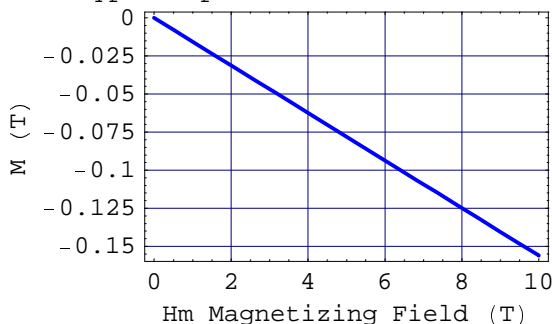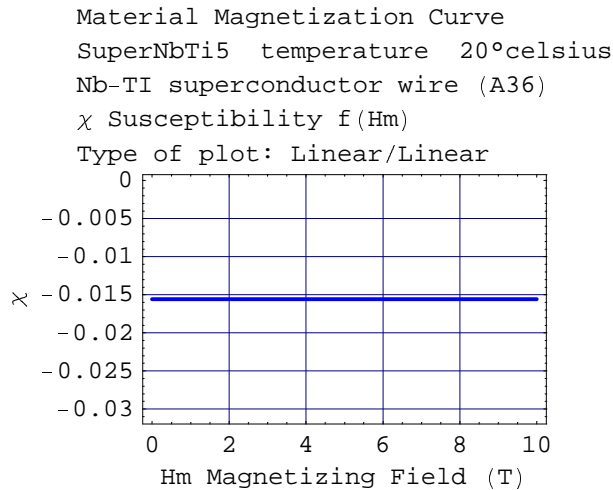
■ B1-4 Soft Ferromagnetic Materials with a Linear Law of Magnetization

■ B1-4-1 Properties

This type of material presents isotropic magnetic properties. It includes 6 different items:

▪ 2 different mild steels, Cobalt metal, Nickel metal, Mu metal, and a special theoretical material.

The relevant properties include the general properties in which the susceptibility data is the value for the low magnetizing range and complementary specifications including the saturation magnetization and the coercive field seen as the continuous reversibility limit.

The example: CobaltLin

```
Properties[CobaltLin]
```

Material properties
```
material name                        CobaltLin
material category                    Ferromagnetic reversible linear isotropic
designation                          Cobalt metal annealed (linear magnetization
```
General Physical Properties
```
specific weight (g/cm3)                    8.860000
dilation coefficient                       0.000013
    (relative linear change)
resistivity (Ohm x cm)                     6.630000e-6
Coefficient temperature resistivity        0.005200
```
Magnetic Physical Properties
```
relative volume magnetic                  40.000000
    (linear) susceptibility
temperature coefficient susceptibility    -0.000300
saturation magnetization (G)           17800.000000
coercive field                             0.700000
(continuous reversibility limit,G)
```

```
Properties[CobaltLin, Temperature → 0]
```

Material properties
material name   CobaltLin
material category   Ferromagnetic reversible linear isotrop
designation   Cobalt metal annealed (linear magnetiza
General Physical Properties
specific weight (g/cm3)   8.860000
dilation coefficient   0.000013
    (relative linear change)
resistivity (Ohm x cm)   6.630000e-6
Coefficient temperature resistivity   0.005200
Magnetic Physical Properties
relative volume magnetic   40.000000
    (linear) susceptibility
temperature coefficient susceptibility   -0.000300
saturation magnetization (G)   17800.000000
coercive field   0.700000
(continuous reversibility limit,G)
Properties at the temperature 0 ºCelsius
linear dilation (variation %)   -0.026000
electrical resistivity (variation %)   -10.400000
relative susceptibility (variation %)   0.600000
effective linear dilation coefficient   0.999740
effective electrical resistivity (ohm-cm)   5.940480e-6
effective volume magnetic susceptibility   40.240000
```

### ■ B1-4-2  Tables and Graphics

The linear modeling of the magnetization curves is applicable only at a low magnetizing field. For high magnetizing fields the magnetization is limited to a maximal strength: the saturation magnetization Ms. Then the susceptibility $\chi$ i.e. the ratio of the magnetization to the magnetizing field, decreases as Ms/Hm. Consequently, the magnetization curves of this model of soft ferromagnetic materials present 2 different shapes with a discontinuity for the critical magnetizing field Hms:

$$\mathbf{Hms} = \frac{\mathbf{Ms}}{\chi_{\mathbf{lin}}} \qquad\qquad (1)$$

- Magnetization curves for Cobalt pure metal: `CobaltLin`. The critical magnetizing field Hms stands at 445 Gauss.

**MaterialMagnetizationTable[CobaltLin]**

**Magnetization Table**

**CobaltLin temperature 20 °Celsius**

**Cobalt metal annealed (linear magnetization)**

**Range for the magnetizing field: 0. -- 890. (G)**

| Hm (G) | M (G) | B (G) | $\chi$ |
|---:|---:|---:|---:|
| 0.000000 | 0.000000 | 0.000000 | 40.000000 |
| 46.842105 | 1873.684211 | 1920.526316 | 40.000000 |
| 93.684211 | 3747.368421 | 3841.052632 | 40.000000 |
| 140.526316 | 5621.052632 | 5761.578947 | 40.000000 |
| 187.368421 | 7494.736842 | 7682.105263 | 40.000000 |
| 234.210526 | 9368.421053 | 9602.631579 | 40.000000 |
| 281.052632 | 11242.105263 | 11523.157895 | 40.000000 |
| 327.894737 | 13115.789474 | 13443.684211 | 40.000000 |
| 374.736842 | 14989.473684 | 15364.210526 | 40.000000 |
| 421.578947 | 16863.157895 | 17284.736842 | 40.000000 |
| 468.421053 | 17800.000000 | 18268.421053 | 38.000000 |
| 515.263158 | 17800.000000 | 18315.263158 | 34.545455 |
| 562.105263 | 17800.000000 | 18362.105263 | 31.666667 |
| 608.947368 | 17800.000000 | 18408.947368 | 29.230769 |
| 655.789474 | 17800.000000 | 18455.789474 | 27.142857 |
| 702.631579 | 17800.000000 | 18502.631579 | 25.333333 |
| 749.473684 | 17800.000000 | 18549.473684 | 23.750000 |
| 796.315789 | 17800.000000 | 18596.315789 | 22.352941 |
| 843.157895 | 17800.000000 | 18643.157895 | 21.111111 |
| 890.000000 | 17800.000000 | 18690.000000 | 20.000000 |

**MaterialMagnetizationCurves[CobaltLin, {B, M, Susceptibility}]**

Material Magnetization Curve
CobaltLin  temperature  20°celsius
Cobalt metal annealed (linear magnetization)
B Induction f(Hm)
Type of plot: Linear/Linear

```
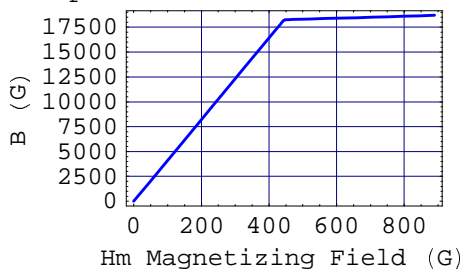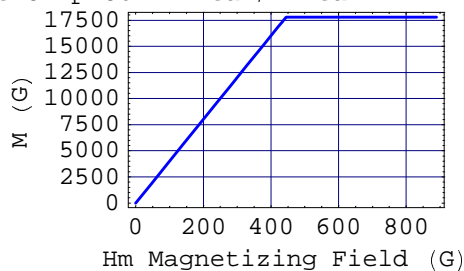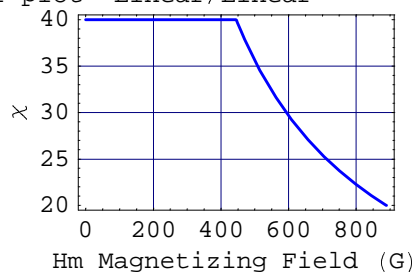Material Magnetization Curve
CobaltLin   temperature   20°celsius
Cobalt metal annealed (linear magnetization)
M Magnetization f(Hm)
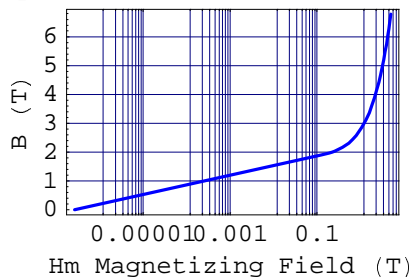Type of plot: Linear/Linear
```



```
Material Magnetization Curve
CobaltLin   temperature   20°celsius
Cobalt metal annealed (linear magnetization)
χ Susceptibility f(Hm)
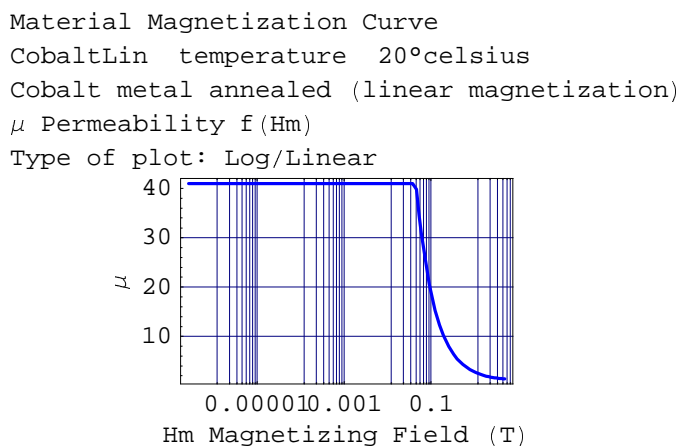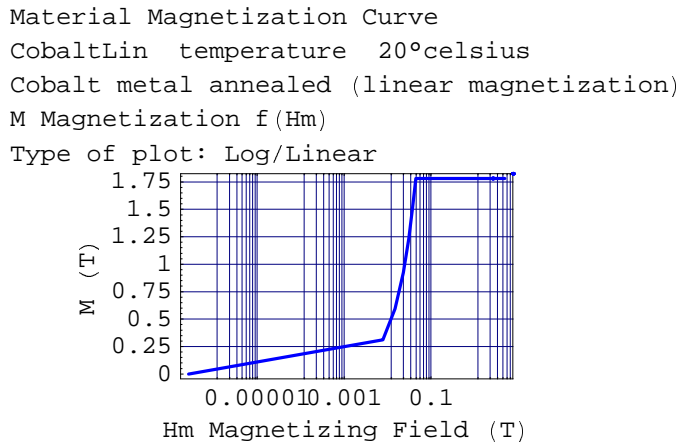Type of plot: Linear/Linear
```


- Magnetization curves for the same material but in a very high magnetizing field up to 5 Teslas. The figure shows that in a low magnetizing field Hm, the induction field B and the magnetization M, are interchangeable (using the proper units). But when the Hm goes beyond the Hms point (445 Gauss) the 2 physical properties become different, the higher the Hm the greater the difference.

```
MaterialMagnetizationCurves[CobaltLin, {B, M, Permeability},
  PlotScale → LogLinearMap, FieldUnit → Tesla, MagnetizingFieldRange → {0, 5}]
```

```
Material Magnetization Curve
CobaltLin   temperature   20°celsius
Cobalt metal annealed (linear magnetization)
B Induction f(Hm)
Type of plot: Log/Linear
```

```
Material Magnetization Curve
CobaltLin   temperature   20°celsius
Cobalt metal annealed (linear magnetization)
M Magnetization f(Hm)
Type of plot: Log/Linear
```



```
Material Magnetization Curve
CobaltLin   temperature   20°celsius
Cobalt metal annealed (linear magnetization)
μ Permeability f(Hm)
Type of plot: Log/Linear
```


■ B1-5 Soft Ferromagnetic Materials with a Non Linear Law of Magnetization

■ B1-5-1 Properties

This type of material presents isotropic magnetic properties. It includes 10 items:

- ▪ 2 different common steels
- ▪ 3 quasi pure iron steels (industrial products)
- ▪ 3 silicon steels and 2 high permeability materials.

The relevant properties include the general properties in which the susceptibility is an average value applicable in the low magnetizing range and complementary specifications including the saturation magnetization and the coercitive field seen as the continuous reversibility limit. These data are used when linear modeling is applied to this material. The other complementary data include the temperature coefficient for the saturation magnetization, the initial values of susceptibility and permeability (spontaneous magnetization) and the specifications at the crest of permeability: HmCr, MCr, χCr and coefficient of linearization (parameter for calculating average susceptibility).

The crest point appears as a very important feature of this type of ferromagnetic material. Crest magnetization MCr has a value around 35% of the saturation magnetization. The crest susceptibility and the crest magnetizing field vary considerably, depending on the material. The lowest HmCr corresponds to the highest χCr.

The example Silicon Steel: `SiSteel2`.

Properties[SiSteel2]

Material properties

material name	SiSteel2
material category	Ferromagnetic reversible non line
designation	Silicon steel sheet, grains NO, n

General Physical Properties

specific weight (g/cm3)	7.650000
dilation coefficient (relative linear change)	0.000012
resistivity (Ohm x cm)	0.000010
Coefficient temperature resistivity	0.001000

Magnetic Physical Properties

relative volume magnetic (linear) susceptibility	5683.886515
temperature coefficient susceptibility	-0.000300
saturation magnetization (G)	20700.000000
coercive field (continuous reversibility limit,G)	0.100000
temperature coefficient saturation mag	-0.000300

Specifications of the maximum of permeability

peak magnetizing field (G)	1.253865
peak magnetization (G)	7918.693612
peak susceptibility (relative to vacuum)	6315.429461
coefficient of linearization for susceptibility	0.900000

Properties at 35° C:

```
Properties[SiSteel2, Temperature → 35]
```

Material properties

material name	SiSteel2
material category	Ferromagnetic reversible non line
designation	Silicon steel sheet, grains NO, n

General Physical Properties

specific weight (g/cm3)	7.650000
dilation coefficient	0.000012
(relative linear change)	
resistivity (Ohm x cm)	0.000010
Coefficient temperature resistivity	0.001000

Magnetic Physical Properties

relative volume magnetic	5683.886515
(linear) susceptibility	
temperature coefficient susceptibility	−0.000300
saturation magnetization (G)	20700.000000
coercive field	0.100000
(continuous reversibility limit,G)	
temperature coefficient saturation mag	−0.000300

Specifications of the maximum of permeability

peak magnetizing field (G)	1.253865
peak magnetization (G)	7918.693612
peak susceptibility (relative to vacuum)	6315.429461
coefficient of linearization for susceptibility	0.900000

Properties at the temperature 35 ºCelsius

linear dilation (variation %)	0.018000
electrical resistivity (variation %)	1.500000
relative susceptibility (variation %)	−0.450000
effective linear dilation coefficient	1.000180
effective electrical resistivity (ohm-cm)	0.000010
saturation field (variation %)	1880.796942
effective volume magnetic susceptibility	5658.309025
effective saturation field (G)	20606.850000

■ B1-5-2 Tables and Graphics

The non linear magnetization curves have a very similar appearance for all materials but with considerable differences for the relevant numerical values. The susceptibility curves are bell-shaped curves, the crest point being the reference point of the curve. A plotting option allows comparing the linear approximation and the real experimental non linear magnetization curves.

    ▪ Silicon Steel tables for: `SiSteel2`.

```
MaterialMagnetizationTable[SiSteel2]
```

Magnetization Table

SiSteel2 temperature 20 °Celsius

Silicon steel sheet, grains NO, middle Si content (S2)

Range for the magnetizing field: 0. -- 7.28375 (G)

Hm (G)	M (G)	B (G)	χ
0.000000	0.000000	0.000000	1663.675569
0.383355	920.313595	920.696950	2400.681221
0.766710	3922.614212	3923.380923	5116.161672
1.150066	7215.441273	7216.591339	6273.939112
1.533421	9347.145310	9348.678731	6095.616850
1.916776	10506.928727	10508.845503	5481.563379
2.300131	11320.769227	11323.069358	4921.793009
2.683486	11900.752545	11903.436031	4434.810253
3.066841	12298.837335	12301.904176	4010.261829
3.450197	12597.724680	12601.174877	3651.306256
3.833552	12841.865029	12845.698581	3349.860787
4.216907	13044.475316	13048.692223	3093.375120
4.600262	13215.085102	13219.685364	2872.680829
4.983617	13360.709256	13365.692873	2680.925954
5.366973	13486.602469	13491.969441	2512.888268
5.750328	13596.759489	13602.509817	2364.519034
6.133683	13694.255494	13700.389177	2232.631776
6.517038	13781.483009	13788.000047	2114.685029
6.900393	13860.320164	13867.220558	2008.627550
7.283749	13932.252337	13939.536086	1912.786019

▪ Magnetization curves for this material:

```
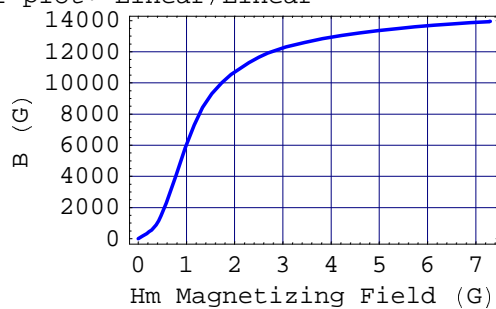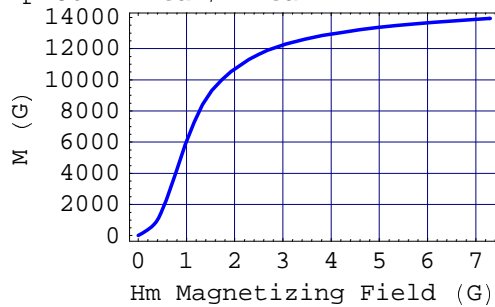MaterialMagnetizationCurves[SiSteel2, {B, M, Permeability}]
```

```
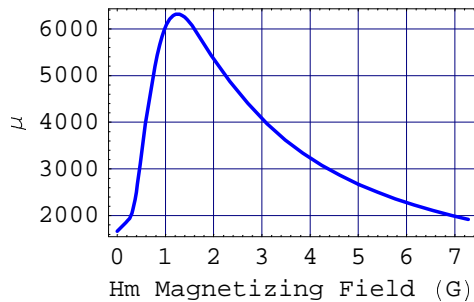terial Magnetization Curve
Steel2   temperature  20°celsius
licon steel sheet, grains NO, middle Si content (S
Induction f(Hm)
pe of plot: Linear/Linear
```

terial Magnetization Curve

Steel2   temperature   20°celsius

licon steel sheet, grains NO, middle Si content (S

Magnetization f(Hm)

pe of plot: Linear/Linear

terial Magnetization Curve

Steel2   temperature   20°celsius

licon steel sheet, grains NO, middle Si content (S

Permeability f(Hm)

pe of plot: Linear/Linear

▪ Comparison of the real magnetization curves and the linear model curves

MaterialMagnetizationCurves[SiSteel2, {B, M, Permeability}, CurveType → Comparison]

erial Magnetization Curve

:eel2   temperature   20°celsius

icon steel sheet, grains NO, middle Si content (

nduction f(Hm)

e of plot: Linear/Linear

erial Magnetization Curve
:eel2 temperature 20°celsius
icon steel sheet, grains NO, middle Si content (
agnetization f(Hm)
e of plot: Linear/Linear

erial Magnetization Curve
:eel2 temperature 20°celsius
icon steel sheet, grains NO, middle Si content (
ermeability f(Hm)
e of plot: Linear/Linear

■ B1-6 Hard Ferromagnetic Materials with High Coercivity (permanent magnet materials)

■ B1-6-1 Properties

This type of material presents strong anisotropic magnetic properties: the permanent magnetization is specific to a certain direction. This set includes 15 different items:

- 6 Ferrite substances,
- 8 Neodymium Iron Boron composites
- 1 Alnico alloy.

The relevant properties include the general properties in which the susceptibility is the relevant (dynamic) value for the principal direction of magnetization and complementary specifications including susceptibility in the transversal direction (orthogonal to the principal direction), remanent magnetization, coercitive field, and temperature coefficients for the susceptibility, the remanent magnetic field and the coercitive field.

The example: standard ferrite material: Ferrit2.

```
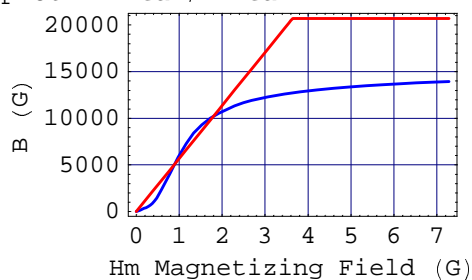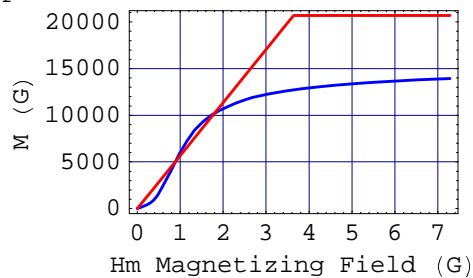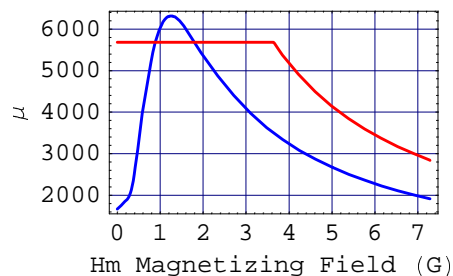Properties[Ferrit2]
```

Material properties

material name	Ferrit2
material category	Permanent magnet
designation	Ferrite O2 (42/32)

General Physical Properties

specific weight (g/cm3)	4.900000
dilation coefficient	0.000013
(relative linear change)	
resistivity (Ohm x cm)	0.000250
Coefficient temperature resistivity	0.001000

Magnetic Physical Properties

intrinsic remanent magnetic field (G)	4200.000000
intrinsic coercive field (G)	-3200.000000
principal dynamic relative susceptibility	0.100000
secondary relative susceptibility	0.200000
temperature coefficient susceptibility	0.000000
temperature coefficient remanent field	-0.002000
temperature coefficient coercive field	0.004000

▪ B1-6-2 Tables and Graphics

Example for the ferrite `Ferrit2`.

```
MaterialMagnetizationTable[Ferrit2]
```

Magnetization Table

Ferrit2 temperature 20 °Celsius

Ferrite O2 (42/32)

Range for the magnetizing field: -4800. -- 4800. (G)

Hm (G)	M principal (G)	B principal (G)	M transversal (G
-4800.000000	-4680.000000	-9480.000000	-960.00000
-4294.736842	-4629.473684	-8924.210526	-858.94736
-3789.473684	-4578.947368	-8368.421053	-757.89473
-3284.210526	-2925.714286	-6209.924812	-656.84210
-2778.947368	3922.105263	1143.157895	-555.78947
-2273.684211	3972.631579	1698.947368	-454.73684
-1768.421053	4023.157895	2254.736842	-353.68421
-1263.157895	4073.684211	2810.526316	-252.63157
-757.894737	4124.210526	3366.315789	-151.57894
-252.631579	4174.736842	3922.105263	-50.52631
252.631579	4225.263158	4477.894737	50.52631
757.894737	4275.789474	5033.684211	151.57894
1263.157895	4326.315789	5589.473684	252.63157
1768.421053	4376.842105	6145.263158	353.68421
2273.684211	4427.368421	6701.052632	454.73684
2778.947368	4477.894737	7256.842105	555.78947
3284.210526	4528.421053	7812.631579	656.84210
3789.473684	4578.947368	8368.421053	757.89473
4294.736842	4629.473684	8924.210526	858.94736
4800.000000	4680.000000	9480.000000	960.00000

Plot of the magnetization curves for the standard ferrite, for the operating range in the positive direction of magnetization.

MaterialMagnetizationCurves[Ferrit2, All]

Material Magnetization Curve
Ferrit2   temperature   20°celsius
Ferrite O2 (42/32)
M Magnetization Principal   f(Hm)

Material Magnetization Curve
Ferrit2   temperature   20°celsius
Ferrite O2 (42/32)
B Induction Principal   f(Hm)

Material Magnetization Curve
Ferrit2   temperature   20°celsius
Ferrite O2 (42/32)
M Magnetization Transversal   f(Hm)

Material Magnetization Curve
Ferrit2 temperature 20°celsius
Ferrite O2 (42/32)
B Induction Transversal f(Hm)

MaterialMagnetizationCurves[Ferrit2, All, MagnetizingFieldRange→HysteresisCycle]

Hysteresis Cycle
Ferrit2 temperature 20°celsius
Ferrite O2 (42/32)
M Magnetization Principal f(Hm)

Hysteresis Cycle
Ferrit2 temperature 20°celsius
Ferrite O2 (42/32)
B Induction Principal f(Hm)

```
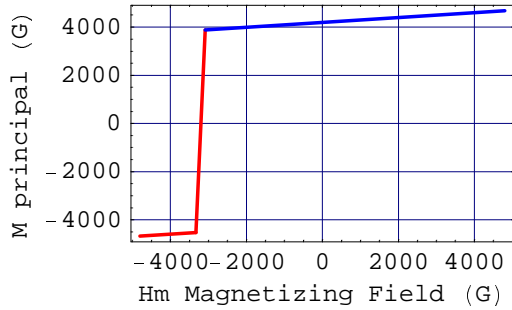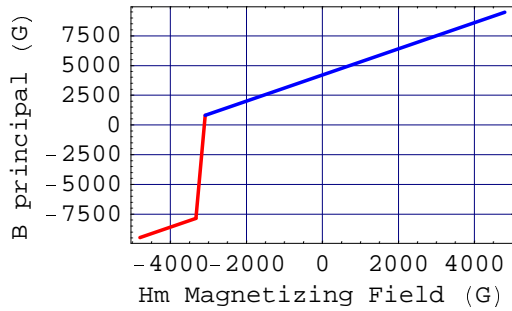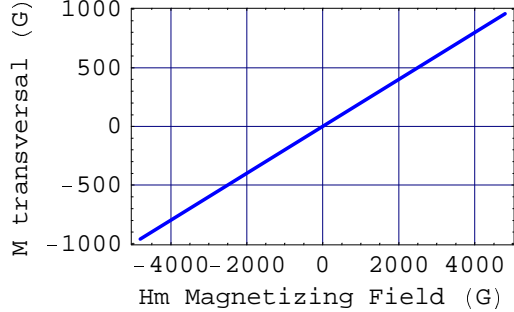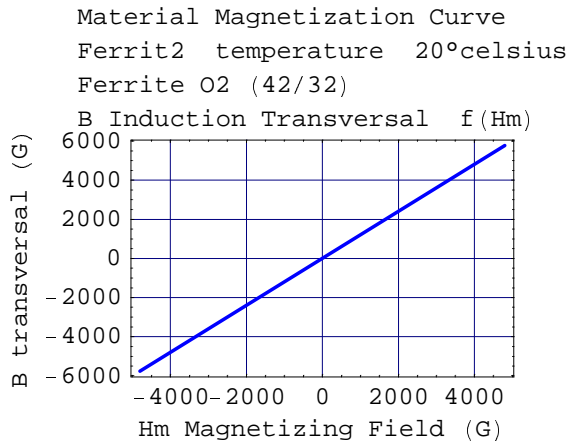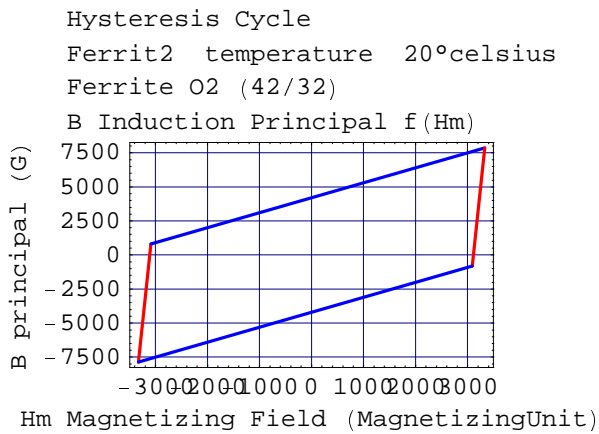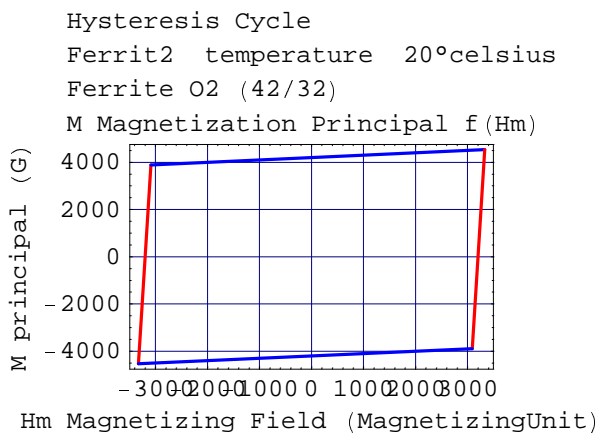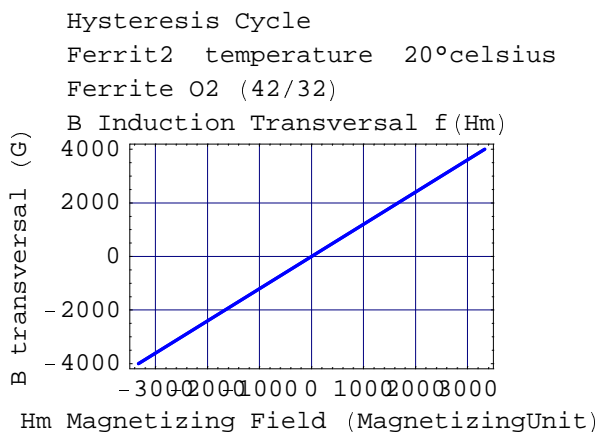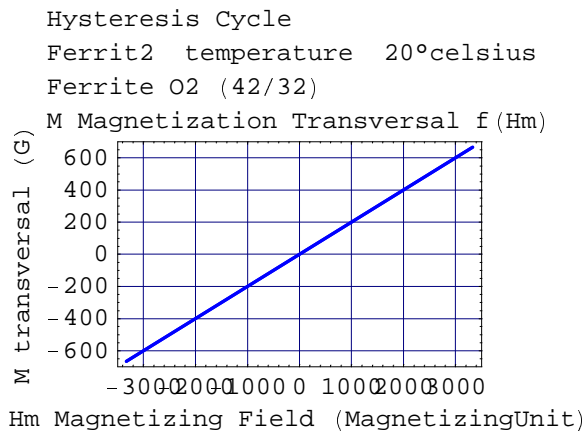Hysteresis Cycle
Ferrit2  temperature  20°celsius
Ferrite O2 (42/32)
M Magnetization Transversal f(Hm)
```

```
Hysteresis Cycle
Ferrit2  temperature  20°celsius
Ferrite O2 (42/32)
B Induction Transversal f(Hm)
```

▪ **B1-7 General Materials**

▪ **B1-7-1 Properties**

This category of material encompasses substances that may play a certain role in the magnetic calculation but are neither current-carrying materials nor ferromagnetic substances. This type of material includes 14 different items:

- ▪ 4 gases: Dry air, Oxygen, Nitrogen and Helium.

- ▪ 3 water substances: Distilled Water, Tap Water, and Sea Water.

- ▪ 4 plastic solid materials: Epox Glass, Phenolic Plastic, PolyEthylene Plastic, Silicon Resin.

- ▪ 2 liquid insulators: PFCarbon, TransformerOil.

- ▪ The vacuum is considered as a medium with specific properties.

The relevant data properties of the general materials are strictly the general properties set.

The example: oxygen gas (strongly paramagnetic): Oxygen.

```
Properties[Oxygen]
```

Material properties
```
material name                     Oxygen
material category                 General material
designation                       Oxygen gaz  normal temperature & pressure
```
General Physical Properties
```
specific weight (g/cm3)                   0.001308
dilation coefficient                      0.001130
    (relative linear change)
resistivity (Ohm x cm)                    1.000000e9
Coefficient temperature resistivity       0.000000
```
Magnetic Physical Properties
```
relative volume magnetic                  1.410000e-7
    (linear) susceptibility
temperature coefficient susceptibility    -0.003400
```

The same but at 60° Celsius

```
Properties[Oxygen, Temperature → 60]
```

Material properties
```
material name                     Oxygen
material category                 General material
designation                       Oxygen gaz  normal temperature & pressu
```
General Physical Properties
```
specific weight (g/cm3)                   0.001308
dilation coefficient                      0.001130
    (relative linear change)
resistivity (Ohm x cm)                    1.000000e9
Coefficient temperature resistivity       0.000000
```
Magnetic Physical Properties
```
relative volume magnetic                  1.410000e-7
    (linear) susceptibility
temperature coefficient susceptibility    -0.003400
```
Properties at the temperature 60 ºCelsius
```
linear dilation (variation %)             4.520000
electrical resistivity (variation %)      0.000000
relative susceptibility (variation %)    -13.600000
effective linear dilation  coefficient    1.045200
effective electrical resistivity (ohm-cm) 1.000000e9
effective volume magnetic susceptibility  1.218240e-7
```

■ B1-7-2 Tables and Graphics

These materials are very similar, in terms of magnetic behavior, to resistive conductors. They have a very low susceptibility which can be either positive (paramagnetic) or negative (diamagnetic).

   ▪ Example for Oxygen gas (paramagnetic).

```
MaterialMagnetizationTable[Oxygen, FieldUnit → Tesla,
 MagnetizingFieldRange → {0, 10}, NumberOfPoints → 11]
```

Magnetization Table

Oxygen temperature 20 °Celsius

Oxygen gaz normal temperature & pressure

Range for the magnetizing field: 0. -- 10. (T)

Hm (T)	M (T)	B (T)	
0.000000	0.000000	0.000000	1.410000e
1.000000	1.410000e-7	1.000000	1.410000e
2.000000	2.820000e-7	2.000000	1.410000e
3.000000	4.230000e-7	3.000000	1.410000e
4.000000	5.640000e-7	4.000001	1.410000e
5.000000	7.050000e-7	5.000001	1.410000e
6.000000	8.460000e-7	6.000001	1.410000e
7.000000	9.870000e-7	7.000001	1.410000e
8.000000	1.128000e-6	8.000001	1.410000e
9.000000	1.269000e-6	9.000001	1.410000e
10.000000	1.410000e-6	10.000001	1.410000e

▪ Plot of some of the magnetization curves:

```
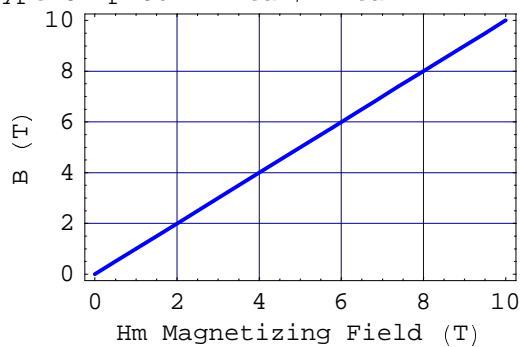MaterialMagnetizationCurves[Oxygen, {B, M, Susceptibility},
 FieldUnit → Tesla, MagnetizingFieldRange → {0, 10}]
```

```
Material Magnetization Curve
Oxygen   temperature  20°celsius
Oxygen gaz  normal temperature & pressure
B Induction f(Hm)
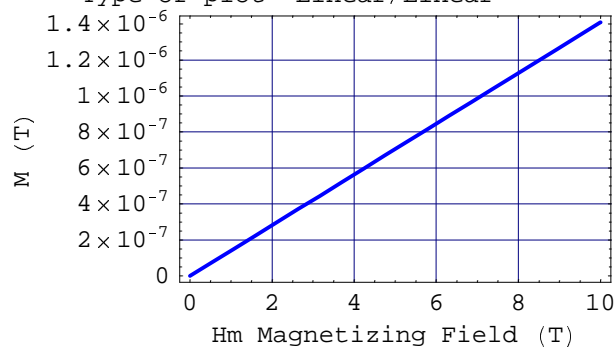Type of plot: Linear/Linear
```

Material Magnetization Curve
Oxygen   temperature   20°celsius
Oxygen gaz   normal temperature & pressure
M Magnetization f(Hm)
Type of plot: Linear/Linear

Material Magnetization Curve
Oxygen   temperature   20°celsius
Oxygen gaz   normal temperature & pressure
χ Susceptibility f(Hm)
Type of plot: Linear/Linear

■ Appendix B2: The Units in *Magnetica*.

■ B2-1 Introduction

For all its internal computations, *Magnetica* uses a comprehensive system of units very close to the International System of units (SI system).

However, the user is free to choose any units for input or output, the conversion between user units and the internal units is entirely automatic. The choice of the units can be made either for a full *Magnetica* session or selectively for any particular function calculation.

Immediate help is available either calling Units on the *Magnetica* palette "MagneticaBasics" or through the *Magnetica* Help Browser.

■ B2-2 Fundamentals on Magnetic Field Units

Magnetic field calculation deals with 3 entities, B, M and H (or Hm) known under different acronyms or names and assigned different units.

  - B: Magnetic induction, also called (even in legal texts) "density of magnetic flux" or "induction field" or "density of lines". The official unit of B in the SI system, legal international system, is Tesla. In the CGS EMU system (for "Electro Magnetic Units") the unit is Gauss. The conversion Tesla/Gauss is simple: 1Tesla = 10 000 Gauss

  - H or Hm: Magnetic field also called "Magnetic field strength". The unit of H in the SI system is the Ampere per meter (Ampere/Meter). In the CGS EMU system the unit is the Oersted.

A first problem arises when we look at a point in space in a vacuum. Clearly, there is only one physical entity that represents the local magnetic field however it is produced, whatever name it is called or whatever the official unit it is assigned. This entity is a vector with a magnitude and a direction.

Consequently, in a vacuum, using B or H to express the magnetic field is just a matter of convention and of a proper conversion factor between units.

The conversion Oersted / Gauss is very easy since they have exactly the same magnitude. The conversion Ampere/-Meter to Tesla is more laborious. The conversion factor is called μ_0 and named vacuum permeability:

$$\mu_0 = 4\ \pi\ 10^{-7} \tag{1}$$

All these denominations and units are used by physicists and engineers according to their current practice. For instance, geophysicists use the Gamma that is simply a Nano Tesla or 10 Micro Gauss.

When we want to describe the magnetic field inside matter, it is necessary to introduce a new entity that represents the magnetic field contribution of the substance, M. Then the distinction between B and Hm becomes meaningful. In a coherent system of units, we have the following relation:

$$\mathbf{B} = \mathbf{M}\ +\mathbf{Hm} \tag{2}$$

This is a **local** relation that expresses the fact that matter, by itself, generates a magnetic field which is added to the magnetic field coming from an external source Hm outside the local volume and the term Induction B means the total magnetic field in the **local** volume of matter.

- Hm: Magnetizing field expresses the idea that it is the cause of the **local** magnetic excitation of the matter that reacts by creating magnetic dipoles whose magnetic volume intensity is M. The response of the matter M to the excitation is specific to each substance and is expressed by the law of magnetization (see Chapter 6-2):

$$\mathbf{M} = \mathcal{X} \times \mathbf{Hm} \qquad\qquad\qquad\qquad\qquad\qquad\qquad (3)$$

- M: Magnetization. This is the most common name but it is also called Magnetic Polarization and given the acronym J. There is no legal definition of the unit of M and therefore all previously mentioned units are in use, depending on the practice.

- $\mathcal{X}$: Susceptibility. It is a dimensionless parameter in a coherent magnetic system of units. In that case $\mathcal{X}$ is called the relative volume magnetic susceptibility or simply relative susceptibility. $\mathcal{X}$ can be expressed with specific dimension units depending on the units for M and Hm. Also, it can be expressed for a given mass of matter instead of a volume. This is the case in most handbooks on material properties. Proper conversion must be done to obtain the dimensionless value.

In the *Magnetica* system of unit conversion, the user is free to choose his/her units according to his/her usual practice.

■ B2-3 The *Magnetica* Units

■ B2-3-1 The Symbols for the *Magnetica* Units

For all types of physical entities concerned by field calculation and applications, there are specific symbols that must be properly used. The immediate help for the function `MagneticaUserUnits` provides the entire list of *Magnetica* symbol names:

```
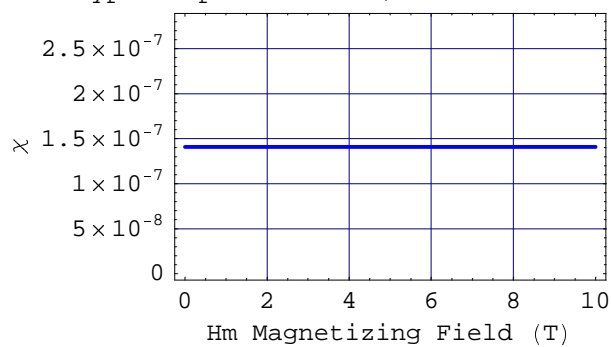? MagneticaUserUnits

MagneticaUserUnits[LengthUnit -> Meter, ...]
   overrides the default Magnetica units with user defined units.
The available units are:

LengthUnit           (for all lengths)
FieldUnit            (for the magnetic field)
PotentialUnit        (for the magnetic potential)
FluxUnit             (for the magnetic flux)
GradientUnit         (for the magnetic field gradient)
InductanceUnit       (for the inductance)
MagneticEnergyUnit   (for the energy)
MagneticMomentUnit   (for the magnetic moment)
CurrentIntensityUnit (for the current intensity)
PowerUnit            (for the power)
VoltageUnit          (for the current voltage)
ResistanceUnit       (for the electrical resistance). Details…
```

The unit values are composed of a symbol name and of an optional prefix that states the value. A blank character (i.e. a space), or the multiplication sign * must be used to separate the prefix from the unit name:

The most common, available prefixes are the following:
```
Pico, Nano, Micro, Milli, Centi, Deci, Kilo, Mega, Giga
```

The units can be given various designations such as :

```
            FieldUnit  Gauss, Tesla, Ampere / Meter
        PotentialUnit  Weber / Meter, Gauss / Meter
            FluxUnit  Weber
        GradientUnit  Gauss / Meter
          LengthUnit  Meter, Inch, Foot, Feet, ...
 CurrentIntensityUnit  Ampere
       InductanceUnit  Henry
   MagneticEnergyUnit  Joule, ...
   MagneticMomentUnit  Ampere Meter ^ 2
```

The function `MagneticaCurrentUnits` displays all the currently used units. If called before any change of units is made we obtain the internal *Magnetica* system of units. They are:

    `MagneticaCurrentUnits[]`

```
Length                  Milli Meter
Magnetic field          Gauss
Magnetic potential      Micro Weber/Meter
Magnetic flux           Micro Weber
Magnetic field gradient Gauss/Meter
Inductance              Micro Henry
Magnetic energy         Joule
Magnetic moment         Micro Ampere Meter²
Current intensity       Ampere
Power                   Watt
Voltage                 Volt
Electrical Resistance   Ohm
```

Some additional units are available only at a local level. For example:

   `MagnetizingFieldUnit:`   for the magnetizing field (Hm)

   `MaterialMagnetizationUnit:`   for the magnetization (M)

   `InductionUnit:`   for the induction field (B=Hm+M)

■ B2-3-2 Levels for Specifying the Units

In *Magnetica*, units can be set in 3 different ways:

- ▪ The global level: units are set by using the function `MagneticaUserUnits` in the following way:

    `MagneticaUserUnits[LengthUnit → Meter, FieldUnit → Milli Tesla]`

- ▪ The function level: units are specified as additional options for the function

    `FieldPoint[system, {10, 12}, LengthUnit → Centi Meter, FieldUnit → Milli Gauss]`

- ▪ The argument level: the unit value is given together with the argument as in the following example:

    `FluxArea[system, {10 Meter, 12 Centi Meter}]`

■ B2-4 Examples of Using *Magnetica* Unit System

■ B2-4-1 Introduction

We show cases of 2 identical magnetic systems but defined with different units for the following applications:

1. Entering data for magnetic systems

2. Displaying punctual field calculations

3. Plotting field curves

■ B2-4-2 Entering Data for Magnetic Systems

Magnetic system made of a single solenoid. This system is built first with the default units and then after some units are changed at the global level

Case 1: magnetic system data

```
solenoid1 = AXSMagneticSystem[
  {AXSSolenoid[AlumElec, 600, 1200, 300, 0, 500, 160, FillingFactor → 0.6]},
  SystemLabel → "generic solenoid"]
```

```
– AXSMagneticSystem :{-solenoid-} –
```

Changing the units for length and magnetic field: these units become the new *Magnetica* units for any subsequent calculation.

```
MagneticaUserUnits[FieldUnit → Tesla, LengthUnit → Meter]
```

```
{FieldUnit → Tesla, LengthUnit → Meter}
```

Case 2: entering data for the same magnetic system but using the new units:

```
solenoid2 = AXSMagneticSystem[
  {AXSSolenoid[AlumElec, 0.6, 1.2, 0.3, 0, 500, 160, FillingFactor → 0.6]},
  SystemLabel → "generic solenoid"]
```

```
– AXSMagneticSystem :{-solenoid-} –
```

We will next verify, by computing the magnetic field at a given point, that these 2 magnetic systems are the same.

■ B2-4-3 Punctual field calculations

Punctual field calculation at coordinates r=z=0.1 Meter. (N.B.: the field unit is currently Tesla)

for Case 1:

```
FieldPoint[solenoid1, {0.1, 0.1}]
```

```
{0.00715304, 0.10485}
```

```
VerboseFieldPoint[solenoid1, {0.1, 0.1}]
```

```
System name: solenoid1
The magnetic field (in Tesla) at the point {0.1, 0.1}(in Meter)
Calculation mode: Automatic Mode
Br          0.007153 T
Bz          0.104850 T
Bm          0.105094 T
```

for Case 2:

```
FieldPoint[solenoid2, {0.1, 0.1}]
```

```
{0.00715304, 0.10485}
```

```
VerboseFieldPoint[solenoid2, {0.1, 0.1}]
```

```
System name: solenoid2
The magnetic field (in Tesla) at the point {0.1, 0.1}(in Meter)
Calculation mode: Automatic Mode
Br          0.007153 T
Bz          0.104850 T
Bm          0.105094 T
```

Changing the units, but only for this calculation:

- Length into Milli Meter

- Field into Micro Tesla

```
FieldPoint[solenoid1, {100, 100}, LengthUnit → Milli Meter, FieldUnit → Micro Tesla]
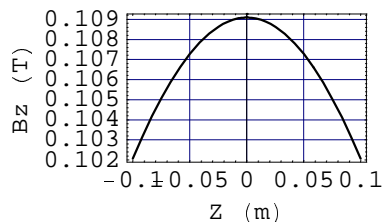```

```
{7153.04, 104850.}
```

■ B2-4-4 Plotting field curves

Plot of the Bz component of the field along an axial line

Using the global units previously defined and still in use: Meter and Tesla:

```
FieldMap[solenoid1, {Bz}, Line[{0, -0.1}, {0, 0.1}]]
```

```
  System name: solenoid1
  generic solenoid
  Bz (T) (Continuous, Automatic Mode)
  between {0., -0.1} and {0., 0.1}
```



```
{- Graphics -}
```

Changing, for this plot:

- the Length unit at the argument level

- the Field unit at the function level

FieldMap[solenoid1, {Bz}, Line[{0, -3 Foot}, {0, 3 Foot}], FieldUnit → Gauss]

System name: **solenoid1**
generic solenoid
Bz (G) (Continuous, Automatic Mode)
between {0., -0.9144} and {0., 0.9144}

{- Graphics -}

■ **Appendix B3: Using *Magnetica* Palettes**

■ **B3-1 Introduction**

Palettes are graphic objects (windows) that contain active buttons enabling you to avoid typing in certain expressions. They allows you to paste codes in your working notebook.

Magnetica provides certain palettes that are also convenient ways to visualize the most common *Magnetica* features. Here is a brief description of each:

- MagneticaBasics Palette: provides a brief overview (approach) to the most commonly used *Magnetica* functionalities.

- MagneticaMaterials Palette: displays the materials available in *Magnetica*.

- MagneticaComponents Palette: facilitates the elaboration of components by using forms.

All the *Magnetica* Palettes are in the sub-menu File ▷ Palettes

■ **B3-2 *MagneticaBasics* Palette**

■ **Location and Appearance:**

Open the "MagneticaBasics" Palette with File ▷ Palettes ▷ MagneticaBasics. If it is correctly installed, something like the following window will appear:

■ **Description**

The MagneticaBasics Palette gives an overview of the main functionalities of *Magnetica*, and allows the user to use them quickly. The top level is divided in 4 sections:

- 2D Axisymmetric systems: components, drawing, field calculations, field maps, for axisymmetric systems.

- 3D Cartesian systems: functionalities for cartesian systems.

- Materials: how to access materials informations and curves.

- Units: how to obtain or modify units.

Click on the "triangles" on the left to open the desired section and to proceed.

To use any function, you just have to click on the buttons to paste the corresponding code in your working notebook. Then, replace the arguments by the relevant values, or variable names and evaluate.

■ B3-3 *MagneticaMaterials* Palette

■ Location and Appearance:

Open the "MagneticaMaterials" Palette with File ▷ Palettes ▷ MagneticaMaterials. If it is correctly installed, something like the following window will appear:

■ Description

The "MagneticaMaterials" Palette presents all the materials available in *Magnetica*. They are grouped by categories. Categories and their meaning are described in Appendix B1 on materials.

■ Usage

1. Open the desired category.

2. Scan with the pointer the grid that appears, to see a description of each material at the bottom of the palette.

3. Click on the button corresponding to the material chosen and its name will be pasted in your working
notebook.

■ B3-4 *MagneticaComponents* Palette

■ Location and aspect:

Open the "MagneticaComponents" Palette with

File ▷ Palettes ▷ MagneticaComponents.

If it is correctly installed, something like the following window will appear:

■ Description

This palette presents one button for each input type of kind of *Magnetica* component. Possible components are either
current-carrying components or magnetic components.

■ Usage

When you click on a button, an other window that contains a form pops up. It describes each argument to give. It must
look like this:

Fill in the edit boxes, and click on the button "Paste Expression" at the end of the form. The expression corresponding to this Magnetica component will be pasted in your working notebook.

■ Appendix B4: Practical Usage and Miscellaneous

■ B4-1 Some *Mathematica* Manipulations

Magnetica proceeds from *Mathematica* which is a powerful but complex calculation and graphics tool as well as an intricate programming language.

But *Magnetica* is designed so that the common or hurried user not have to go into *Mathematica* in depth.To avoid any possible interference, the following rules or tricks are recommended:

- Working Session : a working session is initialized as soon as *Mathematica* (and *Magnetica*) have been called. During a session, all the computer objects are kept in the memory: symbols, expressions, data, numerical results, graphics. This may be useful, but for lengthy calculations this will reduce the efficiency of computation or even abruptly end the session with loss of data if the memory runs out.

You can end a session at any time by either

- exiting *Mathematica*

- starting a new session without quitting *Mathematica* by using the menu command Kernel ▷ Quit Kernel

To avoid unpleasant problems, you are advised:

- to save your notebooks regularly, and particularly if you have many graphics,

- to start new sessions repeatedly.

- Symbols Writing : use only lower case letters. In *Mathematica*, as well as in *Magnetica*, some upper case letters are used for naming functions. In order to avoid any possible conflict, use only lower case letters in your program.

- Expressions Use: any *Mathematica* and Magnetica function is a formal expression producing results either numerical or graphic results. You may be interested either in simply displaying the results or in using them for further calculation. However, remember that adding a semi-colon at the end of an expression cancels its final output. The following rules may be applied:

- in displaying raw results of a numerical expression, do not use a semi-colon as a final character

N[Pi]   ▷ 3.14159

N[Pi];   ▷ nothing

FieldPoint[solenoid1,{100,50}] ▷ {38.5438, 1108.12}

- furthermore *Magnetica* is set with a useful prefix to display not only the final result but also several relevant items of data: this is done with the Verbose prefix as for instance:

VerboseFieldPoint[solenoid1,{100,50}] ▷

```
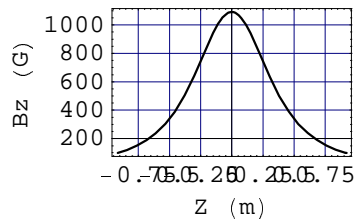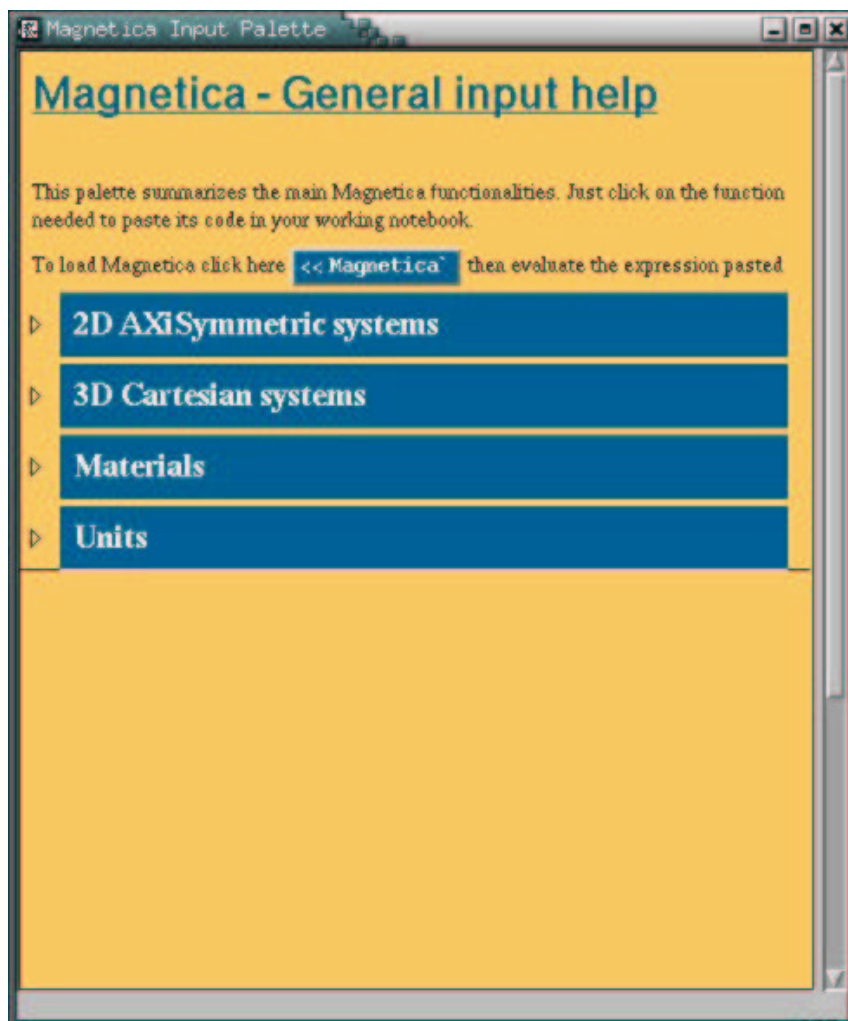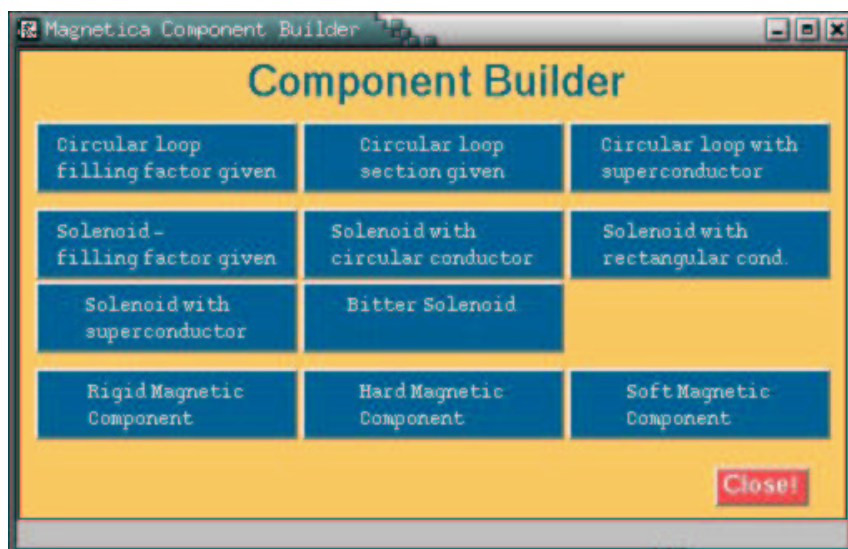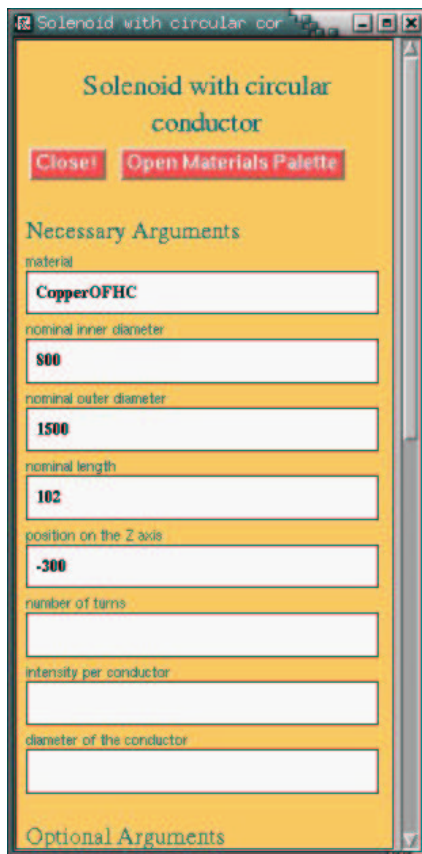System: solenoid1
the magnetic field (in Gauss) at the point {100, 50}(in Milli Meter)
Calculation mode: Automatic Mode
Br          38.543850 G
Bz        1108.118151 G
Bm        1108.788287 G
```

- Numerical Calculation and Display : *Mathematica* offers the possibility of executing calculations at any level of precision, the default situation being the machine precision set to 16/17 decimal digits on most computers. This is the standard precision provided by *Magnetica* for real numbers. But integers have an infinite precision and within the process of calculation, *Mathematica* inclines to maintain this precision. This may result in lengthy calculations.   *Magnetica* carries out all the adhoc transformations when data are entered and you do not have to worry about this problem. But you may be interested in doing side calculations using, *de facto*, *Mathematica.* The following rules may be applied :

1. to convert a number from an integer form to real form:

    - add a dot at the end of the number string or use the *Mathematica* function N: example

    18   ▷   internal integer number

    18.   ▷   internal real number

    N[18] ▷   internal real number

2. to convert a number from real form to integer form use the following *Mathematica* functions: Round, Floor or Ceiling

    - Round[3.1416] ▷ 3

    - Floor[3.1416]   ▷ 3

    - Ceiling[3.1416] ▷ 4

3. to use and display real numbers: the standard (default case) of displaying a real number in *Mathematica* is a 6 digit (or less) expression but a greater number of digits may be set or displayed using the functions N or NumberForm with the indication of the number of digits. The example is given for Pi which is a number of primordial importance in magnetostatics and which, moreover, has a special meaning in *Mathematica*. Naming "Pi" is a special reserved name and although a real irrational number its basic precision is infinite.

    3.1. setting and using Pi with a given precision

    - standard 6 digits precision: N[Pi]   ▷   3.14159

    - 20 digits precision: pinew = N[Pi,20]   ▷ 3.1415926535897932385

    the function N carries out a numerical conversion, the symbol pinew can be given this numerical value for further use. Then this new symbol is kept within the actual working session with this internal precision.

    3.2. Displaying numbers. The function NumberForm does not affect the internal precision of the number but only its display. Consequently the maximum number of digits corresponds to the internal precision. It can be reduced but not increased.

    - NumberForm[pinew,12] ▷ 3.14159265359

3.3 Displaying the number's internal precision: the *Mathematica* Function Precision gives the actual precision in decimal digits. Examples:

- Precision[Pi]   ▷ Infinity

- Precision[pinew] ▷ 20

■ B4-2 Magnetic Field Security Limits

It is known that a **static magnetic field** has a very limited effect on living cells but, depending on its intensity, it may produce definite and important effects on many apparatuses or devices. The following table is given merely for general information and not as a reference table.

Magnetic field intensity is given in Gauss and Milli Tesla. It must be remembered that the Earth's magnetic field intensity is around 0.5 G (0.05mT). Its intensity varies slightly going from the North Pole (or South Pole) to the equator (between 0.6 and 0.3 gauss). But, it may be strongly affected locally by large deposits of iron ore on the Earth's crust.

field limit	equipment
0.5 G (0.05 mT)	medical nuclear camera
1 G (0.1 mT)	linear accelerator, cyclotron, positron emission tomography scanner accurate measuring scale, electron microscope color TV, video display, CT scanners, image intensifiers, ultra sound device, lithotriptor
3 G (0.3 mT)	power transformers, main electrical distribution transformers moving steel equipment : — Vehicular traffic, dumb waiter, loading dock (truck traffic) — fork lift truck, electric transport cart, — elevator, escalator, helicopter
5 G (0.5 mT)	cardiac pacemaker, biosimulation devices, neurosimulator
10 G (1 mT)	magnetic tape, floppy discs, hard copy imager, line printer video cassette recorder, film processor, credit card telephone switching equipment, X – ray tube large steel equipment : — emergency generator, commercial laundry equipment — food preparation area, fuel storage tank, — air conditioning equipment, motor greater than 5 horsepower
30 G (3 mT)	medical MRI equipment : system control cabinet, heat exchanger for mobile system gradient coil cooling
50 G (5 mT)	telephones and most MRI equipment including metal detector for screening
100 G (10 mT)	service medical MRI equipment including Oxygen monitor, pneumatic patient alert control box
200 G (20 mT)	special medical MRI equipment including remote oxygen sensor module, MRI room penetration panel, blower box, magnet rundown unit
300 G (80 mT)	medical MRI brainwave cart

Reference Guide

■ *Magnetica* **Reference Guide**

■ **Introduction**

The Magnetica Reference Guide is a listing and a documentation of all the symbols manipulated in Magnetica. This includes the following:

■ `Functions`: functions are the basic way to manipulate Magnetica. As usual in *Mathematica* the standard syntax is in the form:

        FunctionName[arg1,arg2,...,option1,option2,...]

for each function the syntax is described including meaning of each arguments and possible options.

■ `Options`: many functions can be given optional arguments that are specified in the form of a rule as follows:

        OptionName -> OptionValue,

Option values can be numbers or predefined Values.

■ `Values`: some options and some arguments can be given only predefined values. These are also documented and related functions are listed.

■ **AngularRange**

■ `AngularRange` is an option for `DrawIsoFieldLines`-like functions that allows to choose the range of the polar plot around the origin.

■ `AngularRange` must be given a list of exactly two numeric values: the first angle and the last angle of the range.

■ `AngularRange` values are expressed in Degree.

■ See the Magnetica User Book: Chapter 4.

■ See also: `DrawIsoFieldLines`, `DrawIsoFieldLinesZX`.

■ **AXSBitterSolenoid**

■ `AXSBitterSolenoid[`*material, nominalInnDiameter, nominalOutDiameter, nominalLength, position, number-Turns, intensity, disksWidth* `]` generates a Bitter solenoid.

■ This generates a Magnetica AXiSymmetric Component.

■ This component is similar to the AXSSolenoid, but the current density is inversely proportionnal to the radius (1/r), instead of being constant.

■ Argument 1, the material, must be choosen among the Magnetica Materials and is supposed to be a Current Carrying.

■ Argument 2, is the inner diameter, of the structure (nominal dimension).

■ Argument 3, is the outer diameter, of the structure (nominal dimension).

■ Argument 4, is the lenght of the solenoid, (nominal dimension).

■ Argument 5, is the position of the center of the solenoid on the Z-axis. (temperature independent).

■ Argument 6, is the number of disks.

■ Argument 7, is the current intensity per conductor, signed according to the direction of the current (+,-).

■ Argument 8, is the width of the disks constituting the Bitter Solenoid.

■ Possible option:

`Temperature`: allows to set the actual temperature to another one than the nominal temperature 20°. This value will be used to evaluate the exact dimensions and other temperature dependent properties.

■ Related units:

`LengthUnit`: affects the values given to the inner diameter, the outer diameter, the solenoid length, the position on the Z-axis and the disks width.

`CurrentIntensityUnit`: affects the value given to the current intensity.

■ See the Magnetica User Book: Chapter 3.

■ See also: `AXSMagneticSystem`, `AXSSolenoid`, `LengthUnit`, `CurrentIntensityUnit`.

■ AXSCircularLoop

■ `AXSCircularLoop[`*material, nominalMidDiameter, windingNominalDiameter, position, numberTurns, intensity,* „ `FillingFactor->`*filling*`]` generates a standard current circular loop.

■ `AXSCircularLoop[`*material, nominalMidDiameter, windingNominalDiameter, position, numberTurns, intensity,* `ConductorDiameter->`*diameter*`]` a circular loop, specifying the conductor diameter, instead of the filling factor.

■ `AXSCircularLoop[`*material, nominalMidDiameter, windingNominalDiameter, position, numberTurns, intensity*`]` without any information about the condutor is understood as superconducting wire, since for this kind of material dimensions are fixed properties of the material.

■ This generates a Magnetica AXiSymmetric component.

■ A circular loop is current carrying winding, of circular cross section. For a rectangular cross section see `AXSSolenoid`.

■ Argument 1, the material, must be choosen among the Magnetica Materials and can be either a Current Carrying or a Superconductor.

■ Argument 2 is the diameter of the loop, measured from the center of the winding.

■ Argument 3 is the diameter of the winding, i.e. the section of the loop.

■ Argument 4 is the position of the center of the loop on the Z-axis.

■ Argument 5 is the number of turns of the winding, signed according to the standard orientation right or left hand side (+,-).

■ Argument 6 is the current intensity per conductor, signed according to the direction of the current (+,-).

■ Additionnal information on the condutor dimension can be given by setting (only) one of the following options `FillingFactor` or `ConductorDiameter`.

■ Without specifying neither the filling factor nor the conductor diameter suppose the winding is made of superconducting wire.

■ Other possible options:

`Temperature`: all the dimension arguments are given as nominal values, that is the value at the nominal temperature 20°. This option specify the actual temperature of the component that will be used to evaluate the exact dimensions and other temperature dependent properties.

`FilamentMode`: (`True` or `False`) allows to assume that current is concentrated in the center of the loop. The default value is True.

■ Related units:

`LengthUnit`: affects the values given to the midDiameter, the windingDiameter, the position on the axis, and also to the `ConductorDiameter` option.

`CurrentIntensityUnit`: affects the value given to the intensity.

■ See the Magnetica User Book: Chapter 2.

■ See also: `AXSMagneticSystem`, `AXSSolenoid`, `FillingFactor`, `ConductorDiameter`, `Temperature`, `FilamentMode`, `LengthUnit`, `CurrentIntensityUnit`.

■ **AXSHardMagneticComponent**

■ `AXSHardMagneticComponent[`*material, nominalInnDiameter, nominalOutDiameter, nominalLength, position*`]` generates a cylindrical magnet (bar or ring) whose magnetization vary linearly with the applied field.

■ This generates a Magnetica AXiSymmetric Component. It is a Ferromagnetic component.
■ This component keeps permanently a trace of magnetic field, and the remaining magnetization is referred to by the terms "Remanent Magnetization". (`AXSRigidMagneticComponent` also does so).
■ This component takes into account the linear variation of the magnetization with the applied magnetizing field. (`AXSSoftMagneticComponent` deals with linear and non-linear variation). In order to do this, the component is subdivided into cells each with uniform magnetization.
■ The thinness of the meshing can be adjusted by using the option `ComponentMeshing` (see below). Otherwise the values are determined by the system option `MeshingLevel`.
■ Argument 1, the material, must be choosen among the Magnetica Materials and is supposed to be a Permanent Magnet.
■ Argument 2, is the inner diameter of the structure, that is 0 in case of a bar (nominal dimension).
■ Argument 3, is the outer diameter of the structure (nominal dimension).
■ Argument 4, is the length of the component (nominal dimension).
■ Argument 5, is the position of the center of the component on the Z-axis (temperature independent).
■ Possible options are:
`FillingFactor`: to specify the rate of magnetic material in the component. The default is 1.
`MagnetizationAngle`: to specify the angle between the (oriented) Z-axis and the magnetization direction. The default value is 0.
`RemanentMagnetization`: to be specified if the magnetization must be taken different from the material property.
`Rigid`: (`True` or `False`) allows to consider unvariant magnetization. The default is `False`.
`ComponentMeshing`: allows to specify the number of radial cells and axial cells of the component. The default setting is `Automatic` that tells to refer to the system meshing level.
`Temperature`: allows to set the actual temperature to another one than the nominal temperature 20°. This value is used to evaluate the exact dimensions and other temperature dependent properties.
■ Related units:
`LengthUnit`: affects the values given to the inner diameter, the outer diameter, the length, the position.
`FieldUnit` or MaterialMagnetizationUnit:
■ See the Magnetica User Book: Chapter 7.
■ See also: `AXSMagneticSystem, AXSRigidMagneticComponent, AXSSoftMagneticComponent, FillingFactor, MagnetizationAngle, RemanentMagnetization, Rigid, ComponentMeshing, Temperature,`

■ **AXSLargeMagneticSource**

■ `AXSLargeMagneticSource[`*Bz*`]` simulates an axisymmetric large magnetic source producing at any point a constant magnetic field, directed along the Z-axis, and whose magnitude is *Bz*.

■ This generates a Magnetica AXiSymmetric Component.
■ The only argument specifies the intensity of the magnetic field, the direction being the Z-axis.
■ The argument is expressed in the unit of `FieldUnit`.
■ See the Magnetica User Book: `Chapter 3`.
■ See also: `AXSMagneticSystem, FieldUnit`.

■ **AXSMagneticSystem**

■ AXSMagneticSystem[*{axscomponent1,axscomponent2,...}*] builds a Magnetica object representing a magnetic system.

■ The first argument is a list of Axisymmetric Components.
■ The prefix "AXS" stands for AXiSymmetric.
■ The available axisymmetric components are: AXSCircularLoop, AXSSolenoid, AXSBitterSolenoid, AXSRigidMagneticComponent, AXSHardMagneticComponent, AXSSoftMagneticComponent and AXSLargeMagneticSource.
■ The possible options are the following:
SystemLabel: to specify a label to be attached to that system.
MeshingLevel: is only interesting for systems containing Ferromagnetic Components. It allows to choose the granularity of the meshing for these components. The default value is 2.
LegendrePointPosition: affects all Spherical Harmonics features, including the definition of the SHLimits, the calculation mode etc. It specifies the position of the Legendre point on the Z-axis. The default value is 0.
LegendreRadius: specifies the Legendre radius value for this system. The default setting is Automatic.
LegendreOrder: specifies the number of terms to use in Spherical Harmonics series. The default setting is 20.
■ See the Magnetica User Book: Chapter 2, Chapter 3.
■ See also: AXSCircularLoop, AXSSolenoid, AXSBitterSolenoid, AXSRigidMagneticComponent, AXSHardMagneticComponent, AXSSoftMagneticComponent, AXSLargeMagneticSource, SystemLabel, MeshingLevel, LegendrePointPosition, LegendreRadius, LegendreOrder.

■ **AXSRigidMagneticComponent**

■ AXSRigidMagneticComponent[*material, nominalInnDiameter, nominalOutDiameter, nominalLength, position*] generates a cylindrical or a bar magnet made of rigid (constant) magnetization material.

■ This generates a Magnetica AXiSymmetric Component.
■ This is also a Ferromagnetic Component, (but it is not subdivided into cells).
■ Magnetization is taken from material property or can be overriden with an option (see below). It is then considered unvariant, so the susceptibility is supposed null. For varying magnetization see AXSHardMagneticComponent.
■ Argument 1, the material, must be choosen among the Magnetica Materials and is supposed to be a Permanent Magnet.
■ Argument 2, is the inner diameter of the structure, that is 0 for a bar (nominal dimension).
■ Argument 3, is the outer diameter of the structure (nominal dimension).
■ Argument 4, is the length of the component (nominal dimension).
■ Argument 5, is the position of the center of the component on the Z-axis (temperature independent).
■ Possible options are:
FillingFactor: to specify the rate of magnetic material in the component. The default value is 1.
MagnetizationAngle: to specify the angle between the (oriented) Z-axis and the magnetization direction. The default value is 0.
RemanentMagnetization: to be specified if the magnetization must be taken different from the material property.
Temperature: allows to set the actual temperature to another one than the nominal temperature 20°. This value will be used to evaluate the exact dimensions and other temperature dependent properties.
■ Related units:
LengthUnit: affects the values given to the inner diameter, the outer diameter, the length, the position.
?FieldUnit, MaterialMagnetizationUnit..
■ See the Magnetica User Book: Chapter 7.

■ See also: `AXSMagneticSystem`, `AXSHardMagneticComponent`, `AXSSoftMagneticComponent`, `FillingFactor`, `MagnetizationAngle`, `RemanentMagnetization`, `Temperature`, `LengthUnit`, `FieldUnit`.

■ AXSSoftMagneticComponent

■ `AXSSoftMagneticComponent[`*material, nominalInnDiameter, nominalOutDiameter, nominalLength, position*`]` generates a cylindrical component (bar or ring) with isotropic reversible (linear or non-linear) magnetization.

■ This generates a Magnetica AXiSymmetric Component. It is a Ferromagnetic Component.

■ This component is a reversible magnetization device, that is no remanent magnetization without magnetic field excitation (unlike `AXSRigidMagneticComponent` and `AXSHardMagneticComponent`).

■ This component takes into account the variation (linear or non-linear) of the magnetization with the applied magnetizing field. In order to do this, the component is subdivided into cells, each with uniform magnetization.

■ The thinnes of the meshing can be adjusted by using the option `ComponentMeshing` (see below). Otherwise the values are determined by the system option `MeshingLevel`.

■ Argument 1, the material, must be choosen among the Magnetica Materials and can be any material except Permanent Magnet.

■ Argument 2, is the inner diameter of the component, that is 0 in case of a bar (nominal dimension).

■ Argument 3, is the outer diameter of the component (nominal dimension).

■ Argument 4, is the length of the component (nominal dimension).

■ Argument 5, is the position of the center of the component on the Z-axis (temperature independent).

■ Possible options:

`FillingFactor`: to specify the rate of magnetic material in the component. The default is 1.

`LinearPermeability` (`True` or `False`): to specify if the magnetization function is to be assumed linear. The default is `False`.

`ComponentMeshing`: allows to specify the number of radial cells and axial cells of the component.

`Temperature`: allows to set the actual temperature of the component to another one than the nominal temperature 20°. This value will be used to evaluate the exact dimensions and other temperature dependent properties.

■ Related units:

`LenghtUnit`: affects the values given to the inner diameter, the outer diameter, the length, the position.

■ See the Magnetica User Book: Chapter 8 and Chapter 9.

■ See also: `AXSMagneticSystem`, `AXSHardMagneticComponent`, `AXSRigidMagneticComponent`, `FillingFactor`, `LinearPermeability`, `ComponentMeshing`, `Temperature`, `LengthUnit`.

■ AXSSolenoid

■ `AXSSolenoid[`*material,nominalInnDiameter, nominalOutDiameter, nominalLength, position, numberTurns, intensity,* `FillingFactor->`*filling*`]` generates a standard solenoid.

■ `AXSSolenoid[`*material,nominalInnDiameter, nominalOutDiameter, nominalLength, position, numberTurns, intensity,* `ConductorDiameter->`*diameter*`]` a solenoid, made of round wire, specifying its diameter.

■ `AXSSolenoid[`*material,nominalInnDiameter, nominalOutDiameter, nominalLength, position, numberTurns, intensity,* `ConductorHeightWidth->`*{height,width}*`]` a solenoid, made of rectangular section wire, specifying the section dimensions.

■ `AXSSolenoid[`*material,nominalInnDiameter, nominalOutDiameter, nominalLength, position, numberTurns, intensity*`]` without giving information about the conductor suppose it is a superconducting wire, since for this kind of material dimensions are commercially fixed properties.

■ This generates a Magnetica AXiSymmetric Component.

■ This component is a solenoid, that is a current carrying winding of rectangular cross section.

■ AXSSolenoid is a constant current density winding. For current density varying with the radius (1/r) see AXSBitter-Solenoid.

■ Argument 1, the material, must be choosen among the Magnetica Materials and can be either a Current Carrying or a Supercondutor.

■ Argument 2, is the inner diameter, of the winding, (nominal dimension).

■ Argument 3, is the outer diameter, of the winding, (nominal dimension).

■ Argument 4, is the length of the solenoid, (nominal dimension).

■ Argument 5, is the position of the center of the solenoid on the Z-axis. (temperature independent).

■ Argument 6, is the number of turns of the winding, signed according to the standard orientation right or left hand side (+,-).

■ Argument 7, is the current intensity per conductor, signed according to the direction of the current (+,-).

■ Additionnal information on the conductor dimensions and shape can be specified by (only) one of the following option:

`FillingFactor`: a numeric value between 0 and 1, that specify the filling rate. This settles the average dimensions of the wire.

`ConductorDiameter`: supposes a circular section wire, and specifies its diameter.

`ConductorHeightWidth`: supposes a rectangular section wire, and specifies its dimensions.

■ If none of the previous options is specified, then the solenoid is supposed to be a superconducting solenoid, since dimensions are commercially fixed properties of superconductor materials.

■ Other possible option:

`Temperature`: allows to set the actual temperature to another one than the nominal temperature 20°. This value will be used to evaluate the exact dimensions and other temperature dependent properties.

■ Related units:

`LengthUnit`: affects the values given to the inner diameter, the outer diameter, the solenoid length, the position of the Z-axis, and also the values given to the options `ConductorDiameter`, and `ConductorHeightWidth`.

`CurrentIntensityUnit`: affects the value given to the current intensity.

■ See the Magnetica User Book: Chapter 3.

■ See also: `AXSMagneticSystem`, `AXSBitterSolenoid`, `FillingFactor`, `ConductorDiameter`, `ConductorHeightWidth`, `LengthUnit`, `CurrentIntensityUnit`.

■ CalculationMode

■ `CalculationMode` is an option for almost all field calculation functions and mappings, to punctually modify the calculation mode.

■ The default value is usually `Inherited`, that tells to take into account the global calculation mode.

■ The other possible values are: `VectorPotential`, `SphericalHarmonics` and `Automatic`.

■ Setting another value than `Inherited` forces punctually the use of this value as calculation mode.

■ See the Magnetica User Book: `Chapter 5`.

■ See also: `MagneticaCalculationMode`, `FieldPoint`, `VerboseFieldPoint`, `FieldMap`,

■ ComponentList

■ `ComponentList` is an option for some calculation functions used to define a sublist of components of a given system.

■ The list can be specified with numbers and/or intervals. Intervals are specified with sublist of two elements:

Example: `ComponentList->{1,4,5,6,7,10}` is equivalent to `ComponentList->{1,{4,7},10}`.

■ Other possible values are:

`All`: to select all possible components.

`None`: usually the default value, to avoid any specification.

■ See also: `FieldPoint`, `VerboseFieldPoint`, MagnetizationTable, MagnetizationCurves

■ ComponentMeshing

■ `ComponentMeshing` is an option for `AXSHardMagneticComponent` and `AXSSoftMagneticComponent`, that specifies the number of radial and axial cells to take for the meshing of the component.

■ `ComponentMeshing` must be given a list of two positive integers being respectively the number of radial cells and the number of axial cells as follows:

`ComponentMeshing->`*{nbRadialCells, nbAxialCells}*.

■ `ComponentMeshing->Automatic` tells to refer to value given to the system option `MeshingLevel`.

■ See the Magnetica User Book: Chapter 6, Chapter 7, Chapter 8.

■ See also: `AXSHardMagneticComponent`, `AXSSoftMagneticComponent`, `MeshingLevel`.

■ CondensedForm

■ `CondensedForm` is an option for most "`Verbose`" functions, and some other, that allows to have a more compact display.

■ In `VerboseFieldPoint` if the `ComponentList` option is used, then `CondensedForm` allows to display the infomations in an array form.

■ In the `MagnetizationTable` function, using the condensed form gives a better display for printing.

■ See: `VerboseFieldPoint`, `VerboseFluxArea`, `VerbosePotentialPoint`, `VerboseGradient-Point`, `MagnetizationTable`.

■ ConductorDiameter

■ `ConductorDiameter` is an option for `AXSCircularLoop` and `AXSSolenoid` that allows to specify the wire diameter value.

■ Using this option indicates that the component is made of round wire.

■ `ConductorDiameter` can be given any positive numeric value.

■ The `ConductorDiameter` value is expressed in the unit of `LengthUnit`.

■ See the Magnetica User Book: Chapter 2, Chapter 3.

■ See also: `AXSSolenoid`, `AXSCircularLoop`, `LengthUnit`.

■ ConductorHeightWidth

■ `ConductorHeightWidth` is an option for `AXSSolenoid` that allows to specify the dimensions of the rectangular wire.

■ Using this option indicates that the component is made of rectangular section wire.

■ `ConductorHeightWidth` must be given a list of two positive numeric values as follows:

`ConductorHeightWidth` -> *{height, width}.*

- The `ConductorHeightWidth` values are expressed in the unit of `LengthUnit`.
- See the Magnetica User Book: Chapter 3.
- See also: `AXSSolenoid`, `LengthUnit`.

■ ContourFieldMapContinuous

- `ContourFieldMapContinuous[`*system*, *{fieldfunction1, fieldfunction2,...}*`,Plane[`*point1, point2*`]]`
ContourFieldMap with the automatic distribution of points.

- For detailed information about arguments, options, units,… see `ContourFieldMap`.
- `ContourFieldMapContinuous` choose the appropriate distribution of points, using an internal *Mathematica* algorithm, to produce the best possible accurate display. However it can take a long time for complicated systems. For regular distribution see `ContourFieldMapPoints`.
- Setting the option `NumberOfPoints` to some value indicates the minimum number of points to use for the map.
- See: `ContourFieldMap`, `ContourFieldMapPoints`.

■ ContourFieldMap

- `ContourFieldMap[`*system*, *{fieldfunction1, fieldfunction2,...}*`,Plane[`*point1, point2*`]]` displays a contour plot of each field function, for the given system, on the rectangle defined by its two opposite corners *point1* and *point2*.

- The first argument is a magnetic system: that is the return value of `AXSMagneticSystem` or `CARMagnetic‚ System`.
- If the system is axisymmetric, the points in the directive `Plane`, can be specified either by 3D spatial cartesian coordinates {x,y,z}, or by their 2D axisymmetric coordinates {r,z} (However the two points must be of the same type). For cartesian systems, only cartesian coordinates are possible for points specifications.
- The second argument is the list of the functions to plot: the set of possible functions is different for axisymmetric systems and cartesian systems; it is also different if points specifications are 2D axisymmetric or 3D cartesian:
- In case of axisymmetric system and points described by 2D axisymmetric coordinantes possible functions:
`Br, Bz, Bm, Flux, Potential, GradZZ, GradRZ, GradZR, GradRZ, GradRR.`
- In case of axisymmetric system and points described by 3D cartesian coordinates, possible functions:
`Bx, By, Bz, Bm, Flux, PotentialX, PotentialY, GradZZ.`
- In case of cartesian system:
To be implemented…
- The third argument is the graphical directive `Plane[`*point1, point2*`]` that defines the space where the plot must be done. The points must have exactly two different coordinates (this defines a plane normal to one canonical axis).
- To know more about distribution of points see `ContourFieldMapContinuous` and `ContourFieldMap‚ Points`. `ContourFieldMap` is a shortcut for `ContourFieldMapContinuous`.
- Possible options:
`NumberOfPoints`: allows to choose the number of points to use for the drawing. The default is `Automatic`.
`CalculationMode`: the default value is `Inherited`, that refers to the global calculation mode. The other values coerce the local use of a specific mode and can be: `VectorPotential`, `SphericalHarmonics`, `Automatic`.
- Related units:
`LengthUnit`: to specify the unit of the points.
`FieldUnit, FluxUnit, PotentialUnit, GradientUnit`: to specify the output units.
- See the Magnetica User Book: Chapter 4, Chapter 5.
- See also: `ContourFieldMapContinuous, ContourFieldMapPoints, FieldMap, VectorFieldMap,`

AXSMagneticSystem, NumberOfPoints, CalculationMode, LengthUnit, FieldUnit, FluxUnit, PotentialUnit, GradientUnit.

■ ContourFieldMapPoints

■ ContourFieldMapPoints[*system*, *{fieldfunction1, fieldfunction2,...}*, Plane[*point1, point2*]] ContourField-Map with a regular distribution of points.

■ For detailed information about arguments, options, units,… see ContourFieldMap.

■ ContourFieldMapPoints choose a regularly spaced distribution between the limits of the display. This version of ContourFieldMap is usually faster than ContourFieldMapContinuous but could be less precise.

■ Setting the option NumberOfPoints to some value indicates the exact number of points to use for the map.

■ See: ContourFieldMap, ContourFieldMapContinuous.

■ ContributionType

■ ContributionType is an option for inductance functions, that allows to obtain the proper caracteristic, the ferromagnetic contribution or the total.

■ Possible values are Proper, Ferro, Total.

■ ContributionType->Proper gives the proper inductance information (self, mutual, flux, energy,…).

■ ContributionType->Ferro gives the contribution of all ferromagnetic components to the total inductance information asked for.

■ ContributionType®Total gives the sum of the intrinsic inductance information and the ferromagnetics contribution to the effective global inductance information.

■ See: SelfInductance, SelfFluxInductance, SelfMagneticEnergy,…

■ CurrentIntensityUnit

■ CurrentIntensityUnit is the option used to specify the unit of current intensity.

■ CurrentIntensityUnit can be used in MagneticaUserUnits for global settings, or in any function having a current intensity argument, for local setting.

■ It can also be used for display functions.

■ Possible value: Ampere.

■ This value can be preceded, as a separated word, by any prefix (Milli, Kilo, Micro,…).

■ See the Magnetica User Book: Appendix Units.

■ See also: MagneticaUserUnits, AXSCircularLoop, AXSSolenoid, AXSBitterSolenoid,

■ CurveType

■ CurveType is an option for MaterialMagnetizationCurves that specifies the type of curve to plot for non linear magnetization material.

■ Possible values are Experimental, LinearModel, Comparison.

■ With CurveType®Experimental, the experimental curve is drawn.

■ With CurveType®LinearModel, the approximate linear model is displayed.

■ With `CurveType®Comparison`, both curves are displayed.

■ See: `MaterialMagnetizationCurves`.

■ DrawingRange

■ `DrawingRange` is an option for `DrawSystem`-like functions, that allows to define the range of the drawing.

■ Possible values are `OriginCentered` and `Automatic`.

■ With `DrawingRange ® OriginCentered` the origin `{0,0}` or `{0,0,0}` is always set at the center of the display.

■ With `DrawingRange ® Automatic` the range are automatically determined to show the system in the middle of the display.

■ `DrawingRange` can also be used as an option for `DrawSHLimits`-like functions, and `DrawIsoField„ Lines`-like functions.

■ See: `DrawSystem`, `DrawSystem2D`, `DrawSystem3D`, `DrawSHLimits`, `DrawIsoFieldLines`.

■ DrawIsoFieldLines

■ `DrawIsoFieldLines[`*system*`, `*{fieldvalue1, fieldvalue2,...}*`]` draws, in the default 2D-plane, one isofield line for each *fieldvaluei*, around the magnetic system.

■ Another plane and/or another axes order see the suffixed version of `DrawIsoFieldLines`.

■ The first argument is a magnetic system: that is the return value of `AXSMagneticSystem` or `CARMagnetic„ System`.

■ The second argument is a list of magnetic field values. One line is drawn for each field value corresponding to all points where the magnitude of $\overset{\smile}{B}$ equals this value.

■ The field values are expressed in the unit of `FieldUnit`.

■ `DrawIsoFieldLines` is a polar plot. The plot is done by turning around the origin. The range are then specified by angles (see below the `AngularRange` option).

■ Only lines that are away from the system can be drawn.

■ Possible options are:

`NumberOfPoints`: allows to choose the number of points used for the plot. The default is `15`.

`AngularRange`: allows to select the angular range. The default is `{0,90}`.

`CalculationMode`: the default is `Inherited`, that refers to the global calculation mode. The other values coerce the local use of a specific mode and can be: `VectorPotential`, `SphericalHarmonics`, `Automatic`.

■ Related units:

`FieldUnit`: to specify the magnetic field values.

`LengthUnit`: to specify the unit of the output.

■ See the Magnetica User Book: Chapter 4.

■ See also: `DrawIsoFieldLinesZX`, `AXSMagneticSystem`, `NumberOfPoints`, `AngularRange`, `CalculationMode`, `FieldUnit`, `LengthUnit`.

■ DrawIsoFieldLinesZX

■ `DrawIsoFieldLinesZX[`*system*`, `*{fieldvalue1, fieldvalue2,...}*`]` same as DrawIsoFieldLines but specifying the point of view with the two last letters.

■ The possible functions are: `DrawIsoFieldLinesRZ, DrawIsoFieldLinesZR, DrawIsoField„ LinesXY, DrawIsoFieldLinesXZ, DrawIsoFieldLinesYX, DrawIsoFieldLinesYZ, DrawIso„ FieldLinesZX, DrawIsoFieldLinesZY`.

■ For more details about (arguments, options,…) see `DrawIsoFieldLines`.

■ See also: `DrawIsoFieldLines, AXSMagneticSystem`.

■ DrawSHLimits

■ `DrawSHLimits[`*system*`]` displays the spherical harmonics limits in the default 2D-view.

■ To specify another plane and/or another axes order see the suffixed version of `DrawSHLimits`.

■ The three regions: the convergent region, the spherical harmonics region and the Automatic region are displayed with different colors.

■ See `SHLimits` for more details about the meaning of these regions.

■ See the Magnetica User Book: Chapter 5.

■ See also: `DrawSHLimitsZX, SHLimits, VerboseSHLimits, AXSMagneticSystem`.

■ DrawSHLimitsZX

■ `DrawSHLimitsZX[`*system*`]` displays the spherical harmonic limits in the plane specified by the suffix.

■ All possible functions are: `DrawSHLimitsXY, DrawSHLimitsXZ, DrawSHLimitsYX, Draw„ SHLimitsYZ, DrawSHLimitsZX, DrawSHLimitsZY`.

■ Informations about the meaning of the dispayed regions are given with `SHLimits, DrawSHLimits`.

■ See: `DrawSHLimits, SHLimits`.

■ DrawSystem2D

■ `DrawSystem2D[`*system*`]` gives a 2-dimensional schematic drawing of the system, in the default plane.

■ To specify another plane and/or another axes order see the suffixed version of `DrawSystem2D`.

■ For more details (argument, options,…) see `DrawSystem2DXY,`….

■ See also: `DrawSystem2DZX, DrawSystem3D, DrawSystem, AXSMagneticSystem`.

■ DrawSystem2DZX,DrawSystem2DXY,…

■ `DrawSystem2DZX[`*system*`]` gives a 2-dimensional schematic drawing of the system, in the plane defined by the two last letters.

■ The possible functions are: `DrawSystem2DXY, DrawSystem2DYX, DrawSystem2DXZ, DrawSystem2„ DZX, DrawSystem2DYZ, DrawSystem2DZY`.

■ The suffix of `DrawSystem2D` defines the axis of the drawing, the first letter specifying the horizontal axe, and the second the vertical axe.

■ The argument is a magnetic system: the return value of `AXSMagneticSystem` or `CARMagneticSystem`.

■ Each component color depends on its type:

   : Current Carrying (loops and solenoids).

   : Rigid magnetic magnet.

   : Hard magnetic magnet.

: Soft magnetic component with reversible non linear material.

: Soft magnetic component with reversible linear material.

: Soft magnetic component with other material.

■ Possible options:

`ShowArrows`: allows to display the current direction for current carrying component and the magnetization direction for permanent magnet component. The default is `False`.

`ShowCentralAxes`: allows to display the z-axis. The default is `True`.

`ShowSphericalLimits`: allows to display the limits of valid area for spherical harmonics calculations. The default is `False`.

`DrawingRange`: allows to choose how to determine the range of the display. Default is `Automatic`.

■ See the Magnetica User Book: Chapter 2, Chapter 3….

■ See also: `DrawSystem2D, DrawSystem3D, DrawSystem, AXSMagneticSystem, ShowArrows, Show-CentralAxe, ShowSphericalLimits, DrawingRange`.

■ DrawSystem3D

■ `DrawSystem3D[`*system*`]` gives a 3-dimensional drawing of the system, with the default view.

■ To specify another view and/or another axes order see the suffixed version of `DrawSystem3D`.

■ For more details (argument, options,…) see `DrawSystem3DXYZ`….

■ See also: `DrawSystem3DZXY, DrawSystem2D, DrawSystem, AXSMagneticSystem,`

■ DrawSystem3DZXY,DrawSystem3DZX,…

■ `DrawSystem3DZXY[`*system*`]` gives a 3-dimensional drawing of the system, the axis being specified by the letters order of the suffix of DrawSystem3D.

■ The possible functions are: `DrawSystem3DXYZ, DrawSystem3DXZY, DrawSystem3DYXZ, Draw„System3DYZX, DrawSystem3DZXY, DrawSystem3DZYX, DrawSystem3DXY, DrawSystem3DXZ, DrawSystem3DYX, DrawSystem3DYZ, DrawSystem3DZX, DrawSystem3DZY`.

■ The suffix of DrawSystem3D defines the axes order and the view type.

■ The argument is a magnetic system: the return value of `AXSMagneticSystem` or `CARMagneticSystem`.

■ Each component color depends on its type:

: Current Carrying (loops and solenoids).

: Rigid magnetic magnet.

: Hard magnetic magnet.

: Soft magnetic component with reversible non linear material.

: Soft magnetic component with reversible linear material.

: Soft magnetic component with other material.

■ Possible options:

`ShowCentralAxe` (`True` or `False`): allows to display the z-axis. The default is `True`.

`DrawingRange`: allows to choose how to determine the range of the display. Default is `Automatic`.

■ See the Magnetica User Book: Chapter 2, Chapter 3….

■ See also: `DrawSystem, DrawSystem3D, DrawSystem2D,AXSMagneticSystem, ShowCentralAxe, DrawingRange`.

■ **DrawSystem**

- `DrawSystem[`*system*`]` gives a drawing of the system, with the default view.

- See `DrawSystem2D` and `DrawSystem3D` for more details about arguments, options, views, axes order,…
- `DrawSystemZXY`-like functions are shortcuts to `DrawSystem3DZXY`-like ones.
- `DrawSystemZX`-like functions are shortcuts to `DrawSystem2DZX`-like ones.
- See also: `DrawSystem3DZXY`, `DrawSystem2DZX`, `AXSMagneticSystem`.

■ **MagneticEnergyMatrix**

- `MagneticEnergyMatrix[`*system*`]` returns a summary of all self and mutual magnetic energies of the current carrying components of the magnetic system.

- The only mandatory argument is a magnetic system.
- The option `ComponentsList` allows to select the component involved in the display.
- The output is in the form of a list of two elements:
the first is a matrix given as a list of list, summarizing the self and mutual magnetic energies.
the second is the list of the involved component numbers.
- Possible options:
`ComponentsList`: allows to select the list of components to take into account. The default value is `All`.
`ContributionType`: allows to compute the `Proper` magnetic energies (default), the ferromagnetic contribution (`Ferro`) to the magnetic energies or the `Total`.
`PrecisionLevel`: specifies the precision level among 1, 2 or 3. Default 1.
- Related units:
`MagneticEnergyUnit`: to specify the unit of the output.
- See the Magnetica User Book: Chapter 10, Chapter 11.
- See also: `VerboseMagneticEnergyMatrix`, `SelfMagneticEnergy`, `MutualMagneticEnergy`, `AXSMagneticSystem`, `AXSSolenoid`, `AXSCircularLoop`, `ContributionType`, `PrecisionLevel`, `MagneticEnergyUnit`.

■ **Extended**

- `Extended` is a possible option value for `MagnetizingFieldRange`.

- `Extended` sets the range to include the region where the magnet material has an unstable behavior.
- See: `Operational`, `HysteresisCycle`, `MagnetizingFieldRange`.

■ **FieldMapContinuous**

- `FieldMapContinuous[`*system*`, {`*fieldfunction1, fieldfunction2,…*`},` `Line[`*point1, point2*`]]` FieldMap on a line, with the automatic distribution of points.
- `FieldMapContinuous[`*system*`, {`*fieldfunction1, fieldfunction2,…*`},` `Plane[`*point1, point2*`]]` FieldMap on a plane with the automatic distribution of points.

- For detailed information about arguments, options, units,… see `FieldMap`.
- `FieldMapContinuous` choose the appropriate distribution of points, using an internal *Mathematica* algorithm, to produce the best possible accurate display. However it can take a long time for complicated systems. For regular

distribution see `FieldMapPoints`.
- Setting the option `NumberOfPoints` to some value indicates the minimum number of points to use for the map.
- See: `FieldMap`, `FieldMapPoints`.

■ FieldMap

- `FieldMap[`*system*, *{fieldfunction1, fieldfunction2,...}*, `Line[`*point1, point2*`]]` plots each of the field function, for the given system, on the line defined by *point1* and *point2*.
- `FieldMap[`*system*, *{fieldfunction1, fieldfunction2,...}*, `Plane[`*point1, point2*`]]` plots each of the field function, for the given system, on the rectangle defined by the two opposite corners *point1* and *point2*.

- The first argument is a magnetic system: that is the return value of `AXSMagneticSystem` or `CARMagnetic` `System`.
- If the system is axisymmetric, the points in the directives `Line` or `Plane`, can be specified either by 3D spatial cartesian coordinates {x,y,z}, or by their 2D axisymmetric coordinates {r,z} (However the two points must be of the same type). For cartesian systems, only cartesian coordinates are possible for points specifications.
- The second argument is the list of the functions to plot: the set of possible functions is different for axisymmetric systems and cartesian systems; it is also different if points specifications are 2D axisymmetric or 3D cartesian:
- In case of axisymmetric system and points described by 2D axisymmetric coordinantes possible functions:
`Br`, `Bz`, `Bm`, `Flux`, `Potential`, `GradZZ`, `GradRZ`, `GradZR`, `GradRZ`, `GradRR`.
- In case of axisymmetric system and points described by 3D cartesian coordinates, possible functions:
`Bx`, `By`, `Bz`, `Bm`, `Flux`, `PotentialX`, `PotentialY`, `GradZZ`.
- In case of cartesian system:
To be implemented…
- The third argument is a graphical directive that defines the space where the plot must be done:
Specifying `Line[`*point1, point2*`]` plots the functions on the line between the two points. The output is then a list of 2D graphics, one for each function asked for.
Specifying `Plane[`*point1, point2*`]` plots the functions on the plane defined by its corners. The plane must be normal to one of the canonical axis (this happens when the two points have exactly two different coordinates). The output is then a list of 3D graphics, one for each function asked for.
- See `FieldMapContinuous` and `FieldMapPoints` to know more about distribution of points. `FieldMap` is a shortcut for `FieldMapContinuous`.
- Possible options:
`NumberOfPoints`: allows to choose the number of points to use for the drawing. The default is `Automatic`. (25 for 2D graphics and 15 for 3D graphics).
`CalculationMode`: the default value is `Inherited`, that refers to the global calculation mode. The other values coerce the local use of a specific mode and can be: `VectorPotential`, `SphericalHarmonics`, `Automatic`.
`PlotScale`: allows, for 2D graphics only, to choose a linear or a logarithmic scale on each of the axis. The default is `LinearMap`.
- Related units:
`LengthUnit`: to specify the unit of the points.
`FieldUnit`, `FluxUnit`, `PotentialUnit`, `GradientUnit`: to specify the output units.
- See the Magnetica User Book: Chapter 4, Chapter 5.
- See also: `FieldMapContinuous`, `FieldMapPoints`, `ContourFieldMap`, `VectorFieldMap`, `AXSMagneticSystem`, `NumberOfPoints`, `CalculationMode`, `PlotScale`, `LengthUnit`, `FieldUnit`, `FluxUnit`, `PotentialUnit`, `GradientUnit`.

■ **FieldMapPoints**

■ `FieldMapPoints[`*system, {fieldfunction1, fieldfunction2,...}*, `Line[`*point1, point2*`]]` FieldMap on a line, with a regular distribution of points.

■ `FieldMapPoints[`*system, {fieldfunction1, fieldfunction2,...}*, `Plane[`*point1, point2*`]]` FieldMap on a plane with a regular distribution of points.

■ For detailed information about arguments, options, units,… see `FieldMap`.

■ `FieldMapPoints` choose a regularly spaced distribution between the limits of the display. This version of Field-Map is usually faster than `FieldMapContinuous` but could be less precise.

■ Setting the option `NumberOfPoints` to some value indicates the exact number of points to use for the map.

■ See: `FieldMap, FieldMapContinuous`.

■ **FieldPoint, FieldPointVP, FieldPointSH**

■ `FieldPoint[`*system,targetPoint*`]` gives the magnetic field generated by the system, at the target point, according to the global calculation mode.

■ `FieldPointVP[`*system,targetPoint*`]` gives the magnetic field, using the Vector Potential method for calculation.

■ `FieldPointSH[`*system,targetPoint*`]` gives the magnetic field, using the Spherical Harmonics method for calculation.

■ The first argument is a magnetic system: that is the return value of AXSMagneticSystem or CARMagneticSystem

■ The second argument defines the target point. If the magnetic system is axis symmetric, the target point can be specified either by its 3D spacial cartesian coordinates {*x, y, z*} or by its 2D axisymmetric coordinates {*r, z*}. For cartesian systems, only cartesian coordinates are possible for the target point.

■ The output :

 if the target point is specified with *{r,z}* then the output is in the form *{Br,Bz}*

 if the target point is specified with *{x,y,z}*, then the output is in the form *{Bx, By, Bz}*.

■ `FieldPoint` (without any suffix) computes the magnetic field, according to the global calculation mode. (see `MagneticaCalculationMode`).

■ Possible options are the following:

`ComponentList`: allows to see the contribution of each component specified in the list to the magnetic field value. The output is in the form *{componentNumbers, {fieldpoint1,fieldpoint2,...}, total}*.

`CalculationMode`: the default value is `Inherited`, that refers to the global calculation mode. The other values coerce the local use of a specific mode and can be: `VectorPotential, SphericalHarmonics, Automatic.`

■ Related units are:

`LengthUnit`: to specify the unit of the target point.

`FieldUnit`: to specify the unit of the output.

■ See the Magnetica User Book: `Chapter 4, Chapter 5.`

■ See also: AXSMagneticSystem, ComponentList, CalculationMode, LengthUnit, FieldUnit, VerboseFieldPoint.

■ **FieldPointOrigin**

■ `FieldPointOrigin[`*system*`]` is a shortcut to `VerboseFieldPoint` at the origin `{0,0,0}`.

- `FieldPointOrigin` can be given any of the `VerboseFieldPoint` options.
- See: `VerboseFieldPoint`.

■ **FieldUnit**

- `FieldUnit` is the option used to specify the magnetic field unit.

- `FieldUnit` can be used in `MagneticaUserUnits` for a global setting, or in any function concerned with field unit, for local setting.
- Possible values are: `Gauss, Tesla, Ampere/Meter, Oersted`.
- These values can be preceded, as a separated word, by any prefix (`Milli, Kilo, Micro`,…).
- See the Magnetica User Book: Appendix Units.
- See also: `MagneticaUserUnits, MagneticaCurrentUnits, FieldPoint, VerboseFieldPoint,`

■ **FilamentMode**

- `FilamentMode` is an option for `AXSCircularLoop` that allows to consider the current concentrated at the center of the loop.

- Possible values are `True` or `False`.
- `FilamentMode->True`, the loop is seen as a wire of null radius, that implies an infinite current density. The component is then called a filament loop.
- `FilamentMode->False`, the loop is seen as a real size winding. The component is then called tubular loop.
- See the Magnetica User Book: Chapter 4.
- See also: `AXSCircularLoop`.

■ **FillingFactor**

- `FillingFactor` is an option of most of component functions, that allows to specify the filling rate of conductor or magnetic material.

- Possible values are generally any real number between 0 and 1.
- For component made of wire, it is often impossible to have values greater than $\frac{\pi}{4}$.
- See the Magnetica User Book: Chapter 3, Chapter 7, Chapter 8.
- See also: `AXSSolenoid, AXSCircularLoop, AXSRigidMagneticComponent, AXSHardMagnetic-Component, AXSSoftMagneticComponent`.

■ **FluxArea, FluxAreaVP, FluxAreaSH**

- `FluxArea[`*axssystem*`, `*targetPoint*`]` returns the flux through the area bordered by the circle lying in the plane normal to the z-axis, centered on the z-axis and containing the specified point.
- `FluxAreaVP[`*axssystem*`, `*targetPoint*`]` returns the flux, using the vector potential method for calculation.
- `FluxAreaSH[`*axssystem*`, `*targetPoint*`]` returns the flux, using the spherical harmonics method for calculation.

- The first argument is an axis symmetric system: that is the return value of `AXSMagneticSystem`.
- The second argument defines the target point. It can be specified either by its 3D spatial cartesian coordinates *{x, y, z}* or by its 2D axisymmetric coordinates *{r, z}*.
- The output is in the form of a single element list *{fluxvalue}*.

■ `FluxArea` (without suffix) computes the flux, according to the global calculation mode. (see `Magnetica-CalculationMode`).

■ Possible options are the following:

`ComponentList`: allows to see the contribution of each component, in the specified list, to the flux value. The output then becomes in the form: *{componentsNumber, {flux1, flux2,…}, total}*. The default value is `None`.

`CalculationMode`: the default value is `Inherited`, from the global calculation mode. The other possible values are `VectorPotential`, `SphericalHarmonics` or `Automatic`.

■ Related units are:

`LengthUnit`: to specify the unit of the target point.

`FluxUnit`: to specify the unit of the output.

■ See the Magnetica User Book: `Chapter 4`, `Chapter 5`.

■   See also: `AXSMagneticSystem`, `ComponentList`, `CalculationMode`, `LengthUnit`, `FluxUnit`, `VerboseFluxArea`.

■ FluxInductanceMatrix

■ FluxInductanceMatrix[system] returns a summary of all self flux inductances and mutual inductances of the current carrying components of the magnetic system.

■ The only mandatory argument is a magnetic system.

■ The option `ComponentsList` allows to select the component involved in the display.

■ The output is in the form of a list of two elements:

the first is a matrix given as a list of list, summarizing the self and mutual flux inductances.

the second is a list of the involved component numbers.

■ Possible options:

`ComponentsList`: allows to select the list of components to take into account. The default value is `All`.

`ContributionType`: allows to compute the `Proper` flux inductances (default), the ferromagnetic contribution (`Ferro`) to the flux inductances or the `Total`.

`PrecisionLevel`: specifies the precision level among 1, 2 or 3. Default 1.

■ Related units:

`FluxUnit`: to specify the unit of the output.

■ See the Magnetica User Book: Chapter 10, Chapter 11.

■ See also: `VerboseFluxInductanceMatrix`, `SelfFluxInductance`, `MutualFluxInductance`, `AXSMagneticSystem`, `AXSSolenoid`, `AXSCircularLoop`, `ContributionType`, `PrecisionLevel`, `FluxUnit`.

■ FluxUnit

■ `FluxUnit` is the option used to specify the magnetic flux unit.

■ `FluxUnit` can be used in `MagneticaUserUnits` for a global setting, or in any function concerned with flux unit, for local setting.

■ Possible values: `Weber, Maxwell`.

■ These values can be preceded, as a separated word, by any prefix (`Milli, Kilo, Micro,` …).

■ See the Magnetica User Book: Appendix Units.

■ See also: `MagneticaUserUnits, MagneticaCurrentUnits, FluxArea, VerboseFluxArea,`

▪ GradientPoint, GradientPointVP, GradientPointSH

▪ GradientPoint[*axssystem, targetPoint*] returns the field gradient, of the system, at the the specified target point.

▪ GradientPointVP[*axssystem, targetPoint*] returns the gradient, using the vector potential method for calculation.

▪ GradientPointSH[*axssystem, targetPoint*] returns the gradient, using the spherical harmonics method for calculation.

▪ The first argument is an axis symmetric system: that is the return value of AXSMagneticSystem.

▪ The second argument defines the target point. It can be specified either by its 3D spatial cartesian coordinates *{x, y, z}* or by its 2D axisymmetric coordinates *{r, z}*.

▪ The output is in the form of a list of list (that can be displayed as a matrix with the MatrixForm command): *{{∂Bz/∂z,∂Bz/∂r}, {∂Br/∂z,∂Br/∂r}}*.

▪ GradientPoint (without suffix), computes the gradient according to the global calculation mode. (see MagneticaCalculationMode).

▪ Possible options are the following:

ComponentList: allows to see the contribution of each component, in the specified list, to the gradient value. The output then becomes in the form: *{componentsNumber,{gradient1,gradient2,...},total}*.

CalculationMode: the default value is Inherited, from the global calculation mode. The other possible values are VectorPotential, SphericalHarmonics, or Automatic.

▪ Related units are:

LengthUnit: to specify the unit of the target point.

GradientUnit: to specify the unit of the output.

▪ See the Magnetica User Book: Chapter 4, Chapter 5.

▪ See also: AXSMagneticSystem, ComponentList, CalculationMode, LengthUnit, GradientUnit, VerboseGradientPoint.

▪ GradientUnit

▪ GradientUnit is the option used to specify the magnetic gradient unit.

▪ GradientUnit can be used in MagneticaUserUnits for a global setting, or in any function concerned with potential unit, for local setting.

▪ Possible values: Gauss/Meter, Tesla/Meter,...

▪ These values can be preceded, as a separated word, by any prefix (Milli, Micro,...).

▪ See the Magnetica User Book: Appendix Unit.

▪ See also: MagneticUserUnits, MagneticaCurrentUnits, GradientPoint, VerboseGradient-Point,...

▪ HysteresisCycle

▪ HysteresisCycle is an option value for MagnetizingFieldRange.

▪ HysteresisCycle allows to draw one hysteresis cycle simulation for the given permanent magnet material.

▪ See: MagnetizingFieldRange, Operational, Extended.

■ InductanceCoupling

■ `InductanceCoupling[`*system, {component1, component2}*`]` returns informations related to inductance coupling between *component1* and *component2*, in the magnetic system

- The first argument is a magnetic system.
- The components are specified by their rank in the magnetic system.
- The second argument is a list of two numbers, the first corresponding to the source component, and the second to the target.
- `InductanceCoupling` can be only calculated between current carrying components.
- The output is in the following form:

{{coupling coef air core, coupling coef ferro core},
  {efficiency coef air core, efficiency coef ferro core},
  {leakage coef air core, leakage coef ferro core},
  {transformation coef air core, transformation coef ferro core}}.

- Possible options:

`PrecisionLevel`: specifies the precision level among 1, 2 or 3. Default 1.

- See the Magnetica User Book: Chapter 10, Chapter 12, Chapter 13.
- See also: `VerboseInductanceCoupling`, `AXSMagneticSystem`, `PrecisionLevel`.

■ InductanceMatrix

■ `InductanceMatrix[`*system*`]` returns a summary of all self inductances and mutual inductances of the current carrying components of the magnetic system.

- The only mandatory argument is a magnetic system.
- The option `ComponentsList` allows to select the component involved in the display.
- The output is in the form of a list of two elements:

the first is a matrix given as a list of list, summarizing the self and mutual inductances.

the second is the list of the involved component numbers.

- Possible options:

`ComponentsList`: allows to select the list of components to take into account. The default value is `All`.

`ContributionType`: allows to compute the `Proper` inductances (default), the ferromagnetic contribution (`Ferro`) to the inductances or the `Total`.

`PrecisionLevel`: specifies the precision level among 1, 2 or 3. Default 1.

- Related units:

`InductanceUnit`: to specify the unit of the output.

- See the Magnetica User Book: Chapter 10, Chapter 11.
- See also: `VerboseInductanceMatrix`, `SelfInductance`, `MutualInductance`, `AXSMagnetic-System`, `AXSSolenoid`, `AXSCircularLoop`, `ContributionType`, `PrecisionLevel`, `InductanceUnit`.

■ InductanceUnit

■ `InductanceUnit` is the option used to specify the inductance unit.

■ `InductanceUnit` can be used in `MagneticaUserUnits` for a global setting, or in any function concerned with inductance calculations, for local setting.

- Possible values: `Henry`.
- These values can be preceded, as a separated word, by any prefix (`Milli, Micro`, …).
- See the Magnetica User Book: Appendix Units.
- See also: `MagneticaUserUnits, MagneticaCurrentUnits, SelfInductance, MutualInductance, InductanceMatrix`.

■ Irrelevant

- `Irrelevant` is a symbol returned by some functions to indicate that the computation asked for is meaningless.

- Example: Ferro inductance for a system that does not contain any ferromagnetic component.

■ LegendreCoefficients

- `LegendreCoefficients[`*system*`]` gives the Legendre source coefficients of the system.

- The output is in the form *{centralcoefficients, remotecoefficients}*.
- The limits of the central region and the remote region are given by `SHLimits`, and can be shown with `Draw„ SHLimits`.
- The `LegendreCoefficients` are very dependent on the value of `LegendrePointPosition` given to the system.
- The first coefficient term is the 0 order coefficient and the other terms are sorted up to the 20th order (or less if the `LegendreOrder` was set for the system).
- The first coefficient of the remote region effectively starts at order 2. By convention the orders 0 and 1 are set to 0.
- The coefficients are expressed in the unit of `FieldUnit`.
- To obtain a more detailed output try `VerboseLegendreCoefficients`.
- Possible options are:
`LegendreRadius`: allows to specify the Legendre Coefficients for alternative Legendre radius. It is possible to specify a single value or a list of 2 values for the central and the remote coefficients. The default is `Automatic`.
`LegendreMode`: allows to specify if the Legendre coefficients are expressed as absolute values or relative values. The default is `AbsoluteMode`.
- Related options:
`FieldUnit`: to specify the unit of the output.
`LengthUnit`: to specify the unit of `LegendreRadius`.
- See the Magnetica User Book: Chapter 5.
- See also: `VerboseLegendreCoefficients, SHLimits, DrawSHLimits, AXSMagneticSystem, LegendrePointPosition, LegendreRadius, LegendreMode, FieldUnit, LengthUnit,`

■ LegendreMode

- `LegendreMode` is an option for `LegendreCoefficients` and `VerboseLegendreCoefficients`, that allows to display the absolute coefficients values or the relative coefficients values.

- Possible values are `AbsoluteMode` or `RelativeMode`.
- With `LegendreMode®AbsoluteMode` the absolute values coefficients are displayed.
- With `LegendreMode®RelativeMode` the coefficients are computed relatively to the first coefficient.
- See: `LegendreCoefficients, VerboseLegendreCoefficient`.

■ **LegendreOrder**

■ `LegendreOrder` is an option for `AXSMagneticSystem`, that determines the number of terms to use in spherical harmonics calculations.

■ `LegendreOrder` must be given an integer value between 2 and 20.
■ The default setting is the maximum 20.
■ See the Magnetica User Book: Chapter 5.
■ See also: `AXSMagneticSystem`.

■ **LegendrePointPosition**

■ `LegendrePointPosition` is an option for `AXSMagneticSystem`, that allows to choose the position of the Legendre point on the Z-axis.

■ The `LegendrePointPosition` can be given any signed numeric value.
■ The position of the Legendre point is taken into account in definition of the SHLimits, LegendreCoefficients, and in all field calculations using the `SphericalHarmonics` mode.
■ See the Magnetica User Book: Chapter 5.
■ See also: `AXSMagneticSystem`, `SHLimits`, `LegendreCoefficients`, `VerboseSHLimits`, `Verbose-LegendreCoefficients`.

■ **LegendreRadius**

■ `LegendreRadius` is an option for `AXSMagneticSystem`, `LegendreCoefficients` and `Verbose„LegendreCoefficients` that allows to specifies the Legendre radius.

■ As an option of `AXSMagneticSystem`, it specifies the Legendre radius value for this system that will be taken into account in all calculations. Only a single value (or `Automatic`) is possible.
■ With `Automatic`, the radius is choosen to be in good relation with the real size of the system.
■ As an option of `LegendreCoefficients` or `VerboseLegendreCoefficients` it allows to compute the Legendre coefficients with alternative radii than the one given to the system. A single value or a list of 2 values are possible.
■ The `LegendreRadius` value is expressed in the unit of `LengthUnit`.
■ See the Magnetica User Book: Chapter 5.
■ See also: `AXSMagneticSystem`, `LegendreCoefficients`, `VerboseLegendreCoefficients`, `LengthUnit`.

■ **LengthUnit**

■ `LengthUnit` is the option used to specify the unit of all length type values (dimensions, coordinates,…).

■ `LengthUnit` can be used in `MagneticaUserUnits` for global settings, or in most functions concerned with length unit, for local setting.

■ Possible values are: `Meter`, `Inch`, `Foot`, `Feet`,...

■ These values can be preceded by any prefix (`Milli`, `Kilo`, `Micro`,...).

■ See the Magnetica User Book: Appendix Units.

■ See also: `MagneticaUserUnits`.

■ LinearLogMap

■ `LinearLogMap` is an option value for `PlotScale`.

■ Sets a linear scale on the horizontal axe, and a logarithmic scale on the vertical axe.

■ See `PlotScale`.

■ LinearMap

■ `LinearMap` is an option value for `PlotScale`.

■ Sets a linear scale on both horizontal and vertical axes.

■ See `PlotScale`.

■ LinearPermeability

■ `LinearPermeability` is an option for `AXSSoftMagneticComponent`, that allows to force the magnetization function to be linear.

■ Possible values are `True` or `False`.

■ `LinearPermeability` value has an effect only on components made of non-linear magnetization material.

■ In this case, setting `Permeability->True` tells to use the linear model of the magnetization function that can be shown by MaterialMagnetizationCurves with the appropriate option.

■ See the Magnetica User Book: Chapter 8.

■ See also: `AXSSoftMagneticComponent`, `MaterialMagnetizationCurves`.

■ LogLinearMap

■ `LogLinearMap` is an option value for `PlotScale`.

■ Sets a logarithmic scale on the horizontal axe, and a linear scale on the vertical axe.

■ See `PlotScale`.

■ LogLogMap

■ `LogLogMap` is an option value for `PlotScale`.

■ Sets a logarithmic scale on both horizontal and vertical axes.

■ See `PlotScale`.

■ **MagneticaCurrentUnits**

■ `MagneticaCurrentUnits[]` displays the units that are currently taken into account.

■ Each unit has a current setting either the one specified by the user, or the default one.

■ See the Magnetica User Book: Appendix Units.

■ See also: `MagneticaUserUnits`.

■ **MagneticaUserUnits**

■ `MagneticaUserUnits[`*unitSpecifications*`]` overrides the currently used Magnetica units with those specified.

■ `MagneticaUserUnits[]` resets all units to their default values.

■ Specifications are done as options as in the following example:
`MagneticaUserUnits[LengthUnit->Micro Meter, FieldUnit->Milli Gauss]`.

■ A unit specification is composed with a unit type (LengthUnit, FieldUnit,...) on the left of the arrow and a unit or unit value composed with an eventual prefix (Milli, Micro, Kilo,...) a blank (a space or a *) and unit name (Gauss, Meter,...).

■ See each unit type to get the possible unit values.

■ The possible units are: `LengthUnit, FieldUnit, PotentialUnit, FluxUnit, GradientUnit,` `InductanceUnit, MagneticEnergyUnit, MagneticMomentUnit, CurrentIntensityUnit,` `PowerUnit, VoltageUnit, ResistanceUnit`

■ Most common prefixes are:
`Tera` (10^12), `Giga` (10^9), `Mega` (10^6), `Kilo` (10^3),
`Deci` (10^-1), `Centi` (10^-2), `Milli` (10^-3), `Micro` (10^-6), `Nano` (10^-9), `Pico` (10^-12),

■ See the Magnetica User Book: Appendix Units.

■ See also: `MagneticaCurrentUnits, LengthUnit, FieldUnit, PotentialUnit, FluxUnit,` `GradientUnit, InductanceUnit, MagneticEnergyUnit, MagneticMomentUnit, Current-` `IntensityUnit, PowerUnit, VoltageUnit, ResistanceUnit`,

■ **MagneticEnergyUnit**

■ `MagneticEnergyUnit` is the option used to specify the magnetic energy unit.

■ `MagneticEnergyUnit` can be used in `MagneticaUserUnits` for a global setting, or in any function concerned with magnetic energy calculations, for local setting.

■ Possible values: `Joule`,

■ These values can be preceded, as a separated word, by any prefix (`Milli, Kilo, Micro`,...).

■ See the Magnetica User Book: Appendix Unit.

■ See also: `MagneticUserUnits, MagneticaCurrentUnits, SelfMagneticEnergy, MutualMagnetic-` `Energy, MagneticEnergyMatrix`,...

■ **MagnetizationAngle**

■ `MagnetizationAngle` is an option for `AXSRigidMagneticComponent` and `AXSHardMagnetic` `Component`, that allows to set the angle between the oriented Z-axis and the magnetization direction.

■ `MagnetizationAngle` can be given any numeric value.

■ The `MagnetizationAngle` value is expressed in Degree.

- The default value is 0.
- See the Magnetica User Book: Chapter 6, Chapter 7.
- See also: `AXSRigidMagneticComponent`, `AXSHardMagneticComponent`.

■ **MagnetizationCurves**

■ `MagnetizationCurves[`*system, functions*`]` plots curves of magnetization functions according to the magnetization state of the system.

- The first argument is a magnetic system.
- The second argument is the function to plot or a list of the functions to plot.
- `MagnetizationCurves` displays the magnetization state of the Cells of Ferromagnetic Components (AXSHardMagneticComponent and AXSSoftMagneticComponent).
- When a magnetic system containing components with cells is evaluated, Magnetica computes the magnetization state of all the cells in this system. The `MagnetizationCurves` function intend to display this computed magnetization state.
- The plot is done by examining the component(s) cell by cell, scanning the cross section either radially or axially and by selecting the rows or columns to plot.
- For Hard Components (`AXSHardMagneticComponent`) the possible functions are:

`Mr, Mz, Mm, Mp, Mt, Hr, Hz, Hm, Hp, Ht, Br, Bz, Bm, Bp, Bt, Susceptibility, SusceptibilityT, Permeability, PermeabilityT.`

- For Soft Components (`AXSSoftMagneticComponent`) the relevant functions are only:

`Mr, Mz, Mm, Hr, Hz, Hm, Br, Bz, Bm, Susceptibility, Permeability.`

- The meanings of these functions is described in the following:

`Mr, Mz, Mm`: to get the magnetization of the cells respectively in the Radial direction, in the Axial direction or the Magnitude.

`Hr, Hz, Hm`: to get the magnetizing field at the center of the cells respetively in the Radial direction, in the Axial direction or the Magnitude.

`Br, Bz, Bm`: to get the induction field (H+M) in the cells respectively in the Radial direction, in the Axial direction or the Magnitude.

`Mp, Mt`: to get the magnetization in the Principal and Transversal directions.

`Hp, Ht`: to get the magnetizing field in the Principal and Transversal directions.

`Bp, Bt`: to get the induction field in the Principal and Transversal directions

`Susceptibiliy, SusceptibilityT`: to get the susceptibility in the Principal and the Transversal directions.

`Permeability, PermeabilityT`: to get the permeability in the Principal and the Transversal directions.

- Possible options:

`ComponentsList`: allows to select a list of components to take into account for the plot. The default is `{1}`.

`PlotRows`: allows to scan the component(s) in the axial direction and to specify the rows to plot. This option cannot be used with `PlotColumns`.

`PlotColumns`: allows to scan the component(s) in the radial direction and to specify the columns to plot. This option cannot be used with `PlotRows`.

`ShowComponentsLimits`: when many components are ploted together this option allows to display the limits of the components. Default is `True`.

- See the Magnetica User Book: Chapter 7, Chapter 8, Chapter 9.
- See also: `MagnetizationTable, ComponentsList, AXSMagneticSystem, AXSHardMagneticComponent, AXSSoftMagneticComponent.`

■ **MagnetizationTable**

■ `MagnetizationTable[system]` displays a table summerazing the magnetization state of the cells of the magnetic components of the system.

■ The first argument is a magnetic system.

■ `MagnetizationTable` is only concerned by components with Cells (`AXSHardMagneticSystem` and `AXSSoftMagneticSystem`).

■ When a magnetic system containing components with cells is evaluated, *Magnetica* computes the magnetization state of all the cells in this system. The `MagnetizationTable` function intend to display all relevant informations describing these states.

■ Possible options:

`ComponentsList`: allows to select the component(s) to take into account. Default is `All`.

`CellsList`: allows to select more specifically the cells to display. Default is `All`.

`FullDetails`: to get more or less details. Default is `False`.

`CondensedForm`: to get a compact display. Default `True`.

■ See the Magnetica User Book: Chapter 7, Chapter 8, Chapter 9.

■ See also: `MagnetizationCurves`, `ComponentsList`, `AXSMagneticSystem`, `AXSHardMagnetic-Component`, `AXSSoftMagneticComponent`.

■ **MagnetizingFieldRange**

■ `MagnetizingFieldRange` is an option for `MaterialMagnetizationCurves` and `Material‚MagnetizationTable` that allows to set the range for the magnetizing field.

■ The values are set as *{value1, value2}* where the *valuei* are numeric values.

■ `MagnetizingFieldRange` values are expressed in the unit of `MagnetizingUnit` if it is set, and in the unit of `FieldUnit` otherwise.

■ For permanent magnet material, other possible values are `Operational`, `Extended`, `HysteresisCycle` or `Automatic`.

■ For material from other category, only `Automatic` is possible.

■ See the Magnetica User Book: Appendix Materials.

■ See also: `MaterialMagnetizationCurves`, `MaterialMagnetizationTable`, `Operational`, `Extended`, `HysteresisCycle`, `MagnetizingUnit`, `FieldUnit`.

■ **MaterialMagnetizationCurves**

■ `MaterialMagnetizationCurves[`*material*`, {`*magnetizationFunction1,magnetizationFunction2,...*`}]` plots, for the given material, the magnetization curve of each of the function of *magnetizationFunctioni*.

■ The first argument is a Magnetica material name, see the palette...

■ The second argument is the list of the functions to plot.

■ If the material is a permanent magnet then the possible functions are:

`Mprincipal`, `Bprincipal`, `Mtransversal`, `Btransversal`.

■ If the material is from another category, the possible functions are:

`M`, `B`, `Susceptibility`, `Permeability`, `Reluctivity`.

■ Possible options are:

`MagnetizingFieldRange`: allows to determine the range for the magnetizing field. The default is `Automatic`.

`Temperature`: allows to have the magnetization informations of the material at a specified temperature. The default is `Automatic` (nominal temperature).

`NumberOfPoints`: allows to choose the number of points for the plot. Default is `15`.

`PlotScale`: allows to choose a linear or logarithmic scale on each of the axis.

`CurveType`: For non linear reversible magnetization materials, allows to have the experimental curves, the linear model, or both. Default is `Experimental`.

■ Related Units:

`MagnetizingUnit`: allows to specifically set the unit of the magnetizing field on the horizontal axis of the graphic.

`MagnetizationUnit`: allows to specifically set the unit of `M`,`Mprincipal`,`Mtranversal`.

`InductionUnit`: allows to specifically set the unit of `B`,`Bprincipal`,`Btransversal`.

`FieldUnit`: is the unit that is taken into account when the previous specific units are not set.

■ See the MagneticaUser Book: Appendix Materials.

■ See also: `MaterialMagnetizationTable`, `Properties`, `MagnetizingFieldRange`, `Temperature`, `NumberOfPoints`, `PlotScale`, `CurveType`, `FieldUnit`, `MagnetizingUnit`, `MagnetizationUnit`, `InductionUnit`.

■ MaterialMagnetizationTable

■ `MaterialMagnetizationTable[`*material*`]` gives a table form of the informations used by `Material`„ `MagnetizationCurves`.

■ The first argument is a Magnetica material name, see the palette…

■ Each relevant magnetization function are displayed in a column.

■ Possible options are:

`MagnetizingFieldRange`: allows to determine the range for the magnetizing field. The default is `Automatic`.

`Temperature`: allows to have the magnetization informations of the material at a specified temperature. The default is `Automatic` (nominal temperature).

`NumberOfPoints`: allows to choose the number of rows in the array. Default is `15`.

■ Related Units:

`MagnetizingUnit`: allows to specifically set the unit of the magnetizing field.

`MagnetizationUnit`: allows to specifically set the unit of `M`,`Mprincipal`,`Mtranversal`.

`InductionUnit`: allows to specifically set the unit of `B`,`Bprincipal`,`Btransversal`.

`FieldUnit`: is the unit that is taken into account when the previous specific units are not set.

■ See the MagneticaUser Book: Appendix Materials.

■ See also: `MaterialMagnetizationCurves`, `Properties`, `MagnetizingFieldRange`, `Temperature`, `NumberOfPoints`, `FieldUnit`, `MagnetizingUnit`, `MagnetizationUnit`, `InductionUnit`.

■ MeshingLevel

■ `MeshingLevel` is an option for `AXSMagneticSystem` that specify the granularity of the meshings.

■ The possible values are only: 1, 2, 3 or 4.

■ Only Ferromagnetic Components are affected by this setting (that are `AXSHardMagneticComponent` and `AXSSoftMagneticComponent`).

■ The granularity of the meshing for these components can be choosen to be sparse (1), intermediate (2), high density (3), very high density (4).

■ The higher the meshing level is the higher the cells are numerous and the better is the simulation of the magnetic

behavior of devices, but the time calculation can be important.

- With a low meshing level, the computations are faster but less precise.
- Meshing can be more finely specified at the component level, by using the option ComponentMeshing.
- See the Magnetica User Book: Chapter 6, Chapter 7, Chapter 8.
- See also: `AXSMagneticSystem`, `ComponentMeshing`,

■ **MagneticaCalculationMode**

- `MagneticaCalculationMode[`*mode*`]` sets the global calculation mode to *mode*.
- `MagneticaCalculationMode[]` displays the current calculation mode.

- The global calculation mode is the mode taken into account in all kind of field computations (punctual field functions, field maps,…), when the `CalculationMode` option is set to `Inherited`.
- Possible values for *mode* are:

`VectorPotential`: the vector potential method is used. Computations are possible at any point.

`SphericalHarmonics`: the spherical harmonics series are used. Computations are not possible everywhere. The `SHLimits` function gives the limits of use, and `DrawSHLimits` displays them.

`Automatic`: an internal algorithm switch between the two modes to provide the best result in term of speed and accuracy.

- See the Magnetica User Book: Chapter 4, Chapter 5.
- See also: `VectorPotential`, `SphericalHarmonics`, `CalculationMode`, `SHLimits`, `Verbose-SHLimits`, `DrawSHLimits`.

■ **MutualMagneticEnergy**

- `MutualMagneticEnergy[`*system,{componentnumber1,componentnumber2}*`]` computes the magnetic energy corresponding to the mutual inductance between the components specified.

- The first argument is a magnetic system.
- The components are specified by their rank in the magnetic system.
- The second argument is a list of two numbers, the first corresponding to the source component, and the second to the target.
- Mutual magnetic energy can be only calculated between current carrying components.
- Possible options are:

`ContributionType`: allows to compute the `Proper` mutual magnetic energy (default), the ferromagnetic contribution (`Ferro`) to the mutual magnetic energy or the `Total`.

`PrecisionLevel`: specifies the precision level among 1, 2 or 3. Default 1.

- Related units:

`MagneticEnergyUnit`: to specify the unit of the output.

- See the Magnetica User Book: Chapter 10, Chapter 11.
- See also: `VerboseMutualMagneticEnergy`, `SelfMagneticEnergy`, `MagneticEnergyMatrix`, `AXSMagneticSystem`, `AXSSolenoid`, `AXSCircularLoop`, `ContributionType`, `PrecisionLevel`, `MagneticEnergyUnit`.

■ **MutualFluxInductance**

- `MutualFluxInductance[`*system, {compnumber1,compnumber2}*`]` computes the flux generated by the first component through the geometry of the second.

- The first argument is a magnetic system.
- The components are specified by their rank in the magnetic system.
- The second argument is a list of two numbers, the first corresponding to the source component, and the second to the target.
- Mutual flux inductance can be only calculated between current carrying components.
- Possible options:

`ContributionType`: allows to compute the `Proper` mutual flux inductance (default), the ferromagnetic contribution (`Ferro`) to the mutual flux inductance or the `Total`.

`PrecisionLevel`: specifies the precision level among 1, 2 or 3. Default 1.

- Related units:

`FluxUnit`: to specify the unit of the output.

- See the Magnetica User Book: Chapter 10, Chapter 11.
- See also: `VerboserMutualFluxInductance, AXSMagneticSystem, AXSSolenoid, AXSCircular-Loop, ContributionType, PrecisionLevel, FluxUnit`.

■ MutualInductance

- `MutualInductance[`*system,{component1,component2}*`]` computes the mutual inductance between the two components of the given magnetic system.

- The first argument is a magnetic system.
- The components are specified by their rank in the magnetic system.
- The second argument is a list of two numbers, the first corresponding to the source component, and the second to the target.
- Mutual inductance can be only calculated between current carrying components.
- Possible options:

`ContributionType`: allows to compute the `Proper` mutual inductance (default), the ferromagnetic contribution (`Ferro`) to the mutual inductance or the `Total`.

`PrecisionLevel`: specifies the precision level among 1, 2 or 3. Default 1.

- Related units:

`InductanceUnit`: to specify the unit of the output.

- See the Magnetica User Book: Chapter 10, Chapter 11.
- See also: `VerboserMutualInductance, AXSMagneticSystem, AXSSolenoid, AXSCircularLoop, ContributionType, PrecisionLevel, InductanceUnit`.

■ NumberOfPoints

- `NumberOfPoints` is a common option for most of graphic functions that allows to choose the number of points to use.

- `NumberOfPoints` can be given any positive integer value.
- If the distribution of points is automatically choosen, the value given to `NumberOfPoints` is the minimum of number of points to use.
- If the distribution is taken to be regularly spaced, the value of `NumberOfPoints` is the exact number of points to use.
- See: `FieldMap, ContourFieldMap, DrawIsoFieldLines, MagnetizationCurves,...`

■ Operational

■ `Operational` is a possible option value for `MagnetizingFieldRange`.

■ `Operational` allows to set the range so that the unsafe region is excluded.

■ See: `Extended`, `HysteresisCycle`, `MagnetizingFieldRange`.

■ PlotScale

■ `PlotScale` is an option for most 2D graphics functions, allows to set linear or logarithmic scale on each axes.

■ Possible values are: `LinearMap`, `LogLinearMap`, `LinearLogMap`, `LogLogMap`.

■ The default setting is usually `LinearMap`.

■ See the Magnetica User Book: Chapter 4.

■ See also: `FieldMap`, `MaterialMagnetizationCurves`, `LinearMap`, `LinearLogMap`, `LogLinearMap`, `LogLogMap`.

■ PotentialPoint, PotentialPointVP, PotentialPointSH

■ `PotentialPoint`[*axssystem*, *targetPoint*] returns the azimuthal value of the magnetic vector potential of the system, at the specified target point.

■ `PotentialPointVP`[*axssystem*, *targetPoint*] returns the potential, using the vector potential method for calculation.

■ `PotentialPointSH`[*axssystem*, *targetPoint*] returns the potential, using the spherical harmonics method for calculation.

■ The first argument is an axis symmetric system: that is the return value of `AXSMagneticSystem`.

■ The second argument defines the target point. It can be specified either by its 3D spatial cartesian coordinates *{x, y, z}* or by its 2D axisymmetric coordinates *{r, z}*.

■ The output is in the form of a single list representing the only non-null cylindrical coordinate, i.e. the azimuthal. $\%A_\phi <$.

■ `PotentialPoint` (without suffix) computes the potential, according to the global calculation mode. (See `MagneticaCalculationMode`).

■ Possible options are the following:

`ComponentList`: allows to see the contribution of each component, in the specified list, to the potential value. The output then becomes in the form: *{componentsNumber,{potential1,potential2,...},total}*.

`CalculationMode`: the default value is `Inherited`, from the global calculation mode. The other possible values are `VectorPotential`, `SphericalHarmonics`, or `Automatic`.

■ Related units are:

`LengthUnit`: to specify the unit of the target point.

`PotentialUnit`: to specify the unit of the output.

■ See the Magnetica User Book: `Chapter 4`, `Chapter 5`.

■ See also: `AXSMagneticSystem`, `ComponentList`, `CalculationMode`, `LengthUnit`, `PotentialUnit`, `VerbosePotentialPoint`.

■ PotentialUnit

■ `PotentialUnit` is the option used to specify the magnetic potential unit.

■ `PotentialUnit` can be used in `MagneticaUserUnits` for a global setting, or in any function concerned with potential unit, for local setting.

■ Possible values: `Weber/Meter, Maxwell/Meter,`…

■ These values can be preceded, as a separated word, by any prefix (`Milli, Micro,`…).

■ See the Magnetica User Book: Appendix Units.

■ See also: `MagneticUserUnits, MagneticaCurrentUnits, PotentialPoint, VerbosePotential-Point,`…

■ PowerUnit

■ `PowerUnit` is the option used to specify the unit of power.

■ `PowerUnit` can be used in `MagneticaUserUnits` for a global setting, or in any function concerned with power unit, for local setting.

■ Possible value: `Watt.`

■ This value can be preceded, as a separated word, by any prefix (`Milli`, `Kilo`, `Mega`, …).

■ See the Magnetica User Book: Appendix Units.

■ See also: `MagneticaUserUnits, MagneticaCurrentUnits, VerboseTransformerAnalysis.`

■ PrecisionLevel

■ `PrecisionLevel` is an option for inductance functions, that allows to select the precision computation level.

■ Possible values are: 1,2 or 3.

■ For quick evaluation choose 1.

■ For high precision choose 3.

■ The intermediate, 2, is often enough precise.

■ See: `SelfInductance, SelfFluxInductance, SelfMagneticEnergy,`…

■ Properties

■ `Properties[`*system*`]` displays many informations on the system and its components.

■ `Properties[`*material*`]` displays the caracterictics of the material *material*.

■ With a magnetic system as argument, the option `ComponentList` can be used to obtain more details on some components or on all of them.

■ With a material as argument, the option Temperature can be used to have the properties at a choosen temperature.

■ See the Magnetica User Book: Appendix Materials.

■ See also: `AXSMagneticSystem, ComponentList, Temperature.`

■ RemanentMagnetization

■ `RemanentMagnetization` is an option for `AXSRigidMagneticComponent` and `AXSHardMagnetic`„ `Component`, that allows to set another remanent magnetization value than the one given by the material properties.

■ `RemanentMagnetization` can be given any positive numeric value.

■ The `RemanentMagnetization` value is expressed in the unit of `MaterialMagnetizationUnit` if locally set, otherwise in the unit of `FieldUnit`.

■ See the Magnetica User Book: Chapter 6, Chapter 7.

■ See also: `AXSRigidMagneticComponent`, `AXSHardMagneticComponent`, `FieldUnit`, `Material-MagnetizationUnit`.

■ ResistanceUnit

■ `ResistanceUnit` is the option used to specify the unit of electrical resistance.

■ `ResistanceUnit` can be used in `MagneticaUserUnits` for a global setting, or in any function concerned with electrical resistance unit, for local setting.

■ Possible value: `Volt`.

■ This value can be preceded, as a separated word, by any prefix (`Milli`, `Kilo`, `Mega`, …).

■ See the Magnetica User Book: Appendix Units.

■ See also: `MagneticaUserUnits`, `MagneticaCurrentUnits`, `VerboseTransformerAnalysis`.

■ ResonantCapacitor

■ `ResonantCapacitor[`*self, frequency*`]` gives the needed capacitor to obtain a given resonant frequency in an electric circuit containing a given self inductance.

■ For an electric circuit made of a self inductance expressed in μHenry and a resonant frequency expressed in Hertz, the result is in μFarad.

■ See the Magnetica User Book: Chapter 13.

■ See also: `ResonantFrequency`, `TurnsBrookCoil`.

■ ResonantFrequency

■ `ResonantFrequency[`*self, capacitor*`]` gives the resonant frequency of an electric circuit of given self inductance and capacitor.

■ For an electric circuit made of a self inductance expressed in μHenry and a capacitor expressed in μFarad, the result is in Hertz.

■ See the Magnetica User Book: Chapter 13.

■ See also: `ResonantCapacitor`, `TurnsBookCoil`.

■ Rigid

■ Rigid is an option for AXSHardMagneticComponent that allows to consider constant magnetization.

■ Rigid can only be given True or False.

■ Rigid->True, magnetization is to be considered constant for this component, that is susceptibility is null. The component becomes an AXSRigidMagneticComponent.

■ Rigid->False magnetization vary linearly with the excitation field (default value).

■ See also: `AXSHardMagneticComponent`.

■ **SelfMagneticEnergy**

■ `SelfMagneticEnergy[`*system, componentnumber*`]` computes the magnetic energy corresponding to the self inductance of the component specified.

■ `SelfMagneticEnergy[`*system, {compnumber1,compnumber2,...}*`]` computes the self magnetic energy of each specified component in the given system.

■ `SelfMagneticEnergy[`*system*`,All]` computes the self magnetic energy of all relevant components of the given system.

■ The first argument is a magnetic system.

■ The component are specified by their rank in the magnetic system.

■ Self magnetic energy can only be calculated for current carrying components.

■ If the second argument is only one number, the output is the value of the self magnetic energy corresponding to this component.

■ If the second argument is a list or `All`, then the output is in the form of a list of three elements:

the first is the list of relevant component numbers,

the second is the list of each corresponding self magnetic energy values,

the third is the total, i.e. the sum of these self magnetic energy values.

■ Possible options are:

`ContributionType`: allows to compute the `Proper` self magnetic energy (default), the ferromagnetic contribution (`Ferro`) to the self magnetic energy or the `Total`.

`PrecisionLevel`: specifies the precision level among 1, 2 or 3. Default 1.

■ Related units:

`MagneticEnergyUnit`: to specify the unit of the output.

■ See the Magnetica User Book: Chapter 10, Chapter 11.

■ See also: `VerboseSelfMagneticEnergy, MutualMagneticEnergy, MagneticEnergyMatrix, AXSMagneticSystem, AXSSolenoid, AXSCircularLoop, ContributionType, PrecisionLevel, MagneticEnergyUnit`.

■ **SelfFluxInductance**

■ `SelfFluxInductance[`*system, componentnumber*`]` computes the flux generated by the component through its own geometry.

■ `SelfFluxInductance[`*system, {compnumber1,compnumber2,...}*`]` computes the self flux inductance of each specified component in the given system.

■ `SelfFluxInductance[`*system*`,All]` computes the self flux inductance of all relevant components of the given system.

■ The first argument is a magnetic system.

■ The component are specified by their rank in the magnetic system.

■ Self flux inductance can only be calculated for current carrying components.

■ If the second argument is only one number, the output is the value of the self flux inductance corresponding to this component.

■ If the second argument is a list or `All`, then the output is in the form of a list of three elements:

the first is the list of relevant component numbers,

the second is the list of each corresponding self flux inductance values,

the third is the total i.e. the sum of these self flux inductance.

■ Possible options are:

ContributionType: allows to compute the Proper self flux inductance (default), the ferromagnetic contribution (Ferro) to the self flux inductance or the Total.

PrecisionLevel: specifies the precision level among 1, 2 or 3. Default 1.

■ Related units:

FluxUnit: to specify the unit of the output.

■ See the Magnetica User Book: Chapter 10, Chapter 11.

■ See also: VerboseSelfFluxInductance, AXSMagneticSystem, AXSSolenoid, AXSCircularLoop, ContributionType, PrecisionLevel, FluxUnit.

■ SelfInductance

■ SelfInductance[*system,componentnumber*] computes the self inductance of the specified component in the given magnetic system.

■ SelfInductance[*system, {compnumber1,compnumber2,...}*] computes the self inductance of each specified component in the given system.

■ SelfInductance[*system*,All] computes the self inductance of all relevant components of the given system.

■ The first argument is a magnetic system.

■ the component are specified by their rank in the magnetic system.

■ Self inductance can be only calculated for current carrying components.

■ If the second argument is only one number, the output is the value of the self inductance corresponding to this component.

■ If the second argument is a list or All, then the output is in the form of a list of three elements:

the first is the list of the relevant component numbers,

the second is the list of each corresponding self inductance values,

the third is the total i.e. the sum of these self inductance values.

■ Possible options are:

ContributionType: allows to compute the Proper self inductance (default), the ferromagnetic contribution (Ferro) to the self inductance or the Total.

PrecisionLevel: specifies the precision level among 1, 2 or 3. Default 1.

■ Related units:

InductanceUnit: to specify the unit of the output.

■ See the Magnetica User Book: Chapter 10, Chapter 11.

■ See also: VerboseSelfInductance, AXSMagneticSystem, AXSSolenoid, AXSCircularLoop, ContributionType, PrecisionLevel, InductanceUnit.

■ SHLimits

■ SHLimits[*system*] gives a list of limits of Spherical Harmonics regions, for the given system.

■ The argument is a magnetic system: the return value of AXSMagneticSystem or CARMagneticSystem.

■ The output is a list of pairs representing respectively:

1. the strict limits of the convergent region for the Spherical Harmonics series.

2. the limits for reasonable use of Spherical Harmoncis as only a finite number of terms are taken for calculations.

3. the limits taken when Automatic is set, to switch between Spherical Harmonics and Potential Vector method.

■ These limits depends on the value of LegendrePointPosition given to the system.

■ See the Magnetica User Book: Chapter 5.

■ See also: AXSMagneticSystem, VerboseSHLimits, DrawSHLimits, LegendrePointPosition.

■ ShowArrows

■ `ShowArrows` is an option for `DrawSystem`-like functions, that allows to display the current direction for current carrying component and the magnetization direction for permanent magnet component.

■ Possible values are `True` or `False`.
■ `ShowArrows` can also be used as option for `DrawSHLimits`-like functions, and `DrawIsoFieldLines`-like functions.
■ See: `DrawSystem`, `DrawSystem2D`, `DrawSHLimits`, `DrawIsoFieldLines`.

■ ShowCentralAxe

■ `ShowCentralAxe` is an option for `DrawSystem`-like functions, that allows to display or not the Z-axis.

■ Possible values are `True` or `False`.
■ `ShowCentralAxe` can also be used as option for `DrawSHLimits`-like functions, and `DrawIsoField„Lines`-like functions.
■ See: `DrawSystem`, `DrawSystem2D`, `DrawSystem3D`, `DrawSHLimits`, `DrawIsoFieldLines`.

■ ShowSphericalLimits

■ `ShowSphericalLimits` is an option for `DrawSystem2D`-like functions, that allows to displays the limits for valid `SphericalHarmonics` calculations.

■ Possible values are `True` or `False`.
■ Can also be used with `DrawIsoFieldLines`.
■ See: `DrawSystem2D`, `DrawIsoFieldLines`, `DrawSHLimits`, `SHLimits`.

■ SphericalHarmonics

■ `SphericalHarmonics` is an option value for `CalculationMode` and a possible value for the `Magnetica„CalculationMode` functions.

■ When the calculation mode is set locally or globally to `SphericalHarmonics`, the calculations are done using the spherical harmonics method. They cannot be done everywhere, and the limits of use is given by `SHLimits`.
■ The other possible basic mode is `VectorPotential`.
■ See the Magnetica Book: Chapter 5.
■ See also: `CalculationMode`, `MagneticaCalculationMode`, `SHLimits`, `VectorPotential`.

■ SystemInductance

■ `SystemInductance[`*system*`]` returns the total inductance, the total flux inductance and the total magnetic energy of the current carrying components of the magnetic system.

■ The argument is a magnetic system.
■ The option `ComponentsList` allows to select the component involved in the calculation.
■ The output is in the form *{total inductance, total flux, total magnetic energy}*.
■ Possible options:

`ComponentsList`: allows to select the list of components to take into account. The default is value is `All`.

`ContributionType`: allows to compute the `Proper` inductances (default), the ferromagnetic contribution (`Ferro`) to the inductances, or the `Total`.

`PrecisionLevel`: specifies the precision level among 1,2 or 3. Default 1.

■ Related units:

`InductanceUnit`, `FluxUnit`, `MagneticEnergyUnit`: to specify the unit of the output.

■ See the Magnetica User Book: Chapter 10, Chapter 11.

■ See also: `VerboseSystemInductance`, `InductanceMatrix`.

■ SystemLabel

■ `SystemLabel` is an option for `AXSMagneticSystem` that allows to specify a label to be associated with the system.

■ `SystemLabel` must be specified with a string like: `SystemLable->`*"your description"*.

■ The label specified will appear in every description, drawing or map involving the system.

■ See the Magnetica User Book: Chapter 3.

■ See also: `AXSMagneticSystem`.

■ TableIsoFieldLines

■ `TableIsoFieldLines[`*system, {fieldvalue1, fieldvalue2,...}*`]` gives a table of the points used to draw the lines in `DrawIsoFieldLines` with the same arguments.

■ The arguments are the same as for `DrawIsoFieldLines`.

■ Possible options are:

`NumberOfPoints`: allows to choose the number of points used for the table. The default is `15`.

`AngularRange`: allows to select the angular range. The default is `{0,90}`.

`CalculationMode`: the default is `Inherited`, that refers to the global calculation mode. The other values coerce the local use of a specific mode and can be: `VectorPotential`, `SphericalHarmonics`, `Automatic`.

■ Related units:

`FieldUnit`: to specify the magnetic field values.

`LengthUnit`: to specify the unit of the output.

■ See the Magnetica User Book: Chapter 4.

■ See also: `DrawIsoFieldLines`, `AXSMagneticSystem`, `NumberOfPoints`, `AngularRange`, `CalculationMode`, `FieldUnit`, `LengthUnit`.

■ Temperature

■ `Temperature` is a option for all component functions, some material functions, to locally specify the temperature to take into account.

■ `Temperature` can be given any numeric value and is expressed in Degrees.

■ As an option for a Component function, `Temperature` allows to set the effective temperature of the component, which modify therefore the dimensions values and other temperature dependent values.

■ As an option for Material Properties printing and drawing, it allows to give the properties at the specified temperature.

■ When this option is not used the effective temperature is taken to be the nominal temperature 20°.

■ See the Magnetica User Book: Chapter 2.

■ See also: `AXSCircularLoop`, `AXSSolenoid`, `AXSBitterSolenoid`, `AXSRigidMagneticComponent`, `AXSHardMagneticComponent`, `AXSSoftMagneticComponent`, `Properties`.

■ TransformerAnalysis

■ `TransformerAnalysis[`*system, {compnumber1,voltage1},{compnumber2,voltage2},activeload*`]` returns as a list the results of `VerboseTransformerAnalysis`.

■ This gives a list of the relevant parameters of a monophase transformer for a given frequency and load.

■ The first argument is a magnetic system.

■ The second and third arguments specify the current carrying components constituting the transformer and the voltage of each one.

■ The third argument specifies the active load.

■ Possible options:

`InputFrequency`: to specify the frequency. Default is 50.

`PrecisionLevel`: to specify the precision level among 1,2 or 3. Default 1.

`FullDetails`: to get more or less details. Default `False`.

■ Related units:

`CurrentIntensityUnit`, `ResistanceUnit`, `VoltageUnit`, `FieldUnit`, `PowerUnit`.

■ See the Magnetica User Book: Chapter 13.

■ See also: `VerboseTransformerAnalysis`, `AXSMagneticSystem`, `PrecisionLevel`.

■ TransformerMagneticLosses

■ `TransformerMagneticLosses[`*system*`]` gives the magnetic losses of the system seen as a monophase transformer. It also gives the average induction in the magnetic cells.

■ The first argument is a monophase transformer, that is a magnetic system that does not contain more or less than two current carrying components. Otherwise, the results are not pertinent.

■ The output is in the form *{magnetic losses, average induction of cells}*.

■ The magnetic losses can be calculated only for ferromagnetic structures made of material with known curves of losses.

■ Possible options:

`InputFrequency`: to specify the frequency. Default 50.

■ See the Magnetica User Book: Chapter 10, Chapter 13.

■ See also: `VerboseTransformerMagneticLosses`, `TransformerAnalysis`, `VerboseTranformer-Analysis`.

■ TurnsBrookCoil

■ `TurnsBrookCoil[`*meandiameter, self*`]` gives the number of turns needed to obtain a given self inductance with a coil of given mean diameter, with a Brook type geometry.

■ The mean diameter is supposed to be expressed in Milli Meter, and the self inductance in μHenry.

■ See the Magnetica User Book: Chapter 13.

■ See also: `ResonantFrequency`, `ResonantCapacitor`.

■ **VectorFieldMap**

■ `VectorFieldMap[`*system*`, Plane[`*point1, point2*`]]` displays the magnetic field of the given system, within the rectangle defined by the two opposite corners *point1* and *point2*.
■ `VectorFieldMap[`*system*`, Box[`*point1, point2*`]]` displays the magnetic field of the given system within the cuboid space defined by the two opposite corners *point1* and *point2*.

■ The first argument is a magnetic system: that is the return value of `AXSMagneticSystem` or `CARMagnetic‚ System`.
■ If the system is axisymmetric, the points in the directive `Plane`, can be specified either by 3D spatial cartesian coordinates {x,y,z}, or by their 2D axisymmetric coordinates {r,z} (However the two points must be of the same type). In the directive `Box` and for cartesian systems, only cartesian coordinates are possible for points specifications.
■ Possible options are:
`NumberOfPoints`: allows to choose the number of points to use for the plot. The default is `Automatic`. (15 for `Plane`, and 7 for `Box`).
`CalculationMode`: the default value is `Inherited`, that refers to the global calculation mode. The other values coerce the local use of a specific mode and can be: `VectorPotential`, `SphericalHarmonics`, `Automatic`.
■ Related units:
`LengthUnit`: to specify the unit of the points.
`FieldUnit`: to specify the unit of the output.
■ See the Magnetica User Book: Chapter 4, Chapter 5.
■ See also: `FieldMap`, `ContourFieldMap`, `AXSMagneticSystem`, `NumberOfPoints`, `Calculation-Mode`, `LengthUnit`, `FieldUnit`.

■ **VectorPotential**

■ `VectorPotential` is an option value for `CalculationMode` and a possible value for the `Magnetica‚ CalculationMode` functions.

■ When the calculation mode is set locally or globally to `VectorPotential`, the calculations are done using the vector potential method. They can be done at any point even inside the components.
■ The other possible basic mode is `SphericalHarmonics`.
■ See the Magnetica Book: Chapter 5.
■ See also: `CalculationMode`, `MagneticaCalculationMode`, `SphericalHarmonics`.

■ **VerboseMagneticEnergyMatrix**

■ VerboseMagneticEnergyMatrix[system] prints a detailed and formatted output of the result of `MagneticEnergy‚ Matrix`.

■ Argument and options are the same as for `MagneticEnergyMatrix`.
■ Related units:
`MagneticEnergyUnit`,`CurrentIntensityUnit`: to specify the units of the output.
■ See also: `MagneticEnergyMatrix`.

- **VerboseSHLimits**

- `VerboseSHLimits[`*system*`]` prints a formatted output of `SHLimits`.

- See `SHLimits`.

- **VerboseFieldPoint, VerboseFieldPointVP, VerboseFieldPointSH**

- `VerboseFieldPoint[`*system, targetPoint*`]` prints a detailed and formatted output of the result of `FieldPoint`.
- `VerboseFieldPointVP[`*system, targetPoint*`]` verbose version of `FieldPointVP`.
- `VerboseFieldPointSH[`*system, targetPoint*`]` verbose version of `FieldPointSH`.

- The arguments are the same as for `FieldPoint`.
- The output cannot be managed for further computations. For this, better use `FieldPoint`.
- Possible options:

`ComponentList`: details the contribution, to the magnetic field, of each specified component, and prints the total for these components.

`CalculationMode`: the default value Inherited, tells to take into account the global calculation mode. It can be overriden by `VectorPotential`, `SphericalLimits`, `Automatic` to coerce the local use of a specific mode.

`CondensedForm`: used together with `ComponentList`, allows to dispay the informations in an array form.

- Related units:

`LengthUnit`: to specify the unit of the target point.

`FieldUnit`: to specify the unit of the output.

- See the Magnetica User Book: Chapter 4, Chapter 5.
- See also: `FieldPoint`, `ComponentList`, `CalculationMode`, `LengthUnit`, `FieldUnit`.

- **VerboseFluxArea, VerboseFluxAreaVP, VerboseFluxAreaSH**

- `VerboseFluxArea[`*system, targetPoint*`]` prints a detailed and formatted output of the result of `FluxArea`.
- `VerboseFluxAreaVP[`*system, targetPoint*`]` verbose version of `FluxAreaVP`.
- `VerboseFluxAreaSH[`*system, targetPoint*`]` verbose version of `FluxAreaSH`.

- The arguments are the same as for `FluxArea`.
- The output cannot be managed for further computations. For this, better use `FluxArea`.
- Possible options:

`ComponentList`: details the contribution, to the magnetic flux, of each specified component, and prints the total for these components.

`CalculationMode`: the default value Inherited, tells to take into account the global calculation mode. It can be overriden by `VectorPotential`, `SphericalLimits`, `Automatic` to coerce the local use of a specific mode.

`CondensedForm`: used together with `ComponentList`, allows to dispay the informations in an array form.

- Related units:

`LengthUnit`: to specify the unit of the target point.

`FluxUnit`: to specify the unit of the output.

- See the Magnetica User Book: Chapter 4, Chapter 5.
- See also: `FluxArea`, `ComponentList`, `CalculationMode`, `LengthUnit`, `FluxUnit`.

■ **VerboseFluxInductanceMatrix**

■ `VerboseFluxInductanceMatrix[`*system*`]` prints a detailed and formatted output of the result of `Flux`„`InductanceMatrix`.

■ Argument and options are the same as for `FluxInductanceMatrix`.
■ Related units:
`FluxUnit`,`CurrentIntensityUnit`: to specify the units of the output.
■ See also: `FluxInductanceMatrix`.

■ **VerboseGradientPoint, VerboseGradientPointVP, VerboseGradientPointSH**

■ `VerboseGradientPoint[`*system, targetPoint*`]` prints a detailed and formatted output of the result of `Gradient`„`Point`.
■ `VerboseGradientPointVP[`*system, targetPoint*`]` verbose version of `GradientPointVP`.
■ `VerboseGradientPointSH[`*system, targetPoint*`]` verbose version of `GradientPointSH`.

■ The arguments are the same as for `GradientPoint`.
■ The output cannot be managed for further computations. For this, better use `GradientPoint`.
■ Possible options:
`ComponentList`: details the contribution, to the gradient value, of each specified component, and prints the total for these components.
`CalculationMode`: the default value Inherited, tells to take into account the global calculation mode. It can be overriden by `VectorPotential`, `SphericalLimits`, `Automatic` to coerce the local use of a specific mode.
`CondensedForm`: used together with `ComponentList`, allows to dispay the informations in an array form.
■ Related units:
`LengthUnit`: to specify the unit of the target point.
`GradientUnit`: to specify the unit of the output.
■ See the Magnetica User Book: Chapter 4, Chapter 5.
■ See also: `GradientPoint`, `ComponentList`, `CalculationMode`, `LengthUnit`, `GradientUnit`.

■ **VerboseInductanceCoupling**

■ `VerboseInductanceCoupling[`*system, {component1, component2}*`]` prints a detailed and formatted output of the result `InductanceCoupling`.

■ Arguments and options are the same as for `InductanceCoupling`.
■ Related units:
`CurrentIntensityUnit`, `InductanceUnit`, `FluxUnit`: to specify the units of the printed informations.
■ See the Magnetica User Book: Chapter 12, Chapter 13.
■ See also: `InductanceCoupling`, `AXSMagneticSystem`, `PrecisionLevel`, `CurrentIntensityUnit`, `InductanceUnit`, `FluxUnit`.

■ **VerboseInductanceMatrix**

■ VerboseInductanceMatrix[*system*] prints a detailed and formatted output of the result of Inductance„
Matrix.

■ Argument and options are the same as for InductanceMatrix.
■ Related units:
InductanceUnit,CurrentIntensityUnit: to specify the units of the output.
■ See also: InductanceMatrix.

■ **VerboseLegendreCoefficients**

■ VerboseLegendreCoefficients[*system*] prints many relevant informations and the result of Legendre„
Coefficients in a formatted output.

■ The arguments and the options are the same as for LegendreCoefficients.
■ See LegendreCoefficients.

■ **VerboseMutualMagneticEnergy**

■ VerboseMutualMagneticEnergy[*system,{compnumber1,compnumber2}*] prints a detailed and formatted
output of the result of MutualMagneticEnergy.

■ Arguments and options are the same as for MutualMagneticEnergy.
■ Related units:
MagneticEnergyUnit,CurrentIntensityUnit: to specify the units of the output.
■ See also: MutualMagneticEnergy.

■ **VerboseMutualFluxInductance**

■ VerboseMutualFluxInductance[*system,{compnumber1,compnumber2}*] prints a detailed and formatted
output of the result of MutualFluxInductance.

■ Arguments and options are the same as for MutualFluxInductance.
■ Related units:
FluxUnit,CurrentIntensityUnit: to specify the units of the output.
■ See also: MutualFluxInductance.

■ **VerboseMutualInductance**

■ VerboseMutualInductance[*system,{compnumber1,compnumber2}*] prints a detailed and formatted output of
the result of MutualInductance.

■ Arguments and options are the same as for MutualInductance.
■ Related units:
InductanceUnit,CurrentIntensityUnit: to specify the units of the output.
■ See also: MutualInductance.

■ **VoltageUnit**

■ `VoltageUnit` is the option used to specify the electrical potential difference unit.

■ `VoltageUnit` can be used in `MagneticaUserUnits` for a global setting, or in any function concerned with electrical potential unit, for local setting.

■ Possible value: `Volt`.

■ This value can be preceded, as a separated word, by any prefix (`Milli`, `Kilo`, `Mega`, …).

■ See the Magnetica User Book: Appendix Units.

■ See also: `MagneticaUserUnits`, `MagneticaCurrentUnits`, `VerboseTransformerAnalysis`.

■ **VerbosePotentialPoint, VerbosePotentialPointVP, VerbosePotentialPointSH**

■ `VerbosePotentialPoint[`*system, targetPoint*`]` prints a detailed and formatted output of the result of `PotentialPoint`.

■ `VerbosePotentialPointVP[`*system, targetPoint*`]` verbose version of `PotentialPointVP`.

■ `VerbosePotentialPointSH[`*system, targetPoint*`]` verbose version of `PotentialPointSH`.

■ The arguments are the same as for `PotentialPoint`.

■ The output cannot be managed for further computations. For this, better use `PotentialPoint`.

■ Possible options:

`ComponentList`: details the contribution, to the potential value, of each specified component, and prints the total for these components.

`CalculationMode`: the default value Inherited, tells to take into account the global calculation mode. It can be overriden by `VectorPotential`, `SphericalLimits`, `Automatic` to coerce the local use of a specific mode.

`CondensedForm`: used together with `ComponentList`, allows to dispay the informations in an array form.

■ Related units:

`LengthUnit`: to specify the unit of the target point.

`PotentialUnit`: to specify the unit of the output.

■ See the Magnetica User Book: Chapter 4, Chapter 5.

■ See also: `PotentialPoint`, `ComponentList`, `CalculationMode`, `LengthUnit`, `PotentialUnit`.

■ **VerboseSelfMagneticEnergy**

■ `VerboseSelfMagneticEnergy[`*system,componentnumber*`]` prints a detailed and formatted output of the result of `SelfMagneticEnergy`.

■ `VerboseSelfMagneticEnergy[`*system,{compnumber1,compnumber2,...}*`]` selecting several components.

■ `VerboseSelfMagneticEnergy[`*system*`,All]` selecting all relevant components.

■ Arguments and options are the same as for `SelfMagneticEnergy`.

■ Related units:

`MagneticEnergyUnit`,`CurrentIntensityUnit`: to specify the units of the output.

■ See also: `SelfMagneticEnergy`.

■ VerboseSelfFluxInductance

■ `VerboseSelfFluxInductance[`*system,componentnumber*`]` prints a detailed and formatted output of the result of `SelfFluxInductance`.

■ `VerboseSelfFluxInductance[`*system,{compnumber1,compnumber2,...}*`]` selecting several components.

■ `VerboseSelfFluxInductance[`*system*`,All]` selecting all the relevant components.

■ Arguments and options are the same as for `SelfFluxInductance`.

■ Related units:

`FluxUnit, CurrentIntensityUnit`: to specify the units of the output.

■ See: `SelfFluxInductance`.

■ VerboseSelfInductance

■ `VerboseSelfInductance[`*system,componentnumber*`]` prints a detailed and formatted output of the result of `SelfInductance`.

■ `VerboseSelfInductance[`*system,{compnumber1,compnumber2,...}*`]` selecting several components.

■ `VerboseSelfInductance[`*system*`,All]` selecting all the relevant components.

■ Arguments and options are the same as for `SelfInductance`.

■ Related units:

`InductanceUnit,CurrentIntensityUnit`: to specify the units of the output.

■ See also: `SelfInductance`.

■ VerboseSystemInductance

■ `VerboseSystemInductance[`*system*`]` prints a detailed and formatted output of the result of `VerboseSystem, Inductance`.

■ See the Magnetica User Book: Chapter 10, Chapter 11.

■ See also: `SystemInductance, InductanceMatrix`.

■ VerboseTransformerAnalysis

■ `VerboseTransformerAnalysis[`*system, {compnumber1, voltage1}, {compnumber2, voltage2},activeload*`]` prints a detailed analysis of the system considered as a monophase transformer.

■ The first argument is a magnetic system.

■ The second and third arguments specify the current carrying components corresponding to the transformer, and the voltage of each one.

■ The third argument specifies the active load.

■ Possible options:

`InputFrequency`: to specify the frequency. Default is 50.

`PrecisionLevel`: to specify the precision level among 1,2 or 3. Default 1.

`FullDetails`: to get more or less details. Default `False.`

■ Related units:

`CurrentIntensityUnit, ResistanceUnit, VoltageUnit, FieldUnit, PowerUnit`.

■ See the Magnetica User Book: Chapter 13.

■ See also: `TransformerAnalysis`, `AXSMagneticSystem`, `PrecisionLevel`.

■ **VerboseTransformerMagneticLosses**

■ `VerboseTransformerMagneticLosses[`*system*`]` prints a detailed and formatted output of the result of `TransformerMagneticLosses`.

■ See the Magnetica User Book: Chapter 10, Chapter 13.

■ See also: `TransformerMagneticLosses`, `TransformerAnalysis`, `VerboseTranformerAnalysis`.