EMP 8.0

Finite-element time-domain electromagnetics

PO Box 13595, Albuquerque, NM 87192 U.S.A.
Telephone: +1-505-220-3975
Fax: +1-617-752-9077
E mail: techinfo@fieldp.com
Internet: http://www.fieldp.com

Contents

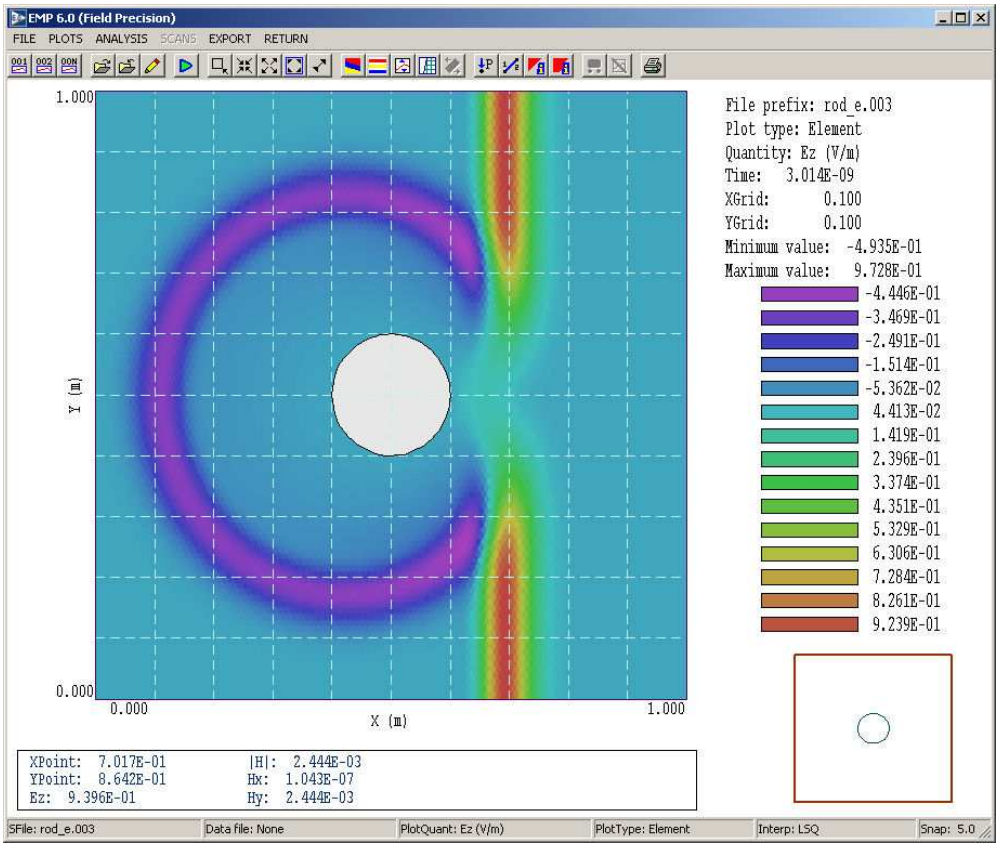

Figure 1: Screenshot – **EMP** in the analysis mode.

1 Introduction

1.1 Program functions

EMP is a versatile tool for calculations of time-dependent electromagnetic fields. The program runs on personal computers operating under any version of Windows. **EMP** follows propagating electromagnetic pulses in two-dimensional planar geometries and three-dimensional cylindrical systems. Applications include simulation of microwave devices, particle accelerators, pulsed-power generators and electromagnetic interference. **EMP** employs unique finite-element methods on conformal triangular meshes for accurate representations of curved and sloped material boundaries. Materials are characterized by values of relative dielectric constant ϵ_r, relative magnetic permeability μ_r and electrical conductivity σ. These quantities may be fixed over a region of the solution volume or may vary with position following specified mathematical functions. You can generate pulses by specifying time-dependent fields on surfaces or by defining source current regions (equivalent to drive current loops or capacitive probes). The program can represent arbitrary time variations of multiple sources through flexible tables with linear or cubic spline interpolations. Material types include vacuum, conductors, or lossy dielectrics and ferrites. In addition to the standard short-circuit and open-circuit boundary conditions, **EMP**

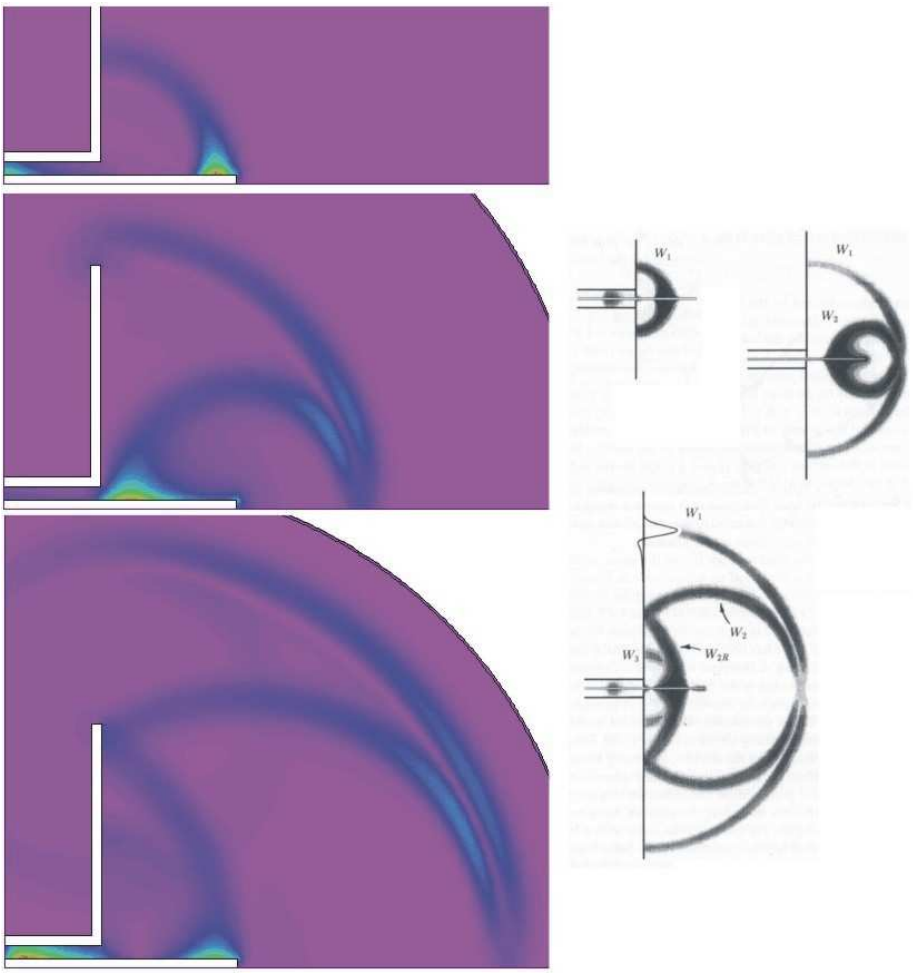

Figure 2: Benchmark test – radiation from a pulsed dipole antenna. Left: **EMP** results, r-z plots of $|\mathbf{E}|$ at 1 ns intervals. Right: cross-section plots of $|\mathbf{E}|$ from J.G. Maloney, G.S. Smith and W.R. Scott, IEEE Trans. Antennas Propagation **AP-38**, 1059 (1990)

supports matched-termination absorbing boundaries to represent open-space conditions.

EMP automatically picks time steps consistent with the Courant stability condition. Run times are short, typically a few seconds on a high-performance computer. The program records spatial distributions of fields and material properties at specified times or temporal variations at up to 20 probe locations. All data files are in structured text format for easy transfer to spreadsheets or mathematical analysis programs. You can quickly analyze spatial data in an interactive, graphical environment (Fig. 1). The program generates quantitative data and a wide variety of plots. The utility program **Probe** is supplied for the analysis of temporal information. With care in preparation, **EMP** can quickly provide results of high accuracy. To illustrate, Fig. 2 shows a comparison of **EMP** results for a pulsed dipole antenna to published results.

1.2 Learning EMP

The size of this manual reflects the broad capabilities of the **Mesh**/**EMP** package. It is not necessary to read the entire document to address most applications. We have organized the chapters to help you get started quickly. You can investigate advanced topics as you gain more experience.

- Browse remaining sections in this chapter to review basics of finite-element calculations. You may also want to read the following chapter on electromagnetic theory. You will probably return to the material to resolve issues (such as boundary conditions) as you get more involved with the programs.
- Scan Chap. 1 of the **Mesh** manual to understand **TC**, the **TriComp** program launcher.
- Be sure to read Chapter 2 of the **Mesh** manual, which reviews essential concepts for conformal meshes in finite-element solutions.
- Chapter 3 of the **Mesh** manual follows a walkthrough example to introduce construction of a solution geometry with the interactive **Mesh** drawing editor. The exercise introduces fundamental tools you will need for you own simulations.
- Move to Chapter 3 of this manual. The walkthrough example covers the steps in creating and analyzing a finite-element solution with **EMP**.
- Run the tutorials described in Chap. 7. The examples give insights into solution options and may serve as templates for your own work.
- As you gain experience you will want to to take advantage of the full range of **Mesh**/**EMP** capabilities. Chapter 4 of the Mesh manual is a comprehensive reference on the **Mesh** drawing editor, while Chap. 5 covers processing, plotting and repair of meshes. Chapters 4 and 6 in this manual summarize the solution and analysis functions of **EMP**. Methods for setting material properties that vary smoothly in space are described in Chap. 5.
- Later chapters in the **Mesh** manual cover advanced techniques. Chapters 6 and 7 show how to make direct entries in scripts to invoke advanced control features. Chapter 8 describes how to create meshes directly from photographic and data images. This feature is useful to model complex or irregular systems that are difficult to describe with simple geometric specifications.

1.3 Finite-element solution procedure

An in-depth understanding of finite-element numerical methods is not necessary to use **EMP**. Nonetheless, it is important to have a clear idea of fundamental concepts to create effective solutions. This section describes background material to understand the steps in an **EMP** solution.

The term *field* indicates a quantity (scalar or vector) defined over a region of space. Examples of fields include the vector electric field **E** in an electrostatic solution, electric and magnetic fields in an electromagnetic solution and the scalar temperature T in a thermal solution. Variations of field quantities are usually described by continuous partial differential equations, such

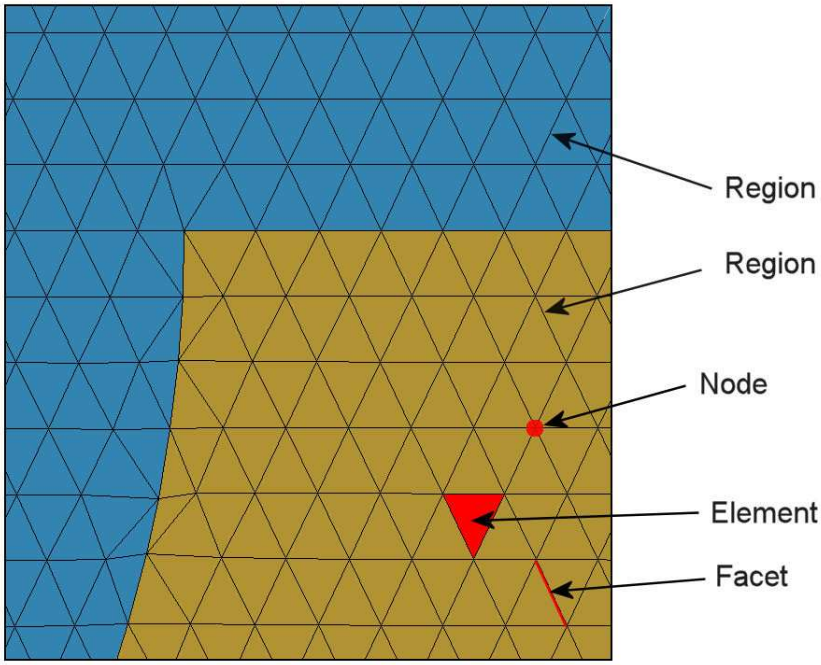

Figure 3: Conformal triangular mesh – definition of terms.

as the Helmholtz equation. These equations can be solved directly by analytic methods if the system geometry and material properties are simple. Analytic solutions are extremely difficult in systems with asymmetric structures or non-linear materials. Furthermore, analytic results are often be expressed in terms of series expansions that must be evaluated numerically. For all but the simplest problems, it is usually quicker and more accurate to employ a direct numerical approach.

The fundamental issue in numerical field solutions is that digital computers cannot directly solve continuous equations. On the other hand, computers are well suited to solving large sets of coupled linear equations. The goal of all numerical field methods (finite-difference, finite-element or boundary-element) is to convert the governing differential equations into a set of coupled linear equations. The solution of the linear equations approaches the results of the differential equation when the set becomes large.

The basis of the finite-element approach is to divide the full solution volume into a number of small volumes, or *elements*. Here, the term *small* indicates that element dimensions are much less than the scale length for variations of field quantities. The division of the volume is called the *computational mesh*. Figure 3 shows the type of mesh used for the two dimensional solutions of **EMP**. The figure defines three terms that will be used throughout this manual:

- **Element**. Volume divisions of the system.
- **Nodes**. Points where elements intersect.
- **Facets.** Surfaces between two elements.

The elements in Fig. 3 have triangular cross-sections. In a planar solution, the cross-section lies in the x-y plane and the elements extend an infinite distance in the z direction. In a cylindrical

solution, an element is a figure of revolution about the axis with a triangular cross-section in the z-r plane.

The mesh in Fig. 3 has the important property of *conformality*. The term means that the triangles have been specially shaped to conform to the boundaries between materials (*regions*). As a result, each element has an unambiguous material identity. The finite-element method is based on two approximations: 1) material properties in an element are uniform and 2) elements are small enough so that the field quantities can be approximated by simple interpolation functions. With these assumptions, the governing differential equation can be integrated over elements surrounding a node to yield a linear equation. This equation relates the field quantity at the node to those at the surrounding nodes. The coupled set has one linear equation for each node in the mesh. In **EMP**, solution of the set representing the Helmholtz equation gives either E_z, rE_θ, H_z or rH_θ (depending on the system symmetry and field polarity). We can then perform two-dimensional interpolations to find the field quantities at intervening points, or take numerical spatial derivatives to find the other field components.

An **EMP** solution involves the following steps::

1. You define the boundaries of material objects in a solution volume for your application. The task is usually accomplished with the drawing editor of **Mesh**. You can also specify control information such as the target element sizes that may affect the accuracy and run time of the solution. The result is a text record (*script*) with sets of line and arc vectors that outline materials and metal boundaries.

2. **Mesh** analyzes the boundary specifications and automatically generates a set of conformal triangles such as those of Fig. 3. The program creates an output file that lists the locations of nodes and the identities of elements.

3. You define the material properties of regions in the solution volume. Usually, this task is performed through an interactive dialog in **EMP**.

4. **EMP** reads the mesh geometry and applies the material parameters to generate the linear equation set.

5. **EMP** advances the coupled equations in time using a centered differencing method with second-order accuracy. The program records field values at all points in space at specified times or records time variations of quantities at specified positions. The resulting files serve as a permanent record of the solution that are always available for analysis.

6. You can use the interactive graphical environment of **EMP** to explore the solution. The program creates a wide variety of plots and performs quantitative calculations of field quantities. You can also write scripts for automatic control of complex analyses.

1.4 Scripts and data files

Mesh and **EMP** read and generate several types of files. For a typical solution it is not necessary for your to deal directly with the files – the user interface takes care of data organization. On the other hand, if you generate a large amount of data it's a good practice to archive application results in individual folders. Furthermore, there are advanced features of **Mesh** and **EMP** that require direct entries to the input scripts. For convenience, all input scripts and

Table 1: **EMP** files

Name	**Function**
`MName.MIN`	**Mesh** input script (definition of foundation mesh and region outlines)
`MName.MLS`	**Mesh** diagnostic listing
`MName.MOU`	**Mesh** output (node locations and element identifies)
`EName.EIN`	**EMP** input script (run control and material properties)
`EName.ELS`	**EMP** diagnostic listing
`EName.001,...`	**EMP** data dumps, field quantities at all positions at specified times
`EName.P01,...`	**EMP** probe files, field quantities over time at specified positions
`AName.SCR`	**EMP** script for automatic data analysis

output data files are in text format. Both **Mesh** and **EMP** feature integrated text editors. Table 1 lists the file types and functions in the **EMP** package. Note that the suffixes indicate the file function.

2 Electromagnetic theory in EMP

2.1 Electromagnetic waves in two dimensions

The reference S. Humphries, **Field Solutions on Computers** (CRC Press, Boca Raton, 1997) gives detailed information on the application of finite-element methods to two-dimensional electromagnetic pulses. This section summarizes the basic theory underlying **EMP**. The program applies the following form of the Maxwell equations:

$$\nabla \times \mathbf{H} = \epsilon \frac{\partial \mathbf{E}}{\partial t} + \mathbf{j}_o + \sigma \mathbf{E}, \tag{1}$$

$$\nabla \cdot \mu \mathbf{H} = 0, \tag{2}$$

$$\nabla \times \mathbf{E} = -\mu \frac{\partial \mathbf{H}}{\partial t}, \tag{3}$$

$$\nabla \cdot \epsilon \mathbf{E} = 0. \tag{4}$$

The magnetic field intensity is related to the magnetic flux density by $\mathbf{H} = \mathbf{B}/\mu$. We assume that the solution volume has no free space-charge. Time-dependent source current densities ($\mathbf{j}_0$) may represent drive structures like coupling loops and capacitive probes. Dielectrics and ferromagnetic materials are isotropic and linear. Furthermore, the magnetic permeability μ, dielectric constant ϵ and conductivity σ are constant in time and have a uniform value within an element. The inclusion of conductivity enables modeling of pulse attenuation and implementation of absorbing boundaries.

Equations 1 through 4 may be combined into individual equations for $\mathbf{E}$ and $\mathbf{H}$. The electric field equation is

$$-\nabla \times \frac{1}{\mu} \nabla \times \mathbf{E} = \frac{\partial}{\partial t} \left(\nabla \times \mathbf{H} \right) = \epsilon \frac{\partial^2 \mathbf{E}}{\partial t^2} + \frac{\partial \mathbf{j}_o}{\partial t} + \sigma \frac{\partial \mathbf{E}}{\partial t}. \tag{5}$$

Using Stoke's theorem, we can rewrite Eq. 5 as

$$-\oint \frac{1}{\mu} \nabla \times \mathbf{E} = \int \int dS \left[\epsilon \frac{\partial^2 \mathbf{E}}{\partial t^2} + \frac{\partial \mathbf{j}_o}{\partial t} + \sigma \frac{\partial \mathbf{E}}{\partial t} \right]. \tag{6}$$

The surface integral on the right hand side extends over the region bounded by the circuital integral.

A planar system has geometric variations in the x and y directions and is uniform along z. The pulse propagation vector lies in the x-y plane. We can divide pulses into two polarization classes:

- E type, with electric field polarization in the z direction, E_z.
- H type with magnetic intensity polarization in the z direction, H_z.

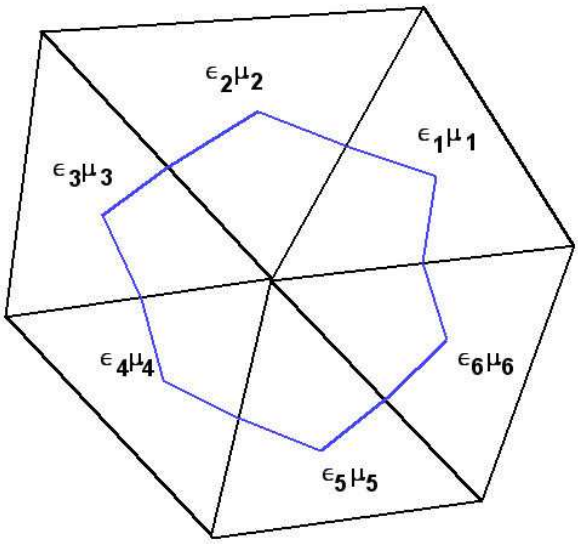

Figure 4: Line integral path through elements surrounding a node.

Equation 6 becomes a scalar equation for E type pulses. Such pulses are created by time-varying current density in the axial direction, j_z.

A finite-element field solution requires the conversion of Eq. 6 to a large number of coupled linear equations. A linear equation may be generated for each node in the solution space by taking surface and line integrals of the governing equation in the x-y plane over the region shown in Fig. 4 (blue line). The right-hand side of Eq. 6 is converted using time-centered difference operators. The solution procedure involves the simultaneous advance of all equations in small time steps Δt to determine new values of E_z at all positions. It is important to recognize that E_z values are defined at nodes while the quantities μ, ϵ, σ and j_z are characteristic of elements.

Knowing E_z at all positions and times, we can find the components of magnetic intensity within the elements by applying numerical differentiation in space and integration in time,

$$H_x = -\frac{1}{\mu}\int_0^t dt' \ \frac{\partial E_z}{\partial y}, \tag{7}$$

$$H_y = \frac{1}{\mu}\int_0^t dt' \ \frac{\partial E_z}{\partial x}. \tag{8}$$

Equations 7 and 8 imply that lines of $\mathbf{H}$ in the x-y plane lie on contours of E_z. This fact is used to create field line plots in **EMP**.

The equation for H type pulses in planar geometry is similar to Eq. 6. Here the numerical solution is carried out for H_z and the subsidiary quantities E_x and E_y are determined from the following equations:

$$E_x = \frac{1}{\epsilon}\left[\int_0^t dt' \ \frac{\partial H_z}{\partial y} + \int_0^t dt' \ j_{ox} + \sigma\int_0^t dt' \ E_x\right], \tag{9}$$

$$E_y = \frac{1}{\epsilon}\left[-\int_0^t dt' \ \frac{\partial H_z}{\partial x} + \int_0^t dt' j_{oy} + \sigma\int_0^t dt' \ E_y\right]. \tag{10}$$

Table 2: Solution types in **EMP**, primary and second field quantities

Type	Function	Primary	Secondary
E	Planar, variations in x and y	E_z	H_x, H_y
	Cylindrical, variations in r and z	rE_θ	H_r, H_z
H	Planar, variations in x and y	H_z	E_x, E_y
	Cylindrical, variations in r and z	rH_θ	E_r, E_z

In this case the electric field lines follow contours of H_z only in regions with no source current and zero conductivity.

Table 2 lists the main and subsidiary field quantities for the four types of solutions in **EMP**. As an example, an H type solution in cylindrical geometry could describe a TEM wave propagating along a coaxial transmission line (Sect. 7.1). Because of the nature of the curl operator, in cylindrical solutions **EMP** determines the quantities rE_θ and rH_θ. It is easy to show that contours of these quantities are parallel to lines of **H** and **E** respectively in the r-z plane. For H type pulses, the correspondence holds only in regions with zero conductivity and source current.

2.2 Courant stability condition and numerical dispersion

The choice of Δt in a time-domain electromagnetic solution is critical. For numerical stability, it must satisfy the Courant condition,

$$\Delta t < \min[\sqrt{\epsilon_i \mu_i} l_i] = \min\left[l_I / v_i\right]. \tag{11}$$

The quantity v_i is the speed of light in element i and l_i is the minimum dimension of the element. Equation 11 states that Δt must be shorter than the time for an electromagnetic disturbance to propagate across the smallest element dimension. If the Courant condition is violated, information can propagate through the mesh faster than the speed of light, violating the principle of causality and leading to a numerical instability. Before beginning a calculation, **EMP** scans all elements and takes l_i as the minimum distance from the element center-of-mass to a vertex. This gives a conservative value for the time step. You can over-ride the choice, but be aware that the choice could lead to an instability that terminates the program.

It is also important to recognize the possibility of *numerical dispersion*. Numerical simulations approximate continuous physical systems through discrete equations. We expect to find some imperfections in the model. As an example, Fig. 5*a* illustrates the long-distance propagation of a Gaussian pulse. A one-dimensional pulse with spatial width 0.067 m travels 9 m. Although energy is conserved, the pulse shape is noticeably distorted. The problem arises because the Gaussian pulse contains a spectrum of traveling Fourier modes. The pulse shape will be preserved only if all modes have the same phase velocity. In the discrete approximation there are small shifts in phase velocity for modes with wavelength comparable to the element size. Therefore, we expect large distortions when the pulse width approaches the element size. Figure 5*b* illustrates declining fidelity with increasing element size for a pulse width of $w_p = 0.067$ m and 2.5 m propagation distance. The three cases correspond to different values of the ratio

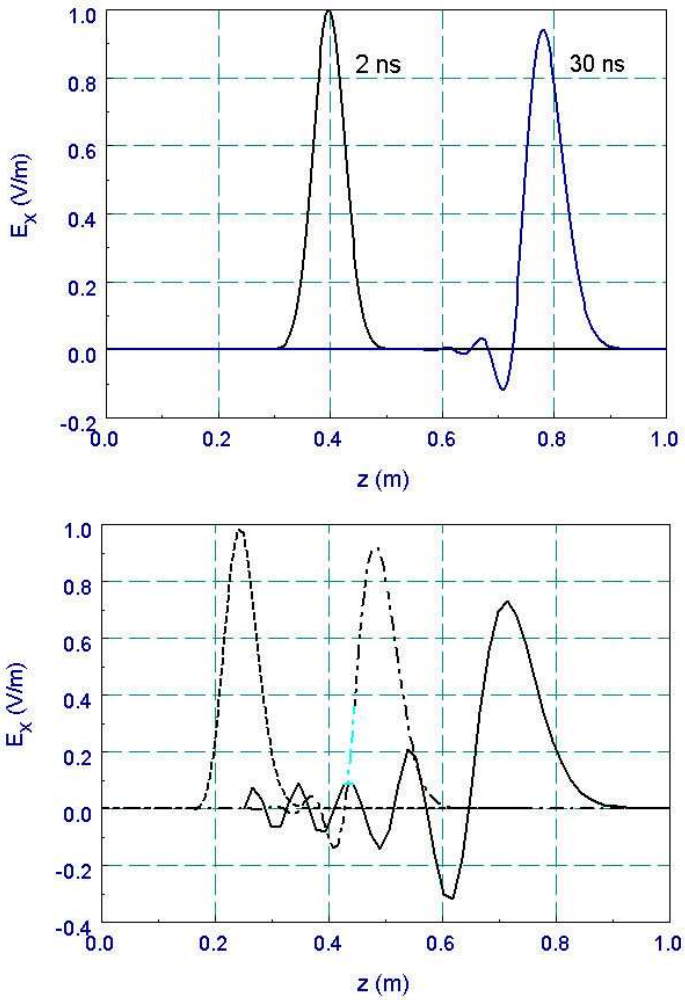

Figure 5: Numerical dispersion in time-domain electromagnetic solution. Gaussian pulse in vacuum, FWHM = 0.067 m. Top: Pulse shapes after 0.6 and 9.0 m propagation for an element width of 0.004 m. Bottom: Pulse shapes after 2.5 m of propagation as a function of element width. Ratio of pulse width to element width: 16.6 (dotted), 8.3 (dashed) and 4.2 (solid).

Table 3: Interpretation of source quantities

Pulse type	Geometry	Quantity	Units
E	Planar	dj_z/dt	A/m^2-s
	Cylindrical	dj_θ/dt	A/m^2-s
H	Planar	j_x, j_y	A/m^2
	Cylindrical	rj_z, rj_r	A/m

of pulse width to element width. Spatial displacements have been added for visibility. As a general rule, dispersion is small if the distance of propagation is less than or comparable to $2w_p^2/w_e$, where w_e is the element width.

2.3 Current sources

Source current density may be assigned to any material region in the solution space. The current density $\mathbf{j}$ is specified by H type solutions and the time derivative $d\mathbf{j}/dt$ for E type solutions. **EMP** assumes uniform spatial distribution and a time dependence given by either a table of values or a mathematical function. Following the discussion of Sect. 2.1, the current-density component dj_z/dt in a planar geometry generates E type pulses. Here, lines of magnetic intensity in the x-y plane circulate around current carrying regions. The regions act like current loops that extend an infinite distance in z. Currents in the x-y plane generate H type pulses. They create field components E_x and E_y through displacement currents and act as capacitive drivers. For H type solutions we must specify an additional parameter for a source region, the direction of current density in the x-y plane. In **EMP**, the quantity β represents the angle of the current density relative to the x axis.

In cylindrical coordinates, an azimuthal current-density variation dj_θ/dt drives E type pulses and current in the z-r plane drives H type pulses. In the latter case, the reference parameter β is the angle of the current density relative to the z axis. Instead of taking a value of $\mathbf{j}$ in H type solutions that is constant over a region, the convention in **EMP** is to use a constant value of the quantity $\overline{\Gamma} = r\mathbf{j}$. This choice ensures that: 1) real current is conserved for radial flow in a region and 2) a source region generates an ideal TEM wave in a uniform coaxial geometry. As an example, suppose we have a source region for H pulses with axial length Δz and a current flux $\overline{\Gamma} = \Gamma_0 \hat{r}$. The total radial current in the region is $I_r = 2\Gamma_0 \Delta z$. Similarly, the axial current through an annular region with inner and outer radii r_i and r_o with an axial flux $\overline{\Gamma} = \Gamma_0 \hat{z}$ equals $I_z = 2\pi\Gamma_0(r_o - r_i)$. Table 3 shows input quantities for source current in the different program solution modes.

2.4 Standard boundary conditions

In the finite-element treatment of **EMP** an unspecified boundary automatically satisfies the special *Neumann* condition. Here, the normal derivative of the primary field component (E_z, rE_θ, H_z or rH_θ) equals zero. For E type pulses, the condition is equivalent to an open-circuit termination on the boundary. In this case, the magnetic intensity $\mathbf{H}$ is normal to the surface and the surface current density equals zero. A pulse is reflected from the boundary with the

same polarity and no change in amplitude. The condition is equivalent to a mirror condition for a symmetric system (equal values of E_z or rE_θ on both sides of the boundary). In contrast, a Neumann boundary is equivalent to a short-circuit condition (perfect conductor) for H type pulses. In this case, the electric field is normal to the surface. The outer boundaries of the solution volume are automatically set to the specialized Neumann condition. You can set internal Neumann boundaries using the *Void* command.

You can also define an internal region with the open-circuit property for E type pulses by assigning a high value of the medium characteristic impedance:

$$\frac{Z}{Z_o} = \sqrt{\frac{\mu_r}{\epsilon_r}} \gg 1.0, \tag{12}$$

where $Z_0 = 377.3\ \Omega$. When defining a high or low imepdance material, be sure to pick values of the relative dielectric constant and magnetic permeability such that

$$\sqrt{\epsilon_r \mu_r} \simeq 1.0. \tag{13}$$

Equation 13 ensures that the speed of light in the medium is close to c. If regions have large differences in the electromagnetic propagation velocity it may not be possible to find a practical time step consistent with the Courant condition. As an example, the values $\epsilon_r = 10^{-6}$ and $\mu_r = 10^6$ are a valid choice for a high-impedance material.

The special *Dirichlet* condition is a second option for external or internal boundaries. This type of boundary is defined by the *Fixed* command. In this case the calculated quantity (E_z, rE_θ, H_z or rH_θ) has the fixed value of 0.0. The special Dirichlet condition is equivalent to a short-circuit boundary (ideal conductor) for E type solutions. Here a pulse is reflected with inverted polarity. For H type pulses, a Dirichlet boundary is equivalent to an open-circuit condition (electric field vector parallel to the surface).

2.5 Absorbing boundaries

We can approximate free-space conditions in **EMP** by placing a resistive termination layer next to a boundary. The idea is similar to a matched termination at the end of a transmission line – the termination acts like an infinite extension of the line. The boundary region beyond the termination layer should have the following properties for the two types of pulses:

- **E type**: special Neumann condition.
- **H type**: fixed condition $H_z = 0.0$ or $rH_\theta = 0.0$.

In most cases the termination layer is located on the outer surface of the solution volume so that the Neumann condition for E type pulses is automatically applied. For H type pulses, it is necessary to add a line region with the *Fixed* condition.

As shown in Fig. 6*a*, a thin resistive layer with conductivity σ and width Δ is located between the boundary and a material with electrical properties ϵ and μ. In the limit that Δ is much smaller than the spatial width of the pulse, the electric field is approximately uniformly over the layer while **H** falls to zero. If the resistor maintains the same conditions on **E** and **H** as an infinite extension of the material, there is no reflected wave from the boundary. We can derive the required value of σ by noting that a field E_x creates a linear current density equal to

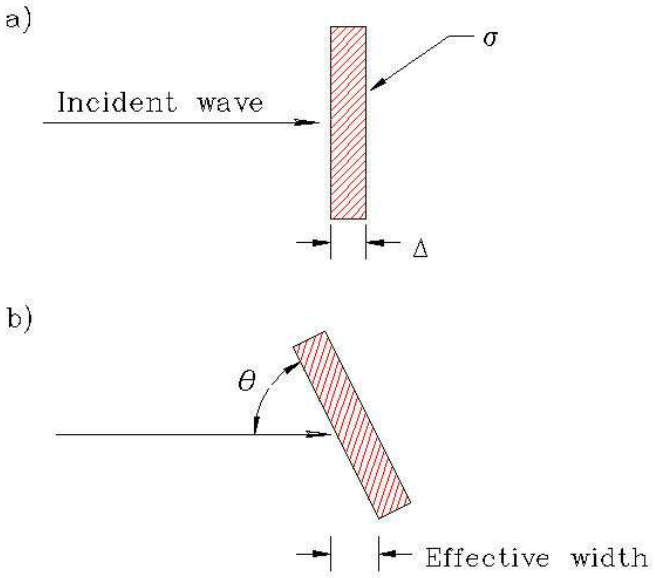

Figure 6: Absorbing boundary for two-dimensional solutions. *a*) Normal incidence. *b*) Incidence at an angle.

$$J_x = \sigma \Delta E_x. \tag{14}$$

The quantity $1/\sigma\Delta$ has units of ohms. Assuming zero magnetic field on the right-hand side of the resistor, the value of magnetic intensity of the left-hand side is $H_y = J_x$. Noting that the ratio E_x/H_y should equal the characteristic impedance of the medium, the matching condition is

$$Z_0 = \sqrt{\frac{\mu}{\epsilon}} = \frac{1}{\sigma\Delta}. \tag{15}$$

The conductivity for a matched termination is

$$\sigma = \frac{1}{\Delta\sqrt{\mu/\epsilon}}, \tag{16}$$

where ϵ and μ are the electrical properties assigned to both the termination layer and the adjacent material.

Figure 7 shows the performance of a termination boundary for normally incident pulses. A Gaussian pulse with spatial width 0.067 m enters from a vacuum region. The figure plots the reflected pulse amplitude as a function of the termination element width Δ. Absorption is better for thin elements that better approximate a lumped element. On the other hand, there is a penalty to using very thin elements because the Courant condition depends on the minimum width. As an example, for the baseline element width of 0.004 m, the termination element absorbs 99.86 per cent of the pulse energy.

The absorber has reduced effectiveness when pulses strike the boundaries a angles less than 90°(Fig. 6*b*). For an angle θ the effective layer thickness increases to $\Delta/\sin\theta$, reducing the surface resistance by a factor of $\sin\theta$. In this case, the termination is under-matched. The predicted reflection coefficient is

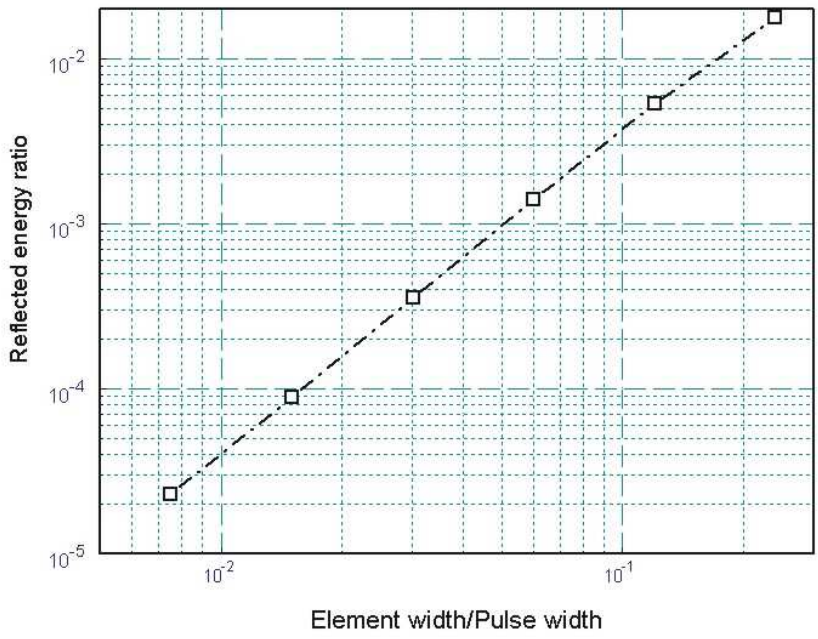

Figure 7: Performance of a matched termination boundary layer. Relative reflected energy as a function of the ratio of layer thickness to pulse spatial width (Gaussian pulse).

$$\frac{E_r}{E_0} = -\frac{1-\sin\theta}{1+\sin\theta}. \tag{17}$$

The degradation in performance is not catastrophic. A matched layer absorbs more than 90% per cent of the incident wave energy for $30^o \leq \theta \leq 90^o$.

Figure 8 shows a test of Eq. 17 using a Gaussian pulse with 0.067 m width incident from vacuum. Figure 18*a* shows results for a 45° corner reflector. The lines are contours of E_z separated by equal intervals. The absorbing layer on the bottom has $\Delta = 0.005$ m and conductivity $\sigma = 0.5301$ S/m. The top-left drive boundary creates the pulse, while the top left edge is an open symmetry boundary. The incident pulse with peak amplitude 1 V/cm is at the bottom-right. The reflected pulse traveling at a right angle has an amplitude of -0.17 consistent with the prediction of Eq. 17. Figure 18*b* shows an example with $\theta = 30^o$. The amplitude of the reflected pulse is -0.37 compared to a prediction of -0.33. One advantage of the termination layer method is that we can improve the performance if we know the general direction of the pulses. For example, if pulses strike the surface at an angle θ, we can optimize absorption by setting the layer conductivity equal to

$$\sigma = \frac{\sin\theta}{\Delta\sqrt{\mu/\epsilon}}\ . \tag{18}$$

Figure 18*c* shows improvement of the solution of Fig. 18*b* with a layer conductivity of $\sigma = 0.2650$ S/m. In this case, the reflection coefficient is only -0.025.

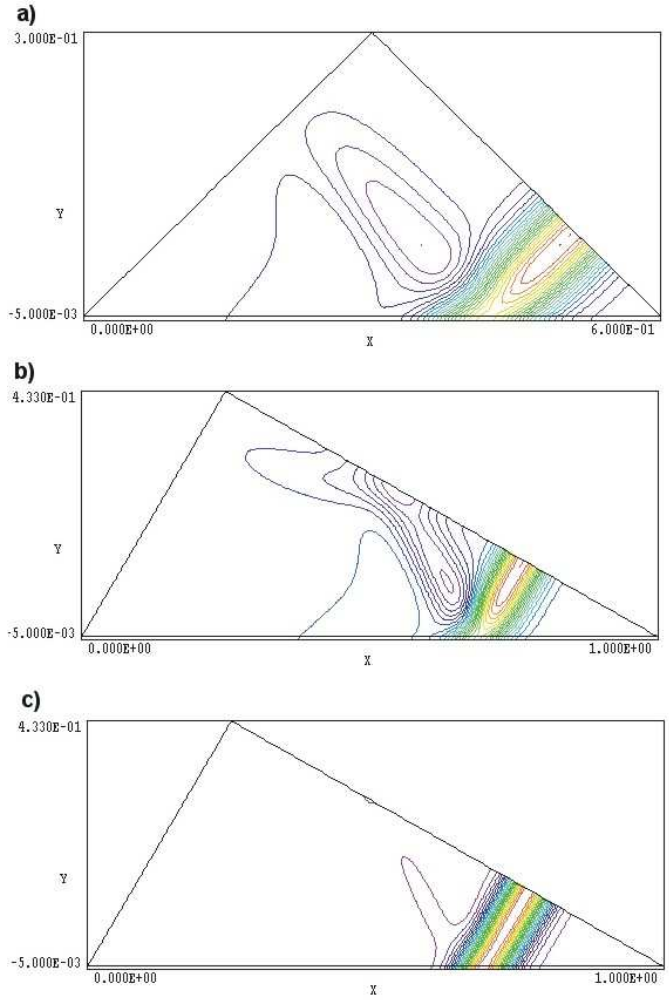

Figure 8: Performance of matched termination boundary as a function of incidence angle. Gaussian pulse in vacuum with spatial width of 0.067m. *a*) Contours of E_z showing incident and reflected pulses at a 45^o corner reflector. Absorbing layer at bottom ($\Delta = 0.005$ m, $\sigma = 0.5301$ S/m) at bottom and symmetry boundary at top right. *b*) Pulse reflection at $\theta = 30^o$. *c*) Improvement of 30^o pulse absorption by adjusting the layer conductivity to $\sigma = 0.2650$ S/m.

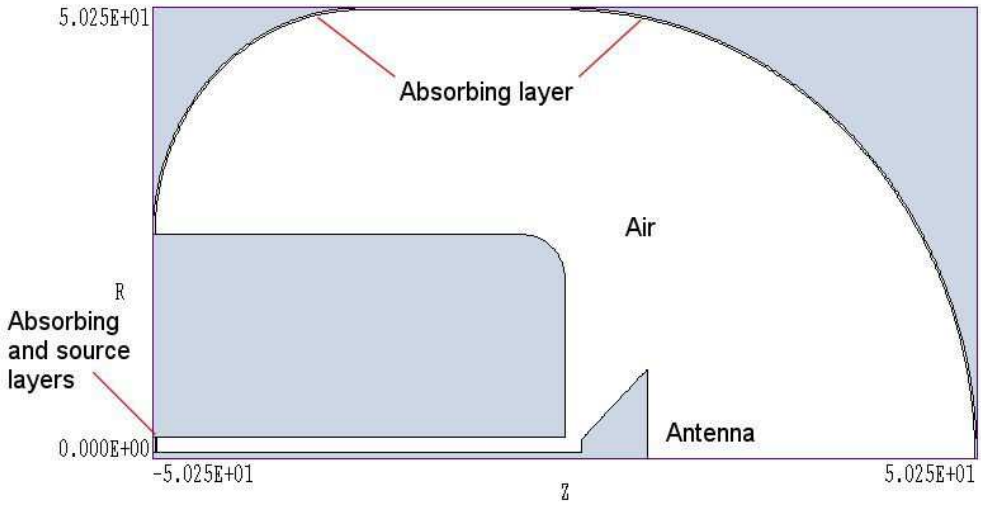

Figure 9: Geometry for the `CONICAL` example. A pulse on a transmission line in a finite-width ground plane drives a conical antenna. Dimensions in cm.

3 Creating and analyzing a solution with EMP

3.1 Mesh generation

As a quick introduction to **EMP** techniques and capabilities, we shall work through an example. Move the files `CONICAL.MIN` and `SQPULSE.SRC` (supplied with the example library) to a working directory such as `\tricomp\buffer`. The first file (input for the **Mesh** program) describes the geometry while the second defines the pulse waveform. All files are in text format – they may be inspected and modified with an editor. Figure 9 shows a z-r plot of the cylindrical system. (Note that parts are figures-of-revolution about the z axis). A source layer carrying a radial current (near the bottom left-hand side) creates a TEM pulse in the 70 Ω, air-filled coaxial transmission line ($r_o = 2.5$ cm, $r_i = 0.78$ cm). The line drives a conical antenna that produces a radiating pulse in the approximately spherical air volume. A matched absorbing layer on the periphery simulates the free-space condition.

The strategy for generating the mesh is influenced by the nature of the electromagnetic pulse solution (Chapter 2 reviews the theory of electromagnetic waves in two-dimensional systems.) A TEM pulse in a coaxial transmission line is an H type solution. **EMP** calculates the quantity rH_θ from a finite-element solution of the Helmholtz equation and then determines the electric field components (E_z, E_r) in terms of derivatives of the primary quantity. In particular, the condition $\partial(rH_\theta)/\partial n = 0$ holds on the surface of a perfect conductor. The Neumann condition automatically applies to all unspecified solution boundaries. Therefore, we use the following strategy for mesh generation:

- Fill the active solution volume (areas not shaded in gray in Fig. 9) with a resistive medium (Region 1) that will comprise the absorption layers in the transmission line and propagation volume. Note that the inner and outer conductors of the transmission line are

excluded from the solution volume.

- Carve out the inside of the solution volume with Region 2 representing air. The process leaves a absorbing layer of uniform thickness 0.25 cm around the propagation volume and also at the end of the transmission line. The line absorber terminates any reflected energy traveling backward.
- Define the thin radial slice that carries the current to initiate the pulse (Region 3).
- Include a line region to represent the fixed condition $rH_\theta = 0.0$. The region has three segments: 1) the axis $r = 0.0$, 2) the outer boundary of the transmission line absorber and 3) the outer boundary of the absorber on the periphery of the soloution volume.

Run **Mesh** and load `CONICAL.MIN`. Pick the *Edit script/graphics* command to enter the drawing editor. Here, you can use the display capabilities to confirm the region assignments:

- Region 1: complex outline of the solution volume including the absorbing layers (Filled)
- Region 2: the vacuum portion of the solution volume excluding the absorbing layers (Filled)
- Region 3: the source layer to initiate the pulse (Filled)
- Region 4: the lines that will assume the condition $rH_\theta = 0.0$ on the axis and adjacent to the absorbing layers (Open).

Because sources should not contact absorbers, there is a small separation between the source and the absorbing layer in the transmission line. Abandon the drawing and return to the main menu. Click the *Process* command. Choose the *Save mesh (MOU)* command. You can now close or minimize **Mesh**.

3.2 Creating the EMP script

Run **EMP** from **TC**. The screen is initially blank and the status bar indicates that the program is waiting for input. Note the prominent tools marked *1*, *2* and *3*. The notation is a reminder of the three steps in an electromagnetic solution:

1. Set up program controls and material properties.
2. Generate and solve the finite-element equations.
3. Analyze the solution.

To start the first operation, we must identify the **Mesh** output file that defines the system geometry. Click the *1* tool or the *Setup* menu command. In the dialog, choose the file `CONICAL.MOU`. **EMP** loads the information and displays the dialog of Fig. 10. Note that the grid contains a row for each mesh region with a notation of the *Fill* status.

The entries in the *Control parameter* section of the dialog have the following functions:

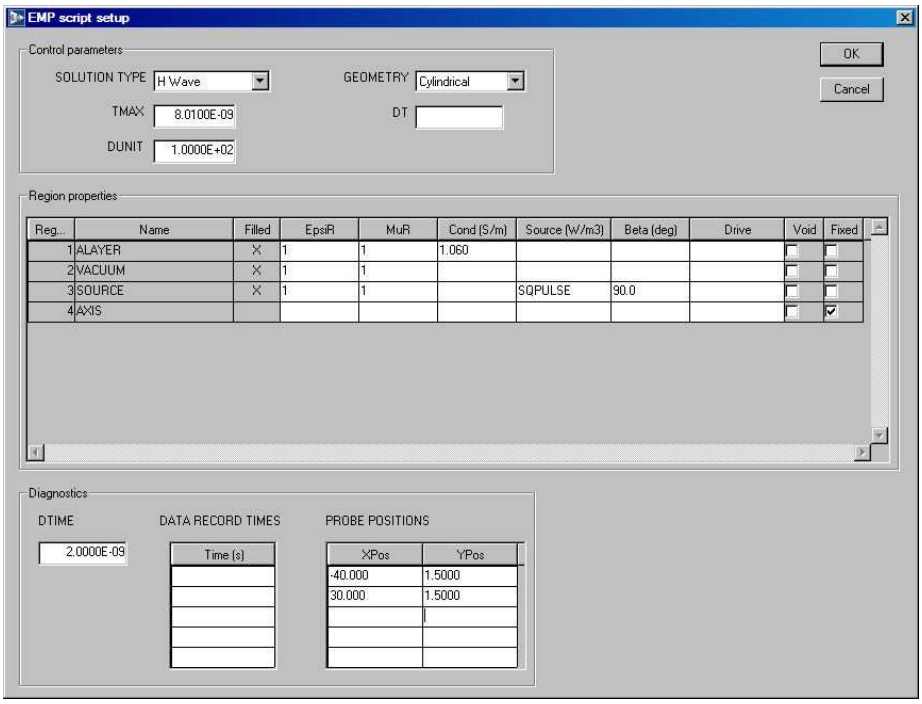

Figure 10: Dialog to set **EMP** control parameters and region properties.

- SOLUTION TYPE. Choose the field polarization. An entry of E designates that the main field component is E_z or rE_θ and H corresponds to H_z or rH_θ.
- GEOMETRY. Set the two-dimensional symmetry of the solution: *planar* or *cylindrical*. A planar solution varies in x-y and has infinite length in z, while a cylindrical solution has symmetry in θ.
- TMAX. The duration of the initial-value solution in seconds.
- DT. The time-step for the difference solution. If you leave the field blank, **EMP** estimates a value that satisfies the Courant stability condition in all elements.
- DUNIT. A factor to convert the units used for coordinates in the **Mesh** file to meters. Choose the appropriate units in the list box. For other units, pick *Custom* and enter a value for *DUnit* directly in the script with an editor. The quantity *DUnit* equals the number of mesh units per meter: 39.37 for inches, 100.0 for cm.

The entries in the column options of the region grid box determine the nature of the corresponding region. The first five columns apply to standard materials: *EpsiR* (ϵ_r, the relative dielectric constant), *MuR* (μ_r, the relative magnetic permeability), *Cond* (σ, the electrical conductivity in S/m), *Source* (the drive current density or its derivative) and *Beta* (the source direction of the current in degrees for H type solutions). The definition of the *Source* quantity depends on the field polarity and solution symmetry. Section 2.3 gives a detailed discussion.

The values shown in Fig. 10 define an H type solution with cylindrical symmetry. Dimensions are in cm and the maximum run time is 8.01 ns. Regions 1, 2 and 3 have the dielectric

properties of air ($\epsilon_r = 1.0, \mu_r = 1.0$). The conductivity in Region 1 is determined from Eq. 16 as

$$\sigma = \frac{1}{Z_0 \Delta}, \tag{19}$$

where Z_0 is the characteristic impedance of Region 2 (377.3 Ω) and the layer thickness is $\Delta = 2.5 \times 10^{-3}$ m. The source current rj_r of Region 3 points in the radial direction ($\beta = 90.0^{\circ}$) and is determined by values in the tabular file `SQPULSE.SRC`. The *Fixed* designation for Region 4 sets the condition $rH_\theta = 0.0$. The entries in the *Diagnostics* section call for four data dumps of spatial information at a time interval of 2.0 ns and two probes at positions [$r = 1.5$ cm, $z = -40.0$ cm] (inside the transmission line) and [$r = 30.0$ cm, $z = 30.0$ cm] (in the pulse propagation volume). When you have completed data entry, click *OK* and save the information in the file `CONICAL.EIN`.

We shall choose a peak source current $I_0 = 2.0$ A. In this case, 1.0 A is lost in the upstream absorber adjacent to the source and 1.0 A travels down the transmission line as a TEM pulse. For the cylindrical solution, we enter rj_r, where the peak value is given by:

$$rj_r = \frac{I_0}{2\pi\Delta}. \tag{20}$$

For a layer thickness $\Delta = 2.5 \times 10^{-3}$ m, the source magnitude is $rj_r = 127.3$ A/m. The table `SQPULSE.SRC` defines a normalized function that rises smoothly from 0.0 to 1.0 over the interval $0.0 \leq t \leq 1.0$ and falls to 0.0 over the interval $4.0 \leq t \leq 5.0$. We must introduce normalization factors for the desired amplitude and duration. We must edit the input file directly to implement this advanced feature. Click the **EMP** command *File/Edit script (EIN)* and choose `CONICAL.EIN`. Modify the *Source* command for Region 3 to read:

```
Source(3) = SQPULSE.SRC (0.20E-9,  127.3)
```

This first number is a multiplication factor for time values input from the file. The resulting full-width of the pulse is 1.0 ns. The second number adjusts the dependent variable. Save the modified file. Table 4 shows the final appearance of the script.

3.3 Finite-element solution

The next step is to generate and to solve the finite-element equations. Click the tool marked *2* or the menu command *Solve*. Accept the default entry of `CONICAL.EIN` in the dialog by clicking `OK`. The screen turns blue to indicate that the program is in solution mode and the status bar reports the progress of the operation. During the run, **EMP** creates five data dumps (`CONICAL.001`,...) and two probe files (`CONICAL.P01`,...). Section 4.7 describes the file formats.

3.4 Solution analysis

To transfer to the analysis menu, click the tool marked *3* or the menu command *Analyze*. The first step is to pick a solution series. Click on the command *File/Series – start file* and choose `CONICAL.001`. After the file is loaded, you should see the plot shown on the top of Fig. 11. It shows electric field lines (contours of rH_θ) at $t = 2.0$ ns. The leading edge of the

Table 4: **EMP** script `CONICAL.EIN`.

```
* ---- CONTROL ----
Mesh = conical
Solution = H
Geometry = Cylin
DUnit =   1.0000E+02
TMax =   8.0100E-09

* ---- REGIONS ----
* Region 1: ALAYER
Epsi(1) =   1.0000E+00
Mu(1) =   1.0000E+00
Cond(1) =   1.0600E+00
* Region 2: VACUUM
Epsi(2) =   1.0000E+00
Mu(2) =   1.0000E+00
Cond(2) =   0.0000E+00
* Region 3: SOURCE
Epsi(3) =   1.0000E+00
Mu(3) =   1.0000E+00
Cond(3) =   0.0000E+00
Source(3) = SQPULSE.SRC
Beta(3) = 90.0
* Region 4: AXIS
Fixed(4)

* ---- DIAGNOSTICS ----
DTime =   2.0000E-09
History =  -4.0000E+01  1.5000E+00
History =   3.0000E+01  1.5000E+00

EndFile
```

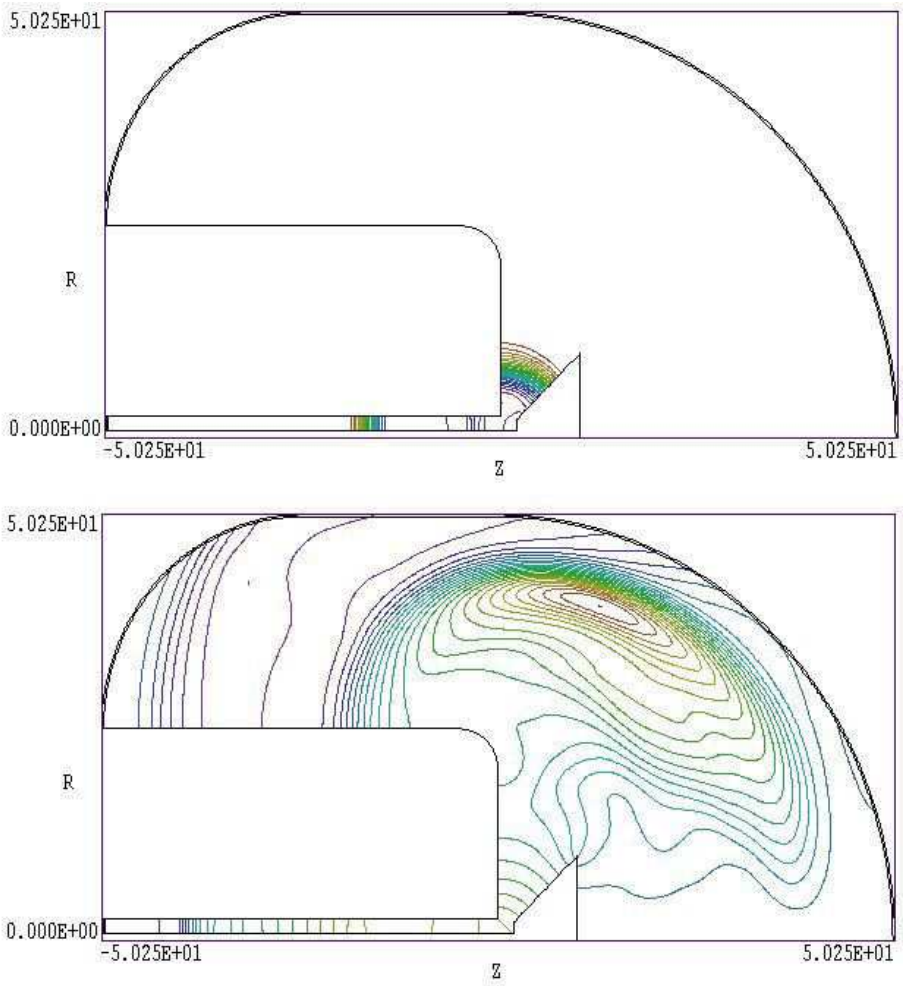

Figure 11: Electric field lines in the `CONICAL` example at 2 ns (top) and 4 ns (bottom).

pulse propagates through the antenna region. You can also see the trailing edge of the pulse in the transmission line and energy reflected from the antenna mismatch moving backward. The electric field distribution at $t = 4.0$ ns is shown on the bottom of Fig. 11. The launched electromagnetic pulse impinging on the absorbing layer is visible as well as energy moving upstream along the ground plane.

Signals from the two probes give quantitative data on field variations. Run the **Probe** program (supplied with the **EMP** package) and click on the command *Load probe file*. After you pick `CONICAL.PO1` the program displays information about the data file. Click *OK* to continue. Pick the command *Plot parameters/Pick plot quantity* and choose E_r. You should see the plot on the top of Fig. 12. For comparison, the theoretical electric field of a 1.0 A pulse at a radius of 1.51 cm in a 70 Ω transmission line is about 4000 V/m. The plot shows the incident pulse, a small negative reflection, and then a positive reflection from energy that is not radiated at the open-circuit antenna termination. The initial negative reflected pulse occurs when the incident pulse meets the antenna which acts like a conical radial transmission line. The characteristic impedance of a conical line is:

$$Z_0 = \sqrt{\frac{\mu}{\epsilon}}\ \frac{\tan\alpha}{2\pi}, \tag{21}$$

where α is the angle of the center conductor. Note that the impedance of the antenna in the example will be lower than this value because the apex is upstream from the ground plane. For $\alpha = 45^{\rm o}$, we find $Z_0 < 60\Omega$ giving a reflection coefficient $R_E > -0.08$. The bottom trace of Fig. 12 shows the radial electric field of the radiated pulse at the second probe ($r = 1.5$ cm, $z = 30.0$ cm).

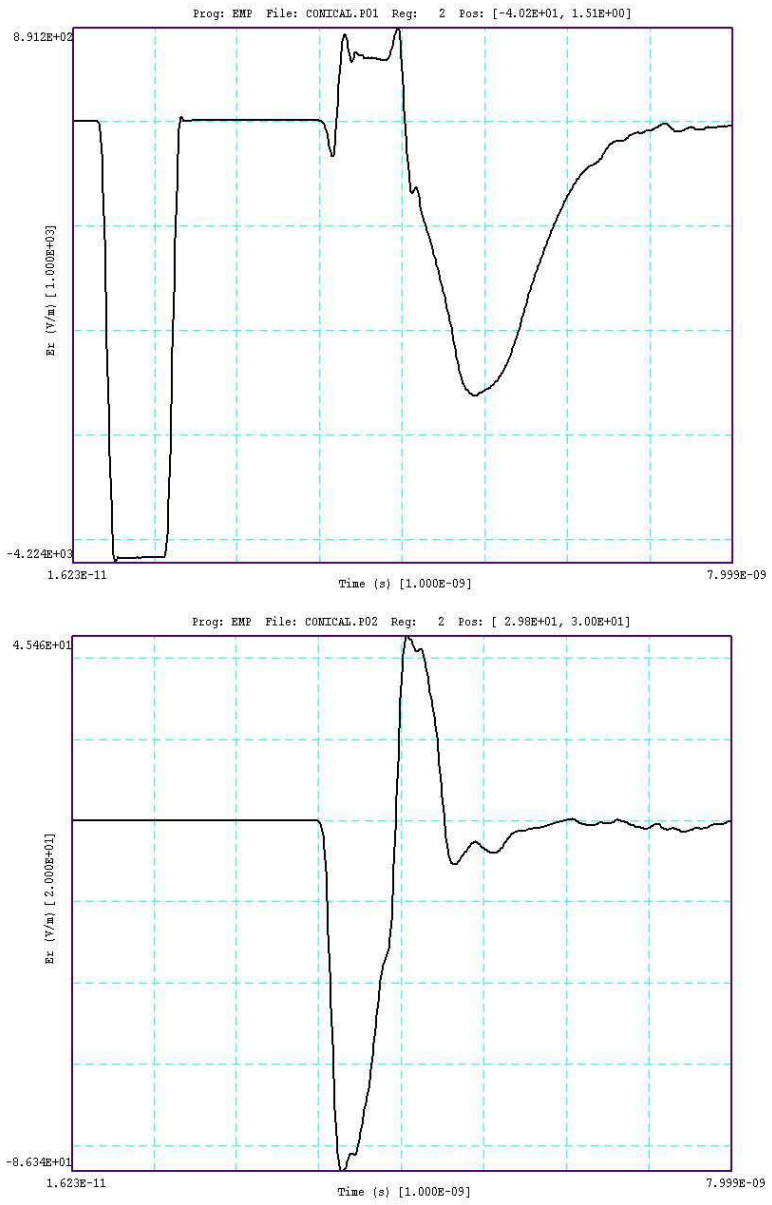

Figure 12: Radial electric field as a function of time for the `CONICAL` example in the transmission line (top) and in free space (bottom).

4 EMP solution reference

4.1 Organizing EMP solutions

The program `emp.exe` has two functions: 1) finite-element calculation of initial-value solutions for electromagnetic pulses and 2) analysis of output dumps (spatial information at specified times). The package also includes the program `probe.exe` to display information in history files (temporal information at specified locations). **EMP** can run in two modes: interactively in a window or autonomously in the background with the option for batch file control. The latter mode allows automatic processing of large or repetitive data sets.

EMP may use the following input files:

- A **Mesh** output file that describes the conformal triangular mesh. The file contains node coordinates and the region numbers of elements and vertices. (Required)
- A command script that sets control parameters for the solution program and describes the physical properties associated with region numbers. (Required)
- Tables that describe waveforms of drives and sources (Optional).

The mesh output file always has a name of the form `MPREFIX.MOU`, where `MPREFIX` is a valid file prefix (1 to 32 characters). The command script must have a name of the form `EPREFIX.EIN`. **EMP** issues an error message if both input files are not available in the working directory. To organize data, the resulting output files have names that begin with `EPREFIX`.

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

- Prepare a **Mesh** script (`MPREFIX.MIN`) following the instructions in the **Mesh** manual. You can create the file directly with a text editor or graphically using the drawing editor.
- Run **Mesh** either interactively from the **TC** program launcher or from the Windows Command Prompt to create the file `MPREFIX.MOU`.
- Prepare a command script (`EPREFIX.EIN`) using either the *Setup* command in the **EMP** main menu (Sect. 3.2) or your own text editor. The allowed file commands are described in this chapter.
- Run **EMP** to create data dumps (`EPREFIX.001, EPREFIX.002,...`) or history files (`EPREFIX.P01, EPREFIX.P02,...`). All output files are in text format. The data dumps contain information on the mesh geometry, the physical properties of regions, and values of computed quantities. In electromagnetic solutions the region properties are the relative magnetic permeability, the relative dielectric constant and the conductivity. The computed field quantities depend on the field polarity and the solution symmetry.

- Analyze data dumps in the *Analysis* menu of **EMP**. You can also transfer the information in the data dumps to your own analysis programs.
- Plot history files using **Probe**.

4.2 EMP script format

The **EMP** script is a text file composed of data lines that contain commands and parameters. You can construct a script using the *Setup* dialog in **EMP** (Sect. 3.2). You can also write and modify scripts with an editor. Direct script editing is required for some advanced **EMP** capabilities. Section 4.3 covers commands that control general program operation. Section 4.4 describes commands to set simple properties of material regions. Advanced methods to define regions with material properties that vary in space are discussed in Chap. 5. Finally, Sect. 4.5 covers diagnostic commands that control the generation of data dumps.

The script must end with the *EndFile* command. The entries on a line may be separated by the standard delimiters introduced in the **Mesh** manual:

- Space [' ']
- Comma [',']
- Tab
- Colon [':']
- Left parenthesis ['(']
- Right parenthesis [')']
- Equal sign ['=']

Any number of delimiters may be used in a line. Blank lines and comment lines are ignored. Comment lines begin with an asterisk (*). **EMP** accepts commands in any order. Table 5 shows an example of complete script.

There are three classes of commands: program control, region properties and diagnostics. Control and diagnostic commands contain a keyword and a value. Region commands set the physical properties associated with elements and nodes. They have the format:

```
Keyword  RegNo  Value
```

Here, the integer *RegNo* is the region number defined in the **Mesh** input file. The string *Keyword* specifies the physical property. The value may be one or more numbers. As an example, the command

```
EPSI(2) = 7.2
```

sets the relatitive dielectric constant in elements with region number 2 to the value 7.2. (**Note**: For back-compatibility, **EMP** recognizes command formats from Version 1.0 through 5.0 where control commands start with the keyword *Set*, region commands with *Region* and diagnostic commands with *Diag*).)

Table 5: Example of an **EMP** control script

```
* EMP Test
* Plane wave absorption, normal incidence
* Gaussian pulse with FWHM of 0.222 ns
* Propagation distance: 0.125 + 0.100 m
* Matched conductance: Sigma = 1/Zo*Deltaz = 0.5301 mhos/m

TMax = 1.95E-9
Geometry = Rect
DUnit = 1.0
Solution = E
dT = 5.0e-12

Mu(1) = 1.0
Epsi(1) = 1.0
Cond(1) = 0.5301

Mu(2) = 1.0
Epsi(2) = 1.0

Drive(3) = GAUSS.SRC

* Propagate 0.125 m
SetTime = 1.084E-9
History = 0.30 0.125

ENDFILE
```

4.3 Control commands

EMP commands divide into three groups: 1) control of the solution, 2) definition of the physical properties of regions, and 3) diagnostics. We begin with the program commands. They begin with a keyword followed by one or more parameters. In the following, each command is written symbolically and as it might appear in a script:

MESH MPrefix
MESH = SparkGap
Specify the **Mesh** output file that defines the geometry of the simulation. The file must have a name of the form `MPREFIX.MOU` and be available in the working directory. If this command does not appear in a script with the name `EPREFIX.EIN`, then **EMP** will seek the default mesh file `EPREFIX.MOU`.

GEOMETRY [Rect,Cylin]
GEOMETRY = Cylin
Specify whether the solution has planar symmetry (variations in x and y with infinite length in z) or cylindical symmetry (variations in r and z, uniform in θ).

DUNIT DUnit
DUnit = 1.0E6
Set the interpretation of spatial coordinates from the mesh input file and the *History* command. The spatial conversion factor equals the number of spatial units per meter. For example, if coordinates in **Mesh** are given in mm, set $DUnit = 1000.0$. (Default, $DUnit = 1.0$).

SOLUTION [E,H]
SOLUTION = H
Set the field polarization of the solution. An E type solution has primary field components E_z (planar) or rE_θ (cylindrical) with secondary components H_x and H_y (or H_z and H_r). An H type solution has primary field components H_z (planar) or rH_θ (cylindrical) with secondary components E_x and E_y (or E_z and E_r).

TMAX TMax
TMAX = 34.5E-9
The real number $TMax$ equals the duration of the solution in seconds.

NSTEP NStep
NStep = 4500
An alternate way to limit the run time by setting the maximum number of computational steps. Enter an integer value. The default is 100,000.

INTERPOLATION [Linear, Spline]
INTERPOLATION = Linear
Sets the interpolation method for input tables that define time variations of drives and sources. If you have a well-behaved table, the *Spline* option gives better accuracy. On the other hand, a spline fit gives poor results if the table contains discontinuous or noisy data. (The default is *Spline*.)

4.4 Region commands

Region commands set the physical properties of the regions of the solution volume defined in **Mesh**. The second entry in all commands is the region number, an integer. This section discusses how to assign uniform properties over the region volume. Chapter 5 covers spatial variations of properties over regions defined by mathematical functions. The first three commands discussed in this section assign properties to the elements of material regions (those with non-zero volume). **EMP** recognizes three properties: relative dielectric constant, relative magnetic permeability and electrical conductivity. The next three commands set boundary conditions. The final three commands add source current density to material regions.

MU RegNo MuR
MU(1) = 250.0
Assigns a value of the relative magnetic permeability ($\mu_r = \mu/\mu_0$) to all elements with the corresponding region number. The default is $\mu_r = 1.0$.

EPSI RegNo EpsiR
EPSI(7) = 7.3
Assigns a value of the relative dielectric constant ($\epsilon_r = \epsilon/\epsilon_0$) to all elements with the corresponding region number. The default is $\epsilon_r = 1.0$.

COND RegNo Sigma
COND(3) = 1.060
Assign a value of electrical conductivity to all elements with the corresponding region number. Enter the conductivity in S/m (siemens per meter). In SI units, the conductivity is the inverse of the volume resistivity, $\sigma = 1/\rho$, with ρ specified in Ω-m. The default is $\sigma = 0.0$.

FIXED RegNo
FIXED(4)
Assign the fixed value of 0.0 to the primary quantity (E_z, H_z, rE_θ or rH_θ) for all nodes of the region. The fixed field value holds throughout the run. In cylindrical solutions, set the axis ($r = 0.0$) to the *Fixed* condition.

VOID RegNo
VOID(8)
Set an internal region to the *Void* condition. The special Neumann condition applies on the

boundaries with standard material regions. In this case, **EMP** treats all nodes and elements of the region as though they were not included in the solution volume. A void region acts as though it has characteristic impedance $\eta = \infty$ for E type pulses and $\eta = 0$ for H type pulses. The main use of a void is to represent internal conducting volumes for H type solutions. You could achieve a similar effect by setting $\epsilon_r = 10^6$ and $\mu_r = 10^{-6}$ manually. The difference is that **EMP** will not try to plot field lines inside a void.

There are two important considerations for using internal voids:

- If you create a void region in a mesh by over-writing a portion of a material region, then **Mesh** will assign the void region number to all nodes including those on the boundary. For a correct **EMP** solution, the boundary nodes should have the region number of the surrounding material. Therefore, you must insert a line region after the definition of the void to change the region number on the boundary. In **EMP** assign properties of the surrounding material to the extra region. The example `RODH` illustrates the setup technique.
- Use internal voids only if the region does not connect to the solution space boundary. Otherwise, define the boundary shape to exclude conducting regions as in the `CONICAL` walkthrough example.

DRIVE RegNo FileName [TMult DMult]
DRIVE(5) = Gauss.DRV

This command defines a region where the primary field quantity follows a specified variation in time. Drive regions are normally used to generate pulses. The string parameter is the full name of a tabular function file in the current directory that specifies the time variation of E_z, H_z, rE_θ or rH_θ. Note that the tabular function routines return values of zero when the time exceeds the largest entry on the table. Therefore, a drive boundary assumes the *Fixed* condition after its active period. To define an open circuit condition after the drive pulse, use the *TExpire* command for the region. When the optional real number parameters *TMult* and *DMult* are present, the program multiplies time and drive quantities as they are loaded from the table. The default is $TMult = 1.0$, $DMult = 1.0$.

A tabular function is a text file consisting of up to 256 entry lines of values for the time and the dependent variable. You can prepare table files with a text editor or spreadsheet. You can also use published data or digitized experimental traces. As an example, Table 6 shows the tabular function `GAUSS.DRV` supplied with the code to define a Gaussian pulse. Note that the file syntax conforms to the same rules as the **EMP** script. The free-form parser accepts real numbers in any format with a choice of delimiters. You can add documenting comment lines starting with an asterisk (*). The end of the data is marked with the *EndFile* command. Although the data lines of the example are ordered by increasing value of the independent variable, this ordering is not required. **EMP** sorts the list before use and records the final order in the listing file, `FPREFIX.ELS`. Furthermore, the independent variable intervals need not be uniform. A table requires a minimum of 5 entries. The interpolation routines return 0.0 for values of the t outside the range of the table.

Table 6: Table to define a Gaussian pulse.

```
* Gaussian pulse shape, 0.222 ns FWHM, amplitude 1.0
* Peak at 0.6667 ns
  0.0000E+00  1.3886E-11
  1.3333E-11  3.7370E-11
  2.6667E-11  9.8581E-11
  4.0000E-11  2.5490E-10
  5.3333E-11  6.4605E-10
  6.6667E-11  1.6050E-09
  ...
  5.3333E-10  3.6787E-01
  5.4667E-10  4.4485E-01
  5.6000E-10  5.2728E-01
  5.7333E-10  6.1261E-01
  5.8667E-10  6.9766E-01
  6.0000E-10  7.7879E-01
  6.1333E-10  8.5213E-01
  6.2667E-10  9.1392E-01
  6.4000E-10  9.6078E-01
  6.5333E-10  9.9005E-01
  6.6667E-10  1.0000E+00
  6.8000E-10  9.9005E-01
  6.9333E-10  9.6080E-01
  7.0666E-10  9.1394E-01
  7.2000E-10  8.5215E-01
  7.3333E-10  7.7881E-01
  7.4666E-10  6.9769E-01
  ...
  1.7000E-09  0.0000E+00
  1.8000E-09  0.0000E+00
  1.9000E-09  0.0000E+00
  2.0000E-09  0.0000E+00
  2.5000E-09  0.0000E+00
  3.0000E-09  0.0000E+00
ENDFILE
```

DRIVE RegNo > FunctionString
DRIVE(5) > 10.5*SIN(6.0E8*$t)
Define the time variation of a drive region from a mathematical function. Section 5.2 gives a detailed description of the command format.

TEXPIRE RegNo TExp
TEXPIRE(7) = 3.5E-9
Define an expiration time for a drive region. The boundary reverts to an unspecified (Neumann) condition after a time limit. The quantity $TExp$ is the expiration time in seconds. The drive condition determined from a table or mathematical function holds in the interval $0.0 \leq t \leq T_{exp}$. Pulses pass through the boundary unimpeded after the expiration time. The main use of this option is to create a pulse inside a volume and then to follow propagation with open space (E type solution) or conducting (H type solution) boundaries.

SOURCE RegNo FileName [TMult SMult]
SOURCE(3) = Coupler.SRC (2.0E-9, 150.0)
Include a spatially-uniform current source with a specified time variation in a material region. Sources represent current loops or capacitive probes that drive RF pulses. The string is the full name of a tabular function file in the current directory that specifies the time variation of the source function. The two optional real-number parameters are multiplication factors for the time ($TMult$) and source ($SMult$) values. With this feature you create a library of normalized waveforms that can be incorporated into a variety of solutions. The default values are $TMult = 1.0$ and $SMult = 1.0$. The interpretation of the source function value in **EMP** depends on the geometry and solution type. Table 3 lists the options. **Important note**: source regions should not be in direct contact with absorbing regions ($\sigma > 0.0$).

SOURCE RegNo > FunctionString
SOURCE(6) > 0.5*(1.0 - COS(6.7E8*$t))
Define the time variation of a source region from a mathematical function. Section 5.2 gives a detailed description of the command format.

BETA RegNo Beta
BETA(2) = 90.0
Set the direction of the region current density in the x-y or z-r plane for H type solutions. The angle β (in degrees) is the direction of the current density relative to the x or z axis. The angle equals $\tan^{-1}(j_y/j)x)$ or $\tan^{-1}(j_r/j_z)$. The default is $\beta = 0.0^{o}$.

TABSHIFT RegNo TMult TOff VMult VOff
TABSHIFT(7) = (1.0, 5.0, 1.0E4, 0.0)
Adjust the interpretation of time-dependent functions from tabular files or mathematical functions for drive and source regions. There are four real number parameters. The first, ($TMult$) is a multiplication factor for values of time and the second ($TOff$) is an offset. The default values are 1.0 and 0.0. The third ($VMult$) is a multiplication factor and the fourth ($VOff$)

is an offset for the dependent variable. This program function allows you to apply normalized tables to a variety of applications. If t is the program time, the value passed to the tabular function or mathematical function is

$$t_{pass} = T_{mult} \times (t - T_{off}). \tag{22}$$

If the interpolation or function returns a value v, the quantity used in the program is

$$v_{pass} = V_{mult} \times v + V_{off}. \tag{23}$$

The command affects only *Drive* or *Source* regions that are associated with a table or mathematical function. Note that the *TabShift* command supplements the $TMult$, $DMult$ and $SMult$ parameters in the *Drive* and *Source* commands.

4.5 Diagnostic commands

Diagnostic commands control data output from the program. **EMP** produces three types of data, all in text format:

- From 1 to 999 data dump files with the names of the form `FPREFIX.001`, `FPREFIX.002`, ... Each file contains a complete record of spatial information (the mesh geometry and distributions of field and material quantities) at a specified time. These files are used as input in the *Analyze* menu of **EMP**.
- A listing file with the name `FPREFIX.ELS` that can be inspected with a text editor. This file contains a variety of information about the **EMP** run, useful mainly to check the validity of input parameters and to diagnose problems.
- From 1 to 20 history files with the names of the form `FPREFIX.P01`, ... , `FPREFIX.P20`. Here, the magnetic and electric field components at one or more probe locations are recorded at each time step of the solution. The information may be inspected with a text editor or displayed with the utility program **Probe** (described in Chap. 8).

SETTIME STime
SETTIME = 2.5E-9
Record a data dump at or immediately after a specified time. The quantity $STime$ is the simulation time (in seconds) for the data record. When using automatic time step selection, it is uncertain that **EMP** will complete a cycle exactly at the specified time. The convention is that the code records a file when the time is equal to or greater than $STime$. You can define up to 100 record times with multiple *SetTime* commands. The only restriction is that the commands must appear in chronological order.

DTIME DTime
DTIME = 1.5E-10
Record data dumps at a uniform time interval. Enter the interval $DTime$ in seconds. As with the *SetTime* command, there is no way to guarantee that the target diagnostic times will fall

exactly on cycle times. **EMP** records a data dump as soon as possible after the calculated target time.

NSTEP NStep
NSTEP = 250
Create data dumps separated by a uniform number of computational steps. The integer $NStep$ is the number of cycles per data dump. You can record data at exact intervals by using this option and setting the time step with the Dt command.

HISTORY XPos YPos
HISTORY ZPos RPos
HISTORY = (2.50, 3.75)
Record field values as a function of time at a specified location. Enter the target coordinates for the probe location in distance units set by $DUnit$. The real number inputs are $[x, y]$ for planar problems and $[z, r]$ for cylindrical problems. The probe coordinates that appear in the history file usually are slightly different from those specified. **EMP** replaces the target coordinates with the center-of-mass coordinates of the element that contains the target point. The spatial derivatives to find secondary quantities are determined by a linear fit using primary field values at the three element nodes. The input file can contain up to 20 *History* commands in any order.

NHSTEP NHStep
NHSTEP = 10
Set the number of computational time steps per record in the history files, where $NHStep$ is an integer. This command allows you to reduce the number of entries in the history file for long runs and small time steps. The default value is $NHStep = 1$.

MOVIE NFrame [Nx Ny] [QMin QMax]
MOVIE 100 800 600 0.0 110.0
Create a set of N_{frame} graphics files in `BMP` format separated by time interval t_{max}/N_{frame}. The plots show electric or magnetic field lines (depending on the solution mode) in the full solution volume. The number of lines is determined by any previous setting in the *Analysis* menu. The integer parameters N_x and N_y give the horizontal and vertical image size in pixels. If they do not appear, the default is $N_x = 1024$ and $N_y = 768$. The two real-number parameters are the upper and lower temperature limits for the generating function. Autoscaling is applied to each frame if they are absent. You can use our utility program **Cecil_B** to stitch the files into a movie in `AVI` format. The program is available at

`http://www.fieldp.com/cecil_b.html`

4.6 Running EMP

EMP can run as an interactive program in a window or as a background task. To run the program from the command prompt, use a command of the form:

```
[ProgPath\]EMP [DataPath\]EPrefix <ENTER>
```

where the file `EPrefix.EIN` and the appropriate **Mesh** output file are available in the data directory. With this capability, you set up extended autonomous **EMP** runs using a DOS batch file or a Perl script.

The remainder of this section discusses commands in the main menu when **EMP** runs in the interactive mode. The program enters this mode when launched from **TC** or run with no input file prefix. The following commands appear in the *File* popup menu:

EDIT SCRIPT (EIN)
EDIT LISTING (ELS)
EDIT FILE

These commands call the internal editor to inspect or to modify **EMP** input and output files. Choosing a file from an alternate directory does not change the working directory. The *Edit script (EIN)* command shows a list of all files with names of the form `FPREFIX.EIN` while *Edit listing (ELS)* displays files with names `FPREFIX.ELS`.

RUN ANALYSIS SCRIPT

An analysis script allows you to perform complex or repetitive analyses on a set of similar solutions. This command displays a dialog listing files with the suffix `SCR`. Pick a file and click *OK*. The script can load data files, open and close data records, and perform any of the quantitative analysis functions described in this chapter. The script command language is described in Sect. 6.5. Note that the analysis script must be in the same directory as the data files.

SETUP

The function of this command is to create an **EMP** script to control an electromagnetic solution. The program first prompts for a **Mesh** output file to define the system geometry. The prefix of the file will be used as the argument of the *Mesh* script command. The program then displays the dialog shown in Fig. 11. The number of regions in the dialog is determined by the **Mesh** file. The functions of the control parameters in the upper box are described in Sect. 4.3. You can enter basic physical properties of regions in the grid box (see Sect. 4.4). You must edit the script directly to invoke advanced functions.

The *Solve* popup menu includes the following two commands.

RUN

Pick an input file (such as `EPrefix.EIN`) to start a solution. The working directory is changed if you pick a file from an alternate directory. The run begins if the file `EPrefix.MOU` or the file specified in the *Mesh* command is present. During the solution, the screen color is blue and the progress is shown in the status bar.

STOP

This command terminates the **EMP** run. Any data dumps created before this time will be available.

SOLUTION DISPLAY

When *solution display* is active, **EMP** plots a screen plot of electric or magnetic field (depending on the solution mode) whenever a data dump is created. The number of lines is determined by the last setting in the *Analysis* menu.

ANALYZE

Call up the analysis menu for plotting and numerical analysis.

EMP MANUAL

Display this manual in your default PDF display sofware. The file `emp.pdf` must be available in the same directory as `emp.exe`.

4.7 Format of the EMP output file

The **EMP** data dump files (`EPREFIX.001, EPREFIX.002, ...`) are in text format. The files have three sections:

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

The header section consists of a title line and 11 data lines:

```
--- Run parameters ---
XMin: -1.270003E-01
XMax:  1.270003E-01
KMax:  101
YMin:  0.000000E+00
YMax:  2.540005E-01
LMax:  101
DUnit:  3.937000E+01
NReg:    5
ICylin:  1
Time:  2.002203E-09
Type: H
```

Lines 1 and 2 list x_{min} and x_{max}, the limits along the horizontal axis (x or z) of the solution volume. Dimensions are recorded in meters. The quantity k_{max} in Line 3 is the number of nodes along the horizontal direction. Lines 4-6 describe the vertical axis (y or r). Line 7 contains the quantity $DUnit$, the conversion factor from dimensions used in **Mesh** to meters. Line 8 gives the number of regions in the solution volume, while Line 9 specifies the symmetry (0: planar, 1: cylindrical). Line 10 is the simulation time when the file was created and Line 11 is the polarity of the solution. The values in Lines 9-11 are used to set labels and calculated quantities in the *Analyze* menu.

The node section (Table 7) consists of 4 title lines and $k_{max} \times l_{max}$ data lines, one for each node of the solution space. Each data line contains the following quantities:

- The indices of the node (K,L)
- The region number of the node ($RgNo$) and region numbers for two associated elements ($RgUp$ and $RgDn$). The upper element lies above the line between nodes (K,L) and (K+1,L) and the lower element lies below the line.
- The coordinates of the node in meters, (x,y) or (z,r).
- The primary field quantity A. The recorded quantity depends on the solution field polarity and symmetry: E_z (V/m), rE_θ (V), H_z (A/m) or rH_θ (A).
- The x or z component of the secondary field quantity in the upper and lower elements ($AxzUp$, $AxzDn$). The recorded quantity depends on the solution field polarity and symmetry: H_x (A/m), H_z (A/m), E_x (V/m) or E_z (V/m).
- The y or r component of the secondary field quantity in the upper and lower elements ($AyrUp$, $AyrDn$). The recorded quantity depends on the solution field polarity and symmetry: H_y (A/m), H_r (A/m), E_y (V/m) or E_r (V/m).
- The material properties in the upper and lower elements, ϵ_r (relative dielectric constant), μ_r (relative magnetic permeability) and σ (electrical conductivity).

The region section consists of four title lines followed by $NReg$ data lines, one for each region. For **EMP** solutions, the section contains the following information for each region:

- $RegNo$: region number.
- Fix: 1 for fixed (Dirichlet) regions, 0 otherwise.
- $Void$: 1 if the region is a void, 0 otherwise.
- $Epsi$: relative dielectric constant for uniform regions.
- Mu: relative magnetic permeability for uniform regions.
- $Cond$: electrical conductivity in S/m for uniform regions.
- $Area$: cross-section area of the region in m^2.
- $Beta$: angle of the current density in degrees relative to the x or z axis (quantity meaningful only for source regions in H type solutions).

Table 7: Example – node section of the **EMP** data dump

```
--- Vertices ---
  k     l  RgNo RgUp RgDn        x              y              A
=====================================================================
 ...
 242     1     4     2     0  1.000000E-01  0.000000E+00  0.000000E+00
 243     1     4     2     0  1.025759E-01  0.000000E+00  0.000000E+00
 244     1     4     2     0  1.051469E-01  0.000000E+00  0.000000E+00
 245     1     4     2     0  1.077111E-01  0.000000E+00  0.000000E+00
 246     1     4     2     0  1.102677E-01  0.000000E+00  0.000000E+00
 ...


       AxzUp          AxzDn          AyrUp          AyrDn          EpsiUp
 ============================================================================
  ...
  -2.411373E-23  0.000000E+00  0.000000E+00  0.000000E+00  8.854188E-12
  -2.283328E-23  0.000000E+00  0.000000E+00  0.000000E+00  8.854188E-12
  -2.068102E-23  0.000000E+00  0.000000E+00  0.000000E+00  8.854188E-12
  -1.793478E-23  0.000000E+00  0.000000E+00  0.000000E+00  8.854188E-12
  -1.490329E-23  0.000000E+00  0.000000E+00  0.000000E+00  8.854188E-12
  ...

      EpsiDn          MuUp           MuDn          CondUp         CondDn
 ============================================================================
   ...
   0.000000E+00  1.256637E-06  0.000000E+00  0.000000E+00  0.000000E+00
   0.000000E+00  1.256637E-06  0.000000E+00  0.000000E+00  0.000000E+00
   0.000000E+00  1.256637E-06  0.000000E+00  0.000000E+00  0.000000E+00
   0.000000E+00  1.256637E-06  0.000000E+00  0.000000E+00  0.000000E+00
   0.000000E+00  1.256637E-06  0.000000E+00  0.000000E+00  0.000000E+00
   ...
```

5 Mathematical functions for drives, sources and material properties

5.1 Function parser

You can define temporal variations of drive and source regions and set spatial variations of material properties over a region using mathematical functions. For this purpose, **EMP** features a flexible and robust algebraic function interpreter. A *function* is a string (up to 230 characters) that may include the following entities:

- Independent variables. For temporal functions, the variable is $t (the adjusted simulation time). Spatial functions are defined in terms of `$x` and `$y` (planar solutions) and `$z` and `$r` (cylindrical solutions).
- Real and/or integer numbers in any valid format (*e.g.*, `3.1415`, `476`, `1.367E23`, `6.25E-02`, `8.92E+04`,...). Integers are converted to real numbers for evaluation.
- Binary operations: `+` (addition), `-` (subtraction), `*` (multiplication), `/` (division) and ∧ (exponentiation).
- Functions: `abs` (absolute value), `sin` (sine), `cos` (cosine), `tan` (tangent), `ln` (normal logarithm), `log` (base 10 logorithm), `exp` (normal exponent) and `sqt` (square root).
- Up to 20 sets of parentheses to any depth.
- Any number of space delimiters.

The parser conforms to the standard algebraic rules and includes error checking features. Errors may include unbalanced parentheses, unrecognized characters and sequential binary operations. As an example, the expression

```
1 - exp(-1.0*(($z^2 + $r^2)/24))
```

corresponds to

$$1 - \exp\left[-\left(\frac{z^2 + r^2}{24}\right)\right]. \tag{24}$$

5.2 Temporal variations of drives and sources

Section 4.4 described how to use the *Drive* and *Source* commands with tabular functions. The commands have alternate forms if you want to set temporal variations using mathematical functions.

DRIVE RegNo > FunctionString
DRIVE(5) > 10.5*SIN(6.0E8*$t)
Define the time variation of a drive region from a mathematical function. The function is a string up to 230 characters in length that follows the '>' symbol. The only allowed variable is the time `$t`.

SOURCE RegNo > FunctionString
SOURCE(6) > 0.5*(1.0 - COS(6.7E8*$t))
Define the time variation of a source region from a mathematical function. The function is a string up to 230 characters in length that follows the '>' symbol. The only allowed variable is the time `$t`.

The time `$t` normally corresponds to the simulation time t. If the script includes a *TabShift* command for the region, **EMP** substitutes the value:

$$\texttt{\$t} = T_{mult} \times (t - T_{off}). \tag{25}$$

The value returned by the function is adjusted according to:

$$f' = V_{mult} \times f + V_{off}. \tag{26}$$

5.3 Spatial variations of material properties

EMP can represent continuous spatial variations of the material quantities ϵ_r, μ_r and σ over regions that follow mathematical functions. Use the following command to set a spatial variation of relative dielectric constant.

EPSI RegNo > Function
EPSI(7) > 1.0 + ($x - 0.25)/5.50
EPSI(4) > 1.0 - 0.625*$z ∧2
Set values of ϵ_r in the elements of a volumetric material region according to a specified function of space. The keyword *Epsi* followed by the region number and the > symbol designates that a function string occupies the remainder of the line. The function may be up to 230 characters in length and follows the format described in Sect. 5.1. For spatial functions, the parser recognizes the variables `$x` and `$y` for planar solutions and $z and $r for cylindrical solutions. The dielectric constant in an element equals the value of the function evaluated at the center-of-mass of the element. You must ensure that the function returns a physically valid value over all elements of the region.

Note that there is no limit on the number of temporal and spatial region functions. Any drive or source region may be associated with a temporal function and the physical properties of all material regions may be defined with spatial functions. Positions are passed to the function in units set by *DUnit* (cm, inches, μm,...). You can model discontinuous functions by dividing a volume into two or more regions.

MU RegNo > Function
MU(3) > 1.0 + exp($x/25.0)
MU(2) > 2.0 - cos(0.05*$z + 2.5)
Set values of μ_r in the elements of a volumetric material region according to a specified function of space.

SIGMA RegNo > Function
SIGMA(2) > 15.2*exp($x∧2/3.4)*exp($y∧2/2.3)
SIGMA(6) > 1.0 + 0.25*$r∧3
Set values of σ (in S/m) over the elements of a volumetric material region according to a specified function of space.

Table 8: **EMP** standard units

Quantity	**Unit**
Spatial dimensions	meters or units set by *DUnit*
Electrical field, **E**	volts/meter (V/m)
Magnetic field, **H**	amperes/meter (A/m)
Current density, **j**	amperes/meter^2 (A/m^2)
Electrical conductivity, σ	siemens/meter (S/m)
Energy density	joules/meter^3 (J/m^3)
Power density	watts/meter^3 (W/m^3)

6 Solution analysis reference

Click the *Analyze* command in the **EMP** main menu to enter the analysis menu. Here, you can load data dumps to create plots or to perform numerical calculations. The menu contains the following main entries: *File*, *Plots*, *Analysis*, *Scans*, *Export* and *Return*. The commands of the *Export* menu (which generate hardcopy output and plot files) are identical to those in **Mesh**. The *Return* command restores the main menu where you can perform additional solutions.

6.1 File menu commands

SET SOLUTION SERIES

A dynamic run may create several data files with the same prefix, `FPREFIX`. Use this command to specify a file prefix for subsequent load operations. Moving to a new directory in the dialog changes the program working directory. The command must be the first activity in an analysis session. **EMP** counts the number of files in the series, records the times and displays the dialog of Fig. 13. Pick a solution file to load and click *OK*.

LOAD SOLUTION

Load a different solution file in a series. **EMP** displays the dialog of Fig. 13. Pick a different solution and click *OK*. This command is active only when a series has been specified.

OPEN DATA RECORD

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

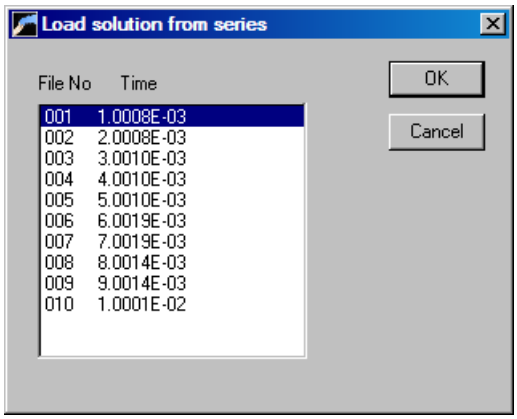

Figure 13: Dialog to pick a solution file from a series

CLOSE DATA RECORD

Close the current data file. Use this command if you want to start a new file. Note that you must close the data file before opening it with the internal editor.

RUN SCRIPT

A script allows you to perform complex or repetitive analyses on a set of similar solutions. This command displays a dialog listing files with the suffix `SCR`. Pick a file and click *OK*. The script can load data files, open and close data records, and perform any of the quantitative analysis functions described in this chapter. The script command language is described in Sect. 6.5. Note that the analysis script must be in the same directory as the data files.

CREATE SCRIPT

Use this command to create an analysis script with the internal text editor. Supply a file prefix *SPrefix* in the dialog – the resulting script will be saved with the name `SPrefix.SCR`. The program opens the file in the editor and writes a reference list of allowed commands. The list follows the *EndFile* command and will be ignored by the script parser. Enter commands above the *EndFile* command.

EDIT SCRIPT
EDIT DATA FILE
EDIT FILE

Use these commands to view or to modify an existing file. The dialog shows files with suffix `SCR` for the *Edit script* command and `DAT` for the *Edit data file* command. Changing directories in the dialog does not change the working directory of the program

6.2 Plot menu commands

Spatial plots show variations of quantities over the two-dimensional space of the simulation. The following plot types are available:

- **Mesh**. Element facets of the computational mesh.

- **Region**. Computational mesh with elements color-coded by region number.
- **Contour**. Lines that follow constant values of a computed quantity. In electromagnetic solutions, contours of the primary field quantity lie along lines of the the secondary field quantity. The correspondence depends on the solution type and symmetry. In E type solutions, contours of E_z or rE_θ lie along lines of $\mathbf{H}$ in the x-y or z-r planes. In H type solutions, contours of H_z or rH_θ lie along lines of $\mathbf{E}$ in the x-y or z-r planes.
- **Element**. Elements of the solution space color-coded according to a computed quantity (such as the field energy density).
- **Vector**. An element plot with orientation lines included in each element to show the local direction of a vector quantity.
- **Surface**. A three-dimensional plot where a computed quantity is represented as height over a region in the x-y or z-r plane. The spatial limits of the plot correspond to the current view window for *Mesh*, *Region*, *Contour*, *Element* or *Vector* plots. For large meshes, you may notice a delay regenerating a *Surface* plot. The program must map the current quantity to a rectangular grid, performing a large number of interpolations.

The *Settings* popup menu contains the following commands.

TYPE

Choose the plot type from the above list. A plot type may not support some plotted quantities. If you receive a message when you switch plot types that the current quantity is not allowed, use the *Quantity* command to pick a valid option.

QUANTITY

A dialog shows a list of available quantities (Table 9) consistent with the current plot type. Note that the specific quantity may depend on the solution type and symmetry. Spatial plots of ϵ_r, μ_r and σ are included to check the validity of assignment of spatial variations from mathematical functions (Chap. 5).

PLOT LIMITS

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

TOGGLE GRID DISPLAY

Use this command to activate or to suppress the display of grid lines in *Mesh*, *Region*, *Contour*, *Element* and *Vector* plots. Grid lines corresponding to the axes ($x = 0.0$ or $y = 0.0$) are plotted as solid lines.

Table 9: Available plot quantities

Plot type	**Quantity** (Depends on solution type, symmetry)
Contour	Lines of **H** or **E**
Element	E_z, rE_θ, H_z or rH_θ (primary quantity) E_z, E_θ, H_z or H_θ (primary field) $\|\mathbf{H}\|$ or $\|\mathbf{E}\|$ (secondary field magnitude) H_x, H_z, E_x or E_z (secondary field, x or z component) H_y, H_r, E_y or E_r (secondary field, y or r component) U (field energy density) P (resistive power density) $\|\mathbf{S}\|$ (Poynting vector) ϵ_r (relative dielectric constant) μ_r (relative magnetic permeability) σ (electrical conductivity)
Vector	$\|\mathbf{H}\|$ or $\|\mathbf{E}\|$ (secondary fields) $\|\mathbf{S}\|$ (Poynting vector)
Surface and scans	E_z, rE_θ, H_z or rH_θ (primary quantity) E_z, E_θ, H_z or H_θ (primary field) $\|\mathbf{H}\|$ or $\|\mathbf{E}\|$ (secondary field magnitude) H_x, H_z, E_x or E_z (secondary field, x or z component) H_y, H_r, E_y or E_r (secondary field, y or r component) U (field energy density) P (resistive power density) $\|\mathbf{S}\|$ (Poynting vector) ϵ_r (relative dielectric constant) μ_r (relative magnetic permeability) σ (electrical conductivity)

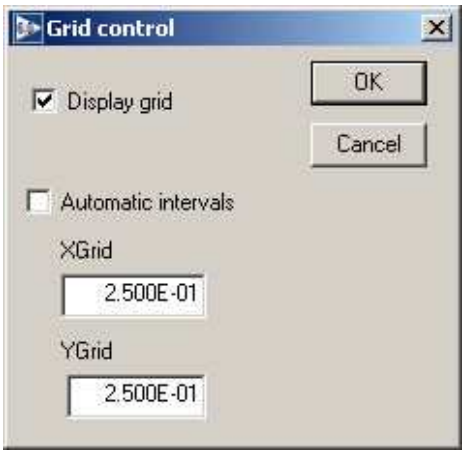

Figure 14: Grid control dialog

GRID CONTROL

This command displays the dialog of Fig. 14 to set properties of the grid. In the default autoscale mode, **EMP** automatically chooses intervals and positions so that lines occur at convenient intervals in x or y (for example, 0.01 rather than 0.01153). The grid intervals change as the view is zoomed. To set the grids manually, uncheck the *Autoscale* box and enter values for the intervals in x and y.

MOUSE/KEYBOARD

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

TOGGLE SNAP MODE

When snap mode is active, the mouse returns the coordinate values closest to an integer multiple of the quantity *DSnap*. In other words, if $DSnap = 0.5$ and the mouse position is (5.4331, -2.6253), the returned coordinates are (5.500, -2.500). By default, snap mode is active. Snap mode is automatically turned off for coordinate input to the commands *Point calculation* and *Element properties*. Otherwise, the program would pick a location closest to the snap point rather than the tip of the cursor arrow, giving misleading results.

SNAP DISTANCE

Set the distance scale *DSnap* for the mouse snap mode.

TOGGLE ELEMENT OUTLINE

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

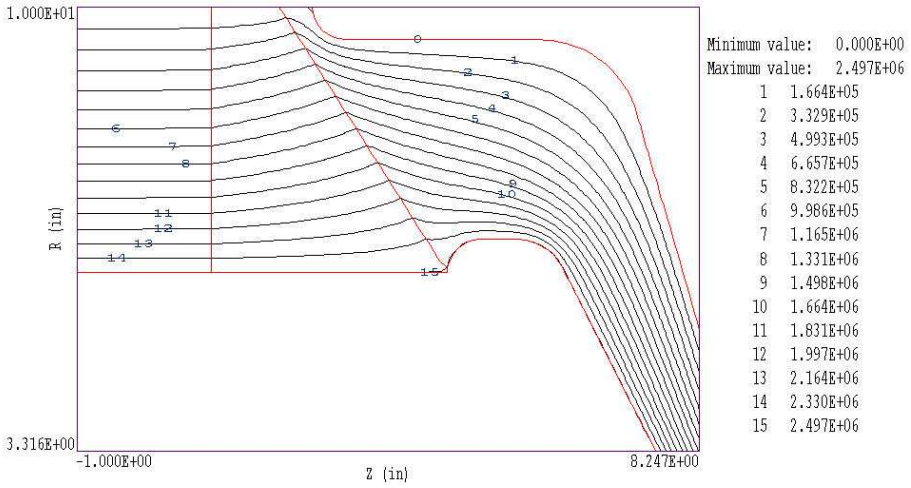

Figure 15: Monochrome contour plot with labels

CONTOUR STYLE

This command is active only when the current plot type is *Contour*. There are four choices: monochrome, monochrome with labels, colored and colored with labels. In the colored mode, the lines are color-coded according to the value of the plotted quantity. A legend is included in the information window to the right of the plot. In the labeled modes, contour lines are numbered according to their values (Fig. 15). Overlapping labels on closely-spaced lines may look better in a zoomed view.

NUMBER OF CONTOURS

Change the number of plotted contour lines. This command is active only when the current plot type is *Contour*.

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

ZOOM WINDOW
ZOOM IN
EXPAND VIEW
GLOBAL VIEW
PAN

The following commands control the appearance of *Surface* plots. The commands are active only when a *Surface* plot is displayed.

ROTATE 3D IMAGE

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

VIEW ANGLE 3D
Set the elevation angle for the view point.

SET GRID 3D
Change the resolution of the surface plot. To create a *Surface* plot, a quantity is mapped to a rectangular grid with dimensions $N_x \times N_y$. The numbers also determine the total number of grid lines in the *Surface* plot. The default values are $N_x = N_y = 40$.

6.3 Analysis menu commands

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

POINT CALCULATION
After clicking on the command, point to any position in the solution volume with the mouse. Note that snap mode is deactivated for coordinate input. The program writes a subset of interpolated quantities to the information area below the plot and also records complete information if a data file is open. To enter point coordinates by keyboard, press the *F1* key or change the program mode with the *Toggle mouse/keyboard* command.

LINE SCAN
The line scan is one of the most useful functions of **EMP**. After clicking on the command, supply two points with the mouse in a view of a *Mesh*, *Region*, *Contour*, *Element* or *Vector* plot to define a scan line. The snap mode is useful in this application (for example, you may want a scan to extend from 0.000 to 5.000 rather than 0.067 to 4.985). To enter point coordinates by keyboard, press the *F1* key or change the program mode with the *Toggle mouse/keyboard* command. The program computes a series of values of field quantities at equal intervals along the line. The information is recorded if a data file is open. The program also makes a screen plot of the currently selected quantity versus distance along the scan and activates the commands in the *Scan plot* menu (Sect.6.4). The default is to include all computed quantities. Use the *Set recorded quantities* command to limit the information.

ELEMENT PROPERTIES
Pick an element with the mouse (or keyboard) and the post-processor writes material and field properties to the screen. The information is recorded if a data file is open.

REGION PROPERTIES
To find the physical properties associated with a region of the solution space, click the mouse close to any arc or line vector of the region. Partial results are shown on the screen and a complete analysis is included in the data file. **EMP** calculates volume integrals of the electric and magnetic field energy density and the total resistive power dissipation in the region.

VOLUME INTEGRAL

No input is needed for this command. **EMP** automatically computes integrals of quantities over the full solution volume and over individual regions. Information is recorded on the screen or in a history file.

MATRIX FILE

EMP can make matrix files of field values and material properties to help you create your own analysis routines. Although information is available in the output file of the solution program, it may be difficult to deal with the conformal triangular mesh. The *Matrix file* command uses the interpolation capabilities of the program to create a text data file of field quantities on a rectangular grid in x-y or z-r. The command displays a dialog where you set the matrix file prefix, the dimensions of the box and the number of intervals along x and y (or z and r). The program creates the file `FPrefix.MTX` in the current directory. The default is to include all computed quantities. Use the *Set recorded quantities* command to limit the information.

The *Settings* popup menu contains the following entries.

INTERPOLATION METHOD

The default interpolation method for the *Point calculation* and *Line scan* commands is a second-order least-squares fit. This method may fail in very small regions or enclosed areas if the program cannot identify enough data points. In this case, toggle to the linear mode. Here, field values are determined by a first order fit in the element that contains the target point. The status bar reports the current interpolation type.

SCAN PLOT QUANTITY

With this command you can choose the quantity to display in screen and hardcopy plots of line scans. Pick the quantity from the list box and click *OK*. This setting has no effect on the data file records which include all field quantities. The available line scan quantities depend on the solution type and symmetry. The choices are listed in Table 9.

NUMBER OF SCAN POINTS

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

SET RECORDED QUANTITIES

The default for data records of line scans and matrix files is to include all plotted quantities. The may result in large files with unnecessary information. Use this command to set the quantities that will be include. In the dialog, activate the check box to include the quantity. There are a few rules:

- Quantity selection is available when the program runs interactively in a window. It does not apply when the program is run in the background.

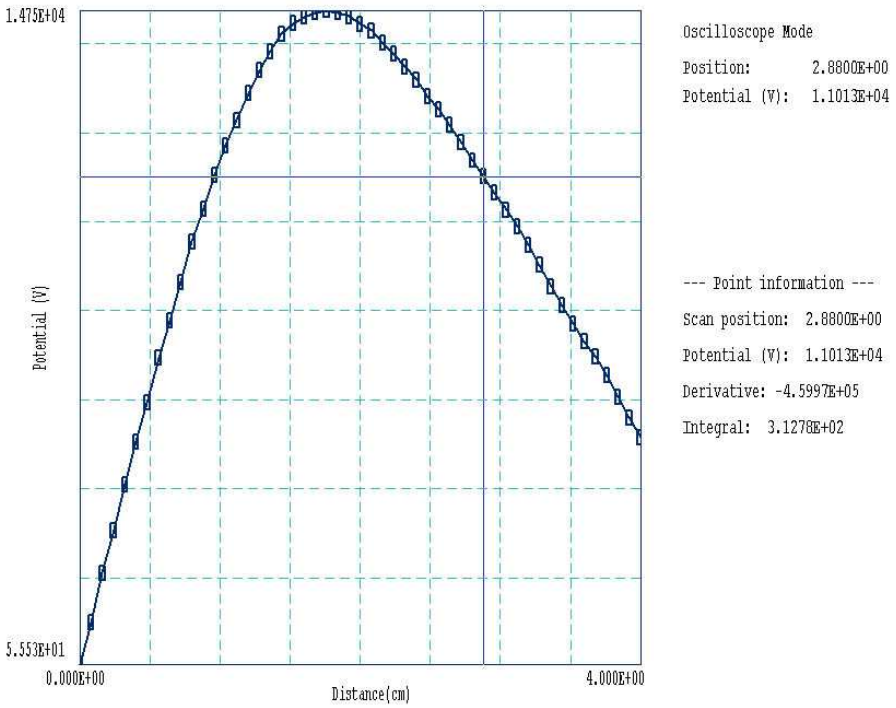

Figure 16: Scan plot in the oscilloscope mode

- In the interative mode, quantity selection applies to direct calls to the *Line scan* or *Matrix file* commands and indirect calls from an analysis script.
- All quantities are active when the program starts.
- The state of active quantities is preserved when a new data file is loaded.

6.4 Scan plot menu

The commands of the *Scan* menu become active when a plot is created following the *Line scan* command.

OSCILLOSCOPE MODE

In the oscilloscope mode, a scan plot assumes characteristics of a digital oscilloscope (Fig. 16). The program superimposes a cross-hair pattern on the graph. Plot values at the intersection are displayed in the information window. Move the marker along the plot by moving the mouse. If you click the left mouse button at a point, the program displays the plot values along with the numerical derivative and integral of the curve. The definite integral is taken from the left-hand side of the plot to the current point. Values are displayed on the screen and written to the data file if open. Press the right mouse button to exit the oscilloscope mode.

TOGGLE SCAN SYMBOLS

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

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

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

6.5 Analysis script commands

Scripts to control analysis sessions have a name of the form `FPrefix.SCR`. They should be in the same directory as the data files. Scripts are text files that follow the **TriComp** syntax conventions. The program ignores blank lines and indentations. Data lines use the standard delimiters and comment lines begin with an asterisk [*]. Processing ends when the *EndFile* command is encountered. To run a script, choose the *Run script* command in the *File* menu. The program shows a list of available scripts. Pick a file and click *OK*. The script operates on the presently loaded data file or you can load other files from within the script. You can also sequentially open one or more data files.

EMP can perform analyses autonomously under script file control from the Command Prompt. If the file `GTest.SCR` is in the data directory, then use a command of the form:

```
[ProgPath]\EMP GTEST <Enter>
```

The main application of the command prompt mode is to generate data files and to perform extended analyses under batch file control.

The following commands may appear in a script:

INPUT FileName
INPUT Convolute.002
Close the current data file and load a file for analysis. The parameter is the full name of the data file. For the command illustrated, the post-processor would load the file `Convolute.001`. You can load several files for sequential analysis.

OUTPUT FPrefix
OUTPUT ACavity
Close the current data file and open an output file `ACavity.DAT`.

POINT X Y
POINT Z R
POINT = (5.65, 10.68)
Perform interpolations at the specified point and write the results to the data file. The two real number parameters are the coordinates of the point in **Mesh** units.

SCAN Xs Ys Xe Ye
SCAN Zs Rs Ze Re
SCAN = (0.00, 0.00) (10.00, 0.00)
Write the results of a line scan between the specified points to the data file. The four real number parameters are the starting and end coordinates in **Mesh** units.

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

ELEMENT X Y
ELEMENT Z R
ELEMENT = (5.65, 10.68)
Write the properties of the element at the specified point to the data file. The two real number parameters are the coordinates of the point in **Mesh** units.

NSCAN NScan
NSCAN = 150
Set the number of points in a line scan. The default is 50 and the maximum number is 500.

REGION RegNo
REGION = 5
Write volume and surface integrals for a region to the data file. The integer parameter is the region number.

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

MATRIX FPrefix Nx Ny Xs Ys Xe Ye
MATRIX FPrefix Nz Nr Zs Rs Ze Re
MATRIX = Switch1 (10, 20) (0.00, 0.00, 5.00, 10.00)
Open a matrix file and record values. The command requires seven parameters: 1) The prefix of the matrix file `FPrefix.MTX` (string), 2) the number of intervals along the x or z direction (integer). 3) the number of intervals along the y or r direction (integer), 4-7) coordinates of the corners of a box in the solution volume (real).

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

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

```
NSCAN 200
OUTPUT COMP
INPUT SWITCH01.001
SCAN 0.00 -50.00 0.00 50.00
INPUT SWITCH01.002
SCAN 0.00 -50.00 0.00 50.00
INPUT SWITCH01.003
SCAN 0.00 -50.00 0.00 50.00
INPUT SWITCH01.004
SCAN 0.00 -50.00 0.00 50.00
ENDFILE
```

Table 10: Examples in the **EMP** library

Example	Technique
`ABSORB`	Defining absorbing layers ion E and H type pulses, normal and non-normal incidence.
`INLINE`	Inline resistor in a coaxial transmission line, comparison to predicted reflection and transmission coefficients (H type, cylindrical)
`RODREFLECT`	Plane wave reflection from a metal rod, definition of an internal metal body in E and H type solutions. Use of an internal void.
`FUNCTEST`	Entering time-dependent drives and sources with table and mathematical functions, effect of *TabShift* command.
`TM01`	Excitation of the TM01 mode in a circular waveguide. Source specified by a mathematical function.
`WGUIDEDIEL`	Excitation of the TM01 mode in a circular dielectric waveguide.
`PARABOLA`	Linear parabolic transmitter. Illustration of the *TExpire* command for a drive region.

7 EMP tutorials

This section contains a number of walkthrough examples that illustrate solution techniques. All input files are contained in the example library included with the program. The library includes a variety of additional examples that illustrate basic setup techniques in **EMP** (Table 10).

7.1 Coaxial transmission lines

This section introduces some useful **EMP** techniques to generate TEM waves in coaxial transmission lines. To begin, consider the simple problem of generating a square-pulse in an ideal coaxia line. The line has inner radius $R_i = 5.0$", outer radius $R_o = 8.0$" and a fill medium of transformer oil with a relative dielectric constant $\epsilon_r = 2.8$. The line has characteristic impedance

$$Z_0 = \frac{1}{2\pi}\sqrt{\frac{\mu}{\epsilon}}\ \ln\left(\frac{R_o}{R_i}\right), \quad (\Omega) \tag{27}$$

and a capacitance per length of

$$c = \frac{2\pi\epsilon}{\ln(R_o/R_i)}. \quad (\mathrm{F/m}). \tag{28}$$

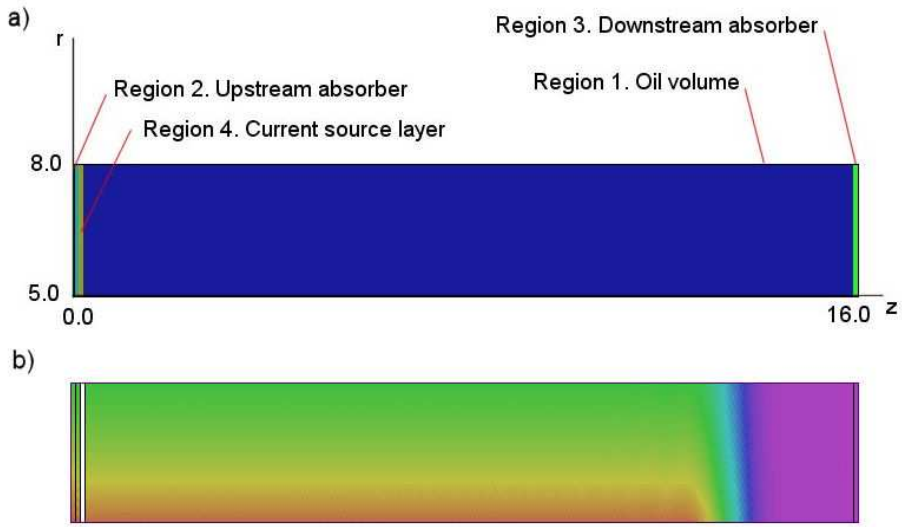

Figure 17: Example `COAX01`. a) Geometry. b) Distribution of $|\mathbf{E}|$ at 2.0 ns.

The system has cylindrical symmetry. A TEM pulse has a radial electric field and an azimuthal magnetic field. The calculation is therefore an H type solution and the primary quantity is rH_θ.

Figure 17 shows the geometry and assignment of regions. The active solution volume has a length of 16.0" in the z direction and extends between R_i and R_o in the radial direction. Thin absorbing layers matched to the line impedance are located upstream and downstream. Boundary layers on the left and right ends are set to the condition rH_θ to complete the definition of the absorbing layer for H type pulses (Sect. 2.5). A radial current source layer just downstream from the absorber drives the pulse. The absorbing and source layers have thickness $\Delta z = 0.1$". The current layer creates a TEM pulse that propagates both upstream and downstream - the upstream pulse is absorbed by the adjacent resistive layer. The downstream resistive layer absorbs the pulse after propagation through the solution volume. The inner and outer radial boundaries assume the default Neumann condition (Sect. 2.4), a perfect conductor for H type pulses.

The **EMP** input file `COAX01.EIN` is listed in Table 11. We shall review the motivations for choosing the parameters in the file. The speed of light in the oil is 1.793×10^8 m/s; therefore, the single pass electromagnetic transit time is 2.27 ns. The run time of 4.01 ns is sufficient to follow the initial pulse and to check reflected waves resulting from non-ideal performance of the absorbing layer. The value $DUnit = 39.37$ converts dimensions in the **Mesh** input file from inches to meters. The solution volume is an ideal dielectric with $\epsilon_r = 2.8$ and $\mu_r = 1.0$. The corresponding impedance for the transmission line is $Z_0 = 16.87\ \Omega$. The absorbing layers have the same dielectric properties and a nonzero conductivity. The matched conductivity for total absorption is given by

$$\sigma = \frac{1}{\Delta z \sqrt{\mu/\epsilon}}. \tag{29}$$

Inserting values in Eq. 29, the matched conductivity is $\sigma = 1.746$ S/m. The resistance of a coaxial resistor is given by

Table 11: Contents of the file `COAX01.EIN`

```
TMax = 4.01E-9
Geometry = Cylin
DUnit = 39.37
Solution = H
* Solution volume
Epsi(1) = 2.8
Mu(1) = 1.0
* Upstream absorber
Epsi(2) = 2.8
Mu(2) = 1.0
Cond(2) = 1.746
* Downstream absorber
Epsi(3) = 2.8
Mu(3) = 1.0
Cond(3) = 1.746
* Source layer
Epsi(4) = 2.8
Mu(4) = 1.0
Source(4) = SMOOTH.SRC (0.50E-9, 125.3)
Beta(4) = 90.0
* Boundary
Fixed(5)
DTime = 1.0E-9
History = 12.0 6.5
History = 4.0 6.5
EndFile
```

$$R = \frac{1}{2\pi\sigma\Delta z}\ \ln(R_o/R_i). \tag{30}$$

Equation 30 gives the value $R = 16.87\ \Omega$, the characteristic impedance predicted by Eq. 27.

Next we consider choosing the magnitude of the radial current in the source layer (Region 4). We want to generate a downstream pulse with a peak current of 1.0 A. Because the layer must drive both upstream and downstream pulses, the total current should be 2.0 A. The following lines in the **EMP** input file define the current source:

```
Source(4) = Smooth.SRC (0.25E-9, 125.3)
Beta(4) = 90.0
```

The file `Smooth.SRC` contains a normalized step function table - the quantity rises from 0.0 to 1.0 over the time interval 0.0 to 1.0. The first factor following the file name converts the rise time to 0.50 ns. The second factor gives a peak total current of 2.0 A. For a radial current density j_r, the total layer current is

$$I = (2\pi r\Delta z)j_r. \tag{31}$$

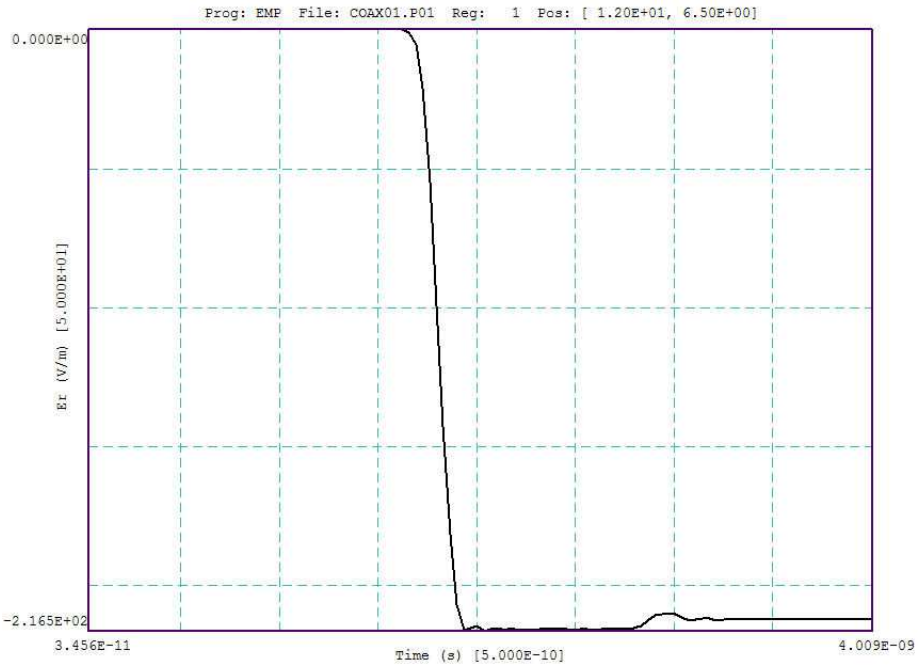

Figure 18: Example `COAX01`: temporal variation of E_r at position $z = 12.0$", $r = 6.5$".

With $I = 2.0$ A and $\Delta z = 2.54 \times 10^{-2}$ m, the quantity rj_r has the value 125.3 A/m. The choice $\beta = 90^o$ sets current flow in the positive radial direction. The resulting electric and magnetic fields in the downstream pulse have negative polarity. Finally, the diagnostic statements specify data dumps at times $t = 1.0, 2.0, 3.0$ and 4.0 ns and probes at positions ($r = 6.5$", $z = 4.0$") and ($r = 6.5$", $z = 12.0$").

The predicted field values for a 1 A pulse at a radius of 6.5" are $H_\theta = -0.9640$ A/m and $E_r = 217.1$ V/m. Figure 18 shows the time-dependent electric field measured by the probe at $z = 12.0$". The field value during the initial pulse propagation is 216.5 V/m. The ratio of the code-predicted electric and magnetic fields is $E_r/H_\theta = 227.3\ \Omega$. The signal delay time between the probes at $z = 4.0$" and 12.0" gives a propagation velocity of 1.783×10^8 m/s, close to the predicted value. The small bump in the signal at 2.9 ns results from imperfect absorption of the pulse leading edge. The reflected energy is about 0.09% of the incident pulse energy. Figure 17*b* shows an element plot of electric field as the pulse front moves down the transmission line. The color changes radially because the electric field magnitude is proportional to $1/r$. There is a region of axially-uniform electric and magnetic fields behind the front. At $t = 3.0$ ns, the uniform region fills the entire solution volume. For this data dump, the volume integral function of the analysis menu gives the following values for electric and magnetic field energy in the solution volume: $U_e = 1.888 \times 10^{-8}$ J and $U_m = 1.863 \times 10^{-8}$ J. The predicted value of electric field energy is $U_e = CV_0^2/2$, where C is the capacitance of the line and V_0 is the pulse voltage (16.87 V). The capacitance per length of the transmission line (Eq. 28) is 3.313×10^{-10} F/m and the length of the solution volume is 16.0" (0.4064 m). The predicted energy is $U_e = 1.916 \times 10^{-8}$ J. In an ideal pulse the electric and magnetic energies should be equal. The predicted power dissipation in each absorbing layers is I^2R, where $I = 1.0$ A and $R = 16.87\ \Omega$. The total is 33.37 W, close to the code calculation of 34.29 W.

The calculation becomes more interesting if we change the downstream absorber to an open-

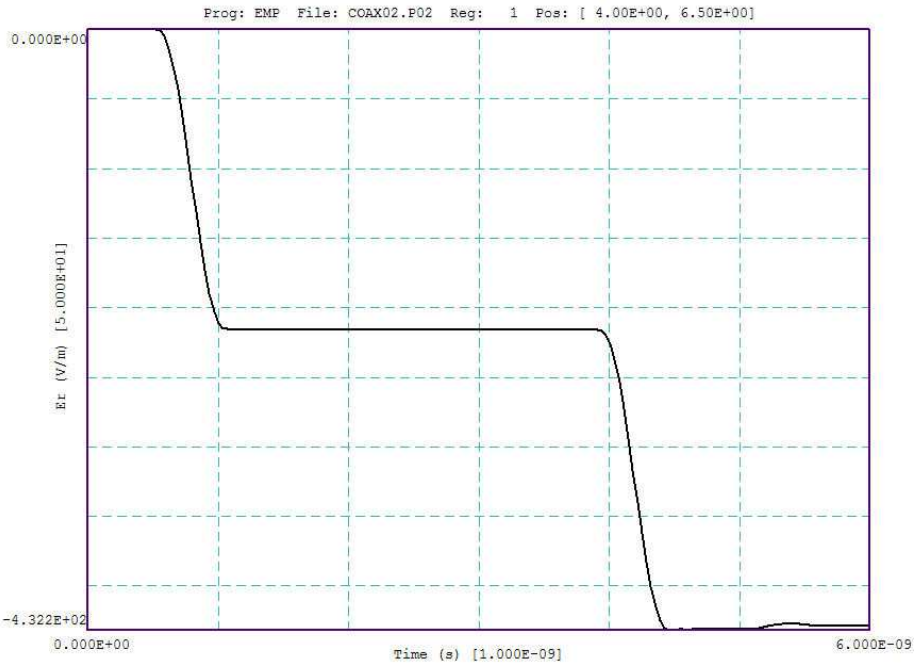

Figure 19: Example `COAX02`: temporal variation of E_r at position $z = 4.0$", $r = 6.5$".

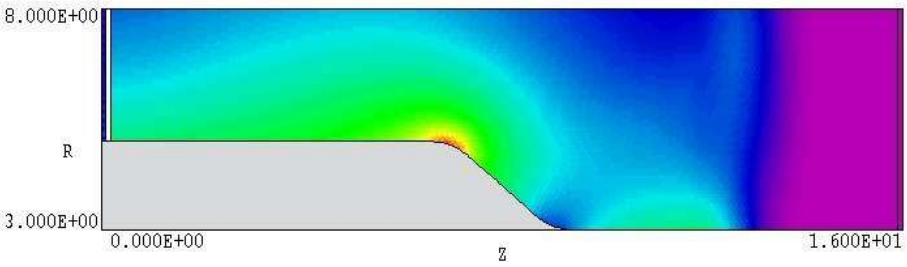

Figure 20: Example `COAX03`: impedance transition in a coaxial transmission line.

circuit termination (example `COAX02`) by setting $\sigma = 0.0$ in Region 3. In this case, the pulse reflects with positive polarity in **E** and negative polarity in **H**. The reflected pulse is absorbed in the upstream resistive layer after a double transit. For long times, the solution is equivalent to a DC voltage on the transmission line. Figure 19 shows the time-dependent electric field measured at position ($z = 4.0$", $r = 6.5$"). As expected, the reflection results in doubling of the electric field and cancellation of the magnetic field. An energy integral for the 6.0 ns solution gives the values $U_e = 7.467 \times 10^{-8}$ J and $U_m = 7.540 \times 10^{-10}$ J. The theoretical values are $U_e = 7.516 \times 10^{-8}$ J and $U_m = 0.0$.

The example `COAX3` (Fig. 20) addresses a calculation that requires a two-dimensional simulation. A coaxial transmission line with impedance $Z_1 = 16.87\ \Omega$ undergoes a transition over a short axial length to a line of impedance $Z_2 = 35.20\ \Omega$. A step pulse with risetime 0.25 ns and current amplitude 1.0 A is created by a current layer near the left-hand boundary. Ideal absorbing layers are placed adjacent to the left and right-hand boundaries. Probes are located near the left-hand boundary ($r = 6.5$", $z = 2.0$") and right-hand boundary ($r = 6.5$", $z = 14.0$").

For an ideal discontinuity we expect the following values of reflection and transmission

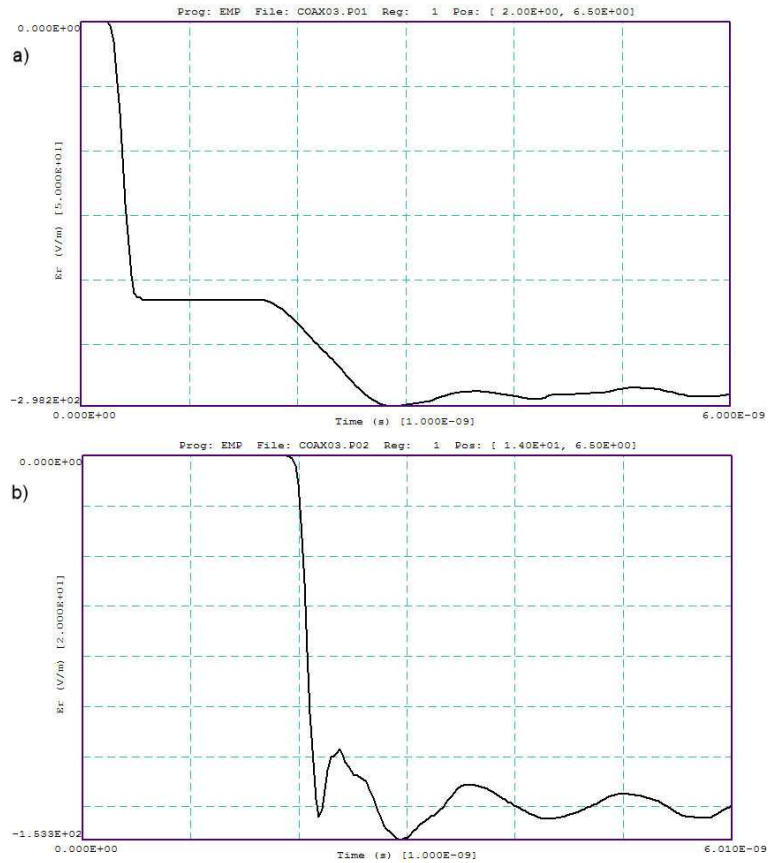

Figure 21: Example `COAX03`: temporal variation of E_r. *a*) Position $r = 6.5$", $z = 2.0$". *b*) Position $r = 6.5$", $z = 14.0$"

coefficients:

$$\rho = \frac{Z_2 - Z_1}{Z_2 + Z_1} = 0.352, \tag{32}$$

$$\tau = \frac{2Z_2}{Z_2 + Z_1} = 1.352. \tag{33}$$

The incident pulse produces an electric field $E_r = 216$ V/m at the upstream probe. The sum of the incident and reflected waves should have an average amplitude $E_r = 292$ V/m. Figure 20*a* shows the code-generated waveform of E_r at the upstream probe. The average level of the long-term pulse is close to the predicted figure. The delayed rise time of and oscillations on the reflected pulses result from the extended length of the transition and two-dimensional wave reflection effects. The color coding of Fig. 20 shows the electric field of the pulse before reacing thedownstream absorber.

The voltage of the transmitted pulse is predicted to be $V_t = V_0 = (16.87)(1.352) = 22.81$ V. The radial electric field is related to the voltage by

$$E_r(r) = \frac{V_0}{r\ \ln(R_o/R_i)}. \tag{34}$$

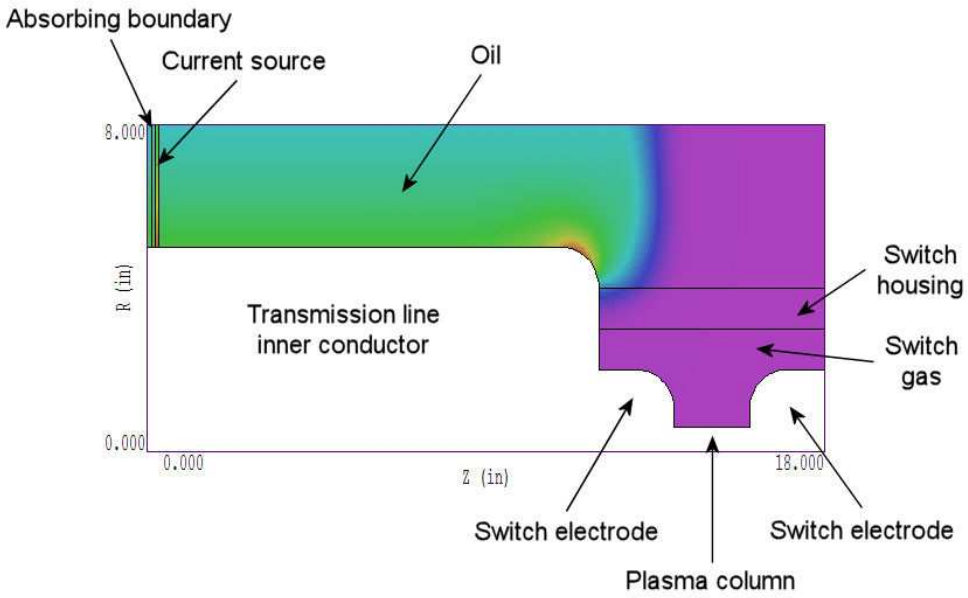

Figure 22: Geometry of the `SWITCH` example, dimensions in inches. Color code show $|\mathbf{E}|$ at 2.0 ns.

The predicted field value is 140.9 V. Figure 21*b* shows the downstream waveform. The average value of E_r is close to the expectated value with superposed oscillations generated at the transition.

As an exercise, modify the coaxial line solutions to investigate pulse reflections from open-circuit and close-circuit terminations. The `TERMINATION` example shows how to model both types of boundaries in one simulation.

7.2 Transients in a high-voltage switch

In this example, we shall use **EMP** to determine the current rise time and transients in a high-voltage Blumlein line controlled by a spark gap[1]. Figure 22 shows the geometry. The transmission line discussed in Sect. 7.1 is connected to ground through a shorting switch. The line (R_i = 5.0", R_o = 8.0") is filled with transformer oil (ϵ_r = 2.8) and has a characteristic impedance of 16.87 Ω. The gas-filled switch has a ceramic housing (ϵ_r = 6.0). The outer boundary, line center conductor, switch electrodes and arc column are represented as perfect conductors (*i.e.*, default Neumann condition for the H type pulse over the boundary of the solution volume).

Because **EMP** handles only dynamic pulses, the key question is how to initiate the solution. A DC charge on the line corresponds to $rH_\theta = 0.0$ at all locations; therefore, the null solution would not evolve in time. We can resolve the problem by noting that a DC charge of voltage V_0 on a transmission line may be represented as the summation of two square pulses with amplitude $V_0/2$ traveling in opposite directions and reflecting from open-circuit terminations

[1]The theory of transmission lines and Blumlein lines as pulse modulators is discussed in S. Humphries, **Principles of Charged-particle Acceleration** (John Wiley and Sons, New York, 1986), Chap. 9. The text is available at `www.fieldp.com/cpa.html`.

at each end of the line. If we suddenly short one end of the line, only the pulse component traveling toward the switch will interact. Therefore, we can estimate the current through and electric field distribution within the switch by generating a dynamic incident step pulse with voltage $V_0/2$.

The properties of the upstream absorber and current source layer are the same as those in Sect. 7.1. The step pulse has a rise time of 0.25 ns and a current of 1.0 A. The simulation run time is extended to 15 ns to investigate long-term behavior. The color coding in Fig. 22 shows |**E**! for the TEM wave arriving at the switch (2.0 ns). Figure 23*a* shows a measurement of the radial electric field in the transmission line near the switch. Initially, the inductive switch acts as an open circuit giving a reflected pulse with positive polarity and voltage doubling. Equivalently, the voltage across the switch equals approximately V_0 at $t = 0.0^+$. The line voltage decreases with time approximately as

$$\frac{V(t)}{V_0} = \exp\left(-\frac{t}{L_s/Z_0}\right). \tag{35}$$

In Eq. 35, L_s is the switch inductance and Z_0 is the characteristic impedance of the transmission line. Inspection of the curve shows that $L_s/Z_0 \cong 2.9$ ns, or $L_s \cong 49$ nH. The oscillations superimposed on the waveform result from the capacitive mismatch of the switch assembly. Figure 23*a* shows the time-variation of H_θ, proportional to the switch current.

7.3 Pulsed electric field probe

This example illustrates how to use **EMP** to model transients for low-frequency pulsed electric fields in media with conductive and dielectric properties (*i.e.*, real and displacement current). Figure 24*a* shows the simulation geometry, a phantom experiment for tumor ablation studies. A cylindrical metal probe with a Telfon sheath is driven by a train of current pulses. The probe is immersed in a medium that simulates tissue ($\epsilon_r = 70.0$, $\sigma = 1.0$ S/m). The medium fills a chamber with a grounded metal wall.electric field, pulsed

In this simulation we use a step-function current pulse with risetime 0.5 ns to investigate both transient and long-term solutions. It is useful to do some preliminary calculations to understand the nature of the solution and to determine parameters to check accuracy. The transit time of an electromagnetic pulse over a distance of 30.0 mm in a medium with $\epsilon_r = 70.0$ is about 1.0 ns. We expect to observe transients on this time scale. Furthermore, we can get a good approximation to the steady-state solution by extending the calculation to 10.0 ns. To benchmark the steady-state result, we perform initial electrostatic calculations with **EStat**. In this case, we assign the fixed potentials $\phi = 1.0$ V to the probe and 0.0 V to boundary (top and right-hand sides in Fig. 24). Equipotential lines for a calculation where the Teflon and tissue are treated as ideal dielectrics ($\sigma = 0.0$) are plotted in Fig. 24*a*. In contrast, Fig. 24*b* shows a solution where the materials are treated as conductors ($\sigma = 10^{-6}$ S/m for the Teflon, 1.0 S/m for the tissue) with no displacement current. This solution would hold at time $t \gg RC$, where R and C are the resistance and capacitance of the system. Taking an integral of electrostatic field energy density over the solution of Fig. 24*a*, we find that $C = 3.052 \times 10^{-11}$ F. Similarly, an integral of power density over the solution of Fig. 24*b* gives $P = 0.04454$ W for a drive voltage of 1.0 V. The corresponding resistance is $R = 22.45\ \Omega$, and the RC time is 0.685 ns. The implication is that the **EMP** solution at $t = 10.0$ ns should be almost identical to the

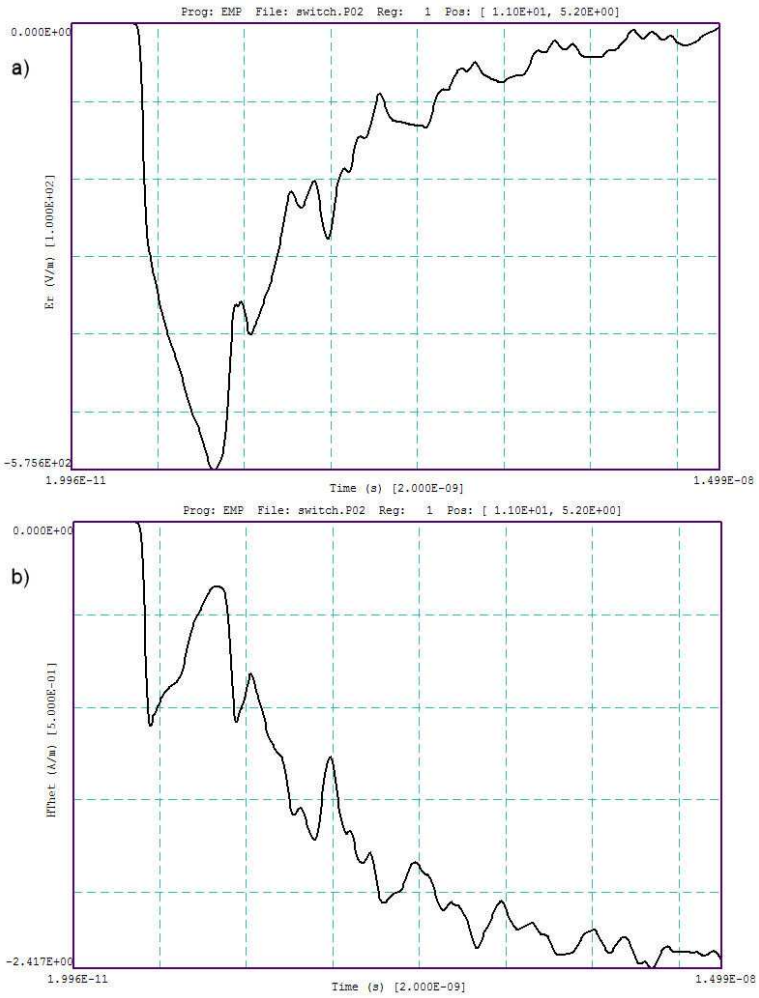

Figure 23: `SWITCH` example: **Probe** display of fields measured at the top of the spark gap [$z = 11.0$", $r = 5.2$"]. *a)* $E_r(t)$. *b)* $H_\theta(t)$.

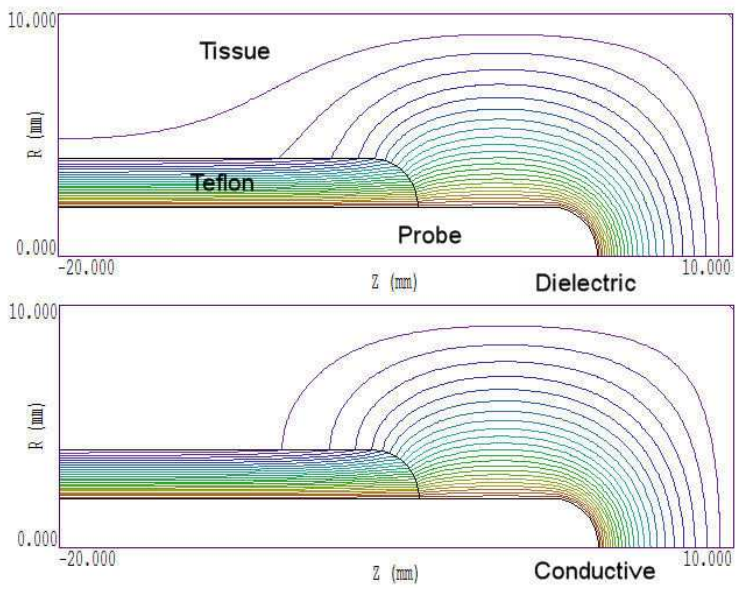

Figure 24: Cylindrical geometry for the `EPULSE` example, dimensions in mm. a) Equipotential lines for a static solution with ideal dielectrics. b) Equipotential lines for a static solution with ideal conductors.

solution of Fig. 24*b*. For a direct comparison, we renormalize the conductive solution for a probe voltage of 22.45 V, equivalent to a drive current of 1.0 A.

Now consider the calculation described by the input files `PULSEDE.MIN` and `PULSEDE.EIN`. In the H type solution, the flow of real and displacement current is in axial and radial directions, creating the primary field component rH_θ. The solution volume includes only the tissue and Teflon regions. The default Neumann condition applies on the metal surfaces of the probe and the top and right-hand boundaries. We add a line region with the condition $rH_\theta = 0.0$ on the portion of the solution volume along the axis ($r = 0.0$). With regard to the pulse drive, we recognize that if a current $I(t)$ flows in the infinite region to the left of the boundary, then the condition

$$rH_\theta(t) = \frac{I(t)}{2\pi}, \tag{36}$$

holds. We can approximate the effect of a long probe by setting a *Drive* boundary condition on the left-hand boundary at some distance from probe. This region is controlled by the command:

```
Drive(3) = SMOOTH.DRV 0.5E-9 0.1592
```

Here, `SMOOTH.DRV` describes a normalized step function with a risetime of 1.0 s. The first real-number parameter changes the risetime to 0.5 ns and the second $(1/2\pi)$ sets the peak current to 1.0 A (Eq. 36).

Figure 25 shows the spatial distribution of electric field at times 0.25, 0.50, 1.00 and 2.00 ns. The effect of the electromagnetic transit time is apparent as the pulse propagates to the probe tip. The plot also shows the effect of the RC time. The electric field is initially concentrated in the Teflon, which has relatively low dielectric constant. Later in time, the field migrates into the conductive tissue. A plot of $|\mathbf{E}|$ at 10.0 ns in indistinguishable from Fig. 24*b*. Figure 26 shows

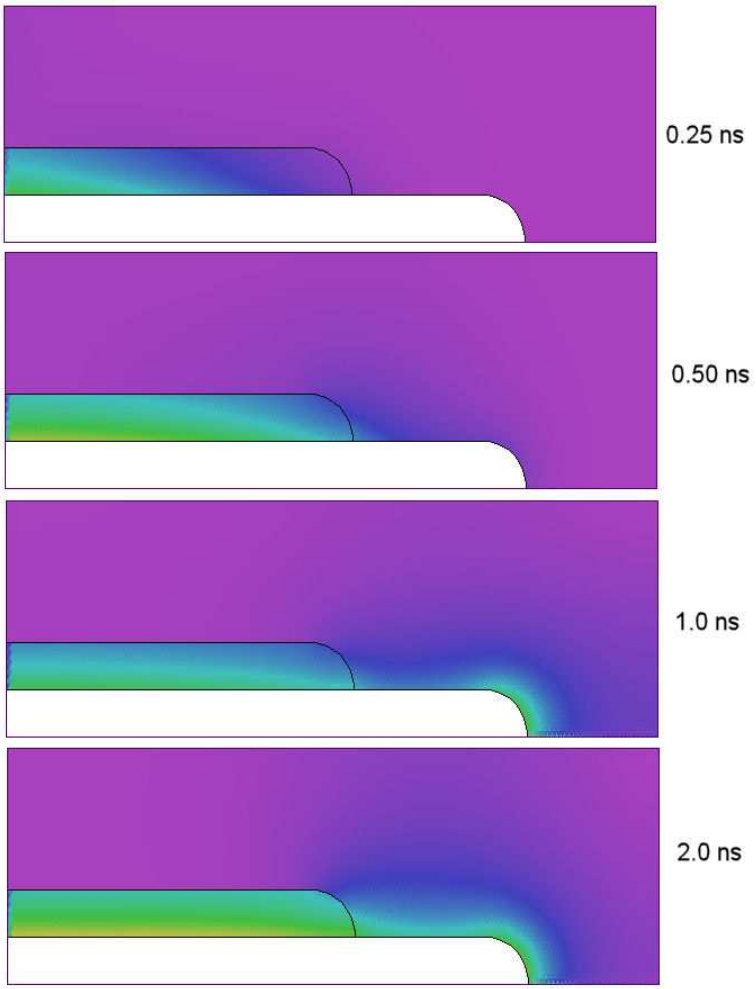

Figure 25: Plots of $|\mathbf{E}|$ for the `PULSEDE` example.

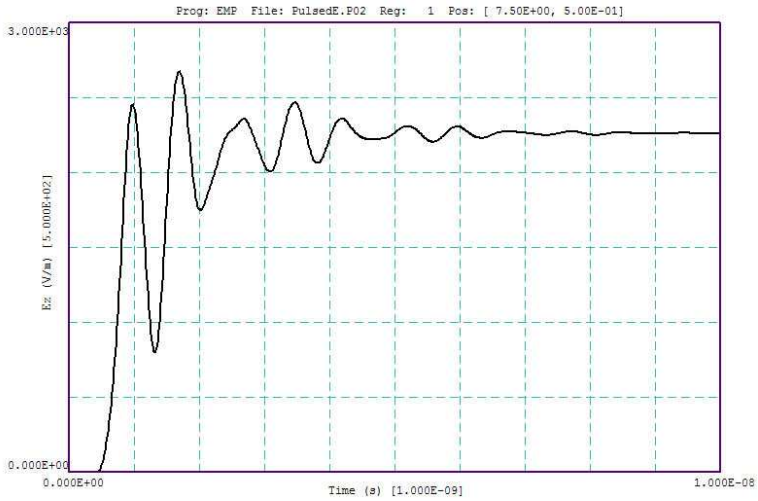

Figure 26: `PULSEDE` example: calculation of $E_z(t)$ at $z = 7.5$ mm, $r = 0.5$ mm.

$E_z(t)$ near the probe tip. Initial transients are damped by the conductivity of the tissue and approach a steady-state level of $E_z = 2266$ V/m over the expected time scale. For comparison, the **EStat** conductive calculation gives a value 2143 V/m.

7.4 Grating with a graded dielectric constant

This example illustrates how to set up a calculation where material properties vary over a region. We investigate the effect of a structure with a periodic variation of dielectric constant in y on a plane pulse. The intent is to generate an approximately planar pulse with amplitude modulation in y. Figure 27*a* shows the geometry. A plane pulse with Gaussian waveform (0.22 ns full width at half maximum) impinges on a dielectric plate of thickness 30 cm with a periodic variation of dielectric constant $\epsilon_r(y)$. The simulation represents one period of the plate in the region $-D \le y \le D$, where $D = 10.0$ cm. The relative dielectric constant varies as

$$\epsilon_r = 1.5 - 0.25\ [1.0 - \cos(\pi y/D)]. \tag{37}$$

The plate acts as a focusing lens about the position of maximum ϵ_r ($y = 0.0$).

The solution space for the E type calculation has symmetry boundaries at the top and bottom and a drive boundary on the left-hand side. The drive produces a normalized pulse with peak amplitude $E_z = 1.0$ V/m. The novel feature of the calculation is the definition of a spatially-varying dielectric constant through the command:

```
* Region 2: LENS
Epsi(2) > 1.5 - 0.25*(1.0 - COS(0.3116*$y))
```

Figure 27*a* shows $\epsilon_r(x, y)$ for the resulting distribution. Figure 27*b* plots lines of $|\mathbf{H}|$ at 1.4 ns after the pulse has passed through the front face of the dielectric. Note the changed shape of the wavefront with resulting focusing. Finally, Fig. 27*c* shows the state of the pulse at 2.0 ns after emerging from the downstream boundary. The amplitude of pulse varies along y from a minimum of 0.683 V/m at the boundaries to 1.223 V/m at the center.

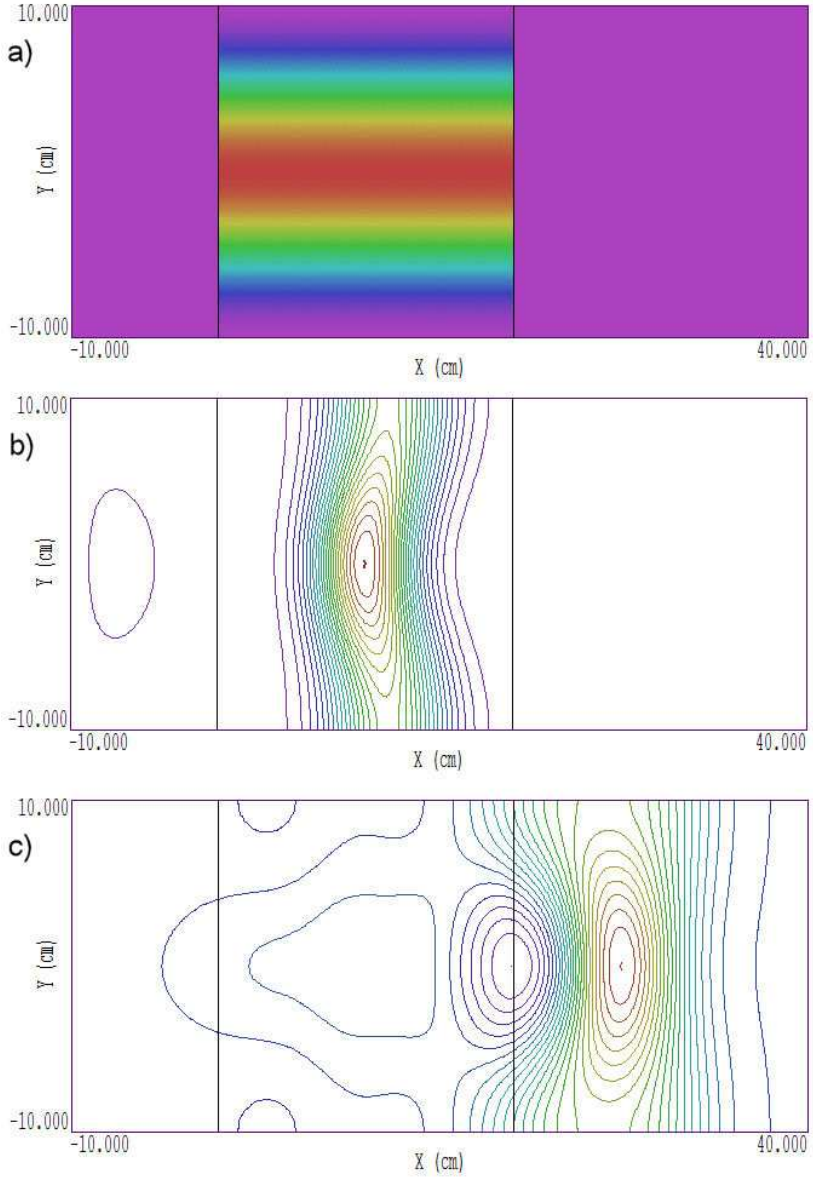

Figure 27: Example `GRATING`, graded dielectric structure to produce an intensity-modulated plane pulse.

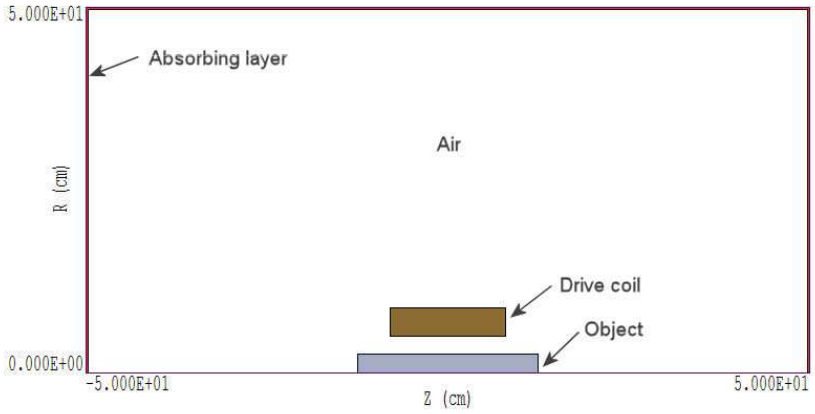

Figure 28: Geometry of example `MAGNEAR`, dimensions in cm.

7.5 Modeling pulsed magnetic fields

The `MAGNEAR` example illustrates how to use **EMP** to determine pulsed magnetic fields in the near-field limit. In particular, we shall calculate the diffusive penetration of fields into a conductive material. The benchmark calculation has a simple geometry for comparison to theory and calculations with **PerMag**. Nonetheless, the example demonstrates the range of techniques in dealing with low-frequency magnetic fields.

Figure 28 shows the geometry. The field is created by a pulsed solenoid in free space with inner radius 5.0 cm, outer radius 8.8 cm and length 16.0 cm. The coil encloses a rod of radius 2.5 cm and length 25.0 cm. In the solution, Region 1 is a matched absorbing layer with $\sigma = 1.060$ S/m to approximate the free-space condition. Region 2 (air) and Region 3 (drive coil) have $\sigma = 0.0$, while the conductivity of the object may be varied to investigate field diffusion.

We shall run the calculation to $t_{max} = 3.0$ ns. This time is much longer than the electromagnetic transit time over the coil region and about twice the transit time over the solution volume. Because the drive current points in the θ direction, the cylindrical solution has E type polarization. Following Sect. 2.3, the source quantity for the coil is dj_θ/dt. We shall use the normalized step function `SMOOTH.SRC`, which smoothly rises to a value of 1.0 at $t = 1.0$. The source command has the form:

```
Source(3) = SMOOTH.SRC 0.1E-9 5.575E12
```

The first parameter modifies the tabular function so that the risetime is $t_{rise} = 0.1$ ns. The second parameter normalizes the derivative of the current so that the total current at 3.0 ns is $I_\theta = 100.0$ A:

$$I_\theta = \left[(t_{max} - t_{rise}) + \frac{t_{rise}}{2}\right] \frac{dj_\theta}{dt} A_c, \tag{38}$$

where the coil cross-section area is $A_c = 6.08 \times 10^{-3}$ m^2.

In the initial run we set $\sigma = 0.0$ and make a comparison to a static field calculation using **PerMag**. A common issue for a static solution is the choice of boundary condition – the free-space condition is not an option. The standard approach is to create a solution volume much larger than the coil and to apply a Dirichlet condition (equivalent to a perfectly-conducting

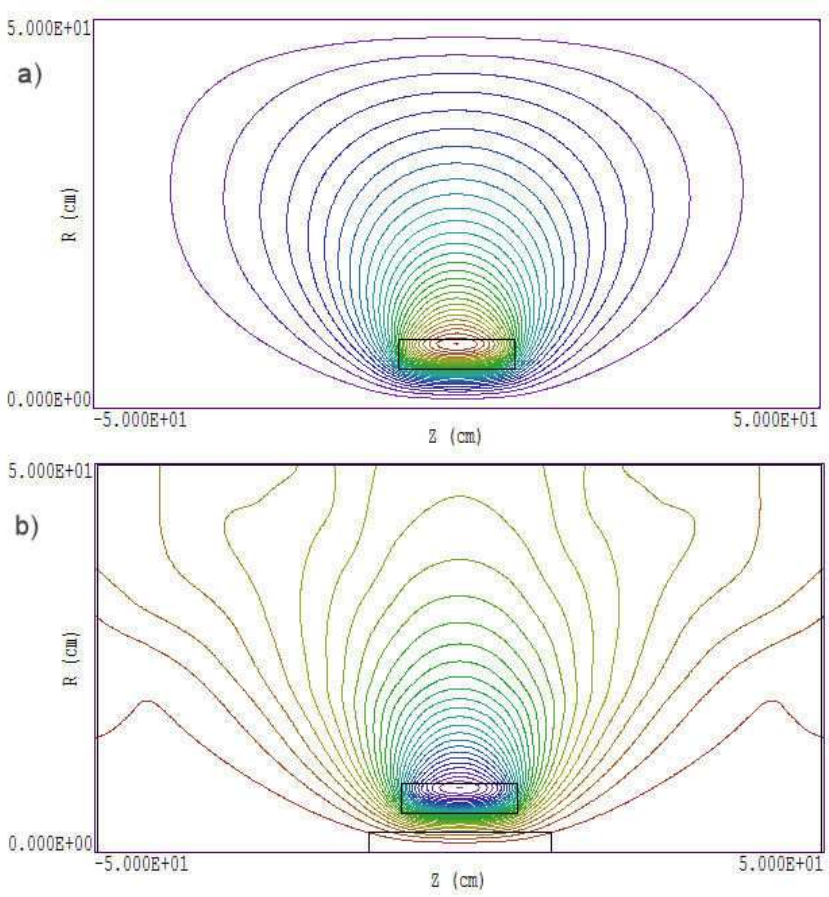

Figure 29: Field line calculations for MAGNEAR example. a) PerMag calculation with 100 A drive. b) EMP calculation at 3.0 ns

wall). Figure 29*a* shows resulting field lines. The **EMP** solution at $t = 3.0$ ns is plotted at the bottom. The large transient from the sharp initial current rise has propagated through the absorbing layer, and the solution is approaching a steady-state. Although there are clear differences in the field lines, they occur at a distance from the coil where the field is quite weak. The magnetic fields near the coil are almost identical in the two solutions. **PerMag** gives a value $|\mathbf{H}| = 471.9$ A/m at [$z = 0.0$ cm, $r = 0.0$ cm], while **EMP** gives 473.4 A/m (an difference of only 0.3%). The EMP solution would approach the ideal free-space solution is we let dj_θ/dt drop to zero and waited a few ns. In contrast to **PerMag**, **EMP** has the interesting feature that the near-field solution is relatively independent of the size of the solution volume. We can verify that long-term field level inside the coil is almost unchanged if we halve the size of the solution volume.

We can observe magnetic field diffusion effects by assigning a non-zero conductivity to the object (Region 4). For a pulsed current of duration Δt, the conductivity for skin depth δ is given approximately by

$$\sigma \cong \frac{2\Delta t}{\mu_0 \delta^2}. \tag{39}$$

For $\Delta t = 3.0$ ns and $\delta = 0.02$ m, Eq. 39 yields the value $\sigma = 12.0$ S/m. Assigning this

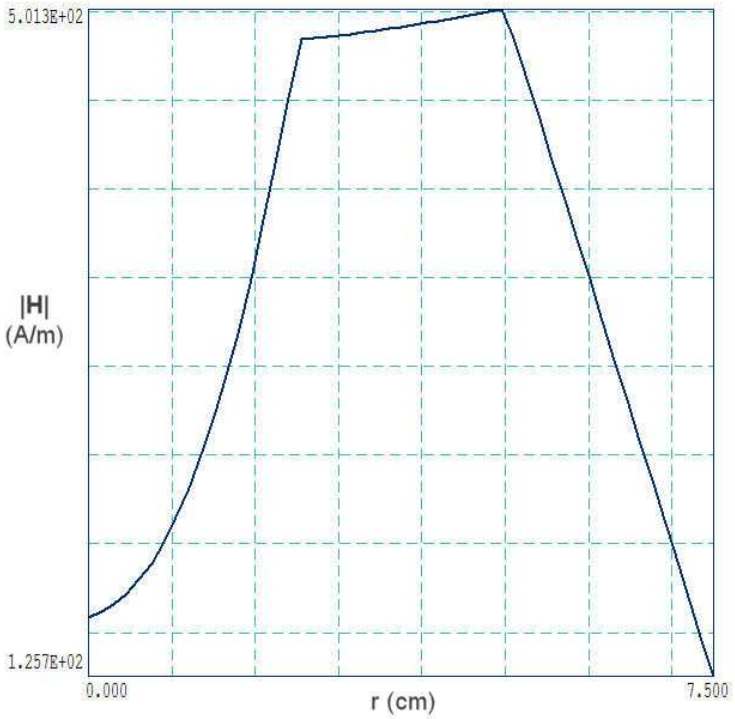

Figure 30: `MAGNEAR` example: radial varation of $|\mathbf{H}|$ at $z = 0.0$ cm and $t = 3.0$ ns.

conductivity to the object gives the field variation show in Figure 29. The figure plots a radial scan of magnetic field at the solution midplane from the axis to a position inside the coil. The field within the conducting object ($r < 2.5$ cm) follows the predicted diffusive pattern.

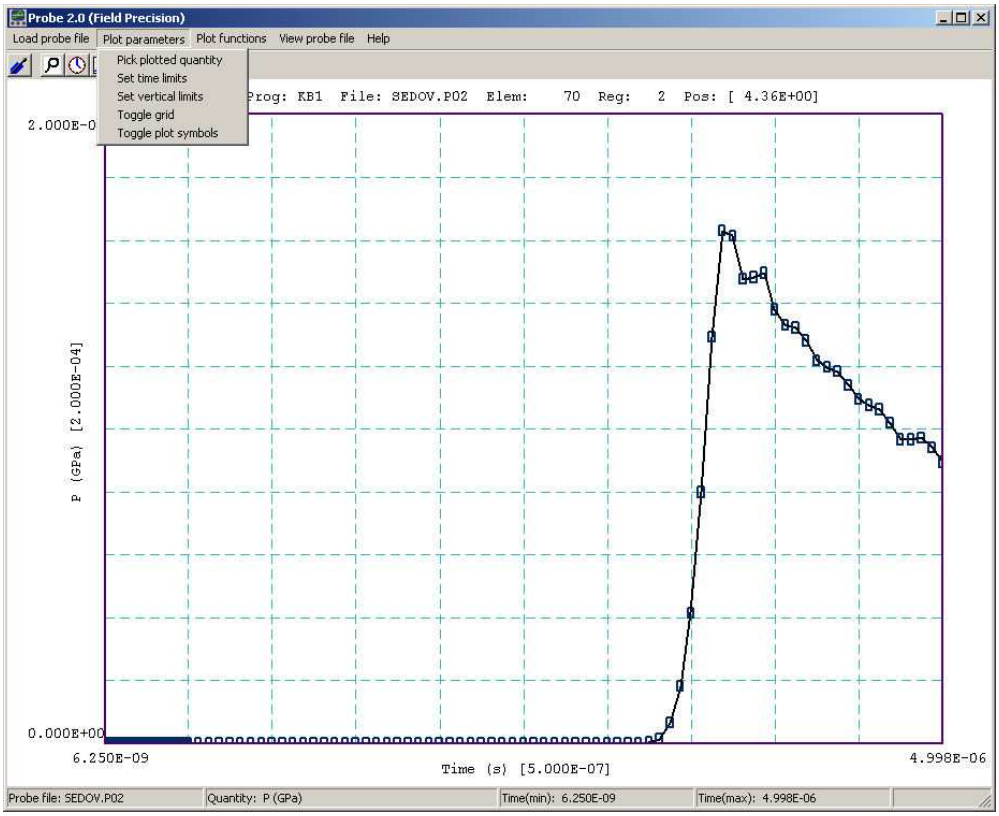

Figure 31: **Probe** screen shot.

8 Probe – history file plot utility

8.1 Introduction

Probe is the universal plotting program for all Field Precision initial-value solution codes. You can set from 1 to 20 probes by specifying positions in the solution program command script. The probes record quantities in an element or at a node as a function of time. The resulting text files have names of the form `FPREFIX.P01`,..., `FPREFIX.P12`, where `FPREFIX` is the run prefix.

Table 12 shows the standard probe file format. The first section is a header that contains the following information:

- Generating program name.
- Dimensionality of the generating program (1, 2 or 3).
- The spatial position of the probe (from 1 to 3 quantities).
- The index of the element sampled by the probe.
- The region number of the element.

- Conversion factors for the probe position and the recorded quantities.
- Labels for the recorded quantities.

Although the solution programs and their output files employ SI units (meters, kilograms,...), the graphical analysis displays often use practical units to make it easier to visualize results and to facilitate automatic grids. **Probe** multiplies file quantities by the conversion factors during the loading process. Note that the quantity *DConv* and conversion factors for positions are equal to *DUnit*, a variable used in many solution programs. After four lines of label information, the remainder of the file consists of data lines. Each line contains the time (in seconds) and one or more element or node quantities. Real numbers are recorded in E15.6 format.

8.2 Loading data files

When you start **Probe** the only active menu option is *Load probe*. Plotting and analysis functions become active when a probe file has been opened. The program displays a dialog showing all files with suffixes of the form `P01`,...,`P12`. Pick a file to analyze and click *OK*. Changing directories in the dialog will change the working directory of the program. If the load is successful, **Probe** creates a default plot of the data (Fig. 31).

The status bar at the bottom of the window contains the name of the probe file, the current plot quantity, and the temporal range of data. The default plot shows the first quantity recorded in the probe file over the full range of time. The horizontal and vertical scales are chosen so that the plot fits on the screen and the grid lines are automatically adjusted so that they lie on even values of the plotted quantity with easily recognized intervals (*e.g.*, 0.02, 0.05, 0.10, ...). The grid intervals are shown in parentheses next to the labels of the horizontal and vertical axes. The title line at the top of the plot shows the following information: generating program, probe file name, element number, region number and position. This information is recorded in hardcopy plots to help you archive your data.

8.3 Plot settings

The commands of the *Plotsettings* menu control the quantities, ranges and appearance of the plot. The screen plot automatically updates whenever you make a change.

PLOTTED QUANTITY

A dialog shows a list of element quantities included in the probe file. Highlight your choice and click *OK*.

TIME LIMITS

By default **Probe** shows the full time-span recorded. You can narrow the range by supplying values for the minimum and maximum time. Uncheck *Autoscale* in the dialog and supply maximum and minimum values. To return to the full range, check the *Autoscale* box.

Table 12: Example of the **Probe** file format

```
Field Precision probe file
Program: KB1
NDimen: 1
XPosition:   4.357E-02
ElementNo:    70
RegionNo:   2
NQuant: 6
DConv: 100.0
QConv1: 100.0
QConv2: 0.001
QConv3: 1.0E-9
QConv4: 1.0
QConv5: 1.0E-6
QConv6: 0.001
QLabel1: x or r (cm)
QLabel2: Rho (gm/cm3)
QLabel3: P (GPa)
QLabel4: T (deg-K)
QLabel5: U (MJ/kg)
QLabel6: v (km/s)

      Time             Re              Rho            Press
       (s)             (m)          (kg/m3)           (Pa)
 ============================================================
  6.250000E-09   4.357041E-02   1.000000E-01  -1.238464E+02
  1.250000E-08   4.357041E-02   1.000000E-01  -1.238464E+02
  1.874999E-08   4.357041E-02   1.000000E-01  -1.238464E+02
  2.499996E-08   4.357041E-02   1.000000E-01  -1.238464E+02
  3.124995E-08   4.357041E-02   1.000000E-01  -1.238464E+02
  ...

          Temp           Epsi           V(Av)
        (deg-K)         (J/kg)         (m/s)
   ============================================
      1.000000E-03   8.620001E+00   0.000000E+00
      1.000000E-03   8.620001E+00   0.000000E+00
      1.000000E-03   8.620001E+00   0.000000E+00
      1.000000E-03   8.620001E+00   0.000000E+00
      1.000000E-03   8.620001E+00   0.000000E+00
      ...
```

VERTICAL LIMITS

In the default mode **Probe** picks a scale to display the full range of the plotted quantity. You can narrow or expand the range by supplying minimum and maximum values. The program returns to full range if you check the *Autoscale* box or if you change quantities using *Pick plotted quantity*.

TOGGLE GRID

Switch between grid and and fiducial lines in the plot.

TOGGLE PLOT SYMBOLS

Include or remove symbols to mark the recorded points.

8.4 Plot functions

The commands of the *Plot functions* menu activate the *Oscilloscope mode* of the program and also send plots to hardcopy devices or plot files.

OSCILLOSCOPE MODE

When you issue this command, **Probe** simulates a digital oscilloscope. As shown in Fig. 32, the mouse cursor changes to a cross-hair pattern when it is inside the plot window. The program adds movable fiducial lines to mark the current point. You can drag the fiducials along the time axis by moving the mouse. A box at the bottom of the plot shows values of the time and plotted quantity at the current position. If you press the left mouse button, the program displays a box with the following information about the current point:

- Time, t.
- Value of the plotted quantity, $V(t)$.
- Derivative of the plotted quantity, $dV(t)/dt$.
- Definite integral of the plotted quantity, $\int_0^t V(t')dt'$.

The definite integral is taken from the time of the first recorded value in the probe file to that of the current point. You can find integrals between points by subtracting values. Other functions of the program are deactivated in the *Oscilloscope mode*. Press the right mouse button or the *Esc* key to return to normal program operation.

SMOOTH DISPLAY

Use this command one or more times to smooth the currently-displayed trace. Smoothing applies to the screen display and exported plots, but does not affect the data values in the probe file.

DEFAULT PRINTER

Probe can port copies of the plot to any installed Windows printer. The program sends output

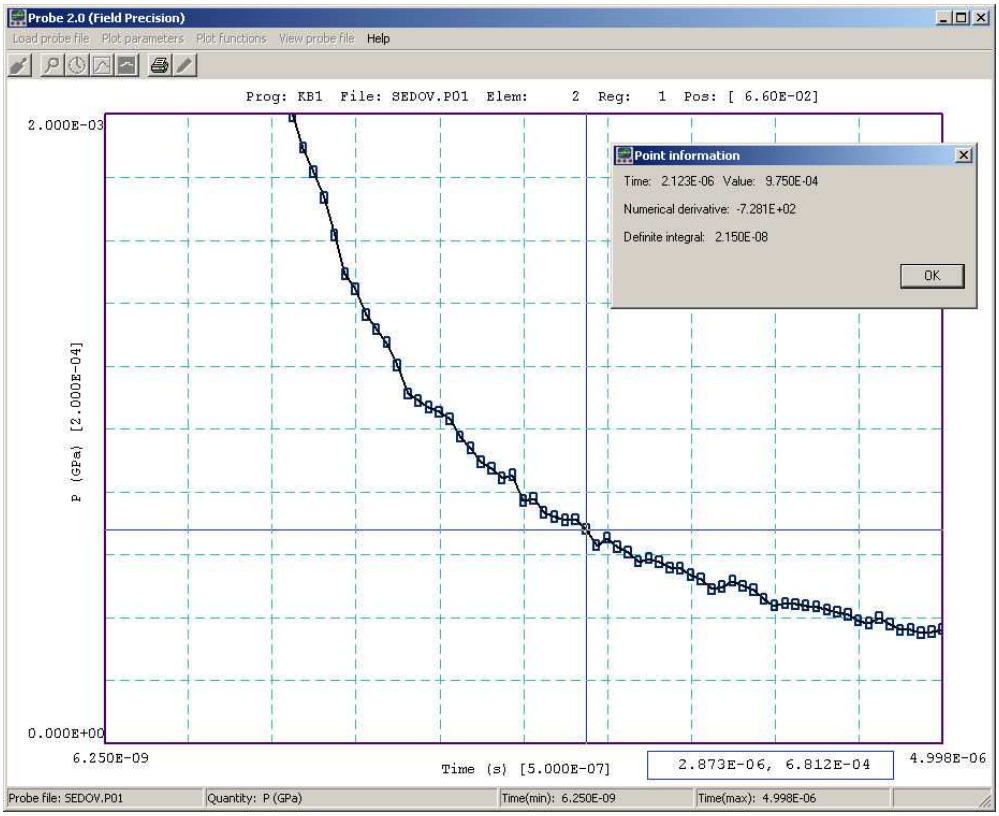

Figure 32: **Probe** in the *Oscilloscope mode*.

to the default printer, so be sure to select the correct device using the *Settings/Printer* function of Windows before making the plot.

SAVE PLOT FILE

Use this command to create a graphics file of the current plot in either Windows Bitmap (BMP) or Portable Network Graphics (PNG) formats. In the dialog, specify the format, the size in pixels and the file prefix. The graphics file is created in the current directory.

COPY TO CLIPBOARD

Copy the plot to the clipboard in in Windows MetaFile format.

8.5 Information

PROBE FILE INFORMATION

Display information on the probe file in a message box (Fig 33). The quantity $NSkip$ in line 7 is used for long files. There is no reason to store more than 1000 points for plots on typical screens and hardcopy devices. When there are less than 1000 data lines, **Probe** loads all points ($NSkip = 1$). When the file contains 1000 to 2000 data lines, the program loads every second

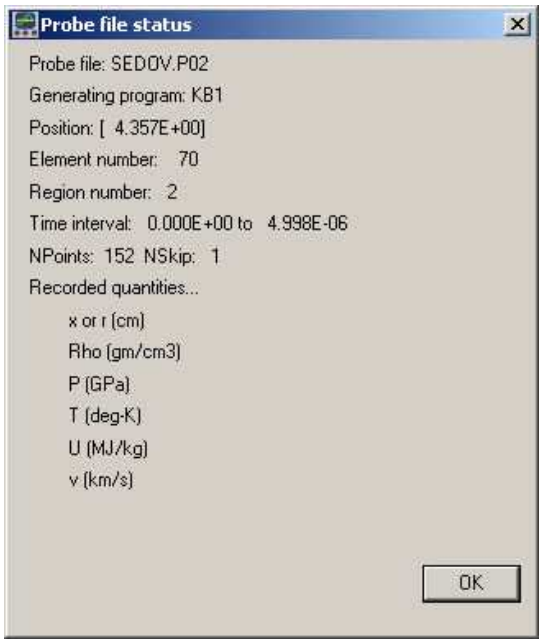

Figure 33: **Probe** file-information message box.

point ($NSkip = 2$), and so forth. In this way the **Probe** can handle probe files of any length without exceeding memory limits.

VIEW PROBE FILES

Load a probe file into the internal editor so you can inspect the data directly. The editor runs in read-only mode so that you cannot change the file. Exit the editor to return to program operation.

PROBE MANUAL

Show this document in your default PDF viewer.

Index