LA-UR-87-115
Issued: January 1987

User's Guide for the POISSON/SUPERFISH Group of Codes

M. T. Menzel, H. K. Stokes
Accelerator Theory and Simulation Group, AT-6,
Los Alamos National Laboratory
Los Alamos, New Mexico 87545

User's Guide for the POISSON/SUPERFISH Group of Codes

Los Alamos National Laboratory
Los Alamos, New Mexico

Contents

Chapter 1

Introduction

Contents 1–1

1.1 Overview of the POISSON/SUPERFISH Programs

The POISSON/SUPERFISH Group Codes are a set of programs written by Ronald Holsinger, with theoretical assistance from Klaus Halbach, to solve two distinct problems—the calculation of magnetostatic and electrostatic fields, and the computation of the resonant frequencies and fields in radio-frequency cavities—in a two-dimensional Cartesian or three-dimensional cylindrical geometry. These codes are widely used for the design of magnets and radio frequency cavities.

These codes are grouped together since they utilize common programs as well as common subroutines. The initial setup of either a magnet or a cavity problem uses two common programs, AUTOMESH and LATTICE, to generate the mesh for the given input geometry. The program TEKPLOT provides graphical output for either of the two problems.

When the mesh has been generated and the appropriate materials and boundary conditions have been specified, the finite difference equations are set up and solved by one of the equation-solving programs. For magnet problems, these programs are PANDIRA, POISSON and/or MIRT; for cavity problems, the programs are SUPERFISH and SFO1.

The programs POISSON and PANDIRA solve electrostatic and magnetostatic problems. PANDIRA also solves permanent magnet problems. The program POISSON solves the discretized problem by the traditional "successive point over-relaxation" (SPOR) method which is very efficient for problems that converge rapidly with this method. However, it is well known that many interesting problems converge slowly or not at all with SPOR. In those cases, the program PANDIRA should be used.

The program PANDIRA solves by a "direct" method and iteration is required only for nonlinear problems. The fact that the mesh is topologically regular means that the coefficient matrix of the finite difference equations has an identical structure for any problem. Because this structure is always the same, a very efficient sparse matrix method is used to solve these systems, which typically contain many thousands of equations.

The program SUPERFISH uses this direct solution method for the eigenvalue problem to determine the resonant frequencies in standing-wave radio-frequency cavities.

A list of all the programs and their functions in the POISSON/SUPERFISH Group Codes is given in Sec. 1.5. The individual programs are described in detail under their chapter listing.

1.2 Structure of the Manual

Since this manual is a user's guide for the POISSON/SUPERFISH programs, the material covered is primarily information that is relevant for the execution of the programs. The user is advised to consult the POISSON/SUPERFISH Reference Manual for information on the theoretical derivations, numerical methods, and an in-depth explanation of the programs.

All the chapters and sections of this manual are largely self-contained; any outside information that is required is clearly indicated by a reference to the necessary section. The user does not have to read the complete chapter; the sections could be skimmed and the subsections that are relevant to the problem should be read.

Chapter 1, the introduction, provides an overview of what the programs do and how they interact and work together. Each of the following chapters describes one of the programs in the group. These chapters are structured in the same manner: they begin with a general introduction followed by sections. Each section gives a general overview of the topic, with detailed information discussed in the subsections that follow.

We suggest that a new user begin by reading Chapter 1, following the step-by-step instructions for the execution of the example in Sec. 1.3 or Sec. 1.4. Then, for each chapter, the user should read both the introduction and the input section, then skip to the examples at the end of the chapter. At that point, the user may refer to sections which are of interest or pertinent to the problem.

Additional examples are given in Chapter 10. The examples there utilize different options to illustrate the great variety of problems that can be solved by these programs.

1.3 POISSON Example—H-Shaped Dipole Magnet

In this section, we use a simple example to illustrate the use of the POISSON Group Codes in the solution of a magnet problem. We are interested in calculating the magnetic field distribution of a long dipole magnet as used in circular particle accelerators. Because the dipole is long, the calculation of the field far from the ends of the magnet is essentially a two dimensional problem and thus solvable with the POISSON programs. Figure 1.1 shows the vertical cross section picture of this H-shaped magnet.

A step-by-step solution of the H-magnet problem is outlined in the subsections below. In order to avoid the use of unfamiliar POISSON terminology at this stage, only brief explanations of the procedures are given. A complete detailed description of the H-magnet problem is given in subsequent chapters.

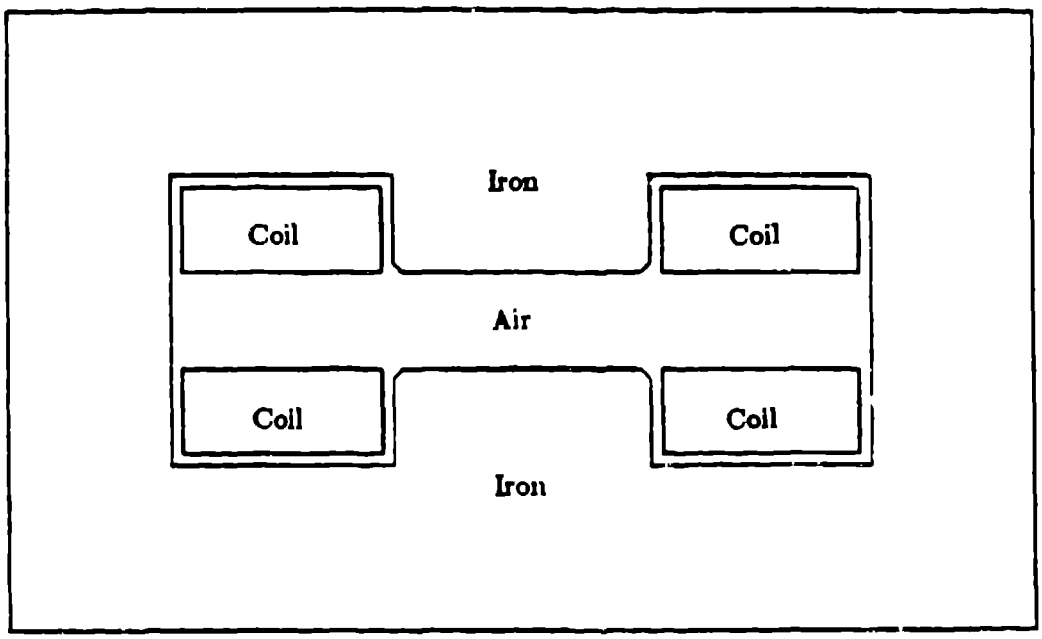


Fig. 1-1: A vertical cross section of an H-shaped dipole magnet showing iron, coil, and air regions.

The terminal output for the execution of the programs on a CRAY computer is also listed. The user only types the underlined quantities; the other text is generated by the executable program.

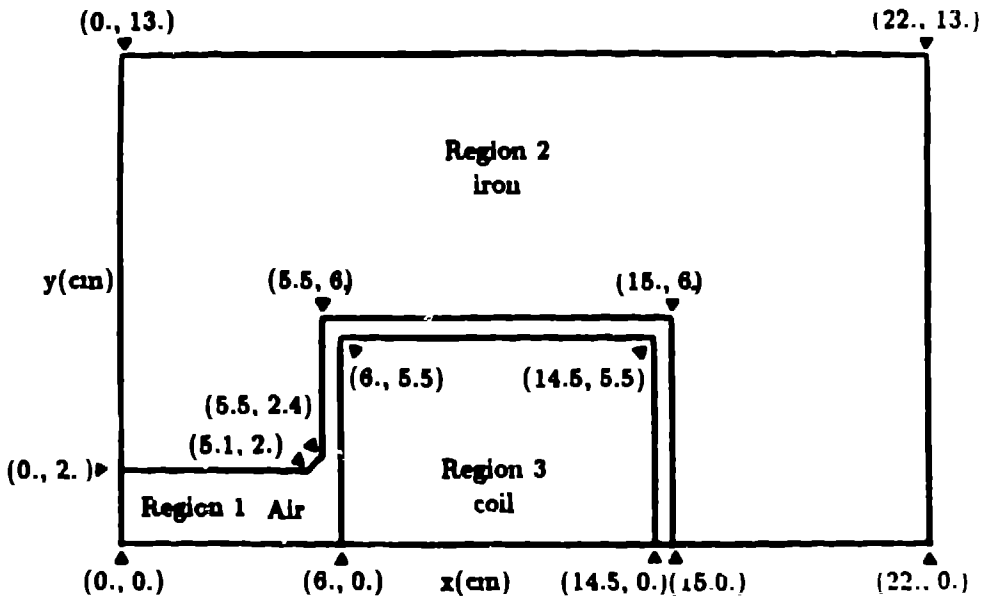
1.3.1 Executing AUTOMESH

The execution of AUTOMESH requires an input file in the format shown below. A successful AUTOMESH run generates an output file, TAPE73, which is the input file for LATTICE.

Because of symmetry options available in the POISSON programs, we only need to input the upper right-hand quadrant of Fig. 1-1. Below is a listing of this input file, HMAG, corresponding to the geometric configuration on the right. A complete, line-by-line, description of HMAG is given in Sec. 2.5. Here we indicate that the input file uses two Fortran NAMELIST entries: $REG–used to input region parameters and $PO– used to input the physical coordinates of the region.

HMAG, Input File for AUTOMESH:

```
h-magnet test, uniform mesh 4/23/85
$reg nreg=3,dx=.45,xmax=22.,ymax=13.,npoint=5$
$po x= 0.0, y= 0.0$
$po x=22.0, y= 0.0$
$po x=22.0, y=13.0$
$po x= 0.0, y=13.0$
$po x= 0.0, y= 0.0$
$reg mat=2,npoint=10$
$po x= 0.0, y= 2.0$
$po x= 5.1, y= 2.0$
$po x= 5.5, y= 2.4$
$po x= 5.5, y= 6.0$
$po x=15.0, y= 6.0$
$po x=15.0, y= 0.0$
$po x=22.0, y= 0.0$
$po x=22.0, y=13.0$
$po x= 0.0, y=13.0$
$po x= 0.0, y= 2.0$
$reg m t=1,npoint=5,
cur=-25455.791$
$po x= 6.0, y= 0.0$
$po x=14.5, y= 0.0$
$po x=14.5, y= 5.5$
$po x= 6.0, y= 5.5$
$po x= 6.0, y= 0.0$
```



Following the listing of HMAG is the execution listing of AUTOMESH using this file.

```
automesh

 ?type input file name
? hmag
region no  1
ok
region no  2
ok
region no  3
ok
stop
automesh ctss time      .380    seconds
cpu=    .120      i/o=    .218      mem=   0.41

all done
```

1.3.2 Executing LATTICE

The execution of LATTICE with the input file, TAPE73, is shown below. There are no changes to the default values (this is designated by typing **s**).

```
lattice

 ?type input file name
? tape73

beginning of lattice execution
dump 0 will be set up for poisson
h-magnet test, uniform mesh  4/23/85
 ?type input values for con(?)
? s

elapsed time  =  0.4 sec.
0iteration converged
elapsed time  =  0.7 sec.
generation completed
dump number 0 has been written on tape35.
stop
lattice  ctss time      1.005     seconds
cpu=     .600     i/o=    .335     mem=    0.70

all done
```

1.3.3 Executing TEKPLOT after LATTICE

After execution of LATTICE, we execute TEKPLOT to verify that our input and the generated mesh is correct. We make two passes through TEKPLOT: the first **go** plots on the screen the input geometry as shown in Fig. 1–2; the second **go** plots the mesh with the geometry as shown in Fig. 1–3. To exit from TEKPLOT, we need to type **-1 s** as given below.

```
tekplot

 ?type input data- num, itri, nphi, inap, nswxy,
? s
input data
num= 0     itri= 0     nphi= 0     inap= 0     nswxy= 0
plotting  prob. name = h-magnet test, uniform mesh 4/23/85 cycle = 0
 ?type input data- xmin, xmax, ymin, ymax,
? s
input data
xmin= 0.000   xmax= 22.000   ymin= 0.000   ymax= 13.000
?type go or no
? go
```

A carriage return (CR) after go clears the screen and plots Fig. 1–2. A second CR clears the plot off the screen and continues execution.

```
type input data- num, itri, nphi, inap, nswxy,
? 0 1 g
input data
num= 0   itri= 1   nphi= 0   inap= 0   nswxy= 0
plotting prob. name = h-magnet test, uniform mesh 4/23/85 cycle = 0
 type input data- xmin, xmax, ymin, ymax
? g
input data
xmin= 0.0000   xmax= 22.000   ymin= 0.0000   ymax= 13.000
 ? type go or no
? go
```

A CR after go clears the screen and plots Fig. 1–3. A second CR clears the screen again and continues with execution. A negative value for the variable NUM terminates execution.

```
 ?type input data- num, itri, nphi, inap, nswxy,
-1 g
cpu=       .031 i/o=    .681 mem=       .053

   all done
```

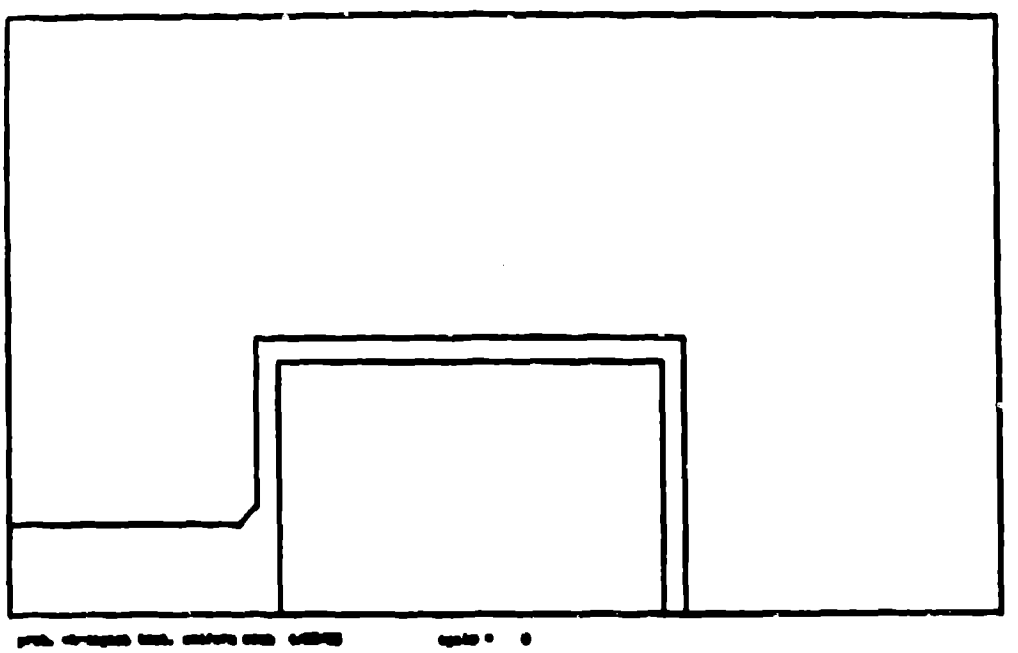


Fig. 1–2: Plot from TEKPLOT of the magnet geometry. This is a verification of the input data to AUTOMESH for the problem "h-magnet test, uniform mesh 4/23/85."

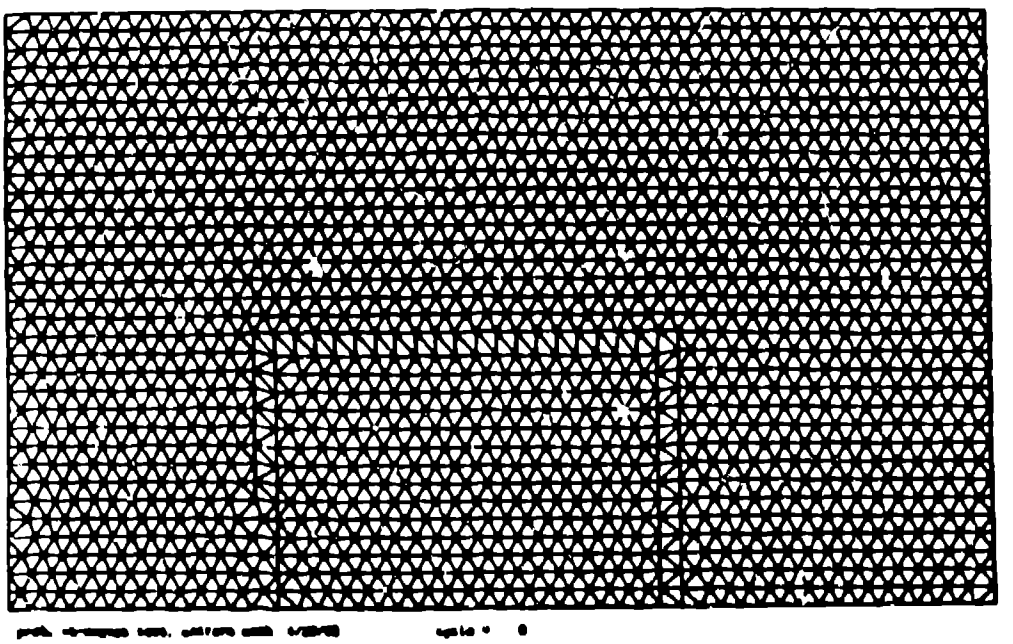


Fig. 1–3: Plot from TEKPLOT of the mesh generated by LATTICE for the problem "h-magnet test, uniform mesh 4/23/85."

1.3.4 Executing POISSON

Since the mesh plot output from TEKPLOT looks good, we proceed to execute POISSON. We choose, by typing tty, to input data from the terminal. We then enter:

0	–	to read dump 0 on TAPE35 generated by LATTICE
*6 0 *46 6	–	to change two of the CON array variables, CON(6) and CON(46), specifying use of μ = finite with use of internal B^2 *vs.* γ and specifying a symmetrical H-magnet, respectively.
−1	–	to terminate POISSON as shown below.

```
poisson
type "tty" or input file name
? tty
 ?type input value for dump num
? 0
beginning of poisson execution from dump number  0
prob. name = h-magnet test, uniform mesh 4/23/85
?type input values for con(?)
*6 0 *46 6 s
elapsed time   =  1.0 sec

0  cycle  amin          amax         residual-air      eta-air   rhoair  xjfact
                       gmax         residual-iron     eta-iron  rhofe
0  0       0.0000e+00  0.0000e+00   1.0000e+00        1.0000    1.0000  1.000
                       4.0000e-03   1.0000e+00        1.0000    1.0000
0  50                    rhoair optimized             0.9903    1.9558  lambda = 9.9976e-01
0  50     -4.7296e+04  0.0000e+00   5.7349e-02        0.9903    1.9558  1.0000
                       3.9026e-03   3.4407e-02        1.0039    1.0000
0  100                   rhoair optimized             0.9717    1.9578  lambda = 9.9978e-01
0  100    -1.0012e+05  0.0000e+00   2.5360e-02        0.9717    1.9578  1.0000
                       2.0634e-02   2.0906e-02        0.9834    1.0000
0  200                   rhoair optimized             0.8960    1.9478  lambda = 1.0003e+00
0  200    -1.1958e+05  0.0000e+00   1.8887e-04        0.8960    1.9478  1.0000
                       4.6352e-02   7.9336e-05        0.8900    1.0000
0  370    -1.1939e+05  0.0000e+00   3.7301e-07        0.9367    1.9478  1.0000
                       4.6392e-02   1.6135e-07        0.9328    1.0000

solution converged in 370 iterations
elapsed time = 3.8 sec.
dump number 1 has been written on tape35
 ?type input value for dump num
? -1
stop
poisson ctss time 4.427 seconds
cpu= 2.831  i/o= 1/081  mem= .514
all done
```

In addition to the output of a binary file, dump 1 of TAPE35, POISSON generates an ASCII output file, OUTPOI. All of the group codes produce a similar output file; the name is formed by "OUT" followed by the first three letters of the program generating the file. That is, AUTOMESH generates OUTAUT, LATTICE generates OUTLAT, etc.

The file OUTPOI contains a summary of input data, a listing of iteration variables as printed at the terminal, and a table listing the calculated field components and their gradients on axis (y=0) for noniron regions only. This table, lines 188--228 of OUTPOI, is shown in Fig. 1-4.

solution converged in 370 iterations
elapsed time = 4.2 sec.
dump number 1 has been written on tape35.
least squares edit of problem , cycle 370
'h' mag symmetry type
stored energy = 1.4249e+03 joules / meter or radian
xjfact= 1.000000

k	l	a(vector)	x	y	bx(gauss)	by(gauss)	bt(gauss)	dby/dy (gauss/cm)	dby/dx (gauss/cm)	aft
1	1	0.000000e+00	0.00000	0.00000	0.000	15212.250	15212.250	0.0000e+00	0.0000e+00	5.3e-04
2	1	-6.829781e+03	0.44898	0.00000	0.000	15210.825	15210.825	0.0000e+00	-6.4635e+00	-3.3e-03
3	1	-1.365822e+04	0.89796	0.00000	0.000	15206.195	15206.195	0.0000e+00	-1.4658e+01	3.7e-03
4	1	-2.048365e+04	1.34694	0.00000	0.000	15197.056	15197.056	0.0000e+00	-2.7097e+01	5.5e-03
5	1	-2.730348e+04	1.79592	0.00000	0.000	15180.665	15180.665	0.0000e+00	-4.7887e+01	6.5e-03
6	1	-3.411342e+04	2.24490	0.00000	0.000	15151.761	15151.761	0.0000e+00	-8.4536e+01	7.8e-03
7	1	-4.090553e+04	2.69388	0.00000	0.000	15100.434	15100.434	0.0000e+00	-1.5084e+02	9.2e-03
8	1	-4.766697e+04	3.14286	0.00000	0.000	15008.404	15008.404	0.0000e+00	-2.7096e+02	1.1e-02
9	1	-5.437198e+04	3.59184	0.00000	0.000	14843.504	14843.504	0.0000e+00	-4.8257e+02	2.4e-02
10	1	-6.097737e+04	4.04082	0.00000	0.000	14554.675	14554.675	0.0000e+00	-8.2909e+02	1.1e-01
11	1	-6.741291e+04	4.48980	0.00000	0.000	14077.744	14077.744	0.0000e+00	-1.3169e+03	5.1e-01
12	1	-7.358221e+04	4.93878	0.00000	0.000	13369.124	13369.124	0.0000e+00	-1.8577e+03	1.8e+00
13	1	-7.938901e+04	5.38776	0.00000	0.000	12435.718	12435.718	0.0000e+00	-2.1782e+03	-7.6e+00
14	1	-8.457337e+04	6.00000	0.00000	0.000	10893.018	10893.018	0.0000e+00	-2.8855e+03	6.7e+00
15	1	-9.114731e+04	6.44737	0.00000	0.000	9613.098	9613.098	0.0000e+00	-2.8434e+03	1.5e+01
16	1	-9.518656e+04	6.89474	0.00000	0.000	8416.533	8416.533	0.0000e+00	-2.5622e+03	5.1e+00
17	1	-9.871191e+04	7.34211	0.00000	0.000	7354.552	7354.552	0.0000e+00	-2.2169e+03	1.9e+00
18	1	-1.017932e+05	7.78947	0.00000	0.000	6439.740	6439.740	0.0000e+00	-1.8967e+03	9.3e-01
19	1	-1.044947e+05	8.23684	0.00000	0.000	5655.367	5655.367	0.0000e+00	-1.6278e+03	5.0e-01
20	1	-1.068700e+05	8.68421	0.00000	0.000	4978.523	4978.523	0.0000e+00	-1.4113e+03	2.8e-01
21	1	-1.089623e+05	9.13158	0.00000	0.000	4387.592	4387.592	0.0000e+00	-1.2403e+03	1.6e-01
22	1	-1.108059e+05	9.57895	0.00000	0.000	3864.280	3864.280	0.0000e+00	-1.1065e+03	9.6e-02
23	1	-1.124277e+05	10.02632	0.00000	0.000	3393.820	3393.820	0.0000e+00	-1.0022e+03	5.6e-02
24	1	-1.138487e+05	10.47368	0.00000	0.000	2964.550	2964.550	0.0000e+00	-9.2096e+02	3.0e-02
25	1	-1.150850e+05	10.92105	0.00000	0.000	2567.346	2567.346	0.0000e+00	-8.5778e+02	1.3e-02
26	1	-1.161495e+05	11.36842	0.00000	0.000	2195.081	2195.081	0.0000e+00	-8.0868e+02	9.6e-04
27	1	-1.170520e+05	11.81579	0.00000	0.000	1842.181	1842.181	0.0000e+00	-7.7002e+02	-9.5e-03
28	1	-1.178000e+05	12.26316	0.00000	0.000	1504.255	1504.255	0.0000e+00	-7.4120e+02	-2.0e-02
29	1	-1.183990e+05	12.71053	0.00000	0.000	1177.824	1177.824	0.0000e+00	-7.1882e+02	-3.4e-02
30	1	-1.188552e+05	13.15789	0.00000	0.000	860.097	860.097	0.0000e+00	-7.0192e+02	-5.7e-02
31	1	-1.191702e+05	13.60526	0.00000	0.000	548.874	548.874	0.0000e+00	-6.8934e+02	-9.7e-02
32	1	-1.193473e+05	14.05263	0.00000	0.000	243.246	243.246	0.0000e+00	-6.7854e+02	-2.4e-01
33	1	-1.193889e+05	14.50000	0.00000	0.000	-59.749	59.749	0.0000e+00	-6.6709e+02	-1.8e+00
34	1	-1.193575e+05	15.00000	0.00000	0.000	-60.019	60.019	0.0000e+00	0.0000e+00	-1.0e+00

Fig. 1-4: A section from the file OUTPOI generated by POISSON for the H-shaped magnet problem 4/23/85, cycle = 370.

1.3.5 Executing TEKPLOT after POISSON

After a successful execution of POISSON, we execute TEKPLOT again. This time we designate dump 1 of TAPE73 and 20 field lines to generate Fig. 1–5.

```
tekplot
 ?type input data- num, itri, nphi, inap, nswxy,
?1 0 20 s
input data
num= 1    itri= 0    nphi= 20    inap= 0    nswxy= 0
plotting prob. name = h-magnet test, uniform mesh 4/23/85 cycle = 370
 ?type input data-  xmin, xmax, ymin, ymax,
? s
input data
xmin= 0.000 xmax= 22.000 ymin= 0.000 ymax= 13.000
?type go or ro
? go
```

A CR after go clears the screen and plots Fig. 1–5. A second CR clears the screen and produces the prompt line. A negative value for the variable NUM terminates execution.

```
?type input data- num, itri, nphi, inap, nswxy,
?  -1 s
tekplot ctss time .561 seconds
cpu= .037 i/o= .430 mem= .044

 all done
```

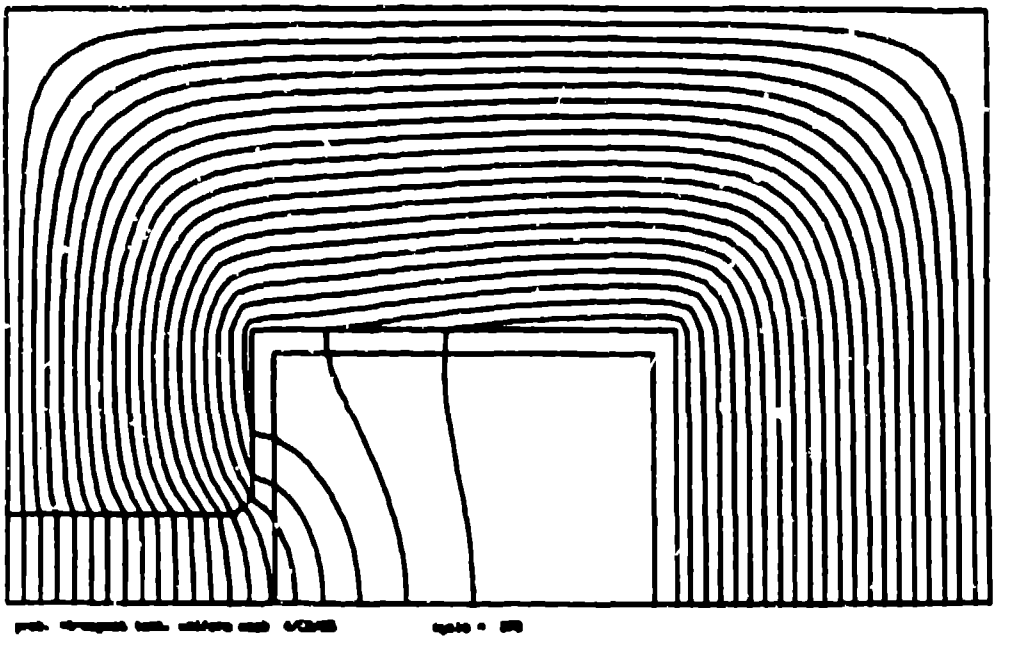


Fig. 1-5: Plot from TEKPLOT of the magnetic field lines generated by POISSON for the problem "h-magnet test, uniform mesh 4/23/85".

1.4 SUPERFISH Example—Modified Pillbox Cavity

In this section, we use a simple example to illustrate the use of the SUPERFISH Codes to find the lowest frequency in a cavity. The structure that we consider is a pillbox cavity with beam pipes entering and exiting the sides, as shown in Fig. 1-6.

A step-by-step solution of this problem is outlined in the subsections below. In order to avoid the use of undefined SUPERFISH terminology at this stage, only brief explanations of the procedures are given. A complete detailed description of a SUPERFISH problem is given in subsequent chapters.

The terminal output for the execution of the programs on a CRAY computer is also listed. The user only types the underlined quantities; the other text is generated by the executable program.

1.4.1 Executing AUTOMESH

The execution of AUTOMESH requires an input file in the format shown below. A successful AUTOMESH run generates an output file, TAPE73, which is the input file for LATTICE.

Taking into account the symmetry of the problem, we need only input a quarter of the geometry in cylindrical coordinates—the default coordinate system for SUPERFISH. Below is a listing of this input file, MODPIL, corresponding to the geometric configuration on the right. The picture

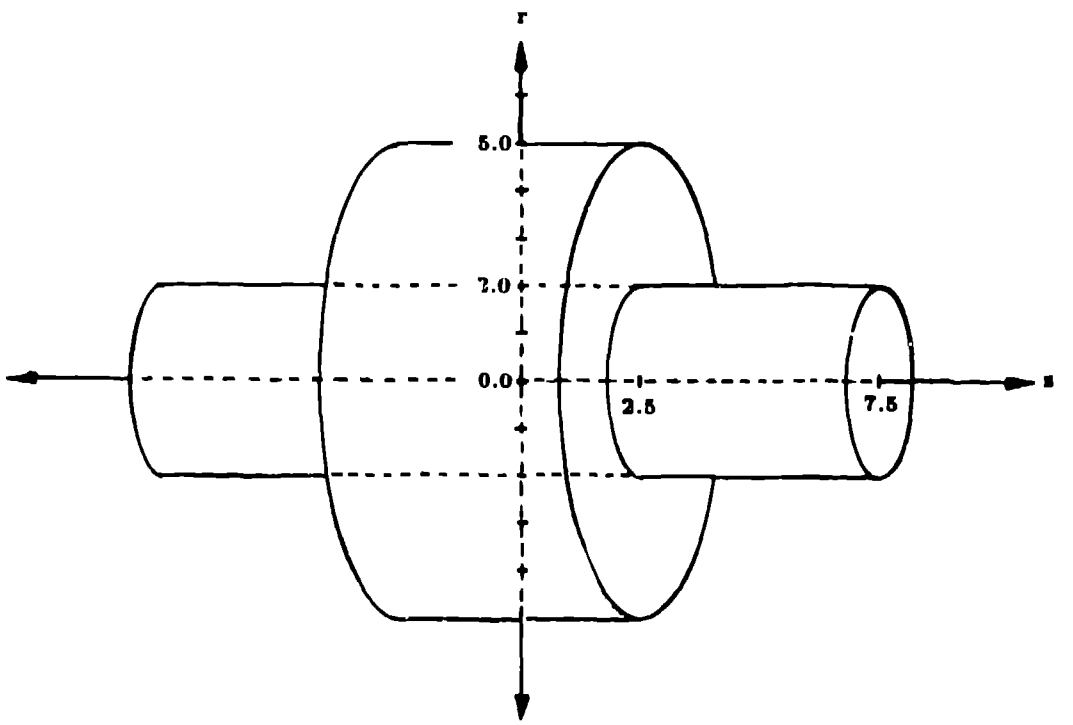
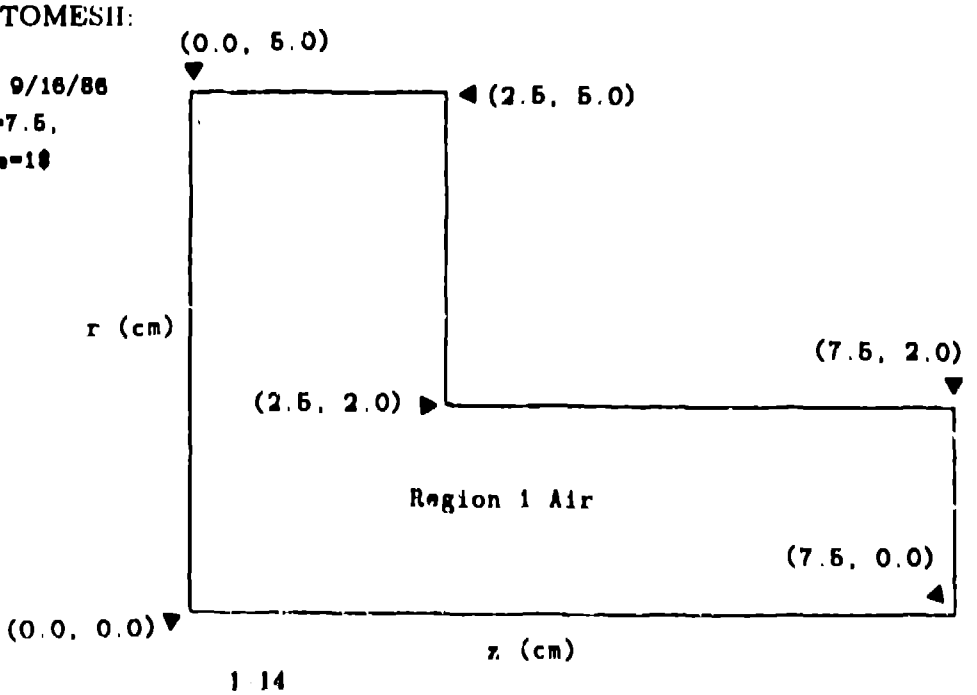


Fig. 1-6: Modified pillbox cavity.

of the full cavity as given in Fig. 1–6, can be visualized by first reflecting the figure through the r-axis and then rotating it around the z-axis.

MODPIL, Input File for AUTOMESH:

```
1 modified pillbox cavity 9/16/86
 $reg nreg=2,dx=.25,xmax=7.5,
 ymax=5.0,npoint=7,ndrive=1$
 $po x= 0.0, y= 0.0$
 $po x= 0.0, y= 5.0$
 $po x= 2.5, y= 5.0$
 $po x= 2.5, y= 2.0$
 $po x= 7.5, y= 2.0$
 $po x= 7.5, y= 0.0$
 $po x= 0.0, y= 0.0$
 $reg npoint=1$
 $po x= 0.0, y= 5.0$
```



A detailed description of an input file to AUTOMESH is given in Chapter 2. Here we indicate that the input file uses two Fortran NAMELIST entries: REG – used to input region parameters, and PO – used to input the physical coordinates of the region. NAMELIST input starts in column 2.

Specifically, MODPIL has a title line with entry starting in column 1 indicating a SUPERFISH run, a first REG entry for an air region followed by PO entries specifying the coordinates of this region with all (x, y) entries corresponding to (z, r) entries, and a second REG entry followed by one PO entry indicating a drive point region.

The execution listing of AUTOMESH using the input file MODPIL is given below.

```
automesh

 ?type input file name
? modpil

region no.  1
logical boundary segment end points
iseg        kb  lb        kd   ld       ke  le
  1          1   1         0    1        1  24
  2          1  24         1    0       11  24
  3         11  24         0   -1       11  10
  4         11  10         1    0       31  10
  5         31  10         0   -1       31   1
  6         31   1        -1    0        1   1

region no.  2
ok
stop
automesh ctss time      .343    seconds
cpu=    .059     i/o=    .228     mem=   0.57

all done
```

1.4.2 Executing LATTICE

The execution of LATTICE using the input file, TAPE73, is shown below. All the default values including the boundary conditions, which specify perfectly conducting metal for upper- and right-hand boundaries and axis of symmetry for bottom- and left-hand boundaries, are correct for our problem. Therefore, we type s (meaning skip) to designate no changes in the input.

```
lattice

?type input file name
? tape73

beginning of lattice execution
dump 0 will be set up for superfis
1modified pillbox cavity 9/16/86
?type input values for con(?)
? s

elapsed time  =  0.3 sec.
0 iteration converged
elapsed time  =  0.3 sec.
generation completed
dump number 0 has been written on tape35.
stop
lattice  ctss time      .532     seconds
cpu=     .137     i/o=    .308     mem=    0.087

all done
```

1.4.3 Executing TEKPLOT after LATTICE

After execution of LATTICE, we execute TEKPLOT to verify that our input and the generated mesh is correct. We make two passes through TEKPLOT: the first **go** plots on the screen the input geometry as shown in Fig. 1–7; the second **go** plots the mesh with the geometry as shown in Fig. 1–8.

```
tekplot

 ?type input data- num, itri, nphi, inap, nswxy,
? s
input data
num= 0     itri= 0     nphi= 0     inap= 0     nswxy= 0
plotting prob. name = modified pillbox cavity 9/16/86
cycle = 0
?type input data- xmin, xmax, ymin, ymax,
? s
input data
xmin= 0.000    xmax= 7.500    ymin= 0.000    ymax= 5.000
 ?type go or no
? go
```

A carriage return (CR) after **go** clears the screen and plots Fig. 1–7. The second CR clears the screen and generates the prompt line:

```
 type input data- num, itri, nphi, inap, nswxy,
? 0 1 s
```

```
input data
num= 0    itri= 1    nphi= 0    inap= 0    nswxy= 0
plotting prob. name = modified pillbox cavity 9/16/86
                      cycle = 0
type input data- xmin, xmax, ymin, ymax,
? g
input data
xmin= 0.000    xmax= 7.500    ymin= 0.000    ymax= 5.000
? type go or no
? go
```

A CR after go clears the screen and plots Fig. 1–8. A second CR clears the screen and generates the prompt line. A negative value for the variable NUM terminates execution.

```
 ?type input data- num, itri, nphi, inap, nswxy,
-1s
tekplot      ctss time 1.186 seconds
cpu=       .105 i/o=    .989 mem=       .093
  all done
```

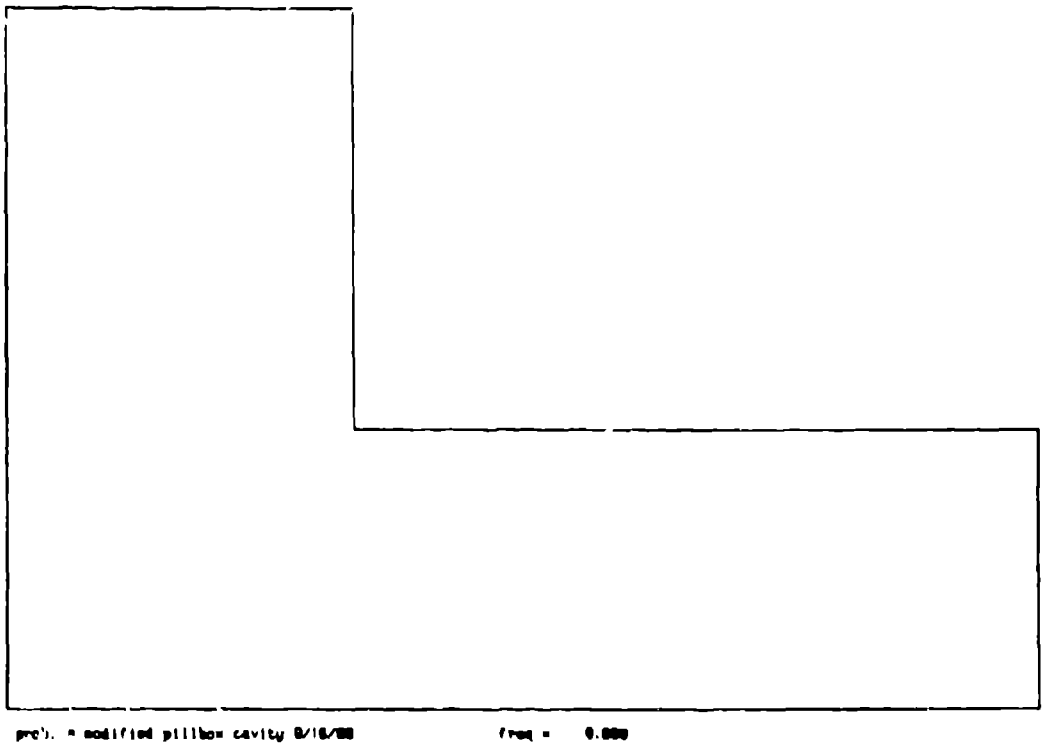


Fig. 1-7: Plot from TEKPLOT of the cavity geometry. This is a verification of the input data for the problem "modified pillbox cavity 9/16/86."

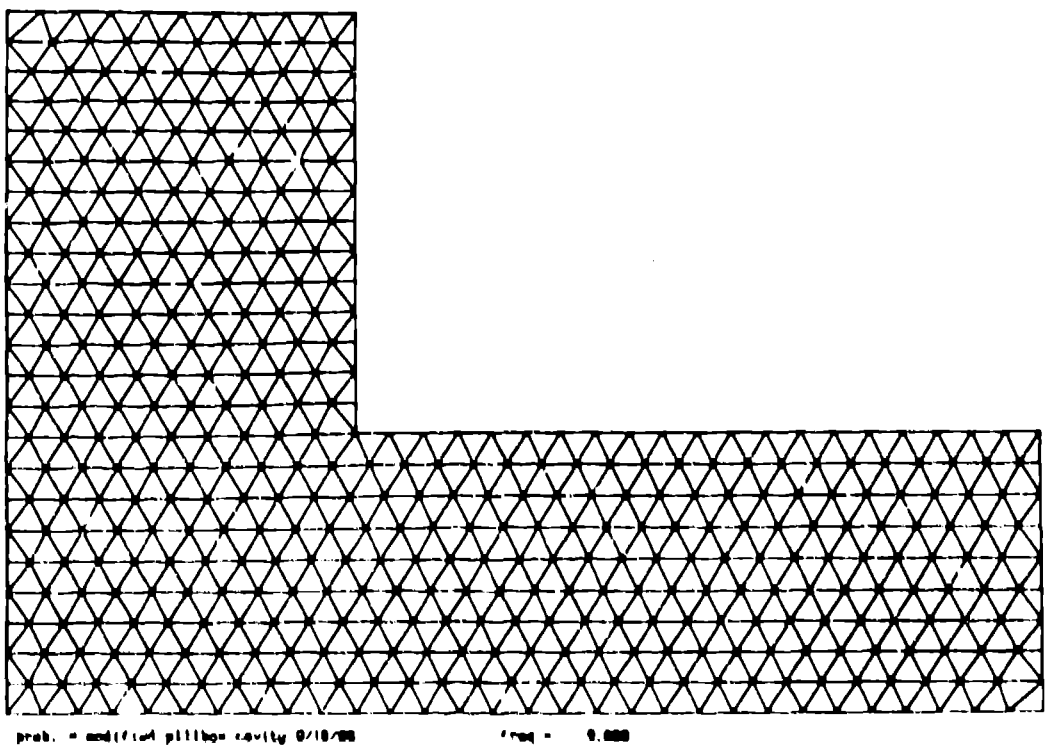


Fig. 1-8: Plot from TEKPLOT of the mesh generated by LATTICE for the problem "modified pillbox cavity 9/16/86."

1.4.4 Executing SUPERFISH

Since the mesh plot output from TEKPLOT looks good, we proceed to execute SUPERFISH (named FISH). We choose, by typing tty, to input data from the terminal. We then enter:

0	–	to read dump 0 on TAPE35 generated by LATTICE
*65 2300. s	–	to input into CON(65) the starting value for the frequency. (Any lower value would do, except the run needs more iterations to converge.)
−1	–	to end the run.

In the output reproduced below, the results of the second iteration cycle have been omitted for the sake of brevity.

```
fish
?type "tty" or input file name
tty
?type input value for dump num
? 0
beginning of superfish execution from dump number  0
prob. name = modified pillbox cavity 9/16/86
?type input values for con(?)
*65 2300. s
elapsed time  =  0.6 sec.
cycle      hmin          hmax          residual
  0  0.0000e+00  0.0000e+00    1.000e+00
     ------------------------------------------------------------
                                              k**2 = 2.3237e-01
                                              freq = 2.3000e+03
solution time =      4.519 sec.
    1  0.0000e+00 9.6312e-01    1.000e+00
      kfix =  1  ifix = 24     delta1 = 7.7380e-02     d1(k**2)= 1.9305e-02

 :
 ------------------------------------------------------------
                     the following improvement
                     using slope = -1 formula with rlx =0.500

                     del k**2 = 9.6523e-03  k**2 = 2.4202e-01
                                           freq = 2.3473e+03
solution time =      3.825 sec.
  3  0.0000e+00     1.1131e+00    1.000e+00
    kfix =  1  ifix = 24     delta1 = -4.2998e-05     d1(k**2)= -7.4418e-06
 ------------------------------------------------------------
                                 delta1(k**2)          d1(k**2)
                        1st deriv.     -5.3543e+00   -1.0815e+00
                        2nd deriv.     -2.7848e+01    6.2168e+00
```

```
                    the following improvement
                    using three point parabola formula
                    del k**2 = -7.4290e-06   k**2 = 2.4843e-01
                                             freq = 2.3782e+03
 solution converged in      3 iterations
 elapsed time = 12.9 sec.
 dump number 1 has been written.
 ?type input value for dump num
 ? -1
 stop
 fish      ctss time      14.443       seconds
 cpu=      .850     i/o=     8.756    mem=      4.836
 all done
```

In addition to the output of a binary file, dump 1 of TAPE35, SUPERFISH generates an ASCII output file, OUTFIS. All the group codes produce a similar output file; the name is formed by "OUT" followed by the first three letters of the program generating the file. That is, AUTOMESH generates OUTAUT, LATTICE generates OUTLAT, etc. Any of these ASCII files may be printed or examined with an editor.

Using dump 1 of TAPE35, the program SF01 can be executed to calculate auxiliary quantities. Chapter 7 gives details.

1.4.5 Executing TEKPLOT after SUPERFISH

After a successful execution of SUPERFISH, we execute TEKPLOT again. This time we designate dump 1 of TAPE35 as the input file and request 30 field lines, as shown below, to generate Fig. 1-9. From the field pattern shown in the figure, it is seen that we have calculated the desired TM_{010} mode.

```
tekplot
?type input data- num, itri, nphi, inap, newxy,
?1 0 30 0
input data
num= 1 itri= 0 nphi= 30 inap= 0 newxy= 0
plotting prob. name = modified pillbox cavity 9/16/86       cycle=3
?type input data xmin, xmax, ymin, ymax,
? 0
input data
xmin= 0.000 xmax= 7.500 ymin= 0.000 ymax= 5.000
?type go or no
? go
```

A CR after go clears the screen and plots Fig. 1-9. A second CR clears the screen and generates the prompt line. A negative value for the variable NUM terminates execution.

```
?type input data- num, itri, nphi, inap, newxy,
?   -1 s
tekplot ctss time 1.093 seconds
cpu= .209 i/o= .807 mem= .077
 all done
```

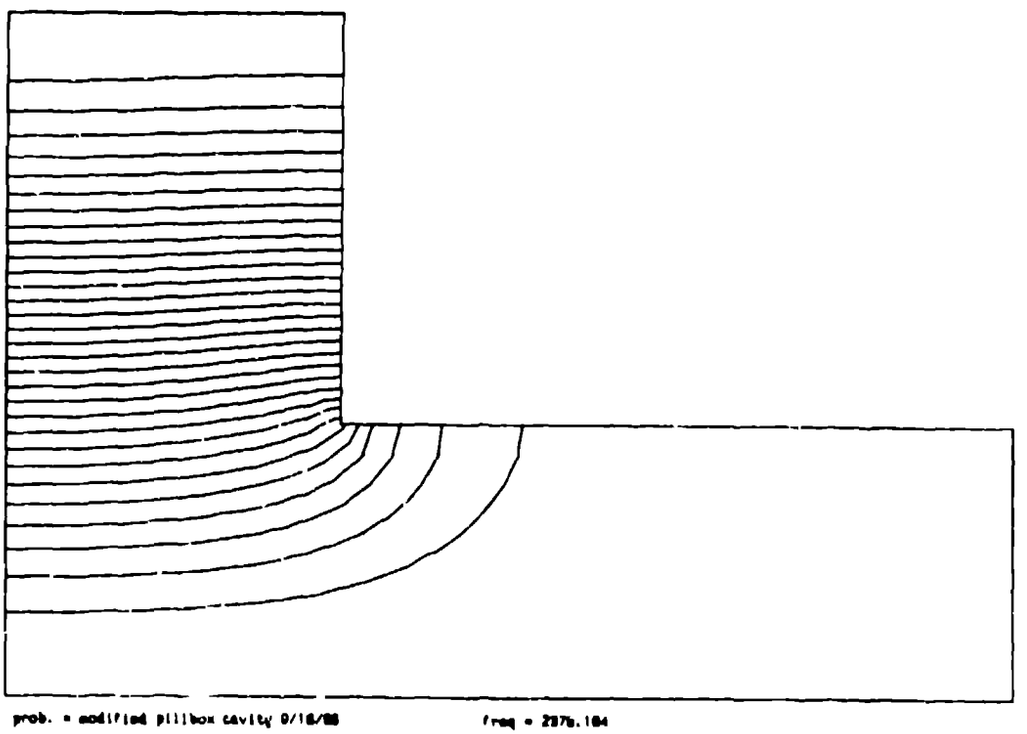


Fig. 1-9: Plot from TEKPLOT of electric field lines (rH_ϕ = const) for TM_{010} mode.

1.5 List of the Programs and Their Functions

1. AUTOMESH – prepares the input for LATTICE from geometrical data describing the problem; that is, it assigns mesh points and generates (x, y) coordinate data for straight lines, arcs of circles, and segments of hyperbolas.

2. LATTICE – generates an irregular triangular mesh (physical mesh) from input data listing the mesh points and the physical coordinates describing the problem, calculates the "point current" term at each mesh point in regions with distributed current density, and sets up mesh point relaxation order. LATTICE writes the information needed to solve the problem using the codes POISSON, PANDIRA, MIRT or SUPERFISH.

3. POISSON – solves, by "successive point over-relaxation," Poisson's (or Laplace's) equation for the vector (scalar) potential with nonlinear isotropic iron (dielectric) and electric current (charge) distributions for two-dimensional Cartesian or three-dimensional cylindrical symmetry. It calculates the derivatives of the potential, namely, the fields and their gradients, calculates the stored energy, and performs harmonic (multipole) analysis of the potential.

4. PANDIRA – Is similar to POISSON except it solves the system equations by a "direct" method, i.e., a direct solution of the block tridiagonal system of difference equations. PANDIRA allows anisotropic materials and $B(H)$ in the second quadrant (negative μ). With this program, permanent magnet and residual field problems may be solved.

5. TEKPLOT – Plots the physical geometry and meshes generated by LATTICE, and equipotential (or field lines) from the output of POISSON, PANDIRA, MIRT, (or SUPERFISH).

6. FORCE – calculates forces and torques on coils and iron regions from POISSON or PANDIRA solutions for the potential.

7. MIRT – optimizes magnet profiles, coil shapes, and current densities based on a field specification defined by the user.

8. SUPERFISH – solves for the TM and TE resonant frequencies and field distributions in an rf cavity with two-dimensional cartesian or three-dimensional cylindrical symmetry. Only the azimuthally symmetric modes are found for cylindrically symmetric cavities. The modes are found one at a time. SUPERFISH also solves for cutoff frequencies and mode patterns of TE and TM waveguide modes.

9. SFO1 – calculates auxiliary quantities useful in the design of cavities from the output of SUPERFISH. These quantities include stored energy, power dissipation on the walls and tube stems, transit time factors, shunt resistance, the quality factor Q, and the maximum electric field on the boundary.

1.6 Physical Units Used in the Programs

1.6.1 Units in AUTOMESH, LATTICE, POISSON, PANDIRA, and MIRT

charge	–	coulombs
current	–	amperes
derivatives of B	–	gauss/cm
electric field E	–	volts/cm
field H	–	oersted
force	–	amp–cm–gauss = 10^{-6} newtons
induction B	–	gauss
μ_0	–	$.4\pi$ gauss–cm/amp
length	–	centimeters or user-defined by CONV in TABLE 2–1 or by CON(9) in TABLE 3–1
scalar potential V	–	volts
stored energy	–	Joules/meter (Cartesian)
	–	Joules/radian (cylindrical)
vector potential A	–	gauss–cm

1.6.2 Constants and Units in SUPERFISH

electric field	–	volts/meter and mega-volt/meter normalized so that: $\int_0^L E_z \, dz / L =$ 1 mega-volt/meter where: E_z = electric field on axis L = length of cavity
frequency	–	megahertz
magnetic field, H_0	–	amp/meter
resistivity of Cu	–	$1.7 \cdot 10^{-6}$ ohm-cm
velocity of light, c	–	$2.997925 \cdot 10^{10}$ cm/sec

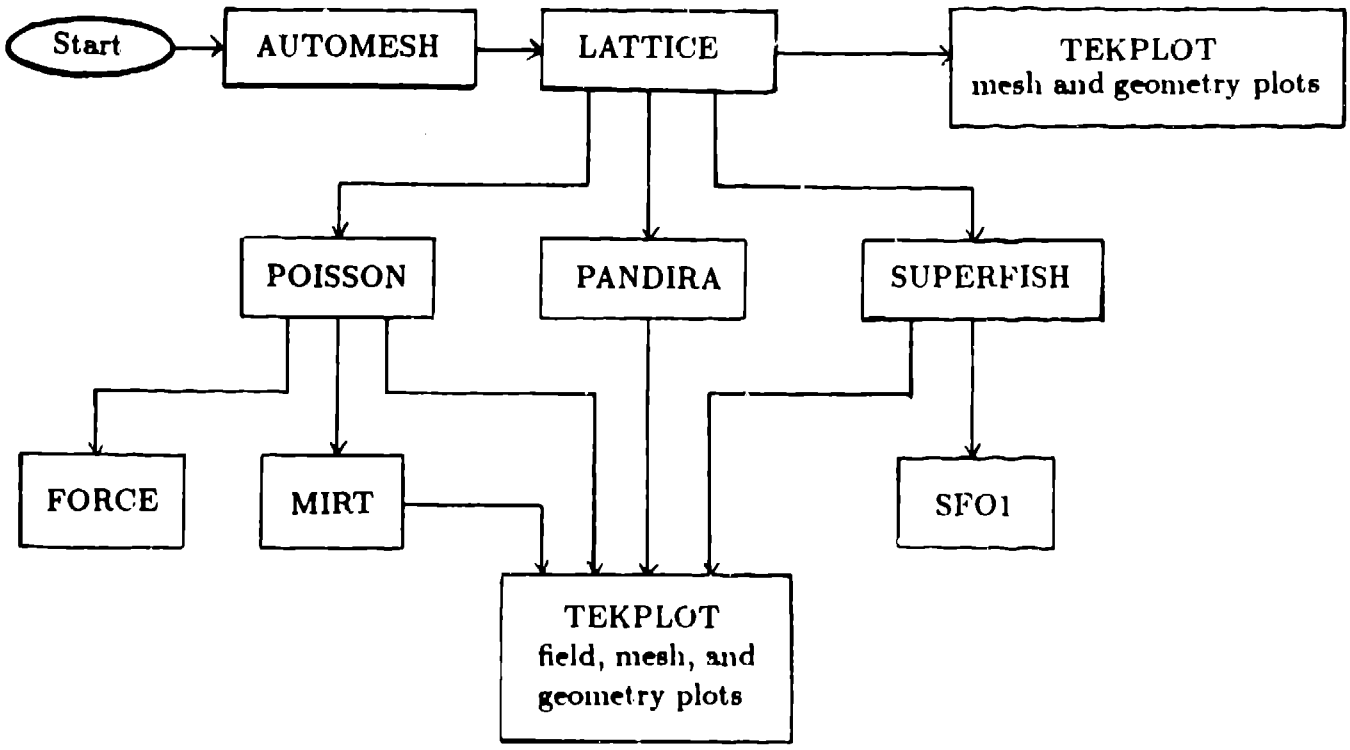


Fig. 1-10: Execution Order of the POISSON/SUPERFISH Programs

1.7 Acknowledgments

The complete POISSON/SUPERFISH Group Codes were developed over a period of 15 years, by Ronald F. Holsinger, now with Field Effects, Inc., and by Klaus Halbach, Lawrence Berkeley National Laboratory. These codes are presently maintained, updated and distributed under the supervision and direction of Richard K. Cooper of the Accelerator Theory and Simulation Group, AT-6, Los Alamos National Laboratory.

We would like to acknowledge DOE-NP and DOE-HEP for their financial support. We would like to thank and acknowledge our co-authors of the POISSON/SUPERFISH Reference Manual:

John L. Warren, Grenfell Boicourt, and Martyn Foss	Bernie Tice
AT-6, Los Alamos National Laboratory	Stanford Linear Accelerator

We would like to extend our apreciation to Pat Byrnes who, with assistance from Phyllis Encinias, did an excellent job typing this manual via LaTeX.

Finally, we would like to thank the many users who brought "coding bugs" to our attention and provided many valuable suggestions.

1.8 References

The most complete published paper describing the application of finite difference methods for the magnetostatic and electrostatic problems solved by the program POISSON and the associated nonuniform triangular mesh generated by the program LATTICE is given below.

1. Alan M. Winslow, "Numerical Solution of the Quasilinear Poisson Equation in a Nonuniform Triangular Mesh," *Journal of Computational Physics* **1** (2) 149-172 (1966).

The reference for the optimizing program MIRT is given below.

2. K. Halbach, "A Program for Inversion of System Analysis and its Application to the Design of Magnets," *Proceedings of the Second International Conference on Magnet Technology* , Oxford, England (July 10-14, 1967), p. 47.

Two published papers describe more or less completely the theory and practice of the program SUPERFISH. The references for these papers are given below.

3. K. Halbach and R.F. Holsinger, "SUPERFISH, a Computer Program for the Evaluation of RF Cavities with Cylindrical Symmetries." *Particle Accelerators* **7** (4) 213-222 (1976).

4. K. Halbach, R.F. Holsinger, W.E. Jule, and D.A. Swenson, "Properties of the Cylindrical RF Cavity Evaluation Code SUPERFISH," *Proceedings of the* 1976 *Proton Linear Accelerator Conference*, Chalk River Nuclear Laboratory, Ontario, Canada report AECL-5677 (September 14–17, 1976), pp. 122–128.

The following paper describes how to use SUPERFISH to analyze traveling wave structures.

5. G.A. Loew, *et al.*, "Computer Calculations of Traveling-Wave Periodic Structure Properties," *IEEE Transactions on Nuclear Science* NS–26 (3) 3701–3704 (1979).

Other reference material:

6. R. Holsinger, "POISSON Group Programs User's Guide," Los Alamos National Laboratory document (February 14, 1981).

7. J. Warren, *et al.*, "POISSON/SUPERFISH Reference Manual," Los Alamos National Laboratory report LA-UR-87-126 (January 1987).

Chapter 2

AUTOMESH

Contents 2-1

2.1 The Job AUTOMESH Performs

AUTOMESH and LATTICE are the mesh generating programs in the POISSON/SUPERFISH Group Codes and are executed prior to any of the other programs. AUTOMESH generates the input to LATTICE, reducing and simplifying the user's work. Basically, the input to AUTOMESH specifies the:

- problem title
- mesh size and description of the regions that define the problem
- lists of coordinates defining the boundary of each region and the type of line that connects each pair of coordinates to the prior pair.

AUTOMESH locates the path in the mesh geometry that most accurately describes the given boundary. This is the data, required by LATTICE, which AUTOMESH writes to the output file TAPE73.

2.2 Input to AUTOMESH

Most of the input data to AUTOMESH is read by the Fortran NAMELIST routine. For a full description of this routine, consult a Fortran manual.

NAMELIST, as used in AUTOMESH, is entered in the following format:

1. A blank in column 1.
2. A $ to delineate the beginning of the NAMELIST entry.
3. A name for this NAMELIST entry, either REG or PO.
4. A list, in any order, of the group variables equated to input constants and separated by commas.
5. A $ to delineate the ending of the NAMELIST entry.

An example of a REG NAMELIST entry, beginning in column 2:

```
 $ REG NREG = 2, DX = .5, XMAX = 20.,
 YMAX = 30., NPOINT = 5 $
```

The input to AUTOMESH consists of 3 data groups described in Sections:

2.2.1 Title Identification
2.2.2 REG NAMELIST (describes the regions)
2.2.3 PO NAMELIST (describes points in the region)

Data groups 2.2.2 and 2.2.3 are repeated for each region.

AUTOMESH assigns meaningful default values to all possible variables. Some of these default values are constant while others are computed by AUTOMESH from the given input data. These latter variables are designated by "AUTOMESH" in the default column of Table 2-1.

There are variables which are unique to the particular problem and must be supplied by the user. The description of these variables is preceded by "□" and "none" and appears in the default value columns of Table 2-1 and Table 2-2. The novice user may ignore all but "□" input variables.

2.2.1 Title Identification

The title is the first line of input to AUTOMESH. Column 1 specifies the type of problem:

- Blank in column 1 : POISSON/PANDIRA/MIRT problem
- Non-blank in column 1 : SUPERFISH problem

Columns 2 through 80 can have any alphanumeric characters. Starting in column 2, AUTOMESH reads by "A FORMAT" 8 computer words (columns 2–65 for a CRAY computer; columns 2–33 for a VAX computer) and uses this information only for title identification in the printed output.

2.2.2 REG NAMELIST

There are twenty-nine region entry values associated with this REG NAMELIST. Some variables are only applicable to the POISSON/PANDIRA/MIRT programs. The user only enters variables that are meaningful to his problem and allows the others to default.

There are some variables that can only be entered in the first REG NAMELIST input and not changed in subsequent REG NAMELISTs or regions. These variables are designated by ◇ before the variable name in Table 2-1.

Table 2-1 lists in alphabetical order all the variables in REG, the default values, and a brief description of their function.

TABLE 2-1
REG NAMELIST VARIABLES

	Variable	Default	Description
◇	CONV	1.0	Conversion factor for coordinate units. CONV = (number of centimeters) per (unit), e.g.: CONV = 1.0 - Centimeters. = 0.1 - Millimeters. = 2.54 - Inches.
	CUR	0.0	Magnet problems - the total current (amperes) in the region. Electrostatic problems - the fixed potential value (volts) on the boundary of the region.
		1.0	SUPERFISH - for drive point regions.
	DEN	0.0	The current density in the region. Magnet problems - amps/length^2 for area region. - amps/length for line region Electrostatic problems - coulombs/length^2 for area region. - coulombs/length for line region.
□ ◇	DX	None	The mesh increment in the horizontal direction.
◇	DY		The mesh increment in the vertical direction.
		$\frac{\sqrt{3}}{2}$ DX	For ITRI = 0 or ITRI = 1. (ITRI described below)
		DX	For ITRI = 2.
	IBOUND		The region's special boundary indicator (see Section 2.2.6).
		1	SUPERFISH problems - for all but drive point regions.
		-1	SUPERFISH problems - for drive point region.
		0	POISSON/PANDIRA/MIRT - for first region.
		1	POISSON/PANDIRA/MIRT - for all succeeding regions.
	IPRINT	0	Special diagnostic printout. IPRINT = 0 - no diagnostic printout. IPRINT ≠ 0 - mesh coordinates printout. IPRINT = 1 - special diagnostic printout in the "path-finding" subroutine, LOGIC.

	Variable	Default	Description
	IREG		The region number for this region.
		1	The first REG input data.
		+1	Incremented by 1 in each succeeding region.
◇	ITRI	0	The type of triangles for LATTICE to use in the mesh generation routine. ITRI = 0 – equal weight, equilateral triangles. ITRI = 1 – equilateral triangles. ITRI = 2 – right triangles.
◇	KMAX	AUTOMESH	The total number of mesh points in the horizontal direction from XMIN to XMAX.
◇	KREG1	0	The total number of mesh points from XMIN to XREG1. If KREG1 is given, XREG1 must be given also. Used for first mesh size change in the horizontal direction. (See Section 2.2.4).
◇	KREG2	KREG1	The total number of mesh points from XMIN to XREG2. KREG2 must be ≥ KREG1. If KREG2 is given, XREG2 must be given. Used for the second mesh size change in horizontal direction. (See Section 2.2.4).
◇	LINX	0	A special indicator for vertical line regions. LINX = 0 – Line regions are added at mesh size change (XREG1, XREG2.) LINX = 1 – No line regions are added at mesh size change (XREG1, XREG2).
◇	LINY	0	A special indicator for horizontal line regions. LINY = 0 – Line regions are added at mesh size change (YREG1,YREG2). LINY = 1 – No line regions are added at mesh size change (YREG1, YREG2).
◇	LMAX	AUTOMESH	The total number of mesh points from YMIN to YMAX.

	Variable	Default	Description
◇	LREG1	0	The total number of mesh points from YMIN to YREG1. If LREG1 is given, YREG1 must also be given. Used for first mesh size change in the vertical direction. (See Section 2.2.4).
◇	LREG2	LREG1	The total number of mesh points from YMIN to YREG2. LREG2 must be ≥ LREG1. If LREG2 is given, YREG2 must be given also. Used for second mesh size change in the vertical direction. (See Section 2.2.4).
	MAT	1	The material code for the region. MAT = 0 – All points inside the region are omitted from the problem. Points on the boundary are set according to IBOUND. MAT = 1 – Air or coil ($k_m = k_e = 1$). = 2 – Iron/dielectric properties from user-defined function or iron with internal permeability table. = 3 – Iron/dielectric properties from user-defined function or input table 1. = 4 – Iron/dielectric properties from user-defined function or input table 2. = 5 – Iron/dielectric properties from user-defined function or input table 3. = 6 – Permanent magnet material with straight line B(H) functions, PANDIRA only. . = 11
◇	NCELL	1	The number of cell cavities in SUPERFISH.
◇	NDRIVE	0	The indicator for drive point region in SUPERFISH. NDRIVE = 0 – AUTOMESH assigns a drive point region. = 1 – Drive point region input. A region with only one coordinate set (NPOINT = 1). AUTOMESH sets CUR = 1.0 and IBOUND = -1 for this region.
[]	NPOINT	None	The number of coordinate sets specifying the boundary points of the region. NPOINT = number of PO entries following this REG entry.

	Variable	Default	Description
□ ◇	NREG	None	The total number of regions for the problem. NREG = total number of REG entries.
□ ◇	XMAX	None	The maximum horizontal dimension of the problem. XMAX must be ≥ than any horizontal boundary coordinates entered with a PO NAMELIST. In Cartesian coordinates, XMAX may have a negative value.
◇	XMIN	0.0	The minimum horizontal dimension of the problem. XMIN must be ≤ than any horizontal boundary coordinates entered with a PO NAMELIST. In Cartesian coordinates, XMIN may have a negative value. For POISSON/PANDIRA/MIRT, if XMIN ≠ 0, see Table 5-1.4, CON(38).
◇	XREG1	XMAX	The location of the first mesh size change in the horizontal direction. KREG1 = 0 mesh size to the right of XREG1 will be approximately double. KREG1 ≠ 0 XREG1 must be given. Mesh size to the right will be computed as described in Section 2.2.4.
◇	XREG2	XMAX	The location of the second mesh size change in the horizontal direction. KREG2 = 0 mesh size to the right will double. KREG2 ≠ 0 XREG2 must be given. Mesh size to the right will be computed as described in Section 2.2.4.
□ ◇	YMAX	None	The maximum vertical dimension of the problem. YMAX must be ≥ than any vertical boundary coordinates entered with PO NAMELIST. In Cartesian coordinates, YMAX may have a negative value.
◇	YMIN	0.0	The minimum vertical dimension of the problem YMIN must be < than any vertical boundary coordinates entered with PO NAMELIST. In Cartesian coordinates, YMIN may have a negative value. For POISSON/PANDIRA/MIRT, if YMIN ≠ 0, see Table 5-1.4, CON(39).

	Variable	Default	Description
◇	YREG1	YMAX	The location of the first mesh size change in the vertical direction. LREG1 = 0 mesh size above YREG1 will approximately double. LREG1 ≠ 0 YREG1 must be given. Mesh size above YREG1 will be computed as described in Section 2.2.4.
◇	YREG2	YMAX	The location of the second mesh size in the vertical direction. LREG2 = 0 mesh size above YREG2 will be double. LREG2 ≠ 0 YREG2 must be given. Mesh size above YREG2 will be computed as described in Section 2.2.4.

where:

- □ Quantities the user must enter.
- ◇ Quantities that can only be entered in the first REG input and cannot be changed in subsequent REG input.

2.2.3 PO NAMELIST

The variables for the PO NAMELIST entry specify the boundary points of the present region. The first PO data set specifies the initial point. Each succeeding set specifies a new boundary point and the type of curve to be drawn from the previous point. The last PO entry must describe the initial point in order to have a closed region. The number of PO entries must equal NPOINT, a REG NAMELIST variable.

The boundary points of the first region must encompass the complete geometry of the problem. Subsequent regions define sections of this geometry. The properties of each new region override the previously defined properties.

AUTOMESH can draw three different curves—straight line, circular arc, and hyperbolic segment—between any two points. The points of a straight line or circular arc may be given in either Cartesian coordinates, (X', Y') polar coordinates, (R', θ').

The prime coordinates correspond to the coordinate axis (X', Y'), which has been displaced by $(X0, Y0)$ from the "standard" coordinate axis (X, Y), see Fig. 2-1 and Fig. 2-2 for an example.

That is:

$$X' = X - X0 \qquad R' = \sqrt{X'^2 + Y'^2}$$

$$Y' = Y - Y0 \qquad \theta' = \tan^{-1} \frac{Y'}{X'}$$

If $(X0, Y0) = (0., 0.)$, then $(X', Y',) \equiv (X, Y)$

When values are assigned to X0 and Y0 in a PO NAMELIST statement, accompanying values of X and Y or R and θ are taken to be relative to this shifted origin.

Circular Arcs

Circular arcs are defined by

$$(X - X0)^2 + (Y - Y0)^2 = R'^2 \equiv X'^2 + Y'^2$$

where $(X0, Y0)$ is the center and R' is the radius of the circle.

The two connecting points given on the $PO entry must satisfy this equation to a relative error of 10^{-3}.

Hyperbolic Segments

AUTOMESH defines hyperbolic branches only in the first quadrant and symmetric about the line $Y = X$ in the (X, Y) coordinate system. The hyperbolic curve is drawn from the previous point to the present point and both points must satisfy, to a relative error of 10^{-3}, the equation

$$2 * X * Y = R^2$$

where R is the minimum distance from the origin, (0,0), to the hyperbolic branch.

A PO entry for a hyperbolic segment can only define the Cartesian coordinate (X, Y) and must specify the value for R.

Hyperbolic arcs are largely used in defining quadrupole magnets.

Cylindrical Coordinates

Cylindrical coordinates (r, z) are entered as the X, Y variables in the NAMELIST group, where:

- $(z \rightarrow X,\ r \rightarrow Y)$ for SUPERFISH problem
- $(r \rightarrow X,\ z \rightarrow Y)$ for POISSON/PANDIRA/MIRT problem

All cylindrical coordinates must have positive values.

Table 2–2 lists the eight PO NAMELIST entries.

TABLE 2-2
PO NAMELIST VARIABLES

	Variable	Default	Description
	New	0	Used to force separation between regions. NEW = 0 – The points on the path of this segment may coincide with points on the path of any previous region. NEW = 1 – The points on the path of this segment are NOT allowed to coincide with points on the path of any previous region. NEW = −1 – The points on the path of this segment are NOT allowed to coincide with points on the path of any previous region, EXCEPT for the starting and end points.
	NT	1	The type of curve to be drawn from previous point to this point. NT = 1 – A straight line. NT = 2 – An arc of a circle with a center at $(X0, Y0)$, radius of R', and defined by: $X'^2 + Y'^2 = R'^2$. NT = 3 – A hyperbolic curve in the first quadrant, symmetric about the line Y = X and defined by $2 * X * Y = R$.
□	R	None	NT = 1 – R' – The radial polar coordinate. NT = 2 – R' – The radius of circle with center at $(X0, Y0)$ (polar coordinates). NT = 3 – R – The minimum distance from the origin to the hyperbolic branch in the first quadrant (Cartesian coordinates). R MUST be entered.
□	THETA	None	NT = 1 – The θ' value of the polar coordinates given in NT = 2 degrees and relative to the X-axis in a counter-clockwise direction. NT = 3 – Not used.

	Variable	Default	Description
□	X	None	The input coordinates. Cartesian coordinates may have positive
□	Y		or negative values. In cylindrical coordinates, both r and z must be ≥ 0.
			NT = 1 - The X', Y' values of the Cartesian coordinate system, (X', Y')
			NT = 2 or r, z values for cylindrical coordinates. When $(X0, Y0) \neq 0$, X, Y are defined RELATIVE to the shifted origin.
			NT = 3 - The X, Y values of the Cartesian coordinate system, (X, Y). R must also be entered.
	X0	None	NT = 1 - The displacement of the (X', Y') axes.
	Y0		NT = 2 - Center of circle.
			NT = 3 - Not used.

where:

□ - Quantities the user must enter for:

NT = 1, (X', Y') or $(R', THETA')$.
NT = 2, (X', Y') or $(R', THETA')$.
NT = 3, (X, Y) and R.

The following two examples illustrate alternate ways of specifying region points.

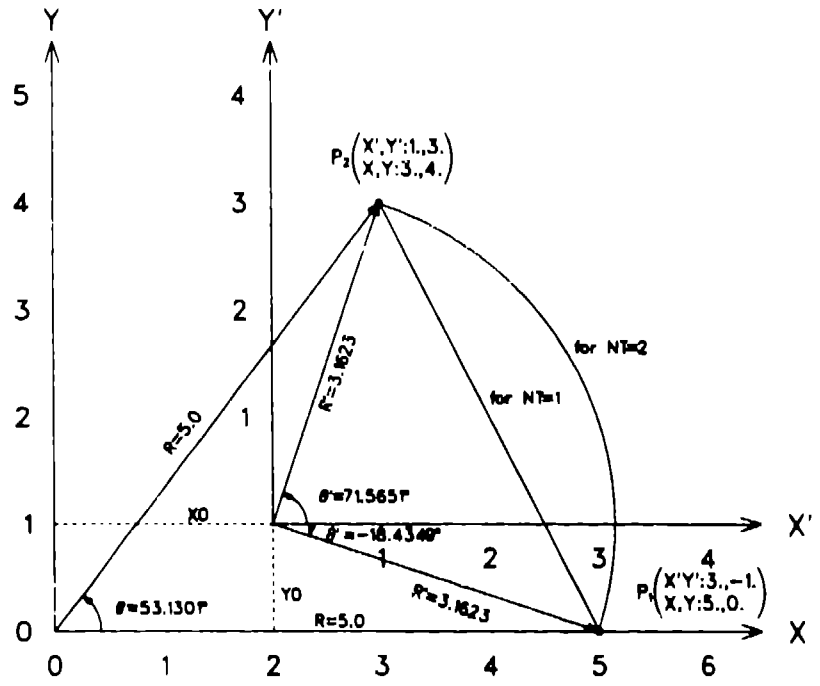


Fig. 2-1. AUTOMESH coordinate system for straight lines and circular arc curves.
Example A -- $ PO entry for P_1 to P_2

EXAMPLE A

The two points, P_1 (5., 0.) and P_2 (3., 4.) of (X,Y) coordinates of Fig. 2-1 may be entered in ANY of the following PO specifications:

For initial point P_1(5.,0.) :

```
$PO X = 5., Y = 0.$
$PO X = 3., Y = -1., X0 = 2., Y0 = 1.$
$PO R = 5., THETA = 0.$
$PO R = 3.1623, THETA = -18.4347, X0 = 2., Y0 = 1.$
```

To point P_2 (3.,4.):

```
$PO X = 3., Y = 4., NT = n$
$PO X = 1., Y = 3., X0 = 2., Y0 = 1., NT = n$
$PO R = 5., THETA = 53.1301, NT = n$
$PO R = 3.1623, THETA = 71.5651, X0 = 2., Y0 = 1., NT = n$
```

where: NT = n, n = 1 straight line curve from P_1 to P_2, (default value).
n = 2 circular arc with center at (X0, Y0) and radius R'.

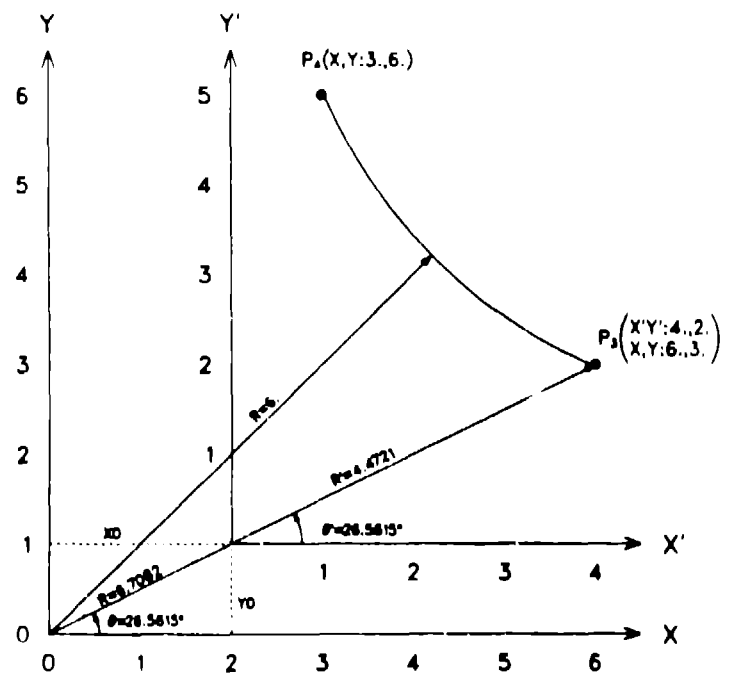


Fig. 2-2. AUTOMESH coordinate system for hyperbolic curves.
Example B - $ PO entry for P_3 to P_4

EXAMPLE B

In Fig. 2-1, any initial point, P_3 (6., 3.), of a hyperbolic segment can be entered in any of the following PO specifications:

```
$PO X = 6., Y = 3. $
$PO X = 4., Y = 2., X0 = 2., Y0 = 1. $
$PO R = 6.7082, THETA = 26.5651 $
$PO R = 4.4721, THETA = 26.5651, X0 = 2., Y0 = 1. $
```

The end point, P_4 (3.,6.), of the hyperbolic segment can only be entered as:

```
$ PO X = 3., Y = 6., R = 6., NT = 3 $
```

2.2.4 Mesh Size Options

AUTOMESH sets the groundwork for LATTICE to generate a triangular mesh of uniform size, varied size, or a combination of uniform and varied mesh sizes.

The user selects the mesh type that is appropriate to the geometry of the problem, keeping in mind that AUTOMESH assigns a new mesh point to two given points, say X_1 and X_2, only if the distance between them is greater than one-half of the increment DX. That is:

If $|X_2 - X_1| > DX/2.$, then X_1 and X_2 are assigned a different mesh point.

For example,

Given coordinates (2., 4.), (2.3, 4.4), and DX = .5, DY = 1.,
X_1 and X_2 will have different mesh points, since $|X_2 - X_1| = .3 > .5/2.$
Y_1 and Y_2 will have the same mesh point, since $| Y_2 - Y_1 | = .4 < 1./2.$

The types of mesh sizes and their corresponding variables that are entered on the first REG input are described below.

Uniform Mesh (Default)

Used: for simple geometries that do not exceed the maximum number of mesh points defined by the program.
Total number of mesh points = (KMAX + 2) * (LMAX + 2).

Input: user must enter DX; DY entry is optional as it has a default value.

Mesh: will generate a uniform triangular mesh of length DX and height DY.

Example: $ REG DX = .5, DY = .5, XMAX = 20., YMAX = 10., ... $
generates uniform triangular mesh of length and height = .5,
see Fig. 2-3(a).

Doubling Mesh

Used: if a fine mesh is desired near the origin and a coarser mesh further away will suffice.

Input: user must enter DX.

Enter one or a combination of XREG1, XREG2, YREG1, YREG2 as long as XREG2 ≥ XREG1 and YREG2 ≥ YREG1.

Mesh: Width of triangular mesh is approximately doubled at XREG1 and again at XREG2. If LINX = 0 (default), vertical line regions are added at XREG1 and XREG2.

Height of triangular mesh is approximately double above YREG1 and again above YREG2. If LINY = 0 (default), horizontal line regions are added at YREG1 and YREG2.

Example $REG DX = .5, DY = .5, XMAX = 20., YMAX = 10., XREG1 = 8., XREG2 = 12., YREG1 = 7., ... $
generates triangular mesh of size:

width =	DX	= 0.5	for	0.0	≤	X	<	8.0
=	2.*DX	= 1.0	for	8.0	≤	X	<	12.0
=	4.*DX	= 2.0	for	12.0	<	X	<	20.0
=	DY	= 0.5	for	0.0	<	Y	<	7.0
height =	2.*DY	= 1.0	for	7.0	<	Y	<	10.0

Vertical line regions added at X = 8.0 and X = 12.0, horizontal line regions added at Y = 7.0, see Fig. 3.2(b) for an example

Variable Mesh

Used: to explicitly define, up to a maximum of three different width and height mesh sizes.

Input: enter any set or combination of sets:
(KREG1, XREG1), (KREG2, XREG2), (KMAX, XMAX), (LREG1, YREG1), LREG2, YREG2), (LMAX, YMAX).

Mesh: generates triangular mesh with

$$\text{width} = \frac{XREG1 - XMIN}{KREG1 - 1} \qquad XMIN \leq X < XREG1$$

$$= \frac{XREG2 - XREG1}{KREG2 - KREG1} \qquad XREG1 \leq X < XREG2$$

$$= \frac{XMAX - XREG2}{KMAX - KREG2} \qquad XREG2 \leq X \leq XMAX$$

height = similar as above but corresponding values (LREG1, YREG1), (LREG2, YREG2), (LMAX, YMAX) are used.

If LINX = 0 (default), vertical line regions are added at XREG1 and XREG2.
If LINY = 0 (default), horizontal line regions are added at YREG1 and YREG2.

Example: $REG DX = .5, DY = .5, XMAX = 20., YMAX = 10., XREG1 = 8., XREG2 = 12., YREG1 = 7., KREG1 = 17, KREG2 = 29, KMAX = 37, ...$

generates triangular mesh of

$$\text{width} = DX = \frac{8.0 - 0.0}{17 - 1} = 0.5, \qquad 0.0 \leq X < 8.0$$

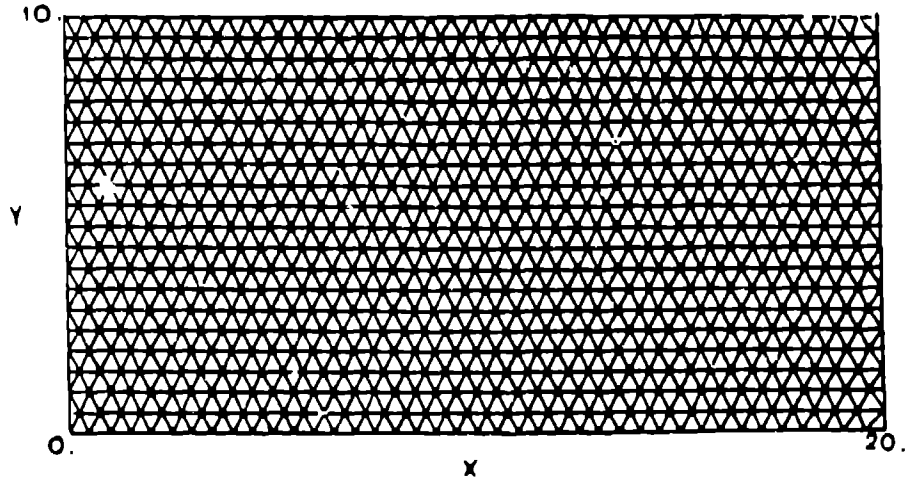
$$= \frac{12.0 - 8.0}{29 - 17} = 0.333\ldots \qquad 8.0 \leq X < 12.0$$

$$= \frac{20.0 - 12.0}{37 - 29} = 1.0 \qquad 12.0 \leq X < 20.0$$

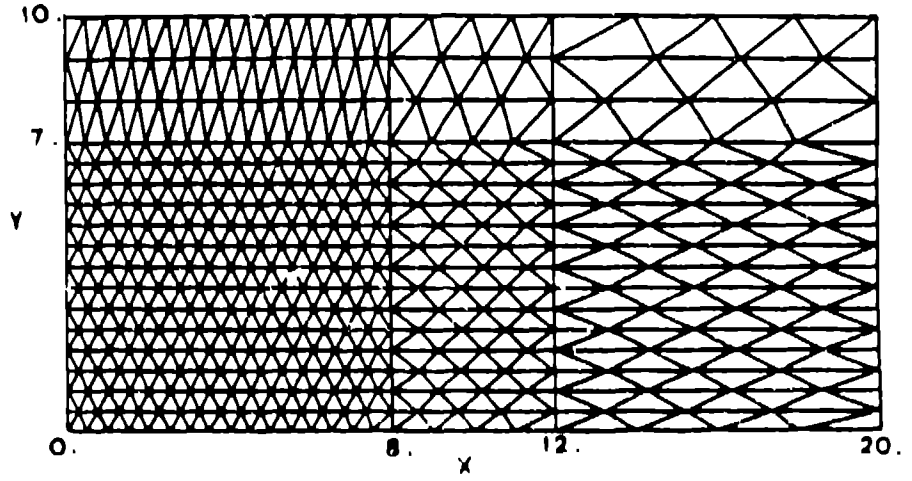
$$\text{height} = DY = 0.5 \qquad 0.0 \leq Y < 7.0$$
$$= 2 * DY = 1.0 \qquad 7.0 \leq Y < 10.0$$

vertical line regions added at X = 8.0 and X = 12.0;
horizontal line region added at Y = 7.0.

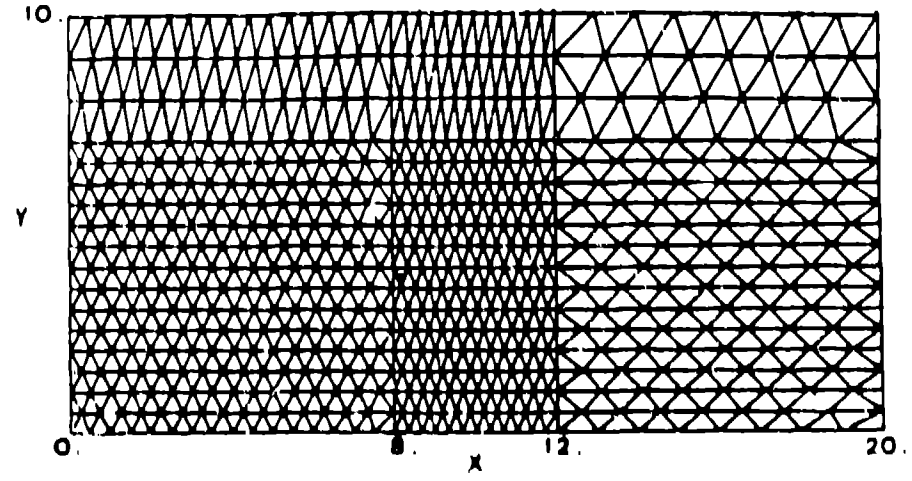
This example is a combination of explicitly defined mesh length and of doubling mesh width, see Fig. 2 3(c).



(a) $REG DX = .5, DY = .5, XMAX = 20., YMAX = 10., ...$



(b) $REG DX = .5, DY = .5, XMAX = 20., YMAX = 10., XREG1 = 8., XREG2 = 12., YREG1 = 7., ..$



(c) $REG DX = .5, DY = .5, XMAX = 20., YMAX = 10., XREG1 = 8., XREG2 = 12., YREG1 = 7., KREG1 = 17, KREG2 = 29, KMAX = 37,...$

Fig. 2-3. Variable mesh corresponding to REG data in AUTOMESH.

2.2.5 Boundary Condition Options

AUTOMESH automatically assigns indicators for the boundary conditions to the upper, lower, right, and left boundaries, defining the rectangular region of the problem, as follows:

	SUPERFISH	POISSON/PANDIRA/MIRT
upper boundary:	1	0
lower boundary:	0	1
right boundary:	1	0
left boundary :	1	0

where:

0 - Indicates Dirichlet boundary conditions, which means electric (for SUPERFISH TM modes) or magnetic (for POISSON/PANDIRA/MIRT) field lines are PARALLEL to the boundary line.

1 - Indicates Neumann boundary conditions, which means electric (for SUPERFISH TM modes) or magnetic (for POISSON/PANDIRA/MIRT) field lines are PERPENDICULAR to the boundary line.

If the above boundaries are inappropriate to the problem, the user is given the option to change them in LATTICE when the code asks for CON changes.

In addition to the above boundary options, AUTOMESH has a special boundary indicator, IBOUND, which may be entered or defaulted in REG NAMELIST.

This parameter is used for three purposes:

1. To indicate the boundary conditions on a boundary which does not coincide with the extreme rectangular boundary of the problem. The boundary of this area or line region is set accordingly:

   IBOUND = 0 - Dirichlet boundary (defined above)
   = 1 - Neumann boundary (defined above)

2. To indicate special fixed potential, IBOUND = -1
   The special fixed potential value is input as the CUR variable in the REG NAMELIST.
   This is the mechanism for setting up electrostatic (scalar potential) problems.
   Scalar potential problems are indicated by setting XJFACT = CON(66) = 0.0
   (See Chapter 5, Table 5-1.5)

3. To indicate a drive point region in SUPERFISH, IBOUND = -1

2.3 Output from AUTOMESH

Automesh generates two output files – TAPE73 and OUTAUT – and, if run interactively, prints output messages at the terminal.

In a successful run, AUTOMESH outputs:

1. To OUTAUT file and to the terminal—
   - no error messages
   - the message: `REGION (--)/OK` for each region, where (--) is the region number supplied by the program.
2. To TAPE73 file — the complete input data to LATTICE.

In this case, the user does not have to be concerned any further with the contents of the output files.

However, if the user has been unsuccessful in correcting the listed errors of an aborted run, some insight might be gained by examining these two ASCII files. These files can be examined with any editor.

2.4 Diagnostic and Error MESSAGES

The error messages from AUTOMESH briefly define the problem and give a possible solution. An explanation of the common messages from AUTOMESH are listed below.

1. `MESH SIZE CHANGE` — usually the mesh is too coarse; user should rerun the problem with a finer mesh; sometimes a slight mesh size change will suffice.

2. `(X1,Y1)/(R1, THETA1)` — the Cartesian/polar coordinates of the previous point (from).
   `(X2,Y2)/(R2, THETA2)` — the Cartesian/polar coordinates of the present point (to).

3. `R` — printed as the value of a variable, means that this variable has been set out of range and not supplied by the user.

4. `REGION (--)/O.K.` — AUTOMESH has successfully found paths for all boundary points in this, (--), region; if errors occur in one region, AUTOMESH proceeds to the next.
   (--) means computer prints out the value.

The first line of all AUTOMESH error messages and explanations are listed in sections 2.4.1, 2.4.2, and 2.4.3.

2.4.1 Error Messages

Messages Containing "ERROR"

1. `--- ERROR --- DATA FOR THIS CIRCLE FROM (X1,Y1)/ (R1, THETA1) TO (X2,Y2)/R2, THETA2) IS INCONSISTENT . . .`
   Either one or both coordinates are not given or the two points with center at $(X0, Y0)$ do not lie on the same circle to a relative accuracy of 10^{-3}. Correct the input data for the listed coordinates. The user should check that the coordinates are given RELATIVE to $(X0, Y0)$. Message from subroutine DATUPS.

2. `--- ERROR --- DATA FOR THIS LINE ARE INSUFFICIENT . . .`
   Either one or both coordinates are not given. Correct the input data for the listed coordinates. Message from subroutine DATUPS.

3. `--- ERROR --- DATA FOR THIS HYPERBOLA FROM (X1,Y1) TO (X2,Y2) IS INCONSISTENT . .`
   Either one or both coordinates are not given, R is not given, or the two points do not lie on the same hyperbolic branch to a relative accuracy of 10^{-3}. Correct input. Message from subroutine DATUPS.

4. --- ERROR --- X/Y IS OUT XMIN, XMAX/YMIN, YMAX LIMITS . . .
The X or Y point printed is less or greater than the given minimum or maximum value for X/Y in the first REG input line. Correct input. Message from DATUPS.

5. --- ERROR --- (KMAX + 2) * (LMAX + 2) = (--) IS GREATER THAN PROGRAM DIMENSIONS OF (--) . . .
The total number of mesh points have exceeded the maximum value dimensioned. Cut mesh size or increase parameter MXDIM and recompile as directed. Message from subroutine SETXY.

6. --- ERROR --- TROUBLE IN FINDING THE PATH OF A POINT . . .
AUTOMESH encountered trouble in both "forward" and or "backward" pass in subroutine LOGIC. To correct, decrease mesh size near the point and try again. Message from main program.

2.4.2 Trouble Messages

Messages Containing "TROUBLE"

1. --- TROUBLE --- DIMENSIONS FOR THE NSEG ARRAYS, EXCEEDED NSG OF (--) . . .
AUTOMESH has exceeded the maximum number of boundary segments dimensioned in the program. Increase parameter NSG and recompile as directed. Used in SUPERFISH problems. Message from subroutine FISHEG.

2. --- TROUBLE --- NPOINT = (--), EXCEEDS DIMENSION OF (--)
The number of PO entries for this region has exceeded the maximum number dimensiond. To correct, increase parameter NPTX and recompile as directed. Message from main program or subroutine INSERT.

3. --- TROUBLE --- THE PROGRAM FOUND THE SAME (K,L) COORDINATES FOR THE FIRST AND LAST POINT OF THIS CURVE . . .
The program has assigned the same mesh point in either vertical or horizontal direction for (X_1, Y_1) and (X_2, Y_2). This usually means mesh size is not fine enough. (See Section 2.2.4.)

3a. Message is printed from subroutine LOGIC. The last [illegible] of the message prints "FORWARD PASS" or "BACKWORD PASS." AUTOMESH executes subroutine LOGIC twice--first in a "forward" search, and a second pass in a "backward" search--to find the path of the current segment. Then the program chooses the path with the smaller number of segments with no errors. A fatal error occurs if BOTH directions encounter "TROUBLE." To correct, decrease mesh size near this point and rerun.

4. --- TROUBLE --- PROGRAM DIMENSIONS 1000 FOR THE KL ARRAYS ARE INSUFFICIENT
   The program has difficulty in finding the path for this segment and thus has exceeded the dimension allocated for storage of the path array. See 3a. above.

5. --- TROUBLE --- LOGICAL PATH IS TRAPPED AT K = (--), L = (--)
   The program cannot find the path for this current segment. See 3a above.

6. --- TROUBLE --- CANNOT FIND A FIXED H-PHI POINT
   The program has problems assigning the default drive point at the upper left-hand corner of the cavity. To correct, input own drive point region by setting NDRIVE = 1 in the first REG entry and then input a region with one point—the drive point, (see the example in Sec. 1.3.1)

7. --- TROUBLE --- TOO MANY END POINTS FOUND FOR THE LINE
   The program has trouble adding a vertical/horizontal line region.

   7a. AUTOMESH could encounter a number of problems in subroutines XLINER/YLINER while attempting to add vertical/horizontal line regions. To correct, change mesh size or set LINX/LINY = 1 in the first REG entry. (This latter option deletes the addition of all vertical/horizontal line regions at horizontal/vertical mesh change locations.)

8. --- TROUBLE --- NO END POINTS FOUND FOR LINE
   The program has trouble finding a mesh point for the end point of the added line region. See 7a. above.

9. --- TROUBLE --- ONLY ONE END POINT FOR THE LINE
   The program has trouble finding an end point for this added line region. See 7a. above.

10. --- TROUBLE --- A POINT WITH (K = KREG) HAS X NOT = TO XREG
    --- TROUBLE --- A POINT WITH (L = LREG) HAS Y NOT = TO YREG
    The program has difficulty adding a vertical/horizontal line region. See 7a. above.

2.4.3 Additional Diagnostic Messages

1. DIMENSION OF 2000 FOR KR, LR . . .
   The program has run into difficulty and has exceeded the maximum number of points dimensioned for a region. Change mesh size and try again. Message from subroutine LOGSEG.

2. DIMENSION OF 3000 INSUFFICIENT FOR KG, LG . . .
   The program has run into difficulty and has exceeded the total number of points dimensioned for all regions. Change mesh size and try again. Message from subroutine SAVAGE.

2.5 Examples of AUTOMESH Runs

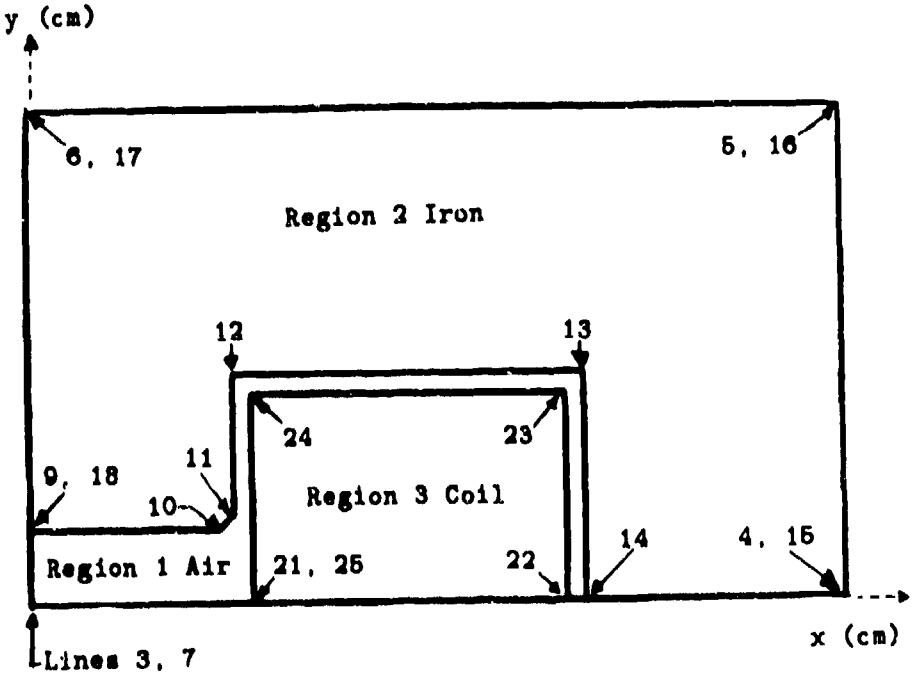
Sections 2.5.1 and 2.5.2 list two input files, HMAG and DTL, a POISSON and a SUPERFISH input respectively. These files utilize a number of various options available in the AUTOMESH input format.

The line numbers listed on the left are not part of the input files, but are used as pointers for the explanation that follows. The terminal output of the execution of AUTOMESH on a CRAY computer is also given. (The input file for a VAX/VMS computer is identical except all input is upper case.)

2.5.1 AUTOMESH Run — H-Shaped Dipole Magnet

The file HMAG describes the cross section of one-fourth of an H-shaped dipole magnet, the upper-right quadrant.

```
Line
No.
 1   h-magnet test, uniform mesh  4/23/85
 2   $reg nreg=3,dx=.45,xmax=22.,ymax=13.,npoint=5$
 3   $po x= 0.0, y= 0.0$
 4   $po x=22.0, y= 0.0$
 5   $po x=22.0, y=13.0$
 6   $po x= 0.0, y=13.0$
 7   $po x= 0.0, y= 0.0$
 8   $reg mat=2,npoint=10$
 9   $po x= 0.0, y= 2.0$
10   $po x= 5.1, y= 2.0$
11   $po x= 5.5, y= 2.4$
12   $po x= 5.5, y= 6.0$
13   $po x=15.0, y= 6.0$
14   $po x=15.0, y= 0.0$
15   $po x=22.0, y= 0.0$
16   $po x=22.0, y=13.0$
17   $po x= 0.0, y=13.0$
18   $po x= 0.0, y= 2.0$
19   $reg mat=1,npoint=5,
20   cur=-25455.791$
21   $po x= 6.0, y= 0.$
22   $po x=14.5, y= 0.0$
23   $po x=14.5, y= 5.5$
24   $po x= 6.0, y= 5.5$
25   $po x= 6.0, y= 0.0$
```



Line
No.

Line 1: Title line, which starts in column 2 (POISSON/PANDIRA/MIRT Problem).

2: First REG entry: nreg = 3 – number of regions.
dx = .45 – horizontal mesh size.
xmax, ymax = 22., 13. – problem's maximum dimensions.
npoint = 5 – number of PO entries that follow.

3–6: Coordinates of points that define the region 1.

7: Same coordinates as line 3 to form a closed region.

8: REG entry for region 2: mat = 2 – implies that the material is iron.

9–18: Coordinates of points that define region 2.

19–20: REG entry for region 3: mat = 1.
cur = –25455.791 – total number of amps. (The sign specifies direction of the current; negative "into the plane of the paper"). These two entries imply this is a coil region.

21–25: Coordinates of points that define region 3.

Below is a listing of the interactive execution of AUTOMESH on a CRAY computer with input file HMAG. (User only types underlined quantities.)

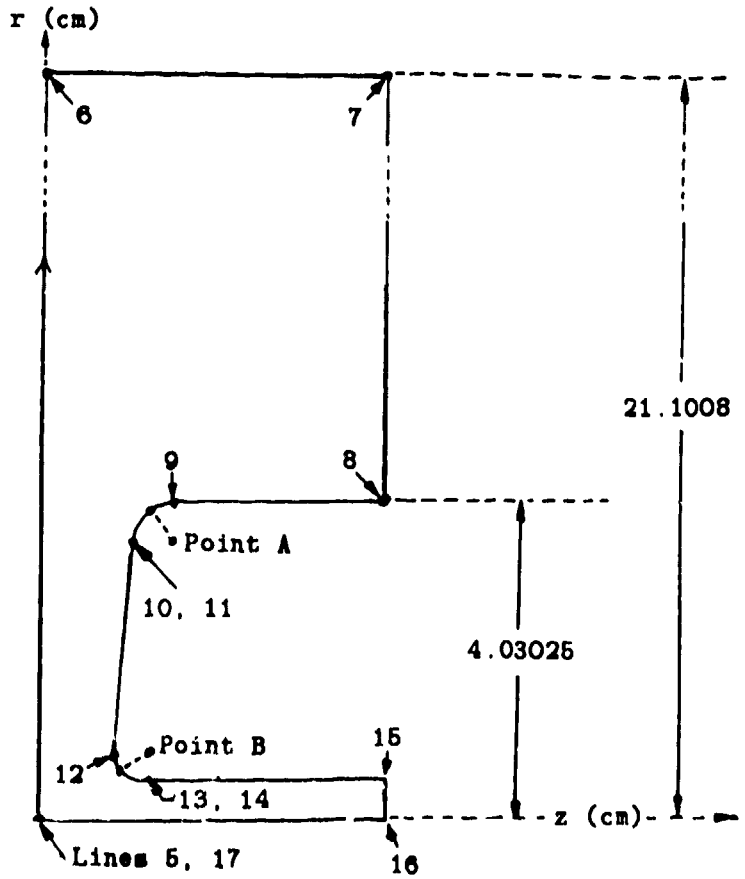
```
 ?type input file name
? hmag
 region no.  1
 ok

 region no.  2
 ok

 region no.  3
 ok
 stop
automesh ctss time        .539     seconds
cpu=      .119   i/o=       .366   mem=       .054

 all done
```

2.5.2 AUTOMESH Run — Drift-Tube Linac Cell

SUPERFISH EXAMPLE: The file DTL describes the cross section of one-fourth of a drift-tube linac cell. (The numbered points on the graph correspond to the input file lines.)

Line No.

```
 1  1superfish dtl test problem
 2   $reg nreg=1,dx=0.25,xmax=4.2843,
 3   ymax=21.1008,yreg1=5.,yreg2=7.,
 4   npoint=11$
 5   $po  x=0.0      , y= 0.0      $
 6   $po  x=0.0      , y=21.1008   $
 7   $po  x=4.2843   , y=21.1008   $
 8   $po  x=4.2843   , y= 4.03025  $
 9   $po  x=1.60454  , y= 4.03025  $
10   $po nt=2, r=0.5   , theta=176.5,
11       x0=1.60454  , y0= 3.53025 $
12   $po  x=0.93936  , y= 0.84484  $
13   $po nt=2, r=0.325, theta=370.0,
14       x0=1.26375  , y0= 0.82b   $
15   $po  x=4.2843   , y= 0.5      $
16   $po  x=4.2843   , y= 0.0      $
17   $po  x=0.0      , y= 0.0      $
```



Line No.

1: This is the title line with a non-blank in column one.

2–4: First REG entry:

nreg	=	1 - number of regions
dx	=	.25 - horizontal mesh size
xmax, ymax	=	4.2843, 21.1008 - problem's maximum dimensions
yreg1, yreg2	=	5., 7. - location at which mesh size doubles
npoint	=	11 - number of PO entries that follow.

5–9: Lists the coordinates, (x corresponds to z; y to r), connected by straight lines.

10–11: Specifies a circular arc with center at Point A (1.60454, 3.53025) and radius (.5). This arc is drawn from the point described by line 9 to the point described by lines 10, 11 (176.5 counterclockwise) and is in polar form.

12: Straight line from the point described by lines 10, 11 to the point described by line 12.

13–14: A circular arc, similar to lines 10–11, with center at Point B.

15–16: Same as lines 5–9.

17: Lists the last coordinate, same as first, to close the region.

Below is a listing of the interactive execution of AUTOMESH on a CRAY computer (user only types underlined quantities).

```
 ?type input file name
? dtl
 region no.  1
 logical boundary segment end points
 iseg   kb   lb       kd   ld       ke   le
    1    1    1        0    1        1   24
    2    1   24        0    1        1   29
    3    1   29        0    1        1   47
    4    1   47        1    0       18   47
    5   18   47        0   -1       18   29
    6   18   29        0   -1       18   24
    7   18   24        0   -1       18   20
    8   18   20       -1    0        7   19
    9    7   19       -1    0        5   17
   10    5   17        1   -1        4    5
   11    4    5        1   -1        6    3
   12    6    3        1    0       18    3
   13   18    3        0   -1       18    1
   14   18    1       -1    0        1    1
 stop
automesh ctss time       .620     seconds
cpu=      .083   i/o=      .443   mem=          .094
 all done
```

Chapter 3

LATTICE

Contents 3-1

3.1 The Job Lattice Performs

In general, LATTICE reads in input data, as it is created by AUTOMESH, completes the mesh generation process, as well as computes and writes a binary file of all information needed for the execution of one of the "equation solving" programs (*i.e.*, POISSON, PANDIRA, MIRT, or SUPERFISH.)

Specifically, LATTICE performs the following tasks:

1. Reads in the properties describing the specific problem, the physical boundary coordinates specifying the geometry, and the corresponding mesh point numbering of the coordinates, for the generation of a regular triangular mesh. See Fig. 3-1(a).
2. Sets up mesh point relaxation order.
3. Distorts the sides of the triangles in the regular triangular mesh by iterating and "relaxing" until the sides of the triangles coincide with the given physical boundaries. See Fig. 3-1(b).
4. Assigns mesh point numbers to all interior coordinates.
5. Identifies and sets up all boundary points for the input boundary conditions.
6. Calculates point current terms in regions with distributed current density.
7. Writes dump 0 to a binary file, TAPE35, which contains all the information needed for the execution of any of the programs: POISSON, PANDIRA, MIRT, SUPERFISH, and TEKPLOT.

3.2 The Input to LATTICE

All POISSON programs, except AUTOMESH, utilize an internal, format-free input subroutine, FREE, to enter input. The FREE format uses special characters to shorten input and save array space. A FREE entry requires all floating point numbers to have a decimal point. Section 3.2.2 outlines in detail all the features of this free-format input and gives an example illustrating these features. However, the beginner or casual user does not need to know all these options since this manual will give and explain the individual format entry needed for the specific input.

It is assumed that the input to LATTICE is TAPE73, which has been generated by AUTOMESH. Therefore, the user normally need not be concerned about the contents or format of the input file. He only needs to examine Section 3.2.1 which describes the various options and changes that can be input to LATTICE through the CON array.

However, in case error messages are received in the execution of either AUTOMESH or LATTICE, it is helpful to examine TAPE73. Section 3.2.3 briefly explains the contents and format of TAPE73.

A user, who wishes (or is forced) to generate his own input to LATTICE because of mesh point limitations, should consult the POISSON/SUPERFISH Reference Manual in Section B.3.2., POISSON/PANDIRA Inputs to LATTICE or Section C.3.2, SUPERFISH Inputs to LATTICE.

3.2.1 The CON Array Variables for LATTICE

The main avenue of communication the user has with all the POISSON/SUPERFISH programs, except AUTOMESH, is through the CON array. The CON array is a one dimensional array of 125 elements specifying various parameters and options. Since some of the variables have different meanings for the POISSON/ PANDIRA/ MIRT codes and SUPERFISH code, the CON elements with the appropriate variables are listed both in Chapter 5—POISSON/ PANDIRA and in Chapter 6—SUPERFISH.

Please note, there are a number of elements in the CON array that must be changed in LATTICE if they are to have any effect on the problem These specific elements with a brief description of their function are listed in Table 3–1

TABLE 3-1
CON VARIABLES FOR LATTICE

	Number	Name	Automesh Default	Lattice Default	Description
⊙	CON (2)	NREG	Computed	None	Total number of regions for the problem $1 \leq NREG \leq 31$.
⊙	CON (9)	CONV	1.0	1.0	Conversion factor for coordinate units. CONV = (no. of centimeters) per (unit), e.g. CONV = 1.0 – Centimeters. = 0.1 – Millimeters. = 2.54 – Inches.
⊙	CON(21)	NBSUP	1 0	0	Indicator for boundary conditions on the UPper, LOwer, RighT and LeFt boundaries of the rectangular region defining the boundary. The two AUTOMESH default values correspond to SUPERFISH or POISSON/ PANDIRA/MIRT problems. 0 - Indicates Dirichlet boundary conditions, which means electric (for SUPERFISH TM modes) or magnetic (for POISSON/ PANDIRA/MIRT) field lines are PARALLEL to the boundary line 1 – Indicates Neumann boundary conditions, which means electric (for SUPERFISH TM modes) or magnetic (for POISSON/ PANDIRA/MIRT) field lines are PERPENDICULAR to the boundary line
⊙	(22)	NBSLO	0 or 1	1	
⊙	(23)	NBSRT	1 0	0	
⊙	(24)	NBSLF	1 0	0	
	CON(32)	IPRINT	None	0	Print options in LATTICE IPRINT = –1 – Prints X, Y coordinates of mesh points IPRINT = 0 no print of mesh points
	CON(36)	NSG	Computed	0	The number of boundary segments for SUPERFISH

Number	Name	Automesh Default	Lattice Default	Description
CON(37)	MAP	None	1	For POISSON/MIRT: A parameter in the conformal transformation $w = z ** MAP/(MAP * RZERO ** (MAP - 1))$. where: RZERO = CON(125). MAP ≠ 1 - the current density is adjusted to conform to the transformed geometry in all closed regions. MAP = 1 - no current density adjustment. Note: if do not want any current density adjustment (user has input the correct density for the transformational geometry), MAP should not be input until execution of POISSON/PANDIRA.
	NCELL	1	1	For SUPERFISH: The number of cells in multicell problems. Not needed until execution of SFO1, but in AUTOMESH it defaults to 1.
CON(70)	ICAL	None	0	Indicator for type of formula to use for calculating current density. ICAL = 0 - Use normal area formula. ICAL = 1 - Use angle formula for calculating the current associated with a point or when accurate fields near coil boundaries are needed.
CON(79)	RHOXY	None	1.6	The starting over-relaxation factor for the irregular mesh generation.
⊙ CON(81)	NOTE	None	1	An indicator for determining the order in which points are relaxed. NOTE = 0 - the order is air points, interface points, then iron points. MUST be used for PANDIRA. (See PANDIRA examples, Secs. 10.4 - 10.6.) NOTE = 1 - the order is (air + interface) points, then iron points. (For SUPERFISH CON(81) = RSTEM and need not be entered until running SFO1.) Default is 1.0.

Number	Name	Automesh Default	Lattice Default	Description
CON(84)	EPSO	None	1.0E – 5	The convergence criterion for mesh generation. If program has trouble converging, increasing EPSO might help
CON(123)	TNEGC	None	0.0	A parameter used in conformal transformation. Input the total negative current in original geometry. LATTICE stores the negative transformed currents.
CON(124)	TPOSC	None	0.0	A parameter used in conformal transformation. Input the total positive current in original geometry. LATTICE stores the positive transformed currents.
CON(125)	RZERO	None	1.0	The scaling factor of the conformal transformation. $w = z$ ** MAP/(MAP * RZERO ** (MAP-1)) where: (MAP = CON(37)) and normally, RZERO = aperture radius.

where:

⊙ — quantities which are most frequently input

In addition to the CON elements listed in Table 3-1, a user may at this time opt to change the default values of any of the other CON elements listed in Chapters 5 and 6. Even though these changes will have no effect in LATTICE, the changes will be carried through to dump 0 on the output file, TAPE35, and be available to the other programs when needed.

The changes in the CON array are entered in the special free-format input as described in Section 3.2.2. The following example illustrates the various options.

Example *9 2.54 *46 6 *21 1 1 1 0 s

inputs: CON(9) = 2.54
CON(21) = 1
CON(22) = 1
CON(23) = 1
CON(24) = 0
CON(46) = 6

explanation: * — occurs before the first numbers of element for which subsequent values will be input. Spaces (or commas) are the delimiters between all input values and star elements. When several elements in a row are to be changed, only the initial element need be indicated by a "*," as done above for *21.

s — designates the end of CON array changes. Since the above input line will be read into CON array which has 125 elements, if an s, meaning skip the rest, is not given, the program will inquire "?" for more input until an "s"is given.

3.2.2 Format-Free Input Routine

The POISSON group programs use the format-free input routine, FREE, which has one of the following calling sequences:

```
CALL FREE (1, RAY1, N1)
CALL FREE (2, RAY1, N1, RAY2, N2)
CALL FREE (3, RAY1, N1, RAY2, N2, RAY3, N3)
```

where RAY1, RAY2, and RAY3 are the array (and/or variable) names into which N1, N2, and N3 values, respectively, are to be input.

The special characters and their functions used in FREE are listed below.

character	function
+ −	Indicates the sign of the number (mantissa or exponents). The "+" sign is optional before the mantissa but needed for the exponent.
.	The decimal point is REQUIRED for all floating point numbers. Floating point numbers may be entered in either fixed format (i.e., xx.xxx) or scientific format (i.e., x.xxxxe+xx)
E	The signed integer number following the letter "E" gives the power of ten to which the previous floating point number is raised. No blank before or after the "E" is allowed. The sign ("+" or "−") of the exponent must follow the "E".
*I	Store the number following "I" into the array location CON(I). Successive input numbers will be stored into array locations: CON(I), CON(I+1), CON(I+2), etc.
RN	Store the last input number, "N" times into the array locations: CON(I), CON(I+1), CON(I+2). . . . through (I+N). A blank separating the previous input number and the "R" is optional.
S	Skip the rest of the N1, N2, or N3 values requested in the call and go on to the next array or return if the current array is the last array in the argument list.
C	Skip the rest of the N1, N2, or N3 values requested in the call, set N1, N2, or N3 equal to the number of values input into the current array and go on to the next array or return.
Blank ,	Blank and comma are the only other non-numeric characters allowed in the input field, and one of these characters must be used to separate input values.

Comments may follow the last "S," "C," or required number of data on any input line. The example below ilustrates all the above features. A and B are dimensioned arrays, and K is a single variable.

```
Calling sequence:   N=100
                    CALL FREE(3,A,5,B,N,K,1)
Input line:        -3,4.  +5.3E-2 R2 S *20 .1R10 C 13 THIS IS AN EXAMPLE
In memory:          A(1) = -3
                    A(2) = 4.0
                    A(3) = 0.053
                    A(4) = 0.053
                    A(5) = unchanged
                    B(1) thru B(19) = unchanged
                    B(20) thru B(29) = 0.1
                    N = 10
                    K = 13
```

3.2.3 TAPE73 — Input to LATTICE

TAPE73, which is generated in AUTOMESH and written with the special free format, is described in the preceding section, Sec. 3.2.2. Normally the user need not know the contents or format of TAPE73. However, if the user is unsuccessful in correcting the listed errors in either AUTOMESH or LATTICE, he might gain some insight by examining TAPE73.

Below are listings, followed by explanations, of the two TAPE73 files generated by AUTOMESH in section 2.5.1., using the two input files HMAG and DTL.

The line numbers listed on the left are not part of TAPE73, but are used as pointers for explanations that follow.

TAPE73 created by input file HMAG

```
 1 h-magnet test, uniform mesh  4/23/85
 2 *2  3 *21 0 1 0 0 *9 1.0000 skip
 3   1   1    0.0000    0.0000 0 0 region
 4   1   1    0.0000    0.0000
 5  50   1   22.0000    0.0000
 6  50  34   22.0000   13.0000
 7   1  34    0.0000   13.0000
 8   1   1    0.0000    0.0000 coun
 9   2   2    0.0000    0.0000 0 1 region
10   1   6    0.0000    2.0000
11  13   6    5.1000    2.0000
12  13   7    5.5000    2.4000
13  14   8    5.5000    2.8000
14  13   9    5.5000    3.2000
15  14  10    5.5000    3.6000
16  13  11    5.5000    4.0000
17  14  12    5.5000    4.4000
18  13  13    5.5000    4.8000
19  14  14    5.5000    5.2000
20  13  15    5.5000    5.6000
21  14  16    5.5000    6.0000
22  35  16   15.0000    6.0000
23  34  15   15.0000    5.6000
24  35  14   15.0000    5.2000
25  34  13   15.0000    4.8000
26  35  12   15.0000    4.4000
27  34  11   15.0000    4.0000
28  35  10   15.0000    3.6000
29  34   9   15.0000    3.2000
30  35   8   15.0000    2.8000
31  34   7   15.0000    2.4000
32  35   6   15.0000    2.0000
33  34   5   15.0000    1.6000
34  35   4   15.0000    1.2000
35  34   3   15.0000    0.8000
36  35   2   15.0000    0.4000
37  34   1   15.0000    0.0000
38  50   1   22.0000    0.0000
39  50  34   22.0000   13.0000
40   1  34    0.0000   13.0000
41   1   6    0.0000    2.0000 coun
42  3 1 -25455.7910    0.0000 0 1 region
43  14   1    6.0000    0.0000
44  33   1   14.5000    0.0000
45  34   2   14.5000    0.3929
46  33   3   14.5000    0.7857
47  34   4   14.5000    1.1786
48  33   5   14.5000    1.5714
49  34   6   14.5000    1.9643
50  33   7   14.5000    2.3571
51  34   8   14.5000    2.7500
52  33   9   14.5000    3.1429
53  34  10   14.5000    3.5357
54  33  11   14.5000    3.9286
55  34  12   14.5000    4.3214
56  33  13   14.5000    4.7143
57  34  14   14.5000    5.1071
58  33  15   14.5000    5.5000
59  14  15    6.0000    5.5000
60  15  14    6.0000    5.1071
61  14  13    6.0000    4.7143
62  15  12    6.0000    4.3214
63  14  11    6.0000    3.9286
64  15  10    6.0000    3.5357
65  14   9    6.0000    3.1429
66  15   8    6.0000    2.7500
67  14   7    6.0000    2.3571
68  15   6    6.0000    1.9643
69  14   5    6.0000    1.5714
70  15   4    6.0000    1.1786
71  14   3    6.0000    0.7857
72  15   2    6.0000    0.3929
73  14   1    6.0000    0.0000 coun
```

Line
No. 1: Title line, with blank in column 1 for POISSON/PANDIRA/MIRT problem.

2: CON variables that have been set by AUTOMESH.
(See Table 3–1 CON Variables for LATTICE for more detail.)

CON(2)	=	3	– NREG – total number of regions.
CON(21)	=	0	– Boundary conditions set up for upper, lower,
CON(22)	=	1	right and left boundaries, respectively, of the
CON(23)	=	0	rectangular region of the problem.
CON(24)	=	0	
CON(9)	=	1.000	– CONV – coordinates are in centimeters.
skip			– The "s" designates end of CON entries; any comments may follow "s".

3: This is the region entry line. LATTICE expects six entries. The variable names are identical to REG NAMELIST variables, so refer to Table 2–1 REG NAMELIST VARIABLES for more detail.

IREG	=	1	– The region number.
MAT	=	1	– The material number for this region. MAT = 1 – air region, when CUR = 0.
CUR	=	0.0000	– The total current, if a coil region.
DEN	=	0.0000	– The current density, if a coil region.
ITRI	=	0	– The type of triangle for the mesh. ITRI = 0 – equal weight, isoceles triangle.
IBOUND	=	0	– The special region boundary indicator. IBOUND = 0 – Dirichlet boundary for this region.
region			– A comment. No "s" is required since the maximum number, (6), of entries is given.

4–7: Each line lists the horizontal and vertical mesh numbers and the corresponding coordinates. (*i.e.*, line 5: horizontal mesh number = 50 for horizontal coordinate = 22. cm., vertical mesh number = 1 for vertical coordinate = 0. cm.)

8: Mesh points and coordinates identical to line 4, to form a closed region.

coun	– The "c" designates both end of entries and to count and store number of boundary point entries for this region.

9: Second region line entry.

IREG	=	2	– Region number 2.
MAT	=	2	– Iron region.
⋮			
IBOUND	=	1	– Neumann boundary for this region.

Line
No. 10–41: Lists mesh point numbers and their corresponding coordinates for region 2 as described in lines 4–8.

42: Third region line entry.

IREG	= 3	– Region number 3.
MAT	= 1	– Coil Region since CUR $\neq$ 0.
CUR	= −25455.7910	– The total current in amps. The sign indicates the direction of the current vector: + "into the plane of the paper"; – "out of the plane of the paper".

43–73: Lists mesh point numbers and their corresponding coordinates for region 3 as described in lines 4–8.

DTL TAPE73

```
 1 isuperfish dtl test problem
 2 *2   4 *21 1 0 1 1 *9 1.0000
 3 *36  14 *37 1 skip
 4    1   1    0.0000   0.0000 0 1 region
 5    1   1    0.0000   0.0000
 6    1  24    0.0000   5.0000
 7    1  29    0.0000   7.0000
 8    1  47    0.0000  21.1000
 9   18  47    4.2843  21.1000
10   18  29    4.2843   7.0000
11   18  24    4.2843   5.0000
12   18  20    4.2843   4.0302
13    8  20    1.8482   4.0302
14    7  19    1.6045   4.0302
15    6  19    1.3634   3.9683
16    6  18    1.1821   3.7977
17    5  17    1.1055   3.5608
18    6  16    1.0916   3.3344
19    5  15    1.0778   3.1081
20    5   6    0.9532   1.0712
21    4   5    0.9394   0.8448
22    5   4    0.9759   0.6741
23    5   3    1.0956   0.5469
24    6   3    1.2637   0.5000
25   18   3    4.2843   0.5000
26   18   1    4.2843   0.0000
27    1   1    0.0000   0.0000
28  2  1   0.0000    0.0000 0 1 region
29   1  24 0.0000  5.0000
30  18  24 4.2843  5.0000 coun
31  3  1   0.0000    0.0000 0 1 region
32   1  29 0.0000  7.0000
33  18  29 4.2843  7.0000 coun
34  4  1   1.0000   0.0000 0 -1 region
35  18  47 4.2843 21.1000 coun
36 14001   1    0   1    1  24
37     1  24    0   1    1  29
38     1  29    0   1    1  47
39     1  47    1   0   18  47
40    18  47    0  -1   18  29
41    18  29    0  -1   18  24
42    18  24    0  -1   18  20
43    18  20   -1   0    7  19
44     7  19   -1   0    5  17
45     5  17    1  -1    4   5
46     4   5    1  -1    6   3
47     6   3    1   0   18   3
48    18   3    0  -1   18   1
49    18   1   -1   0    1   1
```

Line No.

1: Title line with non-blank character in column 1.

2: Comparable to the HMAG file described previously.

3: Additional CON variables for SUPERFISH problem.

- CON(36) NSG = 14 – The number of boundary segments whose parameters are listed in lines 36–49.
- CON(37) NCELL = 1 – The number of cells.

4: Comparable to the HMAG file, line 3, described previously.

5–27: Each line lists the horizontal and vertical mesh numbers and the corresponding coordinates as described in HMAG, line 4–7.

28–33: Addition of two line regions at location where vertical mesh size doubles, followed by their coordinates.

34: Drive point region – IBOUND = –1.

35: Drive point coordinates (upper right hand corner, default)

Line
No. 36: First number is NSG *100 = 14 * 100 = 1400.
The rest of this line and the following 13 lines (a table of NSG values) give:

```
 1    1   0   1   1   24
 1   24   0   1   1   29
 1   29   0   1   1   47
 .    .   .   .   .    .
 .    .   .   .   .    .
 .    .   .   .   .    .
18    1  -1   0   1    1
 ↑    ↑   ↑   ↑   ↑    ↑
 ↑    ↑   ↑   ↑  [K   L ENDING POINT OF SEGMENT ]
 ↑    ↑   ↑   ↑
 ↑    ↑  [ΔK  Δ L FROM STARTING TO 2ND POINT ]
 ↑    ↑
[K   L STARTING POINT OF SEGMENT ]
```

where: K, L are horizontal, vertical mesh point numbering.

3.3 Output from LATTICE

LATTICE generates two output files — TAPE35, dump 0 and OUTLAT — and, if run interactively, prints output messages at the terminal.

In a successful run, LATTICE outputs:

1. To OUTLAT file and to the terminal:
   - no error messages
   - the messages:
     ```
     iteration converged
     elapsed time =  (--)  sec.
     generation completed
     dump number 0 has been written on tape35
     ```
2. To dump 0 on TAPE35:
   - binary information needed for the execution of any of the programs – POISSON, PANDIRA, MIRT, SUPERFISH, and/or TEKPLOT.

In this case, the user need not be concerned any further with the contents of the output files.

However, the user may print or examine the ASCII file, OUTLAT, with any editor, but not the binary file, TAPE35. Most of the relevant information listed in OUTLAT is self-explanatory, including error and diagnostic messages. The last part of OUTLAT lists the complete CON array variable names and values. Those elements that have been changed in the input to LATTICE are flagged by "CON" preceding the element number. For more information of the contents of files OUTLAT and TAPE35, dump 0 consult POISSON/ SUPERFISH Reference Manual.

3.4 Diagnostic and Error Messages

LATTICE lists all of diagnostic and error messages to the output file, OUTLAT, and some to the terminal if run is interactive. An explanation of the common terminology used in these messages is listed below.

1.	K, L	The mesh point numbering for the horizontal and vertical coordinates.
2.	X, Y	The horizontal, vertical coordinates, respectively.
3.	LPRIME, LPRIME	The mesh point numbering for the second of the two coordinates.
4.	(--)	Means the computer prints out the value.

3.4.1 Messages Containing "DATA ERROR"

These messages are issued whenever LATTICE encounters any errors in reading the input file. Mostly, such errors occur when a user creates his own input file for LATTICE. If the input file for LATTICE has been generated by a successful AUTOMESH run, it is unlikely there would be any errors. In any case, the errors issued are self-explanatory. The user need only correct the identified error in the output file and rerun.

LATTICE checks for two types of errors — format and content — and takes different action, accordingly. These two types are described below.

1. --- INPUT DATA ERROR ---
   These types of error messages are printed from the subroutine FREE whenever the input data is not in the special free-format entry. These messages, which are listed both in the file OUTLAT and printed at the terminal, are self-explanatory. The input line that is in error is also listed. If the run is interactive, the user is given an opportunity to re-enter the line in error. If the run is non-interactive, LATTICE aborts immediately; the user should correct the specific error in the input file and rerun the program.

2. --- DATA ERROR ---
   These error messages are written to the file, OUTLAT, from subroutine REREG whenever the routine encounters errors while checking the contents of each input line. The complete input file is processed and, if any error message has been written to OUTLAT, LATTICE will print the following message at the terminal and to OUTLAT before aborting:

   ```
   the errors in the input file that are listed above
   in output file, outlat, have caused this run to be
   aborted.  examine outlat, correct errors in input
   file and rerun or if dimensional problem, recompile.
   ```

   The user should examine all these self-explanatory error messages in OUTLAT and make the recommended changes.

3.4.2 Messages Containing "ERROR EXIT"

1. `--- ERROR EXIT --- TWO MESH DATA POINTS WITH A DIFFERENT K, L HAVE THE SAME X, Y COORDINATES`
   Followed by the printout of two coordinates which have same value but different mesh point numbering.

   1a. LATTICE writes these messages to the file OUTLAT if these errors are encountered while the program is calculating the current density in the subroutine GENOR and function ANGLF. The complete current region mesh points are processed before LATTICE prints to OUTLAT and to the terminal the message below and then terminates.

```
errors listed above in output file, outlat, have
caused this run to abort.  errors probably due to
mesh problems.  reduce mesh size and rerun.
```

2. `--- ERROR EXIT --- IN SUB. ANGLE COST = (--) AT KO = (--) LO = (--)`
   The program has found a cosine value greater than 1.0 at mesh point numbering (KO, LO). See 1a. above.

3. `--- ERROR EXIT --- NWMAX EXCEEDS PROGRAM DIMENSIONS OF (--) ...`
   Prints to OUTLAT and the terminal from subroutine PRELIM and aborts immediately; make the recommended changes and recompile.

3.4.3 Messages Containing "TROUBLE" and "WARNING"

1. `--- TROUBLE ---DIMENSIONS FOR NO. OF SEGMENTS EXCEEDED NSG OF (--) ...`
   Prints to OUTLAT and terminal and immediately aborts. Message from main program; follow instructions and recompile.

2. `--- WARNING ---THE MESH HAS NEGATIVE AND/OR ZERO AREA TRIANGLES`
   LATTICE writes to the file OUTLAT a message whenever it encounters a negative or zero area in subroutine FILPOT, followed by the three coordinates that make up this triangle. The program processes the triangles of all regions before printing above message to OUTLAT and terminating. Message from main program; follow instructions.

3. `--- WARNING ---THE NUMBER OF INTERIOR POINTS=0 ...`
   Message from subroutine SETTLE is self-explanatory.

3.4.4 Miscellaneous Messages

1. THE ABOVE REGION IS NOT CLOSED.
   This message is output to OUTLAT from subroutine REREG and is only a warning. User should check that the same values for the first and last coordinates for this region are specified if a closed region with interior points is desired.

2. ITERATION TERMINATED---MAXIMUM NUMBER OF CYCLES.
   This message is output to OUTLAT from subroutine SETTLE and is only a warning. The mesh generation did not converge to the required accuracy after 100 iteration cycles. Run is continued with present mesh. User could try running the problem with this mesh or cut mesh size and rerun.

3.5 Examples of LATTICE Runs

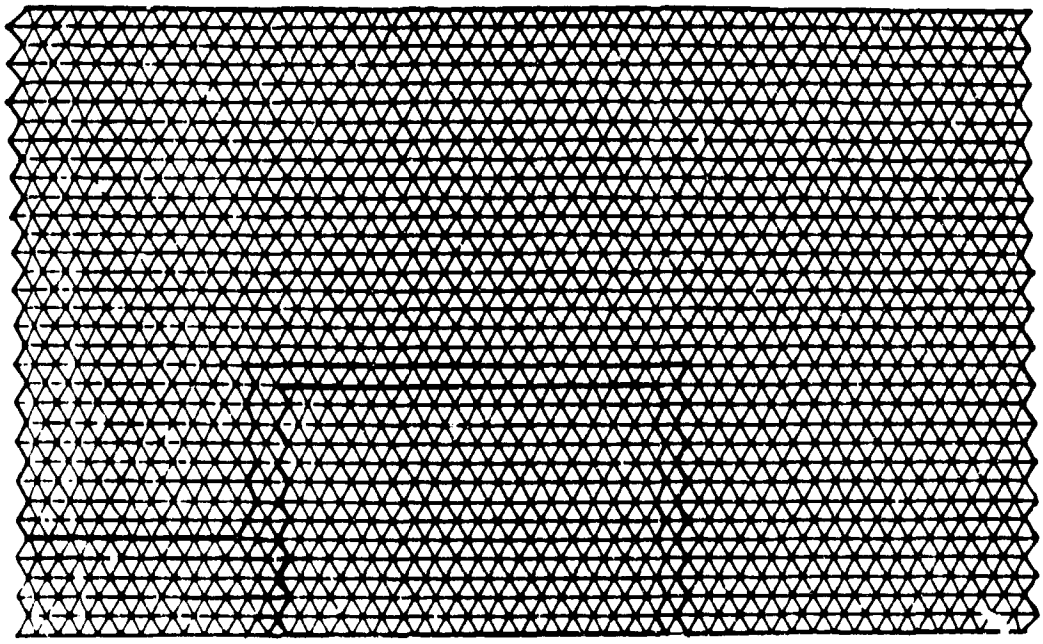
Sections 3.5.1 and 3.5.2 list the execution of LATTICE on the CRAY computer using the TAPE73 files generated by AUTOMESH from the HMAG and DTL input files, respectively. In both problems, since no CON variables need to be changed, an ↵ is typed when LATTICE inquires for CON input.

3.5.1 LATTICE Run — H-Shaped Dipole Magnet

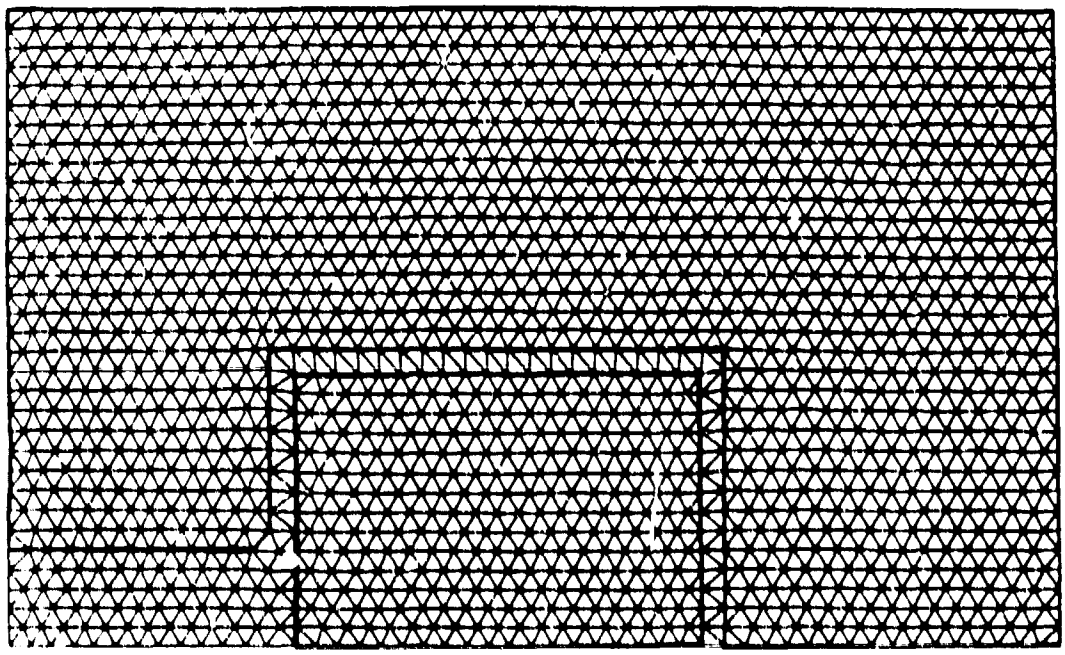
Using the TAPE73 that AUTOMESH generated for HMAG, the user types only the underlined quantities.

```
lattice
 ?type input file name
tape73
 beginning of lattice execution
 dump 0 will be set up for poisson
 h-magnet test, uniform mesh  4/23/85

 ?type input values for con(?)
? ↵
 elapsed time  =  0.6 sec.
0iteration converged
 elapsed time  =  0.8 sec.
 generation completed
 dump number 0 has been written on tape35.
 stop
lattice  ctss time     1.084   seconds
cpu=      .594   i/o      .429  mem=     .062
 all done
```

At this point, the user can instruct TEKPLOT (instructions in Chapter 4) to create a graph of LATTICE's generated mesh as shown in Fig. 3-1(b).



(a) Initial LATTICE regular triangular mesh.



(b) Final LATTICE "relaxed" irregular triangular mesh to coincide with the physical boundaries.

Fig. 3-1: Triangular mesh configuration in LATTICE for the H-shaped magnet.

3.5.2 SUPERFISH Run — Drift-Tube Linac Cavity

Using the TAPE73 that AUTOMESH generated for the DTL input file, the user types only the underlined quantities to execute LATTICE.

```
 lattice
 ?type input file name
?  tape73

 beginning of lattice execution
  dump 0 will be set up for superfis
 1superfish dtl test problem
  ?type input values for con(?)
?  /
  elapsed time  =   0.5 sec.
 0iteration converged
  elapsed time  =   0.5 sec.
  generation completed
  dump number 0 has been written on tape35
  stop
 lattice  ctss  time  .754    seconds
 cpu=      .204    i/o  .491  mem=      .059
  all done
```

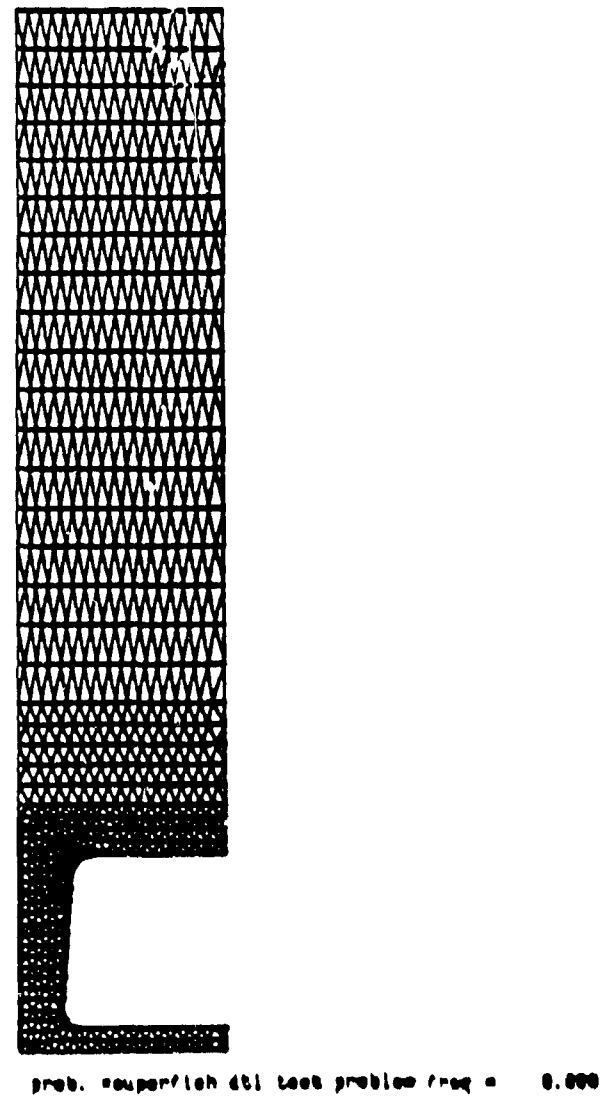


Fig. 3 2: Mesh generated by LATTICE for drift-tube linac cavity.

At completion of LATTICE, the program TEKPLOT (see Chapter 4) creates the graph shown on the right, Fig. 3–2.

Chapter 4

TEKPLOT

Contents 4-1

4.1 The Job TEKPLOT Performs

TEKPLOT is an interactive program which provides a graphical display of the data generated by most of the programs in the POISSON/SUPERFISH Group Programs. It will plot the:

- physical boundaries and mesh resulting from a LATTICE run
- equipotential lines resulting from a POISSON/PANDIRA/MIRT run
- field lines resulting from a SUPERFISH/SFO1 run.

TEKPLOT requires access to the graphics package PLOT10. If PLOT10 is not available on your system, the 9 calls to this package that are described in the beginning of the source file TEKSO will have to be replaced by calls to an available package.

4.2 The Input to TEKPLOT

The input to TEKPLOT is read using the format-free input routine described in Chapter 3 on LATTICE, in Section 3.2.2. The user is not required to read this section; the input to TEKPLOT is fully explained in this section.

Table 4–1 contains a list of all variables used in TEKPLOT for which the user can supply a value. These variables, along with their default values and a brief description, are listed in alphabetical order.

If all the default values in Table 4–1 are sufficient for your purposes, they do not need to be entered. Instead, enter **$** to signal that the default values are acceptable and you want to proceed with execution. However, if you want to change the value of a variable, all preceding values must be entered as well, even if they are default values, and separated by spaces. An **$** may then be entered.

Values for floating point variables must contain a decimal point.

The interactive program TEKPLOT may be viewed as a large loop that creates a desired plot. This loop is executed until a negative value is given to the variable NUM. NUM represents the Dump Number on TAPE35 containing the data to be used in creating the plot.

This loop includes 4 prompts: 3 regular and routine, 1 optional:

```
?type input data- num, itri, nphi, inap, nswxy
?
```

Prompt 1: solicits information concerning the big picture for the current plot. For example, this is where the value for the dump number, the exit from TEKPLOT, the inclusion of the mesh, the receipt of the optional prompt, and the orientation of the X and Y axes are determined.

```
?type input data- xmin, xmax, ymin, ymax
?
```

Prompt 2: gives the user an opportunity to choose which section of the larger picture will be plotted. The smaller the section, the more the detail.

Optional Prompt:

```
?type input data- amin, amax
?
```

allows the user to choose the minimum and maximum values of the equipotential/field lines. To receive this prompt, the variable INAP in Prompt 1 must be set.

```
?type go or no
?
```

Prompt 3: provides the user with the opportunity to correct the given input for the current plot. A reply of go will clear the screen and create the plot, a reply of no will return the user to Prompt 1 to repeat the input process.

TABLE 4-1
REG VARIABLES FOR TEKPLOT

Variable	Default	Description
AMIN	0	The minimum value for the equipotential/field lines to be plotted.
AMAX	0	The maximum value for the equipotential/field lines to be plotted.
INAP	0	This flag indicates whether the additional prompt is desired to set the minimum (AMIN) and maximum (AMAX) values for the equipotential/field lines to be plotted. This indicates that the user is interested in a particular range. INAP = 1 - read in user-defined values for AMIN and AMAX. INAP = 0 - accept default values for AMIN and AMAX. Values plotted are: (AMIN + DELTA) to (AMAX - DELTA) in steps of DELTA where DELTA = (AMAX - AMIN)/(NPHI + 1).
ITRI	0	This flag indicates whether the triangular mesh is plotted. ITRI = 0 - triangular mesh is not plotted. ITRI = 1 - triangular mesh is plotted.
NPHI	0	The value of this variable specifies the number of equipotential lines plotted. The program does not plot the minimum or maximum potential values, one of which is usually a point. For most problems, a good number for NPHI is between 20 and 30. POISSON/PANDIRA/MIRT - equipotential lines are 2-dimensional field lines or flux surfaces. SUPERFISH - rH_ϕ = constant lines which are proportional to electric field lines.
NSWXY	0	This flag indicates whether an interchange of the x and y axes on the plot is desired. NSWXY = 0 - no interchange. NSWXY = 1 - interchange.
NUM	0	This variable indicates the location (dump number on TAPE35), of the data to be used in creating the plot.
XMAX XMIN YMAX YMIN	XMAX XMIN YMAX YMIN	The values for these variables specify the limits for each plot, which may be any part of the problem. Setting these limits in TEKPLOT does not affect the problem definition. If these values are allowed to default, the entire problem area will be plotted.

4.3 The Output from TEKPLOT

In addition to the generation of plots sent to the user's terminal, TEKPLOT creates a file called OUTTEK. This file is only created if field lines have been drawn and contains a list of the plotted potential values. This file is for the user's information and is rarely used.

4.4 Error Message

There is only one error message for TEKPLOT. It is sent to the terminal with the current line of data from the subroutine FREE whenever the input data is not in the special free-format:

```
--- INPUT DATA ERROR ---
```

Following this notice, the user is provided with information detailing the nature of the error. The user is then provided with the opportunity to re-enter the line.

```
RETYPE LINE
```

4.5 Examples of TEKPLOT Runs

Sections 4.5.1 and 4.5.2 describe the execution of TEKPLOT on a CRAY computer using the file TAPE35 generated by POISSON Group Programs from the original input files to AUTOMESH, namely HMAG and DTL.

4.5.1 TEKPLOT Run — H-Shaped Dipole Magnet

After AUTOMESH and LATTICE have been run, the user may check the problem geometry and the appearance of the triangular mesh.

```
tekplot
 ?type input data- num, itri, nphi, inap, nswxy,
? /
 input data
 num= 0   itri= 0   nphi= 0   inap= 0   nswxy= 0
 plotting prob. name = h-magnet test, uniform mesh 4/23/85        cycle = 0
 ?type input data- xmin, xmax, ymin, ymax,
? /
 input data
 xmin= 0.0000 xmax= 22.000 ymin= 0.0000 ymax= 13.000
 ?type go or no
? go
```

To check the problem geometry, values for all the variables are allowed to default. A carriage return after go will clear the screen and produce Fig. 4–1.

TEKPLOT will only plot CLOSED regions. If the region is not plotted, check the input file to AUTOMESH to assure a closed region (first and last point of the region must be the same).

When the user is ready to continue, a carriage return will clear the screen and TEKPLOT will continue with execution.

To produce a graph of the triangular mesh, the following is entered:

```
 ?type input data- num, itri, nphi, inap, nswxy,
? 0 1 /
 input data
 num= 0   itri= 1   nphi= 0   inap= 0   nswxy= 0
 plotting prob. name = h-magnet test, uniform mesh 4/23/85      cycle = 0
 type input data- xmin, xmax, ymin, ymax
? /
 input data
 xmin= 0.0000   xmax= 22.000   ymin= 0.0000   ymax= 13.000
 ?type go or no
? go
```

Although the default value for NUM is 0, the variable ITRI must be set, therefore both values are entered and all other values are allowed to default by entering the /. A carriage return after go will clear the screen and produce Fig. 4 2.

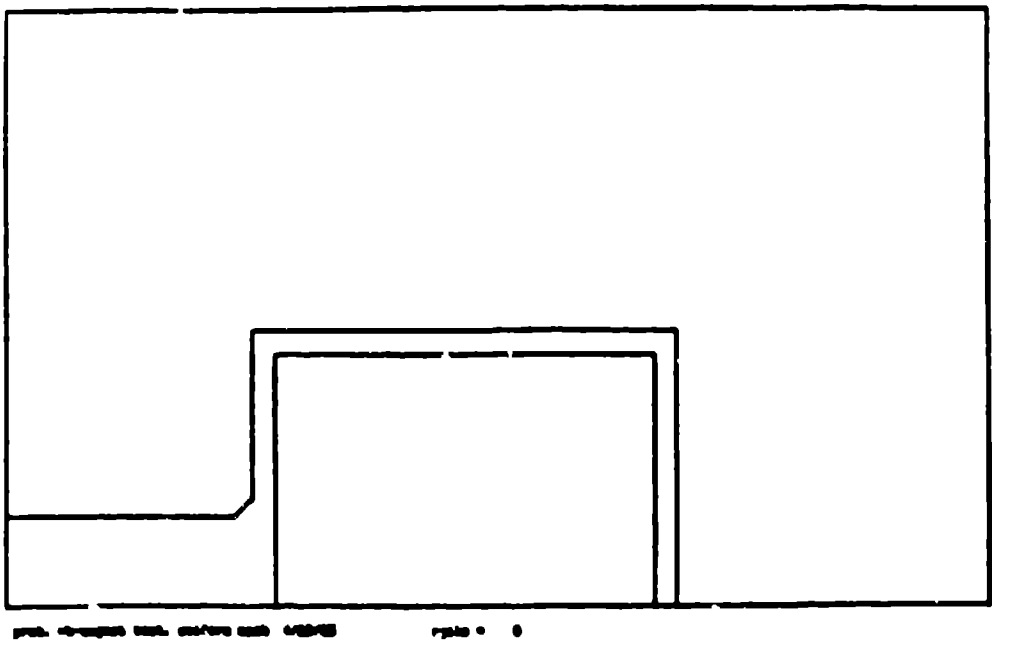


Fig. 4–1: Geometry outline for the H-shaped dipole magnet.

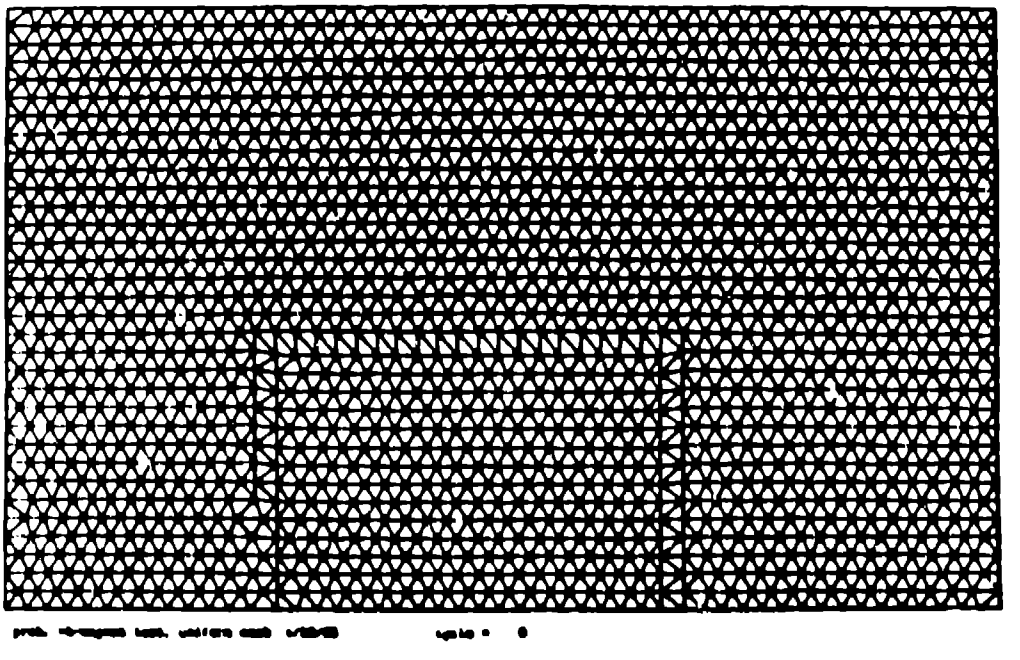


Fig. 4–2: Mesh for the H-shaped dipole magnet.

When the user is ready to continue, a carriage return will clear the screen and TEKPLOT will continue with execution To exit TEKPLOT, a negative value is assigned to the variable NUM and g is entered to avoid further entries.

```
 ?type input data- num, itri, nphi, inap, nswxy,
? -1 g
 tekplot      ctss time     .766   second:
 cpu=      .031    i/o= .681    mem= .053
  all done
```

After the execution of POISSON, the user can plot the field lines.

```
tekplot
 ?type input data- num, itri, nphi, inap, nswxy,
? 1 0 20 g
 input data
 num= 1   itri= 0   nphi= 20   inap= 0   nswxy= 0
 plotting prob. name = h-magnet test, uniform mesh  4/23/85        cycle= 370
 ?type input data data- xmin, xmax, ymin, ymax,
? g
 input data
 xmin= 0.000  xmax= 22.000  ymin= 0.000  ymax= 13.000
 ?type go or no
? go
```

After POISSON has been run, only dump numbers 1 or greater are of interest. The triangular mesh is not desired and, for this example, we chose to look at 20 field lines.

A carriage return after go will clear the screen and produce Fig. 4–3.

When the user is ready to continue, a carriage return will clear the screen and TEKPLOT will continue with execution. To exit TEKPLOT, a negative value is assigned to the variable NUM and g is entered to avoid further entries.

```
 ?type input data- num, itri, nphi, inap, nswxy,
? -1 g
 tekplot  ctss time       .561    seconds
 cpu=      .037   i/o=      .480  mem=     .044
  all done
```

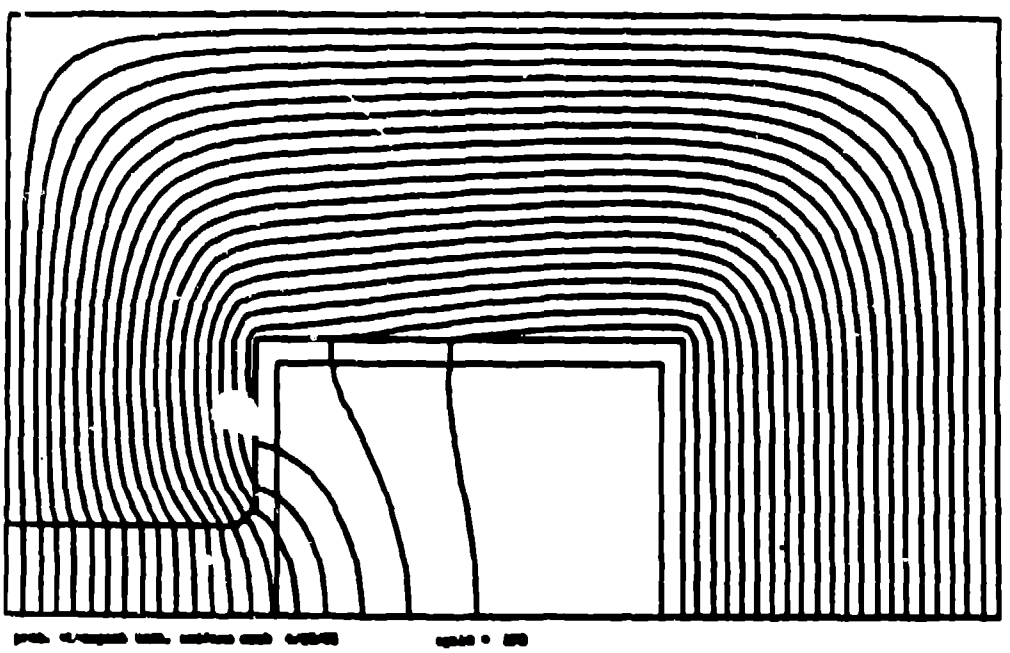


Fig. 4–3: Field lines for the H-shaped dipole magnet.

4.5.2 TEKPLOT Run — Drift-Tube Linac Cell

After AUTOMESH and LATTICE have been run, the user may check the problem geometry and the appearance of the triangular mesh.

To check the problem geometry, values for all the variables are allowed to default. A carriage return after go will clear the screen and produce Figure 4–4.

```
tekplot
 ?type input data- num, itri, nphi, inap, nswxy,
?  g
 input data
 num= 0   itri= 0   nphi= 0   inap= 0   nswxy= 0
  plotting prob.  name = superfish dtl test problem       cycle =   0
 ?type input data- xmin, xmax, ymin, ymax
?  g
 xmin= 0.0000 xmax= 4.284 ymin= 0.0000 ymax= 21.101
 ?type go or no
?  go
```

TEKPLOT will only plot CLOSED regions. If the region is not plotted, check the input file to AUTOMESH to assure a closed region (first and last point must be the same).

When the user is ready to continue, a carriage return will clear the screen and TEKPLOT will continue with execution.

To produce a graph of the triangular mesh, the following is entered:

```
 ?type input data- num  itri, nphi, inap, nswxy,
?  0 1 g
 input data
 num=  0   itri= 1   nphi= 0    inap= 0   nswxy= 0
 plotting prob. name = superfish dtl test problem            cycle = 0
 ?type input data data- xmin, xmax, ymin, ymax,
?  g
 input data
 xmin= 0.000  xmax= 4.284  ymin= 0.000  ymax= 21.101
 ?type go or no
?  go
```

Although the default value for NUM is 0, the variable ITRI must be set; therefore both values are entered and all other values are allowed to default by entering the g.

A carriage return after go will clear the screen and produce Fig. 4 5.

When the user is ready to continue, a carriage return will clear the screen and TEKPLOT will continue with execution. To exit TEKPLOT, a negative value is assigned to the variable NUM and g is entered to avoid further entries.

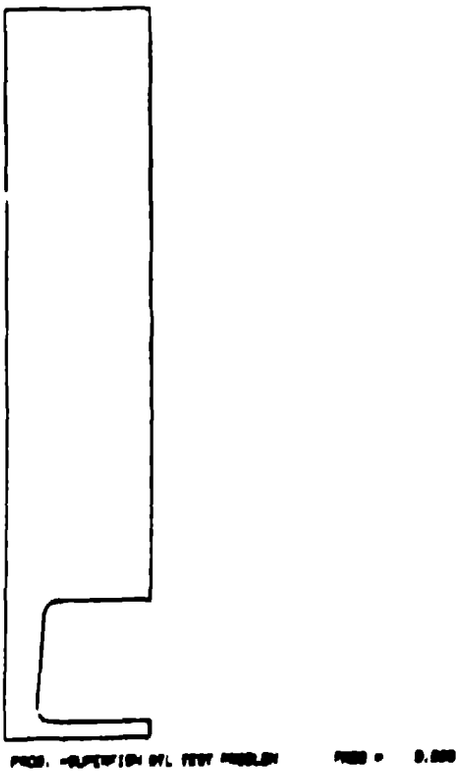


Fig. 4–4: Geometry outline for the drift-tube linac cell.

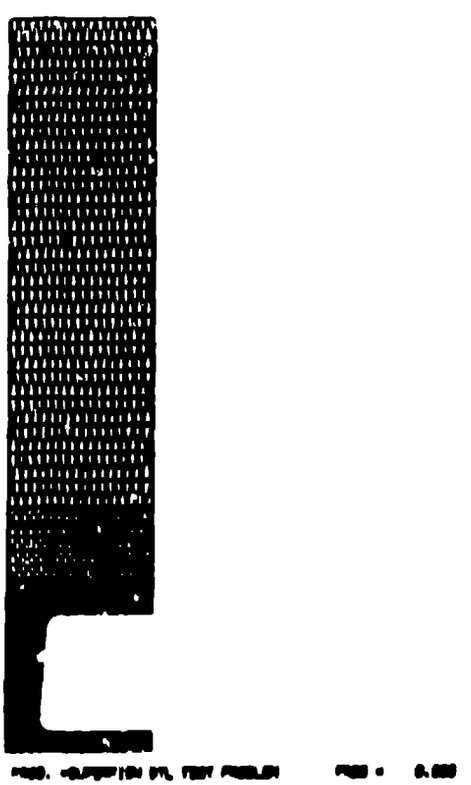


Fig. 4–5: Mesh for the drift-tube linac cell.

```
?type input data- num, itri, nphi, inap, nswxy,
? -1 $
 tekplot      ctss time 1.575 seconds
 cpu=      .128 i/o= 1.327 mem= .120
  all done
```

After the execution of SUPERFISH, the user can plot the electric field lines (rH_ϕ = const.).

After SUPERFISH has been run, only dump numbers 1 or greater are of interest. The triangular mesh is not desired and, for this example, we chose to look at 20 equipotential lines.

```
tekplot
 ?type input data- num, itri, nphi, inap, nswxy,
? 1 0 20 $
 input data
 num= 1   itri= 0   nphi= 20   inap= 0   nswxy= 0
 plotting prob.  name = superfish dtl test problem      cycle = 3
 ?type input data- xmin, xmax, ymin, ymax,
? $
 input data
 xmin=   0.0000   xmax= 4.284   ymin=   0.0000   ymax=   21.101
 ?type go or no
? go
```

A carriage return after go will clear the screen and produce Fig. 4-6.

When the user is ready to continue, a carriage return will clear the screen and TEKPLOT will continue with execution. To exit TEKPLOT, a negative value is assigned to the variable NUM and $ is entered to avoid further entries.

```
?type input data- num, itri, nphi, inap, nswxy,
? -1 $
 tekplot      ctss time 1.250 seconds
 cpu=      .298 i/o= .840 mem= .112
  all done
```

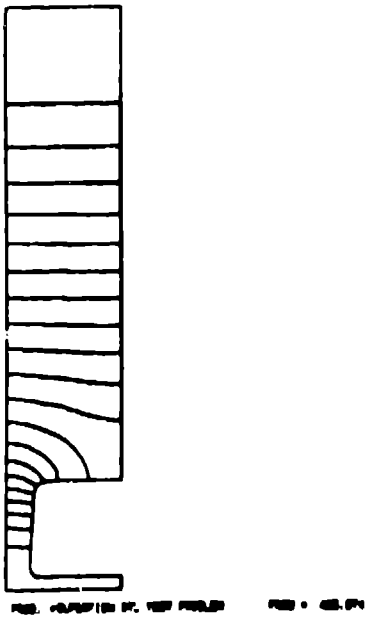


Fig. **Electric field lines (rH_ϕ = const.) for the drift-tube linac cell.**

Chapter 5

POISSON/PANDIRA

Contents 5-1

5.1 The Job POISSON/PANDIRA Perform

POISSON and PANDIRA are two independent programs which solve the Poisson equation for magnetostatic (electrostatic) problems with nonlinear isotropic iron (dielectric) and electric current (charge) distributions for two dimensional Cartesian or three-dimensional cylindrical geometry. POISSON utilizes the successive point over-relaxation method for the solution of the equations while PANDIRA directly solves the block tridiagonal system of difference equations. After solving the equations, both programs compute the derivatives of the potential to obtain the fields and their gradients and calculate the stored energy.

In addition, these programs have the option to:

- solve fixed potential problems.
- use infinite or a fixed constant permeability/permittivity value.
- read up to three different permeability tables or use internal table (very low carbon steel).
- perform harmonic analysis of the potential.
- do conformal mapping (POISSON only).
- solve permanent magnet problems (PANDIRA only).
- use anisotropic materials (PANDIRA only) for solution of residual field problems.

In general, either POISSON or PANDIRA may be used to calculate any of the above quantities, except those specifically identified. If either POISSON or PANDIRA has trouble converging, we recommend using the other program.

The input, output and execution of POISSON and PANDIRA are nearly identical. Thus they will be discussed together in this chapter and any differences will be clearly identified.

5.2 Terminology Used in POISSON/PANDIRA

"word1"("word2")	–	when describing magnetostatic and electrostatic quantities, "word1" refers to a magnetostatic quantity; "word2" in parentheses refers to an electrostatic quantity.
air region	–	a region in which the relative permeability (permittivity) = 1.0
iron(dielectric) region	–	a region in which the relative permeability (permittivity) is a constant $\neq$ 1.0 defined by a linear function or given by a nonlinear table.

5.3 Standard Input to POISSON/PANDIRA

All input, both standard and optional, to POISSON and PANDIRA are read in by the internal format-free subroutine, FREE. A FREE entry assumes, according to Fortran conventions, that variables beginning with I, J, K, L, M, and N are integers and all others are floating point variables; all floating point numbers must be entered with a decimal point. Each particular POISSON/PANDIRA entry will be given and its format explained below, but if more information is desired for FREE, refer to Sec. 3.2.2. Data may be entered either from an input file by specifying the name of the file or directly from the terminal by typing "tty."

Standard input refers to input data that is required (vs. optional input) for the execution of the programs. This standard input consists of two groups of input data–the NUM input and the CON array input.

5.3.1 NUM Input

NUM is an integer value specifying the dump number of TAPE 35 that the executing program reads and processes if NUM $\geq$ 0, or terminates if NUM $<$ 0. This is the first and last line entry for all the POISSON Group Codes, except AUTOMESH and LATTICE.

LATTICE generates dump 0 after a successful run. POISSON/PANDIRA begin execution with the reading in of NUM (NUM = 0, for start of new problem) and at completion generate a new file with dump number incremented by one (i.e., NUM + 1 $\rightarrow$ NUM). This feature allows a user to continue a run that has not converged or to calculate and print out auxiliary quantities that had not been done in the previous run.

In summary:

NUM = 0 – start of a new POISSON/PANDIRA run.
NUM > 0 – continue from previous POISSON/PANDIRA run.
NUM < 0 – terminate execution of POISSON/PANDIRA run.

Since only one value, NUM, is expected for this entry, no "s" is needed to indicate end of this input.

5.3.2 CON Array Input

The CON array—a one dimensional array of 125 elements specifying various parameters and options is the same CON array discussed in LATTICE, Sec. 3.2.1. The CON input values to POISSON/PANDIRA can change previously defined CON values, which were read from TAPE 35 dump ($\geq$ 0), or specify values which are used only in POISSON/PANDIRA runs and were not input in LATTICE or a previous POISSON/PANDIRA run. Unless all 125 entries are given, this entry, which may consist

of more than one line, must end with an "s." (An example of a CON array entry is given at the end of Sec. 3.2.1.)

A complete list of the CON array for POISSON/PANDIRA programs is given in Table 5–1. The CON elements are grouped according to function to facilitate in identifying all variables that perform a given function. To make the group complete, the CON variables, which must be input to LATTICE if they are to have any effect, are also included, but are identified with "LATTICE input only."

A complete numerical and an alphabetical list of all 125 elements for POISSON/PANDIRA/MIRT variables are given in Appendix C.

TABLE 5-1
CON VARIABLES FOR POISSON/PANDIRA

⊙ – denotes quantities most frequently input

	Number	Name	Default	Description
	5-1.1 Basic Problem Parameters			
	CON(2)	NREG		LATTICE input only.
⊙	CON(6)	MODE	−2	The permeability code in iron. MODE = −2 – μ-infinite in iron. = −1 – μ-finite, constant and defined by CON(10) = FIXGAM, ($\gamma = 1.0/\mu$). = 0 – Option a function of REG NAMELIST parameter MAT. (See Table 2-1.) MAT = 2 – μ-finite and defined by internal table (very low-carbon steel) or user-defined constant permeability/permittivity or user-defined stacking (fill) factor 3 ≤ MAT ≤ 5 – μ-finite and defined by table1 – table3. or user-defined constant permeability/permittivity or user-defined stacking (fill) factor. 6 ≤ MAT ≤ 11 – permanent magnet material with a user-defined $B(H)$ function.
	CON(7)	STACK	1.0	Stacking or fill factor for iron regions using MAT=2 (See above CON(6).)
	CON(9)	CONV	1.0	LATTICE input only.
	CON(10)	FIXGAM	.004	The value of γ (= $1.0/\mu$) used in a μ-finite but constant solution. [CON(6) = MODE = −1] (See example, Sec. 10.7.) Also used to initialize γ for μ-finite and variable [CON(6) = MODE = 0].
	CON (18)	NPERM	0	The number of permeability functions to be read in as data. NPERM ≠ 0 – must set CON(6) = MODE = 0 and the first optional input data listing the permeability/permittivity functions and/or tables must follow the CON array entry (see Sec. 5.4.1).

	Number	Name	Default	Description
⊙	CON(19)	ICYLIN	0	Coordinate system indicator ICYLIN = 0 – Cartesian (x,y) coordinates = 1 – cylindrical (r, z) coordinates (horizontal → x → r) (vertical → y → z)

5–1.2 Boundary Conditions

	CON(20)	INPUTA	0	The number of special fixed potential values to be read in as data by POISSON/PANDIRA. INPUTA > 0 – second optional input data listing mesh points and fixed potential values must be given (see Sec. 5.4.2).
	CON(21)	NBSUP	0	LATTICE input only.
	CON(22)	NBSLO	1	LATTICE input only.
	CON(23)	NBSRT	0	LATTICE input only.
	CON(24)	NBSLF	0	LATTICE input only.
	CON(50)	IHDL	100000	The number of cycles between making a quasi-integral $H \cdot dl$ calculation around the Dirichlet boundary during POISSON iteration. Decreasing IHDL sometimes speeds the convergence, particularly for nonsymmetric "H" magnets.

5–1.3 Limits, Printed Output and TAPE35 Dump Options

	CON(30)	MAXCY		Maximum number of iteration cycles.
			100000	POISSON. (If not converging, decrease MAXCY and rerun to get a dump).
			20	PANDIRA. (If terminates before convergence, increase MAXCY and continue from current dump).
⊙	CON(31)	IPRFQ	0	An indicator for the cycle iteration print frequency for POISSON only. IPRFQ = 0 – POISSON determines frequency print. IPRFQ > 0 – Prints every IPRFQ cycles. IPFRQ must be a multiple of IVF's CON(87).

	Number	Name	Default	Description
⊙	CON(32)	IPRINT	0	An indicator for additional printout IPRINT = −1 – LATTICE input only. = 0 – no additional printout. = 1 – print the vector potential array. = 2 – print the |B| in iron regions. = 4 – print the B_x, B_y in iron regions = sum – a combination of any of the above three options (i.e. IPRINT = 7 = 1 + 2 + 4 will give all three options).
	CON(34)	INACT	-1	An indicator used in interactive POISSON/PANDIRA run to allow user interaction. INACT = −1 – no interaction. = 1 – program stops at each iteration cycle, queries the user and proceeds according to typed value: GO – continues to next iteration. IN – inquires for new CON values before proceeding to next iteration. NO – run terminates and results are written on TAPE35 and on OUTPOI/OUTPAN.
	CON(35)	NODMP	0	An indicator to write TAPE35 dump at completion of POISSON/PANDIRA run. NODMP = 1 – do not write dump. = 0 – write dump.

Number	Name	Default	Description
5-1.4 Edit Options (Field Gradients)			
CON(38)	XORG	0.0	The real part of z_o used to specify the origin in the polynomial expansion for vector potential. $A(x,y) = Re[\sum c_n(z - z_o)^n]$, the derivatives of which give the field and gradient. XORG = 0.0 for cylindrical coordinates. NOTE: For programs PRIOR to 11/10/86 if XMIN $\neq$ 0., user MUST set XORG = XMIN for correct field calculation. XMIN is a REG NAMELIST parameter, (Sec. 2.2.2), and not the XMIN of CON(54).
CON(39)	YORG	0.0	The imaginary part of z_o described in CON(38). NOTE: For programs PRIOR to 11/10/86 if YMIN $\neq$ 0., user MUST set YORG = YMIN for correct field calculation. YMIN is a REG NAMELIST parameter, (Sec. 2.2.2), and not the YMIN of CON(55).
CON(42)	KMIN	1	The mesh point limits of the region in which the fields and gradients are to be calculated and written on file OUTPOI/OUTPAN for noniron regions only. [Use CON(32) for IRON regions]. Default value writes fields and gradients at all mesh points on horizontal axis (L = 1). To get values for all geometry set LTOP to value of LMAX [KMAX, LMAX values listed as CON(3), CON(4) in files OUTLAT and OUTPOI/OUTPAN.] See example, Sec. 10.2.
CON(43)	KTOP	KMAX	
CON(44)	LMIN	1	
⊙ CON(45)	LTOP	1	

	Number	Name	Default	Description
⊙	CON(46)	ITYPE	2	A code specifying the problem symmetry. For Cartesian symmetry: ITYPE = 1 – no symmetry. ITYPE = 2 – midplane symmetry. ITYPE = 3 – elliptic aperture quadrupole. ITYPE = 4 – symmetric quadrupole. ITYPE = 5 – skew elliptic aperture quadrupole. ITYPE = 6 – symmetric "H" magnet or elliptical aperture sextupole. ITYPE = 7 – symmetric sextupole. ITYPE = 8 – elliptic aperture octupole. ITYPE = 9 – symmetric octupole. For all of the above symmetry codes, except ITYPE = 1 or = 5, field lines are perpendicular to the x-axis. For ITYPE = 5, the x-axis is a field line. For cylindrical symmetry: ITYPE = 1 – no symmetry. ITYPE = 2 – midplane symmetry. vector problems—field lines perpendicular to r-axis. scalar problems—potential(v) lines perpendicular to r-axis. ITYPE = 3 – midplane symmetry for scalar problem only. r-axis is a v=constant line. NOTE: If in doubt to the type of symmetry, use ITYPE = 1 or = 2 and set boundary conditions by CON(21)–CON(24), (see Sec. 2.2.5 and Table 3-1). For further detail on problem symmetry, consult POISSON/PANDIRA Reference Manual, (Sec. B.5.3.2).
	CON(47)	W2ND	0.125	The weight factor for the second nearest neighbors used in determining the c_n in the polynomial expansion for the vector potential $A(x,y) = Re[\sum c_n(z - z_m)^n]$.

Number	Name	Default	Description
CON(48)	ISECND	1	Indicator for use of first or second neighbors in determining the c_n above ISECND = 1 – first and second = 0 – first neighbors only. (Use this option if a problem has trouble converging.)
CON(54)	XMIN	0.0	The vertical and horizontal limits of the region in which the fields and the gradients are calculated for the computed values of (x,y) or (r,z), (not necessarily on a mesh point) and written on file OUTPOI or OUTPAN for noniron regions only. The coordinates are computed by starting from XMIN, YMIN and incrementing by DX, DY where: DX = (XMAX-XMIN)/(KTOP-1) [KTOP=CON(43)] DY = (YMAX-YMIN)/(LTOP-1) [LTOP=CON(45)] up to XMAX, YMAX (see example, Sec. 10.2).
CON(55)	XMAX	0.0	
CON(56)	YMIN	0.0	
CON(57)	YMAX	0.0	

5–1.5 Current Adjustment

CON(8)	BDES	1.0E+15	The value of the field, |B| = BDES at mesh location [KBZERO = CON(40), LBZERO = CON(41)]. If BDES ≠ 1.0E+15, the current factor, XJFACT = CON(66) will be adjusted so that |B| = BDES within a tolerance XJTOL = CON(67) (see example, Sec. 10.2).
CON(40)	KBZERO	1	The vertical and horizontal mesh coordinates specifying the location of BDES [CON(8)] for adjusting the current factor.
CON(41)	LBZERO	1	
⊙ CON(66)	XJFACT	1.0	The factor by which all currents and current densities (except current filaments) will be scaled. If BDES = CON(8) is input, then current will be adjusted as described above (see example, Sec. 10.2). XJFACT = 0. – a scalar potential problem (no current). (See example, Sec. 10.8.)
CON(67)	XJTOL	1.0E-4	The tolerance on the determination of XJFACT = CON(66) for BDES = CON(8).

	Number	Name	Default	Description
	5-1.6 Current Options			
	CON(49)	NFIL	0	The number of current filaments to be read in as data by POISSON/PANDIRA. NFIL > 0 – third optional input data listing mesh points and current filaments must be given (see Sec. 5.4.3).
	CON(70)	ICAL	0	LATTICE input only.
⊙	CON(101)	IPERM	0	Indicator for permanent magnet problem in PANDIRA only. IPERM = 0 – not permanent magnet problem. IPERM = 1 – a permanent magnet problem. The vector potential is initialized by either a current region or by current filaments [CON(49)] which the user MUST input. (See PANDIRA examples, Secs. 10.5 and 10.6.)
	5-1.7 Over Relaxation Factors			
	CON(74)	RHOPT1	1.9	See CON(75) = RHOAIR.
	CON(75)	RHOAIR	1.9	The over-relaxation factor in POISSON for air and interface points and for iron points with a constant, but finite permeability. RHOAIR = RHOPT1 – optimizes RHOAIR during iteration. RHOAIR ≠ RHOPT1 – RHOAIR not optimizes; uses value assigned.
	CON(77)	RHOFE	1.0	The over-relaxation factor in POISSON for iron points with a finite variable permeability.
	CON(78)	RHOGAM	0.08	The under-relaxation factor in POISSON for γ (= 1./permeability) for finite variable permeability.
	CON(79)	RHOXY	1.6	LATTICE input only.
	CON(80)	ISKIP	1	The number of cycles between recalculating the γ during a finite variable permeability solution.

	Number	Name	Default	Description
⊙	CON(81)	NOTE	1	LATTICE input only. For PANDIRA, NOTE must be set to 0 in LATTICE.

5-1.8 Convergence Criteria

CON(84)	EPSO	1.0E-5	LATTICE input only.
CON(85)	EPSILA	5.0E-7	The convergence criterion for the potential solution of air and interface points and for iron points with a finite, but constant permeability.
CON(86)	EPSILI	5.0E-7	The convergence criterion for the potential solution of iron points with finite variable permeability. NOTE: For problem to converge, both values printed under columns: "residual-air" "residual-iron" in file OUTPOI and the terminal must be less than EPSILA and EPSILI respectively. If printed values are near EPSILA/EPSILI and solution is not converging, increasing EPSILA/EPSILI will force program to converge, with less accuracy.
CON(87)	IVERG	10	The number of cycles between convergence test. The default value of 10 should not be altered if the option to optimize the over-relaxation factor CON(15) = RHOAIR is used.

5-1.9 Harmonic Analyses Parameters

CON(110)	NTERM	0	The number of coefficients to be obtained in the harmonic analyses of the potential. $0 \leq$ NTERM ≤ 14. See harmonic analysis examples in Secs. 10.3 and 10.9. (For complete discussion of harmonic analyses, refer to POISSON/SUPERFISH Reference Manual.)

Number	Name	Default	Description
CON(111)	NPTC	0	The number of equidistant points on the arc of a circle with its center at the origin, at which points the vector potential is to be interpolated. Fourier analysis of the vector potential at these points yields the harmonic coefficients. NPTC should be approximately equal to the number of mesh points adjacent to the arc. $0 \leq$ NPTC ≤ 101.
CON(112)	RINT		The radius of the arc of a circle at which the vector potential is to be calculated for harmonic analysis. RINT should be less than the radius to nearest singularity (pole or coil) by at least one mesh space.
CON(113)	ANGLE		The final angle, in degrees, that defines the arc of the circle with radius RINT = CON(112).
CON(114)	RNORM		The aperture radius or other normalization radius used in the harmonic analysis.
CON(115)	ANGLZ		The initial angle, in degrees, that defines the arc of the circle with radius RINT = CON(112). Both ANGLE and ANGLZ are measured from the x-axis

5-1.10 Conformal Transformation Factors

CON(37)	MAP	1	For POISSON: A parameter in the conformal transformation $\omega = z * *$MAP/[MAP*RZERO**(MAP-1)] where: RZERO = CON(125) MAP = 1 – no conformal transformation MAP $\neq$ 1 – conformal transformation with no current density adjustment if MAP = 1 in LATTICE [see CON(37) in Table 3-1].
CON(123)	TNEGC	0.0	LATTICE input only.
CON(124)	TPOSC	0.0	LATTICE input only.
CON(125)	RZERO	1.0	LATTICE input only.

5.3.3 Example of Minimum Input

The minimum data required for the execution of a POISSON/PANDIRA run consists of either a file or a terminal entry with the following three lines:

```
 0
 s
-1
```

where:

0 = NUM	–	reads dump 0 of TAPE35.
s	–	no changes in the CON array. (If any CON variables need to be input, they are entered here.)
−1 = NUM	–	terminates run after convergence.

5.4 Optional Input

There are three sets of optional input data to the POISSON/PANDIRA programs. The particular optional input is designated by a CON array variable and follows the CON array entry in the following order:

1. Permeability/permittivity function input	if CON(18) $\neq$ 0.
2. Fixed potential input	if CON(20) > 0.
3. Current filament input	if CON(49) > 0.

These inputs are discussed in detail in the sections below.

5.4.1 Permeability/Permittivity Function Input

The first optional data are designated by NPERM = CON(18) $\neq$ 0 and allow the permeability functions in the iron regions to be defined in several different ways and/or to input constant permittivity values. This input also permits, in PANDIRA only, the specification of anisotropic materials and the definition of a straight line B(H) for permanent magnet problems. These various features are described in detail.

A. The internal table may be used with up to a maximum of four different stacking factors, to define four different regions. Also, this option defines up to four different γ (= 1.0/(relative μ)) for problems that use μ-finite-but-constant values or up to four different permittivity (epsilon = relative) for dielectric materials. These features are exercised by:

1. Setting the two variables in CON array:

   CON(18) = NPERM = – no. of stacking factors or
   = – no. of fixed γ or ϵ values to be read in

   CON(6) = MODE = 0.

2. Entering |NPERM| lines specifying the values for the three variables:

   MATER STACK FXGAM,

   where:

   MATER – The material code to which the input permeability function applies (MATER ≡ MAT variable in REG NAMELIST—TABLE 2-1). $2 \leq$ MATER ≤ 5.

   STACK – Stacking (fill) factor for this material.

   FXGAM – Fixed γ for μ-finite but constant value or fixed ϵ for dielectric material.

This is FREE routine input entry so uses blank or commas as delimiters and an "s" if less than three values are entered.

EXAMPLE A—Stacking and γ Input

```
Line
No.
 1    *18 -2 *6 0 ... s
 2    3 0.8 s
 3    5 1.0 .004
```

where:

Line 1: CON(6) = MODE = 0
CON(18) = NPERM = −2 – indicating | −2| = 2 stacking and/or fixed γ/*epsilon* will be input.

Line 2: specifies that regions with material code 3 will use the internal permeability table with a stacking factor of 0.8. (Note an "s" is used to indicate end of line entry since the third input value is not given.)

Line 3: specifies that regions with material code 5 will use a fixed gamma value = .004 for their permeability/permittivity function.

B. POISSON/PANDIRA allow up to three different permeability tables to be read in for use with different iron regions. (For examples, see Secs. 10.3, 10.4, 10.10, 10.11.) In addition, PANDIRA permits specification of anisotropic materials. To exercise these options:

1. Set the two variables in the CON array:

   CON(18) = NPERM – no. of input permeability tables ($1 \leq$ NPERM ≤ 3)
   CON(6) = MODE – must be set to zero.

2. Enter values for the three variables:

   MATER STACK MTYPE

   where:

MATER – The material code to which the input permeability tables apply (MATER ≡ MAT variable in REG NAMELIST—TABLE 2-1). $3 \leq$ MATER ≤ 5.

STACK – The stacking (fill) factor for this material.

MTYPE – The type of input table values.

MTYPE = ±1 – input table values are (B,γ).
= ±2 – input table values are (B,μ).
= ±3 – input table values are (B,H).

MTYPE > 0 – SKIP to data group 4.
MTYPE < 0 – anisotropic material, input data group 3.

3. Input values into the five variables which define the anisotropic properties:
   ANISO GAMPER XOA YOA PHAXIS,
   where:

   ANISO – The direction angle (in degrees) of the "easy axis" relative to the horizontal axis in the counterclockwise direction (default = 0.0).
   GAMPER – The $\gamma(= 1.0/\mu)$ perpendicular to the "easy axis" (default = 1.0).
   XOA – These three variables are used when the "easy axis" cannot
   XOB be defined by ANISO. (XOA, XOB) is the center of a circular
   PHAXIS arc and PHAXIS is the angle between the radius vector and "easy axis." (For more detail consult Reference Manual.) (Default XOA = XOB = .90; PHAXIS = 1111.0.)

4. Input table values, a maximum of 50 entries per table, which define the permeability function for regions with this material code (MATER). These values are entered a pair per line, according to MTYPE specification, with a "c" (a FREE symbol to designate a last entry and to "count" the number of entries) following the last pair. In order to ensure convergence, the input values should generate a smooth function of γ(B).

   Entries 2-5 are repeated |NPERM| times.

Below is an example of input permeability tables. The lines listed on the left are not part of the example, but are used as pointers for the explanations that follow.

EXAMPLE B—Input Permeability Tables

```
Line
No.
 1    *18 2 *6 0 ... s
 2    3  0.95  1
 3    0.0              0.000250
 4    9.0E+3           0.000260
       :                :
```

```
5    7.5E+4          0.714258 c
6    4 1.0 2
7    0.0          1236.0
8    2.4E+3       1232.0
⋮     ⋮              ⋮
     2.25E+4        11.3 c
```

where:

Line 1: NPERM = CON(18) = 2 – two tables to read in.
MODE = CON(6) = 0
s = – end of CON array entry.

Line 2: MATER = 3 – the material code to which the first read in permeability table applies.
STACK = 0.95 – stacking factor for the first read in table.
MTYPE = 1 – table will be given as (B, γ).

Line 3–5: lists the (B, γ) values, B in gauss. The "c" designates last entry of this table.

Line 6: MATER = 4 – the material code to which 2nd input permeability table applies.
STACK = 1.0 – no stacking.
MTYPE = 2 – 2nd table will be given as (B, μ).

Line 7– on: Similar to lines 3–6, except now input (B, μ) pairs.

C. PANDIRA permits the specification of anisotropic materials and the definition of a straight line $B(H)$ input for solution of permanent magnet problems. (For examples, see Secs. 10.5 and 10.6.) To exercise this option:

1. Set the three variables in the CON array:

   CON(18) = NPERM – no. of straight line $B(H)$ input. (0 < NPERM ≤ 6).
   CON(6) = MODE = 0
   CON(101) = IPERM = 1 – indicator for permanent magnet problems.

   (Note: CON(81) = 0 in LATTICE for PANDIRA run).

2. Enter values for the three variables:

   MATER STACK MTYPE

   where:

   MATER – The material code to which the input straight line $B(H)$ (MATER = MAT variable in REG NAMELIST—TABLE 2 -1). 6 ≤ MATER ≤ 11.

   STACK – The stacking (fill) factor for this material.

MTYPE – Indicator for permanent magnet. It MUST be negative (numerical value insignificant).

3. Input values into the five variables which define the anisotropic properties:
   ANISO GAMPER XOA YOA PHAXIS,
   where:
   ANISO – The direction angle (in degrees) of the "easy axis" relative to the horizontal axis in the counterclockwise direction (default = 0.0).
   GAMPER – The $\gamma(= 1.0/\mu)$ perpendicular to the "easy axis" (default = 1.0).
   XOA – These three variables are used when the "easy axis" cannot be
   XOB defined by ANISO. (XOA, XOB) is the center of a circular arc and
   PHAXIS PHAXIS is the angle between the radius vector and "easy axis" (For more detail consult Reference Manual.) (default XOB = XOB = 0.90; PHAXIS = 1111.0).
4. Input values into the two variables which define straight line $B(H)$ function for this material

   HCEPT BCEPT

   where:

   HCEPT – the H-axis intercept in the second quadrant given in oersted; HCEPT is a negative number with the value of "H-coercive."
   BCEPT – the B-axis intercept is gauss; BCEPT is the "residual induction."

EXAMPLE C—Permanent Magnet Input

```
Line
No.
 1    *18 2 *6 0 *101 1  ...  *
 2       6       1.0            -1
 3      90.      1.0            *
 4   -9000.   9000.
 5       8       1.0            -1
 6     135.      1.0            *
 7   -9000.   9000.
```

where:

Line 1:	NPERM =	CON(18) =	2	–	two permeability functions to read in.
	MODE =	CON(6) =	0		
	IPERM =	CON(101) =	1	–	permanent magnet.
		* =		–	end of CON array.
Line 2:		MATER =	6	–	the material code to which the first read in permeability function applies. Since 6 ≤ MATER ≤ 11, designates permanent magnetic material.
		STACK =	1.0	–	no stacking factor.

An example of this option:

```
*20 3 ... s
[permeability function input, if CON(18) = NPERM ≠ 0, follows]
1  3  1.0E+5
1  9  2.0E+5
4  2  1.5E+5
```

where:

CON(20) = INPUTA = 3 – three fixed potential values are input.
s – designates end of CON array entry.

The list of mesh points (K, L) and their fixed potential values follow after CON array or if NPERM ≠ 0, after permeability function entries.

5.4.3 Current Filament Input

Current filaments may be input to POISSON/PANDIRA by setting the CON array variable NFIL = CON(49) = no. of current filaments. This is the last optional input and precedes the second NUM entry line. (See Sec. 5.3.1.) A list of NFIL lines for the following three variables per line is entered:

K L CFIL

where:

K, L – are the mesh numbers for the horizontal (K) and vertical (L) coordinates.
CFIL – the current in amperes at the (K,L) mesh coordinate.

If the (K,L) for the particular (X,Y) are unknown, executing LATTICE with CON(32) = IPRINT = −1, will list in output file, OUTLAT, the complete list of the coordinates with their corresponding (K,L) mesh.

An example of this option:

```
*49 2 ... s
[permeability function input, if CON(18) = NPERM ≠ 0 and/or fixed
potential input, if CON(20) = INPUTA ≠ 0]
10  25    100.0
16  18   -100.0
```

where:

CON(49) = NFIL = 2 – two current filaments are input
s – designates end of CON array entry.

The list of mesh points (K,L) and the current, in amperes, are the last entries to POISSON/PANDIRA just prior to the second NUM line entry (Sec. 5.3.1).

5.5 PANDIRA Input Summary

As has been pointed out, PANDIRA input is similar to POISSON except for a few changes which are summarized below.

A. PANDIRA runs for all types of POISSON problems, the user needs only to set in:

LATTICE - CON(81) = C

B. PANDIRA runs for permanent magnet problems, the user needs to set in:

1. AUTOMESH – a current line region. The location of line and the value of current are immaterial.
2. LATTICE - CON(81) = 0
3. PANDIRA -

CON array variables:

CON(6) = 0 – must be set to zero.
CON(101) = 1 – indicator for permanent magnet problem.
CON(18) = no. of straight line B(H) input

Followed by B(H) parameters that are defined in Sec. 5.4.1–C.

For example of PANDIRA permanent magnet runs, see Secs. 10.5 and 10.6.

5.6 Output from POISSON/PANDIRA

POISSON/PANDIRA generate two types of output files—TAPE 35 with dump numbers > 0, and OUTPOI (from POISSON)/OUTPAN (from PANDIRA). In addition, if run interactively, the programs print out messages and iteration cycle data at the terminal.

In a successful run POISSON/PANDIRA output:

1. To OUTPOI/OUTPAN and to the terminal—
   - no error messages
   - the message:

     ```
     solution converged in (--) iterations
     elapsed time = (--) sec.
     dump number 1 has been written on tape35.
     ```

2. To TAPE35, dump no. (no. $\geq$ 1) binary information that is needed to execute TEKPLOT to plot flux lines or to continue a POISSON/PANDIRA run. (Note: even though POISSON and PANDIRA generate files of the same names, these files cannot be used interchangeably by the other program).

OUTPOI/OUTPAN are ASCII files and may be printed or examined with any editor. Most of the information listed in these files, including the error mesages, if any, are self-explanatory. Both OUTPOI and OUTPAN have similar default output listings which contain:

- a complete list of the CON array variable names and values. Those elements that have been changed in input to either LATTICE or POISSON/PANDIRA are flagged by "CON" preceding the element number.
- a list of all permeability tables, including the internal table if it is used or not, stacking factors, and/or other permeability/permittivity functions that were input.
- the iteration cycle data as printed on the terminal.
- a table listing (see Fig. 5-1 or Fig. 5-2) for mesh points on axis (l=1) for noniron regions:

  k, l – mesh point coordinates
  a(vector) – vector potentials
  x, y – the physical coordinates (in user input units)
  bx, by – field components, B_x, B_y
  bt – total field = $|B| = \sqrt{B_x^2 + B_y^2}$
  dby/dy, dby/dx – the field gradients, $\partial B_y/\partial y$, $\partial B_y/\partial x$
  afit – the difference of the solved vector potential, a(vector)-column3, and the vector potential

computed from the least square fit polynomial that is used to obtain the field components, B_x and B_y and the gradients, $\partial B_y/\partial y$ and $\partial B_y/\partial x$. afit values show the accuracy of the fit which theoretically should be = 0.

The user has options, through the CON array, to specify additional output to OUTPOI/OUTPAN. (See example in Chapter 10.)

5.7 Error Messages in POISSON/PANDIRA

POISSON/PANDIRA list all their error messages, with recommended correctional instructions, to the output file OUTPOI/OUTPAN and some to the terminal, if the run is interactive. The majority of these error messages are identical to both programs since these programs use some of the same or similar subroutines. These messages are listed below with the notation "word 1"/"word 2" where "word 1" refers to POISSON and "word 2" refers to PANDIRA.

5.7.1 Message Containing "Input Data Error"

1. `--- INPUT DATA ERROR ---`
   These types of error messages are printed from the subroutine FREE whenever the input data is not in the special free-format entry. These messages which are listed both in the file, OUTPOI/OUTPAN, and printed at the terminal are self-explanatory. A print of the input line that is in error is also listed. If run is interactive, FREE prints "*retype line*" and gives the user the opportunity to enter the line that was in error. If noninteractive, POISSON/PANDIRA immediately abort. The user should correct the specific error in the input file and rerun.

5.7.2 Message Starting with "Error Exit"

1. `--- ERROR EXIT--- (KMAX + 2) * (LMAX + 2) = (--) IS GREATER THAN PROGRAM DIMENSIONS OF (--) ...`
   The total number of mesh points have exceeded the maximum value dimensioned. Cut mesh size or increase parameter MXDIM and recompile as directed. Message from subroutine RDUMP/PDUMP.

2. `--- ERROR EXIT--- NWMAX EXCEEDS PROGRAM DIMENSION OF (--) ...`
   NWMAX has exceeded MAXDIM/2. Cut mesh size or increase parameter MXDIM and recompile as directed. Message from subroutine RDUMP/PDUMP.

3. `--- ERROR EXIT--- THE MESH HAS NEGATIVE AND/OR ZERO AREA TRIANGLE ...`
   This error message was issued in LATTICE also. Follow recommended procedure. Message from subroutine RDUMP/PDUMP.

4. `--- ERROR EXIT--- MATERIAL CODE .GT.5...`

The user has erroneously input stacking factor [NPERM = CON(18) = neg. value] with material code greater than six. Check Sec. 5.4.1 for limitations on MATER. Message from subroutine TABLE/PTABLE.

5.7.3 Message Ending with "Error Exit"

All these messages come from the common subroutine TABIN.

1. NAME OF MATERIAL IS LESS THAN OR EQUAL TO 1, OR GREATER THAN 11 ... --- ERROR EXIT ---
   The material code, MATER must be input as $2 \leq$ MATER ≤ 11. Check Sec. 5.4.1.

2. THE NUMBER OF INPUT TABLES IS GREATER THAN FOUR --- ERROR EXIT ---
   In addition to the internal table, three or more tables may be input for a total maximum of four tables. User has tried to input more than three.

3. GAMMA = H/B, AND B = 0.0 ... --- ERROR EXIT ---
   The user has input an H vs. B table with a B value of 0.0. Correct element of the given table and rerun.

4. YOU HAVE EXCEEDED THE MAXIMUM DIMENSIONS ALLOWED FOR THE GAMMA VS B TABLES
   ---ERROR EXIT---
   The user has exceeded the maximum of 50 entries per table for the given table. Eliminate one or more of the entries for the table and rerun.

5.7.4 Message with "Data Error"

1. --- DATA ERROR --- ITYPE = CON(46) = (--) CANNOT BE ZERO OR NEG ...
   Follow recommended procedure. (See Table 5-1.)

5.8 Error Messages in PANDIRA Only

In addition to the common error messages in both POISSON and PANDIRA, given in Sec. 5.6, PANDIRA has the following additional error messages.

5.8.1 Messages Starting with "Error Exit"

1. --- ERROR EXIT --- NOTE = CON(81) = 1
   PANDIRA requires CON(81) to be set to 0 in LATTICE. Rerun LATTICE with this correction. Message from main program.

2. --- ERROR EXIT --- NO. INTERFACE CURRENT POINTS .GT. DIMENSIONED ARRAY OF (--) ...
   Present dimension = 400. Increase parameter value INMX in PANDIRA only and recompile. Error message from subroutine RHANDS.

3. --- ERROR EXIT --- SUM OF INTER. AND IRON A'S IS ZERO
   The program has found that the sum of the vector potentials for the interface and iron points is zero for this iteration. This will result in division by zero, so the run is aborted. Error message from subroutine RHANDS. To correct: try cutting mesh size, check that the iron region is a closed region in input file to AUTOMESH or if running permanent magnet problem check that have included a current line region (see example 10.5 and 10.6).

4. --- ERROR EXIT --- NAMAX EXCEEDS PROGRAM DIMENSIONS OF (--) ...
   NAMAX has exceeded MAXDIM/2. Cut down on the number of mesh points or increase parameter MXDIM and recompile as directed. Message from subroutine SWIND.

5. --- ERROR EXIT --- NROW = MINO (KMAX,LMAX) = (--) EXCEEDS MATRIX DIMENSIONS OF (--)
   The storage needed for the matrix inversion has exceeded the dimensioned arrays. Cut down the mesh points or increase parameter IMX and recompile POILIB and PANDIRA as directed. Message from subroutine TRIBES.

5.9 Example of POISSON/PANDIRA Runs

Sections 5.9.1 and 5.9.2 list the execution of POISSON and PANDIRA on the CRAY computer for the H-shaped Dipole Magnet. The solution output of OUTPOI/OUTPAN giving the table listing of the calculated field components and their gradients on axis are shown in Fig. 5–1 for POISSON run and in Fig. 5–2 for PANDIRA. As can be seen from these two figures, there is excellent agreement of the vector potentials [A(vector)] and the B fields in the region of interest—near the origin—between these runs.

The user only types the underlined quantities for the execution of the programs as given below.

5.9.1 Executing POISSON Run---H-Shaped Dipole Magnet

We use the LATTICE generated dump 0 of TAPE3 and only the "standard" input to execute POISSON. We choose, by typing tty, to input data from the terminal. We then enter:

```
          0     --  to read dump number (NUM) = 0 on TAPE35 generated by LATTICE
 *6 0 *46 6     --  to change two of the CON array variables, CON(6) and CON(46),
                    specifying use of μ = finite with use of internal B² vs. γ and
                    specifying a symmetrical H-magnet, respectively.
         -1     --  reads NUM < 0 to terminate run.
```

```
poisson
 ?type ''tty'' or input file name
? tty
 ?type input value for dump num
? 0
```

```
beginning of poisson execution from dump number  0
prob. name = h-magnet test, uniform mesh 4/23/85
?type input values for con(?)
*6 0 *46 6 s
elapsed time  =  1.0 sec

0  cycle      amin         amax     residual-air    eta-air  rhoair   xjfact
                           gmax     residual-iron   eta-iron rhofe
0       0  0.0000e+00  0.0000e+00   1.0000e+00    1.0000   1.0000   1.000
                       4.0000e-03   1.0000e+00    1.0000   1.0000
0      50                      rhoair optimized    0.9903   1.9558   lambda = 9.9976e-01
0      50 -4.7296e+04  0.0000e+00   5.7349e-02    0.9903   1.9558   1.0000
                       3.9026e-03   3.4407e-02    1.0039   1.0000
0     100                      rhoair optimized    0.9717   1.9578   lambda = 9.9978e-01
0     100 -1.0012e+05  0.0000e+00   2.5360e-02    0.9717   1.9578   1.0000
                       2.0634e-02   2.0906e-02    0.9834   1.0000
0     200                      rhoair optimized    0.8960   1.9478   lambda = 1.0003e+00
0     200 -1.1958e+05  0.0000e+00   1.8887e-04    0.8960   1.9478   1.0000
                       4.6352e-02   7.9336e-05    0.8900   1.0000
0     370 -1.1939e+05  0.0000e+00   3.7301e-07    0.9367   1.9478   1.0000
                       4.6392e-02   1.6135e-07    0.9328   1.0000

solution converged in 370 iterations
elapsed time = 3.8 sec.
dump number 1 has been written on tape35
 ?type input value for dump num
stop
poisson ctss time 4.427 seconds
cpu= 2.831   i/o= 1/081    mem= .514

all done
```

5.0.2 PANDIRA Run—H-shaped Dipole Magnet

PANDIRA run must have CON(81) = NOTE = 0. Since this CON variable is one that must be changed in LATTICE, we rerun LATTICE, and input this value as shown.

```
lattice
 ?type input file name
?tape73
beginning of lattice execution
dump 0 will be set up for poisson
 h-magnet test, uniform mesh    4/23/85
 ?type input value for con(?)
? *81 0 s
```

```
 elapsed time =    0.5 sec.
0iteration converged
 elapsed time =    0.7 sec.
 generation completed
dump number 0 has been written on tape35.
stop
lattice ctss time      1.029      seconds
cpu=     .593    i/o=     .341    mem=       .094

all done
```

To execute PANDIRA, we use the same input as in POISSON and again we choose, by typing tty, to input data from the terminal. We then enter:

0	–	to read dump number (NUM) = 0 on TAPE35 generated by LATTICE.
*6 0 *46 6	–	to change two of the CON array variables, CON(6) and CON(46), specifying use of μ = finite with use of internal B^2 *vs.* γ and specifying a symmetrical H-magnet, respectively.
−1	–	reads NUM < 0 to terminate run.

```
pandira
 ?type ''tty'' or input file name
tty
?type input value for dump num
? 0
beginning of pandira execution from dump number  0
prob. name = h-magnet test,uniform mesh  4/23/85
?type input values for con(?)
*6 0 *46 6 s
 elapsed time  =      1.0 sec.
cycl      amin          amax         residual
                         bmax          residual-fe    eta-fe
  0   0.0000e+00    0.0000e+00
                    0.0000e+00    1.0000e+00    1.0000
solution time =      2.1 sec.
  1  -1.1872e+05    0.0000e+00
                    5.1937e+04    2.777e-02     1.0000
solution time =      2.1 sec.
  2  -1.2260e+05    0.0000e+00
                    2.7844e+04    2.490e-02     0.9167
solution time =      2.0 sec.
  3  -1.2063e+05    0.0000e+00
                    2.4503e+04    1.801e-02     0.7191
```

```
solution time =      2.0 sec.
    4  -1.1968e+05  0.0000e+00
                     2.2385e+04   7.631e-03   0.4207
solution time =      2.1 sec.
    5  -1.1943e+05  0.0000e+00
                     2.2251e+04   2.345e-03   0.3068
solution time =      2.0 sec.
    6  -1.1939e+05  0.0000e+00
                     2.2243e+04   5.235e-04   0.2232
solution time =      2.0 sec.
    7  -1.1939e+05  0.0000e+00
                     2.2244e+04   3.234e-05   0.0618
solution time =      2.0 sec.
    8  -1.1939e+05  0.0000e+00
                     2.2244e+04   3.381e-06   0.1045
solution time =      2.0 sec.
    9  -1.1939e+05  0.0000e+00
                     2.2244e+04   2.773e-10   0.0001
  solution converged in      9 iterations
  elapsed time = 21.0 sec.
  dump number 1 has been written on tape35.
  ?type input value for dump num
? -1
stop
pandira      ctss time      22.518       seconds
cpu=     5.154     i/o=    12.965    mem=      4.399
  all done
```

```
 solution converged in  370 iterations

 elapsed time  =    4.2 sec.

 dump number  1 has been written on tape35.
1least squares edit of problem , cycle   370

'h' mag  symmetry type
stored energy =  1.4219e+03 joules / meter o   adian

 xjfact=  1.000000                          bx       by       bt       dby/dy       dby/dx
 k l   a(vector)      x        y          (gauss) (gauss)  (gauss)  (gauss/cm)   (gauss/cm)    afit
 1 1  0.000000e+00  0.00000 0.00000   0.000 15212.250 15212.250 0.0000e+00  0.0000e+00  5.3e-04
 2 1 -6.829781e+03  0.44898 0.00000   0.000 15210.825 15210.825 0.0000e+00 -6.4635e+00 -3.3e-03
 3 1 -1.365822e+04  0.89796 0.00000   0.000 15206.195 15206.195 0.0000e+00 -1.4658e+01  3.7e-03
 4 1 -2.048366e+04  1.34694 0.00000   0.000 15197.056 15197.056 0.0000e+00 -2.7097e+01  5.5e-03
 5 1 -2.730348e+04  1.79592 0.00000   0.000 15180.665 15180.665 0.0000e+00 -4.7887e+01  6.5e-03
 6 1 -3.411342e+04  2.24490 0.00000   0.000 15151.761 15151.761 0.0000e+00 -8.4536e+01  7.8e-03
 7 1 -4.090583e+04  2.69388 0.00000   0.000 15100.434 15100.434 0.0000e+00 -1.5084e+02  9.2e-03
 8 1 -4.766697e+04  3.14286 0.00000   0.000 15008.404 15008.404 0.0000e+00 -2.7096e+02  1.1e-02
 9 1 -5.437198e+04  3.59184 0.00000   0.000 14843.504 14843.504 0.0000e+00 -4.8257e+02  2.4e-02
10 1 -6.097737e+04  4.04082 0.00000   0.000 14554.675 14554.675 0.0000e+00 -8.2909e+02  1.1e-01
11 1 -6.741291e+04  4.48980 0.00000   0.000 14077.744 14077.744 0.0000e+00 -1.3169e+03  5.1e-01
12 1 -7.358221e+04  4.93878 0.00000   0.000 13369.124 13369.124 0.0000e+00 -1.9577e+03  1.8e+00
13 1 -7.938901e+04  5.38776 0.00000   0.000 12435.718 12435.718 0.0000e+00 -2.1782e+03 -7.6e+00
14 1 -8.657337e+04  6.00000 0.00000   0.000 10893.018 10893.018 0.0000e+00 -2.8856e+03  6.7e+00
15 1 -9.114731e+04  6.44737 0.00000   0.000  9613.098  9613.098 0.0000e+00 -2.8434e+03  1.6e+01
16 1 -9.518666e+04  6.89474 0.00000   0.000  8416.533  8416.533 0.0000e+00 -2.5622e+03  5.1e+00
17 1 -9.871101e+04  7.34211 0.00000   0.000  7354.552  7354.552 0.0000e+00 -2.2169e+03  1.9e+00
18 1 -1.017933e+05  7.78947 0.00000   0.000  6439.740  6439.740 0.0000e+00 -1.8967e+03  9.3e-01
19 1 -1.044947e+05  8.23684 0.00000   0.000  5665.367  5665.367 0.0000e+00 -1.6278e+03  5.0e-01
20 1  1.068700e+05  8.68421 0.00000   0.000  4978.523  4978.523 0.0000e+00 -1.4113e+03  2.8e-01
21 1 -1.089623e+05  9.13158 0.00000   0.000  4387.692  4387.692 0.0000e+00  1.2403e+03  1.6e-01
22 1 -1.108059e+05  9.57895 0.00000   0.000  3864.280  3864.280 0.0000e+00 -1.1065e+03  9.6e-02
23 1 -1.124277e+05 10.02632 0.00000   0.000  3393.820  3393.820 0.0000e+00 -1.0022e+03  5.6e-02
24 1 -1.138487e+05 10.47368 0.00000   0.000  2964.550  2964.550 0.0000e+00 -9.2096e+02  3.0e-02
25 1 -1.150860e+05 10.92105 0.00000   0.000  2567.346  2567.346 0.0000e+00 -8.5778e+02  1.3e-02
26 1 -1.161495e+05 11.36842 0.00000   0.000  2195.081  2195.081 0.0000e+00 -8.0868e+02  9.6e-04
27 1 -1.170520e+05 11.81579 0.00000   0.000  1842.181  1842.181 0.0000e+00 -7.7062e+02  9.5e-03
28 1 -1.178000e+05 12.26316 0.00000   0.000  1504.255  1504.255 0.0000e+00 -7.4126e+02  2.0e-02
29 1 -1.183996e+05 12.71053 0.00000   0.000  1177.824  1177.824 0.0000e+00 -7.1982e+02  3.4e-02
30 1  1.188662e+05 13.15789 0.00000   0.000   860.097   860.097 0.0000e+00 -7.0192e+02  5.7e-02
31 1  1.191702e+05 13.60526 0.00000   0.000   548.874   548.874 0.0000e+00 -6.8934e+02  9.7e-02
32 1  1.193473e+05 14.05263 0.00000   0.000   243.246   243.246 0.0000e+00 -6.7864e+02  2.4e-01
33 1  1.193889e+05 14.50000 0.00000   0.000    59.749    59.749 0.0000e+00 -6.6709e+02  1.5e+00
34 1  1.193575e+05 15.00000 0.00000   0.000    60.019    60.019 0.0000e+00  0.0000e+00  1.0e+00
```

Fig. 5 1: A section of output from the file OUTPOI for the H-shaped magnet problem of 4/25/85, cycle = 370.

```
least squares edit of problem , cycle     9

'h' mag  symmetry type
stored energy =  1.4249e+03 joules / meter or radian

 xjfact=  1.000000                          bx        by        bt        dby/dy       dby/dx
 k  l     a(vector)     x         y        (gauss)   (gauss)   (gauss)   (gauss/cm)   (gauss/cm)    sfit
 1  1  0.000000e+00  0.00000  0.00000     0.000  15212.137  15212.137  0.0000e+00   0.0000e+00  4.6e-04
 2  1 -6.829730e+03  0.44898  0.00000     0.000  15210.712  15210.712  0.0000e+00  -6.4653e+00 -2.5e-03
 3  1 -1.365812e+04  0.89796  0.00000     0.000  15206.083  15206.083  0.0000e+00  -1.4662e+01  4.9e-03
 4  1 -2.048350e+04  1.34694  0.00000     0.000  15196.946  15196.946  0.0000e+00  -2.7105e+01  7.1e-03
 5  1  2.730328e+04  1.79592  0.00000     0.000  15180.557  15180.557  0.0000e+00  -4.7897e+01  8.5e-03
 6  1 -3.411317e+04  2.24490  0.00000     0.000  15151.654  15151.654  0.0000e+00  -8.4547e+01  1.0e-02
 7  1 -4.090553e+04  2.69388  0.00000     0.000  15100.328  15100.328  0.0000e+00  -1.5085e+02  1.2e-02
 8  1 -4.766683e+04  3.14286  0.00000     0.000  15008.300  15008.300  0.0000e+00  -2.7097e+02  1.4e-02
 9  1 -5.437159e+04  3.59184  0.00000     0.000  14843.405  14843.405  0.0000e+00  -4.8258e+02  2.7e-02
10  1 -6.097693e+04  4.04082  0.00000     0.000  14554.583  14554.583  0.0000e+00  -8.2909e+02  1.1e-01
11  1 -6.741244e+04  4.48980  0.00000     0.000  14077.663  14077.663  0.0000e+00  -1.3169e+03  5.2e-01
12  1 -7.358170e+04  4.93878  0.00000     0.000  13367.055  13367.055  0.0000e+00  -1.8566e+03  1.7e+00
13  1  7.938847e+04  5.38776  0.00000     0.000  12443.430  12443.430  0.0000e+00  -2.2157e+03 -7.4e+00
14  1 -8.657280e+04  6.00000  0.00000     0.000  10907.134  10907.134  0.0000e+00  -2.5916e+03 -3.6e+01
15  1  9.114671e+04  6.44737  0.00000     0.000   9041.105   9041.105  0.0000e+00  -2.4917e+03 -2.5e+01
16  1  9.510595e+04  6.89474  0.00000     0.000   8444.068   8444.068  0.0000e+00  -2.2768e+03 -3.5e+01
17  1 -9.871128e+04  7.34211  0.00000     0.000   7374.312   7374.312  0.0000e+00  -1.9080e+03 -3.8e+01
18  1 -1.017926e+05  7.78917  0.00000     0.000   6455.361   6455.361  0.0000e+00  -1.5776e+03 -4.0e+01
19  1 -1.044940e+05  8.23684  0.00000     0.000   5668.198   5668.198  0.0000e+00  -1.3034e+03 -4.0e+01
20  1  1.068693e+05  8.68421  0.00000     0.000   4989.374   4989.374  0.0000e+00  -1.0837e+03 -4.0e+01
21  1 -1.089616e+05  9.13158  0.00000     0.000   4396.981   4396.981  0.0000e+00  -9.1060e+02 -4.1e+01
22  1 -1.108052e+05  9.57895  0.00000     0.000   3872.570   3872.570  0.0000e+00  -7.7529e+02 -4.1e+01
23  1  1.124270e+05 10.02632  0.00000     0.000   3401.283   3401.283  0.0000e+00  -6.6905e+02 -4.1e+01
24  1 -1.138479e+05 10.47368  0.00000     0.000   2971.404   2971.404  0.0000e+00  -5.8774e+02 -4.1e+01
25  1 -1.150843e+05 10.92105  0.00000     0.000   2573.778   2573.778  0.0000e+00  -5.2380e+02 -4.1e+01
26  1  1.161488e+05 11.36842  0.00000     0.000   2201.263   2201.263  0.0000e+00  -4.7399e+02 -4.1e+01
27  1  1.170512e+05 11.81579  0.00000     0.000   1848.286   1848.286  0.0000e+00  -4.3520e+02 -4.1e+01
28  1  1.177902e+05 12.26316  0.00000     0.000   1510.492   1510.492  0.0000e+00  -4.0497e+02 -4.1e+01
29  1  1.183988e+05 12.71053  0.00000     0.000   1184.500   1184.500  0.0000e+00  -3.8128e+02 -4.1e+01
30  1  1.188644e+05 13.15789  0.00000     0.000    867.827    867.827  0.0000e+00  -3.6208e+02 -4.1e+01
31  1  1.191604e+05 13.60526  0.00000     0.000    559.635    559.635  0.0000e+00  -3.4325e+02 -4.0e+01
32  1  1.193405e+05 14.05263  0.00000     0.000    265.806    265.806  0.0000e+00  -2.9804e+02 -3.9e+01
33  1  1.193881e+05 14.50000  0.00000     0.000     34.285     34.285  0.0000e+00   1.8571e+02  2.0e+01
34  1  1.193567e+05 15.00000  0.00000     0.000     39.444     39.444  0.0000e+00   2.1618e+01  2.1e+00
```

Fig. 5-2: A section of output from the file OUTPAN for the H-shaped magnet problem of 4/28/88, cycle = 0.

5.9.3 Executing TEKPLOT after POISSON/PANDIRA

After a successful execution of POISSON/PANDIRA, we execute TEKPLOT again. This time we designate dump 1 of TAPE35 and 20 field lines, as shown below, to generate Fig. 5–3. (Either POISSON or PANDIRA run produces identical plots.)

```
tekplot
 ?type input data- num, itri, nphi, inap, nswxy,
?1 0 20 s
input data
num= 1    itri= 0    nphi= 20    inap= 0    nswxy= 0
plotting prob. name = h-magnet test, uniform mesh 4/23/85 cycle = 370
 ?type input data-  xmin, xmax, ymin, ymax,
? s
input data
xmin= 0.000 xmax= 22.000 ymin= 0.000 ymax= 13.000
?type go or no
? go
```

A CR after go clears the screen and plots Fig. 5–3. A second CR clears the screen and produces the prompt line.

```
?type input data- num, itri, nphi, inap, nswxy,
?   -1 s
tekplot ctss time .581 seconds
cpu= .037 i/o= .180 mem= .044

 all done
```

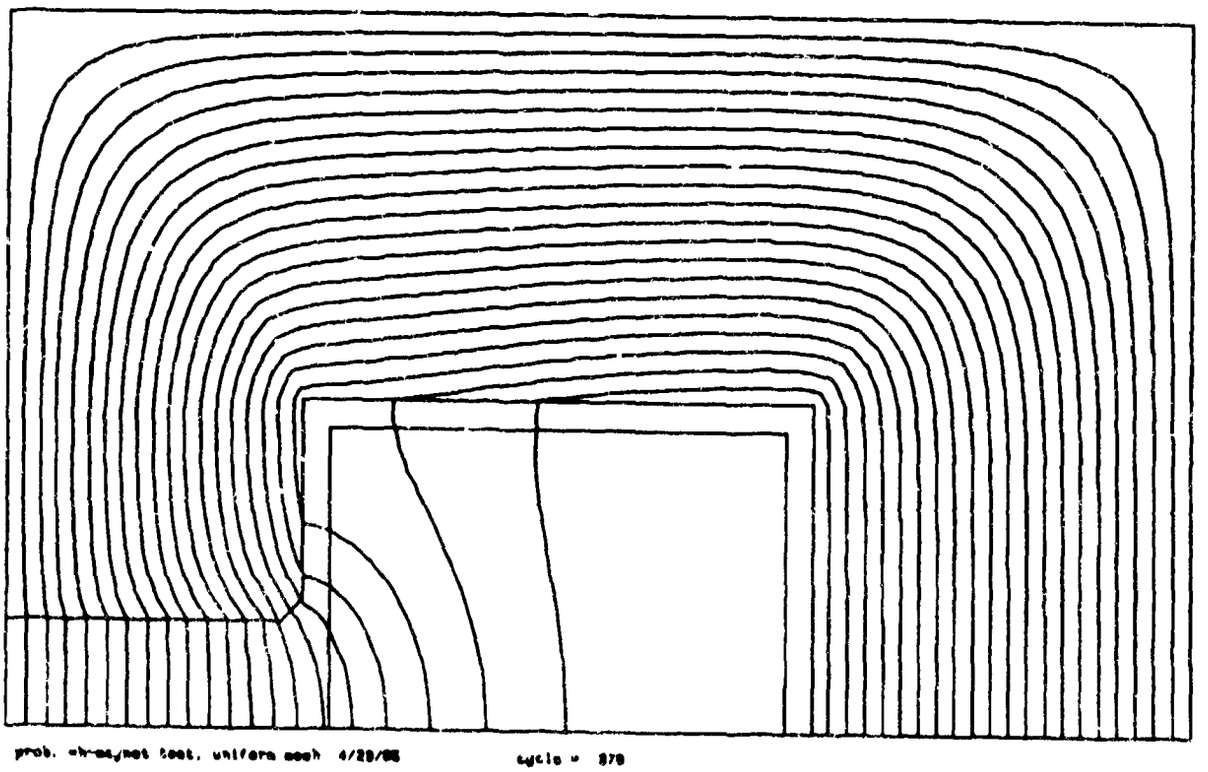


Fig. 5-3: Plot from TEKPLOT of the magnetic field lines generated by POISSON/PANDIRA for the problem "h-magnet test, uniform mesh 4/23/85".

Chapter 10

POISSON/PANDIRA Examples

Contents 10-1

10.1 Introduction

The examples in this chapter utilize various options that are available in the POISSON/ PANDIRA programs. The input to all problems are defined by both a COSMOS file for the Cray and a parallel command procedure for the VAX. Assuming that the programs specified in the procedural files are in the user's directory (if not, see Appendix A), the user should enter

for CRAY:

```
cosmos i=filename
```

where: filename – the name of COSMOS file

for VAX:

```
@FILENAME
```

where: FILENAME – the name of VAX command file

In Section 10.2, we give a line-by-line description of all files used. In subsequent sections, we will only comment on the specific options used. The numbers in the figures correspond to selected line numbers of the input file.

The examples used are actual work problems that have either originated here at LANL or were sent to us for solution by outside users.

All entries must be in lower case for CRAY and upper case for VAX.

All the field line plots are generated by executing TEKPLOT using dump 1 of TAPE35 and 20 field lines exactly as done in Sec. 4.5 – Examples of TEKPLOT Runs.

10.2 POISSON -- H-Magnet with Options

The H-shaped magnet as described in Sec. 1.3 is executed with the following options:

- calculates fields and gradients in a region on mesh points
- continues from previous POISSON dump number
- adjusts current to produce a given field
- calculates fields and gradients in a region on specifed increments of X and Y
- calculates the field components and the total fields in iron

Given the input file, HMAG, to AUTOMESH, and the COSMOS/VAX file CHMAG which specifies the above options, the user types

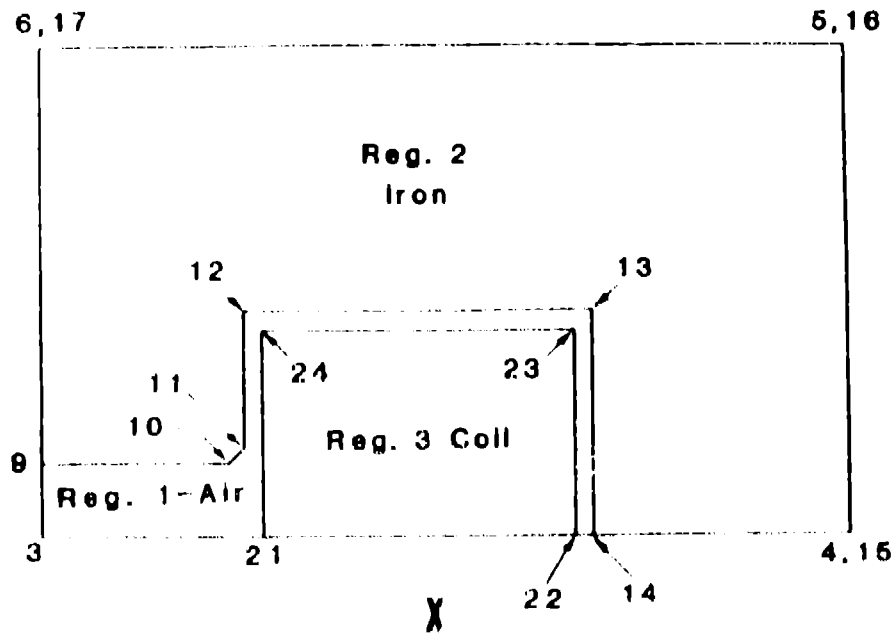
`cosmos i=chmag`	for CRAY
`@CHMAG`	for VAX

For completeness we include the HMAG file and the H-magnet geometry as given in Sec. 2.5.1.

The file HMAG describes the cross section of one-fourth of an H-shaped dipole magnet, the upper-right quadrant.

HMAG

```
Line
No.
 1   h-magnet test, uniform mesh  4/23/85
 2   $reg nreg=3,dx=.45,xmax=22.,ymax=13.,npoint=5$
 3   $po x= 0.0, y= 0.0$
 4   $po x=22.0, y= 0.0$
 5   $po x=22.0, y=13.0$
 6   $po x= 0.0, y=13.0$
 7   $po x= 0.0, y= 0.0$
 8   $reg mat=2,npoint=10$
 9   $po x= 0.0, y= 2.0$
10   $po x= 5.1, y= 2.0$
11   $po x= 5.6, y= 2.4$
12   $po x= 5.5, y= 6.0$
13   $po x=15.0, y= 6.0$
14   $po x=15.0, y= 0.0$
15   $po x=22.0, y= 0.0$
16   $po x=22.0, y=13.0$
17   $po x= 0.0, y=13.0$
18   $po x= 0.0, y= 2.0$
19   $reg mat=1,npoint=5,
20   cur=-25455.791$
21   $po x= 6.0, y= 0.$
22   $po x=14.5, y= 0.0$
23   $po x=14.5, y= 5.5$
24   $po x= 6.0, y= 5.5$
25   $po x= 6.0, y= 0.0$
```



Line 1: Title line, which starts in column 2 (POISSON/PANDIRA/MIRT Problem).
2: First REG entry: nreg = 3 – number of regions.
dx = .45 – horizontal mesh size.
xmax, ymax = 22., 13. – problem's maximum dimensions.
npoint = 5 – number of PO entries that follow.
3–6: Coordinates of points that define the region 1.
7: Same coordinates as line 3 to form a closed region.
8: REG entry for region 2: mat = 2 – implies that the material is iron.
9–18: Coordinates of points that define region 2.
19–20: REG entry for region 3: mat = 1.
cur = -25455.791 – total number of amps. (The sign specifies direction of the current; negative "out of the plane of the paper"). These two entries imply this is a coil region.
21–25: Coordinates of points that define region 3.

The procedural file, CHMAG, with explanation is given below.

Line No.	COSMOS File, chmag	VAX File, CHMAG.COM
1	*automesh	$RUN AUTOMESH
2	hmag	HMAG
3	*lattice	$RUN LATTICE
4	tape73	TAPE73
5	s	S
6	*poisson	$RUN POISSON
7	tty	TTY
8	0	0
9	*6 0 *46 6 *43 4 1 3 s	*6 0 *46 6 *43 4 1 3 S
10	1	1
11	*8 16000. *40 3 2 s	*8 16000. *40 3 2 S
12	1	1
13	*43 6 1 6 *55 2.5 *57 5. *32 6 s	*43 6 1 6 *55 2.5 *57 5. *32 6 S
14	-1	-1
15	*/	$EXIT

where:
Line 1 2: executes AUTOMESH with the input file HMAG.
3 5: executes LATTICE with TAPE73 that was generated by AUTOMESH, and no changes in the CON array (designated by "s").
6: executes POISSON.
Line 7: TTY must begin in column 1. TTY designates that data for POISSON follows.
Line 8: the 0 indicates that POISSON will read in binary data from dump 0 on TAPE35 that was produced by LATTICE.

Line 9: CON array entries. Note that for a COSMOS file, this must NOT begin in column 1.

CON (6) = 0 – use internal permeability table for MAT = 2.
CON (46) = 6 – symmetric H-magnet.
CON (43) = 4 – calculates fields and auxiliary data in a rectangular non-iron
CON (44) = 1 region on mesh coordinates (K, L) where K and L are the
CON (45) = 3 vertical and horizontal mesh points with K from 1 (the default value for CON(42)) to 4 and L from 1 to 3.
s – designates the end of CON array entries.

Completion of this run generates, in the output file OUTPOI, the values specified above as shown in Fig. 10.2–1.

Line 10: the 1 indicates that POISSON will read in dump 1 which was written after the completion of the above run, to continue with new CON parameter changes that follow.

Line 11: CON array entry specifying current adjustment.

CON (8) = 16000. – The current factor XJFACT is to be adjusted to
CON (40) = 3 produce a field of 16000 gauss at mesh point (3, 2)
CON (41) = 2
s – designates the end of CON array entries.

Completion of this run generates the output in OUTPOI shown in Fig. 10.2–2, where $|B| = bt = 15999.267$ at mesh coordinate (3, 2) and XJFACT = 1.079816. The input current is multiplied by this factor to produce the required current:

New Current = −25455.791 * 1.079816 = −27487.570 amps.

Line 12: the 1 indicates that POISSON will read in dump 1 again since we wish to go back to previous run (other than continue from above run which wrote dump 2).

Line 13: CON array entry.

CON (43) = 6 – calculates the same quantities as described in Line 9
CON (44) = 1 except it is calculated in a rectangular region of (X, Y)
CON (45) = 6 coordinates where X goes from 0.0 to 2.5 and Y goes
CON (55) = 2.5 from 0.0 to 5.0 in increments of:
CON (57) = 5.0

$$\Delta X = \frac{2.5 - 0.}{6 - 1} = .5 \qquad \Delta Y = \frac{5.0 - 0.}{6 - 1} = 1.0$$

as seen in Fig. 10.2–3, the fields are printed only in non-iron regions. Even though it was specified that Y would have values up to 5.0, the printout only goes up to 2.0 since an iron region starts at Y = 2.0.

	CON (32) = 6	- calculates and prints B_x, B_y, and $\|B\|$ in iron region
		Figure 10.2–4 shows the first part of the output for $\|B\|$. Two values of the fields are printed at each mesh point, one using upper (u) and one the lower (l) triangle in computing the field at that point. Taking an average of the two is a good approximation of the field at any given point (for more detail, see the Reference Manual)
	s	- designates the end of CON array entries.
Line 14	-1	- designates the end of the POISSON execution.
Line 15:	*/	- end of COSMOS file.
	$EXIT	- end of VAX command file

```
least squares edit of problem , cycle   370

'h' mag  symmetry type
stored energy =  1.4249e+03 joules / meter or radian

xjfact=  1 000000                                  bx       by        bt        dby/dy      dby/dx
k l    a(vector)     x        y                    (gauss) (gauss)   (gauss)   (gauss/cm)  (gauss/cm)   afit
1 1   0 000000e+00 0 00000 0 00000    0 000 15212 250 15212 250  0 0000e+00  0 0000e+00   5 3e-04
2 1  -6 829781e+03 0 44898 0 00000    0 000 15210 825 15210 825  0 0000e+00 -6 4635e+00  -3 3e-03
3 1  -1 365822e+04 0 89796 0 00000    0 000 15206 195 15206 195  0 0000e+00 -1 4658e+01   3 7e-03
4 1  -2 043365e+04 1 34694 0 00000    0 000 15197 056 15197 056  0 0000e+00 -2 7097e+01   5 5e-03
1 2   0 000000e+00 0 00000 0 39394    0 000 15213 302 15213 302  5 2584e+00  0 0000e+00   2 6e-04
2 2  -4 181175e+03 0 27484 0 39533   -1 471 15212 832 15212 832  5 5239e+00 -3 4631e+00  -1 7e-03
3 2  -1 069198e+04 0 70285 0 39718   -4 124 15210 091 15210 092  7 0908e+00 -9 6364e+00   6 7e-03
4 2  -1 746259e+04 1 14807 0 39846   -8 008 15203 981 15203 983  1 0745e+01 -1 8548e+01   3 7e-03
1 3   0 000000e+00 0 00000 0 78788    0 000 15216 242 15216 242  9 4312e+00  0 0000e+00   4 8e-04
2 3  -7 566413e+03 0 49720 0 79299   -4 960 15215 194 15215 195  1 0985e+01 -4 4442e+00  -1 2e-02
3 3  -1 445525e+04 0 95007 0 79628  -10 851 15212 132 15212 136  1 5552e+01 -9 4604e+00  -4 1e-03
4 3  -2 124061e+04 1 39619 0 79811  -19 620 15206 481 15206 494  2 4655e+01 -1 6487e+01  -1 1e-03
```

Figure 10.2–1: H-magnet output in air region on mesh coordinates (K, L). No current adjustment.

```
lleast squares edit of problem , cycle  1180

'h' mag  symmetry type
stored energy =  1 6148e+03 joules / meter or radian

xjfact= 1 079816                                   bx       by        bt        dby/dy      dby/dx
k l    a(vector)     x        y                    (gauss) (gauss)   (gauss)   (gauss/cm)  (gauss/cm)   afit
1 1   0 000000e+00 0 00000 0 00000    0 000 16002 182 16002 182  0 0000e+00  0 0000e+00   6 2e-04
2 1  -7 184376e+03 0 44898 0 00000    0 000 16000 289 16000 289  0 0000e+00 -8 6595e+00  -4 2e-03
3 1  -1 436698e+04 0 89796 0 00000    0 000 15994 221 15994 221  0 0000e+00 -1 9022e+01   4 9e-03
4 1  -2 154569e+04 1 34694 0 00000    0 000 15982 541 15982 541  0 0000e+00 -3 4173e+01   6 6e-03
1 2   0 000000e+00 0 00000 0 39394    0 000 16003 590 16003 590  7 0602e+00  0 0000e+00   3 1e-04
2 2  -4 398362e+03 0 27484 0 39533   -1 968 16002 952 16002 953  7 3534e+00 -4 6944e+00   1 9e-03
3 2  -1 124718e+04 0 70285 0 39718   -5 438 15999 266 15999 267  9 0890e+00 -1 2867e+01   9 3e-03
4 2  -1 836875e+04 1 14807 0 39846  -10 289 15991 220 15991 224  1 3123e+01 -2 4114e+01   4 8e-03
1 3   0 000000e+00 0 00000 0 78788    0 000 16007 571 16007 571  1 2898e+01  0 0000e+00   6 2e-04
2 3  -7 988794e+03 0 49720 0 79299   -6 723 16006 037 16006 039  1 4642e+01 -6 4742e+00  -1 6e-02
3 3  -1 520651e+04 0 95007 0 79628  -14 374 16001 628 16001 634  1 9692e+01 -1 3444e+01  -5 0e-03
4 3   2 234363e+04 1 39619 0 79811   25 193 15993 739 15993 759  2 9771e+01 -2 2630e+01  -1 5e-03
```

Figure 10.2–2: H-magnet output in air region on mesh coordinates (K, L). Current adjustment at given mesh point (3, 2).

least squares edit of problem , cycle 380

'h' mag symmetry type
stored energy = 1.4249e+03 joules / meter or radian
xjfact= 1.000000

k	l	a(vector)	x	y	bx (gauss)	by (gauss)	bt (gauss)	dby/dy (gauss/cm)	dby/dx (gauss/cm)	afit
1	1	-4.927671e-04	0.00000	0.00000	0.000	15212.225	15212.225	0.0000e+00	0.0000e+00	4.9e-04
2	1	-7.605820e+03	0.50000	0.00000	0.000	15210.451	15210.451	0.0000e+00	-7.2718e+00	7.8e+02
3	1	-1.520977e+04	1.00000	0.00000	0.000	15204.558	15204.558	0.0000e+00	-1.7003e+01	1.6e+03
4	1	-2.280936e+04	1.50000	0.00000	0.000	15192.451	15192.451	0.0000e+00	-3.2967e+01	2.3e+03
5	1	-3.040045e+04	2.00000	0.00000	0.000	15169.500	15169.500	0.0000e+00	-6.1944e+01	3.1e+03
7	1	-3.797554e+04	2.50000	0.00000	0.000	15126.270	15126.270	0.0000e+00	-1.1731e+02	-2.9e+03
1	4	1.318073e-05	0.00000	1.00000	0.000	15218.391	15218.391	1.0946e+01	0.0000e+00	-1.3e-05
2	3	-7.609068e+03	0.50000	1.00000	-5.777	15217.623	15217.624	1.2700e+01	-3.0919e+00	4.4e+01
3	3	-1.521735e+04	1.00000	1.00000	-13.375	15215.189	15215.195	1.8565e+01	-6.7795e+00	7.6e+02
4	3	-2.282390e+04	1.50000	1.00000	-25.413	15210.618	15210.639	3.1131e+01	-1.1756e+01	1.6e+03
6	4	-3.042748e+04	2.00000	1.00000	-46.587	15203.106	15203.177	5.6637e+01	-1.8742e+01	-1.2e+03
7	4	-3.802630e+04	2.50000	1.00000	-86.474	15191.268	15191.515	1.0924e+02	-2.9722e+01	-2.8e+02
1	6	5.427195e-02	0.00000	2.00000	0.000	15230.845	15230.845	1.3903e+01	0.0000e+00	-5.4e-02
2	6	-7.615107e+03	0.50000	2.00000	-5.139	15230.855	15230.856	1.1230e+01	3.9043e+00	1.1e+03
3	6	-1.523123e+04	1.00000	2.00000	-11.800	15234.000	15234.005	1.5750e+01	8.5139e+00	2.3e+03
5	6	-2.284961e+04	1.50000	2.00000	-21.290	15240.176	15240.191	2.2561e+01	1.5340e+01	-3.0e+03
6	6	-3.047228e+04	2.00000	2.00000	-34.024	15251.904	15251.942	2.7693e+01	3.2344e+01	-1.9e+03
7	6	-3.810420e+04	2.50000	2.00000	-50.200	15278.913	15278.995	3.4283e+01	7.9167e+01	-7.6e+02

Figure 10.2-3: H-magnet output in air region on coordinates (X, Y).

1the following is a map of /b/(kg) upper ---cycle 380
/b/(kg) lower

0 1

k . . . .

l		1	2	3	4	5	6	7	8	. . .	19	20
34	u											
	l	0.00	2.36	4.30	6.12	7.85	9.49	10.85	12.01	. . .	15.85	15.98
33	u	2.37	3.92	5.58	7.25	8.87	10.30	11.52	12.52	. . .	15.92	16.05
	l	2.04	3.63	5.37	7.11	8.81	10.23	11.47	12.51	. . .	15.91	16.04
32	u	3.17	3.98	5.34	6.86	8.43	9.77	11.02	12.07	. . .	15.86	15.98
	l	3.17	3.85	5.21	6.77	8.38	9.76	10.99	12.12	. . .	15.84	15.97
31	u	4.96	5.76	6.97	8.36	9.53	10.65	11.72	12.57	. . .	15.91	16.04
	l	4.95	5.72	6.93	8.34	9.54	10.65	11.76	12.63	. . .	15.90	16.03
30	u	6.42	6.77	7.59	8.68	9.62	10.53	11.51	12.28	. . .	15.84	15.97
	l	6.42	6.77	7.59	8.65	9.58	10.52	11.54	12.39	. . .	15.82	15.96
29	u	8.19	8.66	9.35	10.00	10.70	11.50	12.17	12.79	. . .	15.89	16.03
	l	8.19	8.62	9.23	9.90	10.66	11.51	12.25	12.89	. . .	15.88	16.02
28	u	9.77	9.92	10.26	10.67	11.18	11.76	12.26	12.71	. . .	15.82	15.95
	l	9.77	9.88	10.17	10.58	11.10	11.68	2.22	12.78	. . .	15.80	15.94

Figure 10.2-4: H-magnet output of $|B|$ in iron for coordinates (K, L).

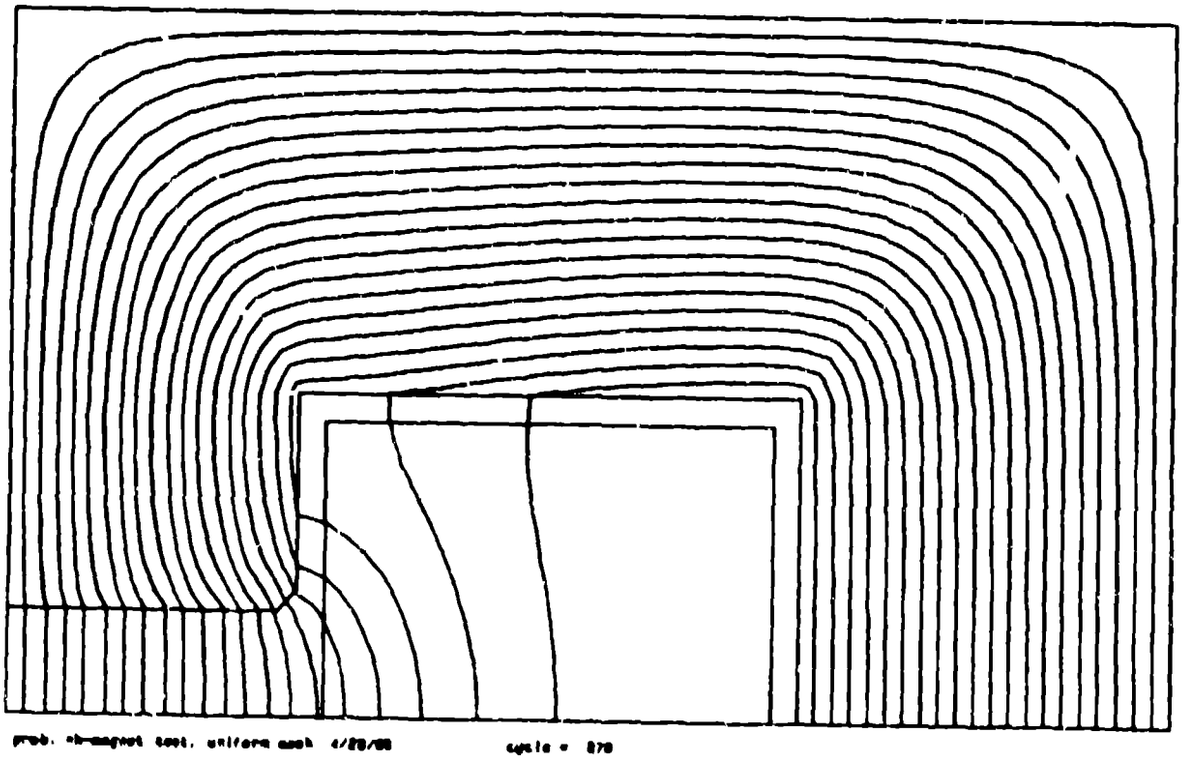


Figure 10.2-5: **Field lines TEKPLOT for the H-shaped dipole magnet. No Current adjustment.**

10.3 POISSON -- Quadrupole Magnet with a Hyperbolic Pole Tip

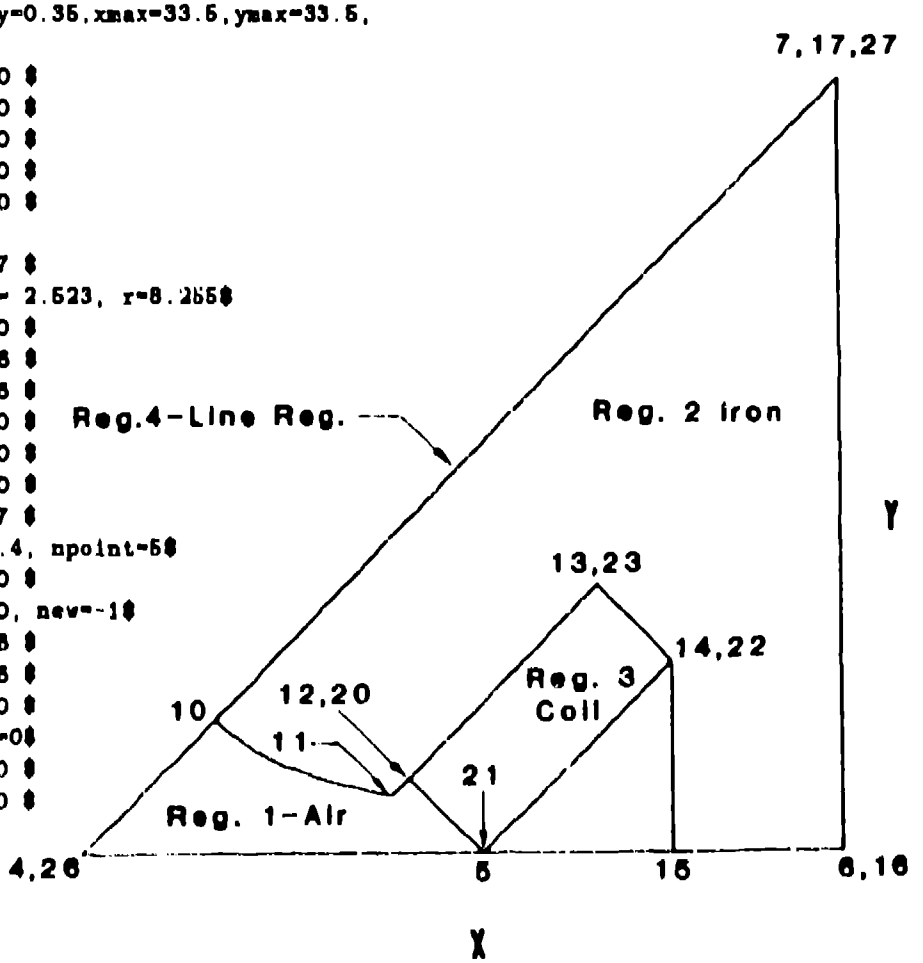
The magnetic fields and their gradients are calculated for a quadrupole magnet utilizing the following options:

- quadrupole symmetry
- REG parameters: NT = 3, NEW and IBOUND
- input of a permeability table
- output fields on specified lines

Taking into account the symmetry of a quadrupole we need only calculate one-eighth of the magnet. Below is the AUTOMESH input file, QUAD, corresponding to the geometrical configuration on the right.

QUAD

```
 1  quad with hyperbolic curve, input table 9/12/86
 2  $reg nreg=4,dx=0.35,dy=0.35,xmax=33.5,ymax=33.5,
 3       npoint=5$
 4  $po x= 0.000, y= 0.000 $
 5  $po x=17.444, y= 0.000 $
 6  $po x=33.080, y= 0.000 $
 7  $po x=33.080, y=33.080 $
 8  $po x= 0.000, y= 0.000 $
 9  $reg mat=3, npoint=9$
10  $po x= 5.837, y= 5.837 $
11  $po nt=3, x=13.507, y= 2.523, r=8.255$
12  $po x=14.214, y= 3.230 $
13  $po x=22.470, y=11.486 $
14  $po x=25.700, y= 8.256 $
15  $po x=25.700, y= 0.000 $
16  $po x=33.080, y= 0.000 $
17  $po x=33.080, y=33.080 $
18  $po x= 5.837, y= 5.837 $
19  $reg mat=1, cur=11416.4, npoint=5$
20  $po x=14.214, y= 3.230 $
21  $po x=17.444, y= 0.000, nev=-1$
22  $po x=25.700, y= 8.256 $
23  $po x=22.470, y=11.486 $
24  $po x=14.214, y= 3.230 $
25  $reg npoint=2, ibound=0$
26  $po x= 0.000, y= 0.000 $
27  $po x=33.080, y=33.080 $
```



where:

Line 9: MAT = 3 – the material code for region 2 will be a user-defined permeability/permittivity function. In this case it will be an input permeability table.

Line 11: NT = 3 – designates that the hyperbola $2 * X * Y = R^2 = 8.255^2$ will be drawn from previous point (5.837, 5.837) to this point (13.507, 2.523).

Line 21: NEW = -1 – was added after a first run produced a glitch on the boundary line from (14.214, 3.230) to (17.444, 0.0) by picking up a mesh point from the horizontal boundary of region 1. By setting NEW = -1, it ensures that points on the boundary do not coincide with any points on previously defined boundary region except for starting and end points.

Line 25–27: Region 4 is used to define the boundary condition on the 45° boundary from (0.0, 0.0) to (33.08, 33.08) as Dirichlet boundary (magnetic field lines are parallel). In a non-rectangular region, one must use IBOUND to define boundary conditions. (See Sec. 2.2.5 - Boundary Condition Options.)

The procedural file, CQUAD, designates that POISSON will be executed with input from PQUAD. Given the files, QUAD, CQUAD and PQUAD, the user types:

`cosmos i = cquad` for CRAY

`@CQUAD` for VAX

Line No.	COSMOS File, cquad	VAX File, CQUAD.COM
1	*automesh	$RUN AUTOMESH
2	quad	QUAD
3	*lattice	$RUN LATTICE
4	tape73	TAPE73
5	s	S
6	*poisson	$RUN POISSON
7	pquad	PQUAD
8	*/	$EXIT

PQUAD FILE

```
 1  0
 2  *18 1 *6 0 *46 4 s
 3  3 1.0 1
 4   0.0000000e+00   0.0017513135
 5   0.1142000e+04   0.0017513135
 6   0.2953000e+04   0.0010159504
 7   0.5114000e+04   0.0007821666
 8   0.8476000e+04   0.0007078644
 9   0.9667000e+04   0.0007241130
10   0.1057800e+05   0.0007562580
11   0.1131900e+05   0.0007951022
12   0.1194000e+05   0.0008376209
13   0.1245100e+05   0.0008834703
14   0.1291200e+05   0.0009293680
15   0.1331300e+05   0.0009764671
16   0.1365400e+05   0.0010253255
17   0.1393500e+05   0.0010764263
18   0.1421600e+05   0.0011254924
19   0.1444700e+05   0.0011767475
20   0.1461800e+05   0.0012313603
21   0.1478900e+05   0.0012846865
22   0.1502000e+05   0.0013315579
23   0.1513100e+05   0.0013879251
24   0.1525200e+05   0.0014423770
25   0.1542300e+05   0.0014912019
26   0.1559400e+05   0.0015389351
27   0.1570500e+05   0.0015918497
28   0.1618000e+05   0.0018542555
29   0.1684000e+05   0.0023752969
30   0.1715000e+05   0.0029154519
31   0.1736000e+05   0.0034566194
32   0.1762000e+05   0.0039729837
33   0.1785000e+05   0.0044862167
34   0.1820000e+05   0.0054945055
35   0.1895000e+05   0.0079176564
36   0.1950000e+05   0.0102564103
37   0.2020000e+05   0.0148588410
38   0.2065000e+05   0.0193798450
39   0.2095000e+05   0.0238663484
40   0.2160000e+05   0.0370370370
41   0.2190000e+05   0.0456621005
42   0.2300000e+05   0.0869565217 count
43  1
44  *18 0 *30 0 *42 19 19 1 68
45  *110 10 10 1. 90. 1. s
46  -1
```

where:

Line 1: 0 – to read dump number (NUM) = 0 on TAPE35 generated by LATTICE. This is the first required POISSON entry.

Line 2: CON array is the second required POISSON entry where:
- CON(18) = 1 – one permeability table to be read in.
- CON(6) = 0 – must be set to zero if CON(18) $\neq$ 0.
- CON(46) = 4 – symmetrical quadrupole.
- s – designates end of CON array entries.

Line 3: CON(18) $\neq$ 0 – indicates optional input:
- 3 – material code for which input permeabilty applies.
- 1.0 – stacking factor.
- 1 – input table given as (B, γ).

Lines 4–42: the values of (B, γ) with last value having a "c" (count) to indicate end of table. (See Sec. 5.4.1 B for more detail.)

Line 43: 1 – POISSON will read in dump 1, which was written after completion of the above run, to continue with new CON entries that follow.

Line 44:	CON array entries:	
	CON(18) = 0	– no new table to read in. (The read in table from previous run is in.)
	CON(30) = 0	– no further iteration (since problem converged in previous run).
	CON(42) = 19 CON(43) = 19 CON(44) = 1 CON(45) = 68	– calculates fields and auxiliary data on all non-iron points of the vertical line from mesh coordinates (19, 1) to (19, 68). These are listed in Fig. 10.3–2, up to l = 13. In the OUTPOI file the fields are given only up to l = 17 since iron region begins after that.
Line 45:	CON(110) = 10 (111) = 10 (112) = 1. (113) = 90. (114) = 1.	harmonic analysis parameters as described in Table 5–1.9. here we request: 10 coefficients and 10 points on the arc of circle starting at 0°, (Default CON(115)), up to 90° with radius = 1.0 (CON(112)). 1. - CON(114) - normalization radius.
Line 46:	-1	– designates the end of POISSON execution.

Fig. 10.3–1 lists the output harmonic values from the file OUTPOI. As seen, the coefficients begin at n = 2 and go up in steps of 4. The missing coefficients are zero due to the quadrupole symmetry. For complete detail on harmonic analysis see Reference Manual Section B.13.3.

```
harmonic analysis
integration radius =   1.00000
table for interpolated points
       n     angle   x coord   y coord   kf   lf       vec.pot.

       1    0.0000    1.0000    0.0000    4    1    2.08094e+02
       2   10.0000    0.9848    0.1736    4    1    1.95570e+02
       3   20.0000    0.9397    0.3420    4    2    1.59487e+02
       4   30.0000    0.8660    0.5000    4    2    1.04268e+02
       5   40.0000    0.7660    0.6428    3    3    3.58140e+01
       6   50.0000    0.6428    0.7660    3    3   -3.55994e+01
       7   60.0000    0.5000    0.8660    3    3   -1.02283e+02
       8   70.0000    0.3420    0.9397    3    3   -1.57878e+02
       9   80.0000    0.1736    0.9848    3    2   -1.95635e+02
      10   90.0000    0.0000    1.0000    3    2   -2.08943e+02
1table for vector potential coefficients
Onormalization radius =   1.00000
       a(x,y) = re( sum (an + i bn) * (z/r)**n )
        n         an             bn          abs(cn)
        2     2.0776e+02     0.0000e+00     2.0776e+02
        6     7.0685e-01     0.0000e+00     7.0685e-01
       10     7.6068e-02     0.0000e+00     7.6068e-02
       14     2.6021e-02     0.0000e+00     2.6021e-02
       18    -1.0182e-01     0.0000e+00     1.0182e-01
       22     2.6021e-02     0.0000e+00     2.6021e-02
       26     7.6068e-02     0.0000e+00     7.6068e-02
       30     7.0685e-01     0.0000e+00     7.0685e-01
       34     2.0776e+02     0.0000e+00     2.0776e+02
       38     2.0776e+02     0.0000e+00     2.0776e+02
table for field coefficients
normalization radius =   1.00000
       (bx - i by) = i * sum n*(an + i bn)/r * (z/r)**(n-1)
        n      n(an)/r        n(bn)/r      abs(n(cn)/r)
        2     4.1552e+02     0.0000e+00     4.1552e+02
        6     4.2411e+00     0.0000e+00     4.2411e+00
       10     7.6068e-01     0.0000e+00     7.6068e-01
       14     3.6429e-01     0.0000e+00     3.6429e-01
       18    -1.8328e+00     0.0000e+00     1.8328e+00
       22     5.7246e-01     0.0000e+00     5.7246e-01
       26     1.9778e+00     0.0000e+00     1.9778e+00
       30     2.1206e+01     0.0000e+00     2.1206e+01
       34     7.0639e+03     0.0000e+00     7.0639e+03
       38     7.8949e+03     0.0000e+00     7.8949e+03
```

Figure 10.3-1: Harmonic analysis listing of OUTPOI for the quadrupole magnet problem.

```
least squares edit of problem , cycle  1750

symm qua symmetry type
stored energy =  3.2544e+02 joules / meter or radian

xjfact=  1.000000
k   l  a(vector)      x         y        bx(gauss)  by(gauss) bt(gauss) dby/dy(gauss/cm) dby/dx(gauss/cm) aflt
19   1  8.189508e+03  6.27984  0.00000      0.000  -2607.309  2607.309   0.0000e+00    -4.1464e+02    1.7e-02
19   2  7.727194e+03  6.11021  0.35398   -146.825  -2537.033  2541.278   -1.7080e-01   -4.1473e+02    7.6e-04
19   3  8.113152e+03  6.29040  0.70787   -293.510  -2611.845  2628.285   -4.4300e-01   -4.1464e+02    1.8e-04
19   4  7.543931e+03  6.11991  1.06197   -440.442  -2541.312  2579.197   -5.2161e-01   -4.1481e+02    1.1e-03
19   5  7.829619e+03  6.30118  1.41583   -587.140  -2616.723  2681.785   -6.8666e-01   -4.1493e+02    1.7e-03
19   6  7.153326e+03  6.12984  1.77002   -734.253  -2545.853  2649.622   -6.4311e-01   -4.1510e+02    2.5e-03
19   7  7.339293e+03  6.31247  2.12403   -881.117  -2621.906  2768.000   -6.7693e-01   -4.1525e+02    3.6e-03
19   8  6.555627e+03  6.14018  2.47815  -1028.397  -2550.587  2750.071   -5.7370e-01   -4.1530e+02    5.1e-03
19   9  6.642720e+03  6.32467  2.83260  -1175.431  -2627.381  2878.328   -4.5120e-01   -4.1536e+02    7.4e-03
19  10  5.751486e+03  6.15112  3.19611  -1322.335  -2556.474  2877.328   -3.9673e-01   -4.1518e+02    1.0e-02
19  11  5.741555e+03  6.33831  3.54110  -1469.678  -2633.225  3015.598   -3.1686e-02   -4.1495e+02    1.5e-02
19  12  4.743763e+03  6.16273  3.89237  -1616.389  -2560.518  3027.486   -2.5434e-01   -4.1421e+02    2.7e-02
19  13  4.642707e+03  6.35445  4.24741  -1762.469  -2639.781  3174.074    3.6528e-01   -4.1345e+02    9.1e-02
```

Figure 10.3-2: A section of listing of OUTPOI for the quadrupole magnet problem.

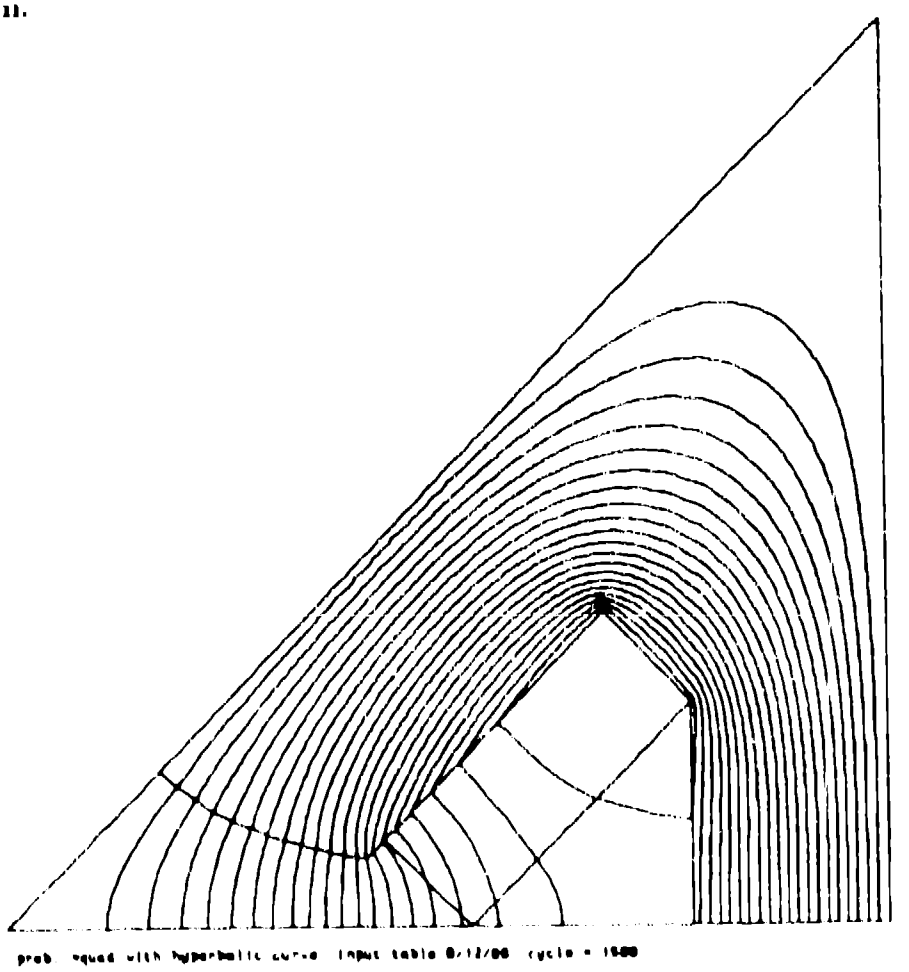


Figure 10.3-3: Field lines TEKPLOT for the quadrupole magnet.

10.4 PANDIRA -- Quadrupole Magnet with Hyperbolic Pole Tip

This is the same problem as 10.3, but now we use PANDIRA to solve for the magnetic fields and auxiliary quantities. The files QUAD and FQUAD are identical. The procedural file, CQUAD2, differs only in line 5 as listed below.

Line No.	COSMOS File, cquad2	VAX File, CQUAD2.COM
1	*automesh	$RUN AUTOMESH
2	quad	QUAD
3	*lattice	$RUN LATTICE
4	tape73	TAPE73
5	*81 0 s	*81 0 S
6	*pandira	$RUN PANDIRA
7	pquad	PQUAD
8	*/	$EXIT

For PANDIRA run CON(81) = 0. Please note that in the COSMOS file "*" must not be in column 1 since a "*" in column 1 designates an executable file to COSMOS. Given the files QUAD, PQUAD and CQUAD2, the user types:

cosmos i = cquad2	for CRAY
@CQUAD2	for VAX

PANDIRA produces almost identical results. This is seen by comparing listings of Fig. 10.4-1 of PANDIRA to Fig. 10.3-2 of POISSON.

```
least squares edit of problem , cycle     8

symm qua symmetry type
stored energy =  3.2646e+02 joules / meter or radian

xjfact=  1.000000                                  bx          by          bt          dby/dy      dby/dx
  k  l   a(vector)      x          y               (gauss)     (gauss)     (gauss)     (gauss/cm)  (gauss/cm)   afit
 19  1  8.190186e+03  6.27984  0.00000              0.000    -2607.514   2607.514     0.0000e+00  -4.1467e+02  1.7e-02
 19  2  7.727806e+03  6.11021  0.35398           -146.837    -2537.233   2541.478    -1.7114e-01  -4.1476e+02  7.7e-04
 19  3  8.113795e+03  6.29040  0.70787           -293.533    -2612.061   2628.492    -4.4378e-01  -4.1467e+02  1.8e-04
 19  4  7.544529e+03  6.11991  1.06197           -440.476    -2541.513   2579.400    -5.2288e-01  -4.1485e+02  1.1e-03
 19  5  7.830240e+03  6.30118  1.41583           -587.184    -2616.930   2681.997    -6.8818e-01  -4.1496e+02  1.7e-03
 19  6  7.153893e+03  6.12984  1.77002           -734.309    -2546.065   2649.831    -6.4479e-01  -4.1513e+02  2.5e-03
 19  7  7.339877e+03  6.31247  2.12403           -881.184    -2622.113   2766.218    -6.7904e-01  -4.1528e+02  3.6e-03
 19  8  6.556149e+03  6.14018  2.47815          -1028.376    -2550.790   2750.289     6.7579e-01  -4.1533e+02  5.1e-03
 19  9  6.643251e+03  6.32467  2.83260          -1175.521    -2627.592   2878.557     4.5359e-01  -4.1539e+02  7.4e-03
 19 10  5.751947e+03  6.15112  3.18611          -1322.4      -2555.679   2877.558    -3.9889e-01  -4.1522e+02  1.0e-02
 19 11  5.742016e+03  6.33831  3.54110          -1469.792    -2633.438   3015.839    -3.3716e-02  -4.1498e+02  1.6e-02
 19 12  4.744145e+03  6.16273  3.89237          -1616.516    -2560.726   3027.739     2.5618e-01  -4.1424e+02  2.7e-02
 19 13  4.643083e+03  6.35445  4.24741          -1762.607    -2639.995   3174.328     3.6473e-01  -4.1348e+02  9.1e-02
```

Figure 10.4-1: A section of listing of OUTPAN for the quadrupole magnet problem.

10.5 PANDIRA -- A Ring Dipole Permanent Magnet

Figure 10.5-1 shows a vertical cross section of a ring dipole magnet composed of eight identical permanent magnet (PM) slabs. We wish to calculate the fields at the center which according to experimental measurements should be ~ 10 kgauss.

Taking advantage of the symmetry, we only need to compute one-quarter of the magnet.

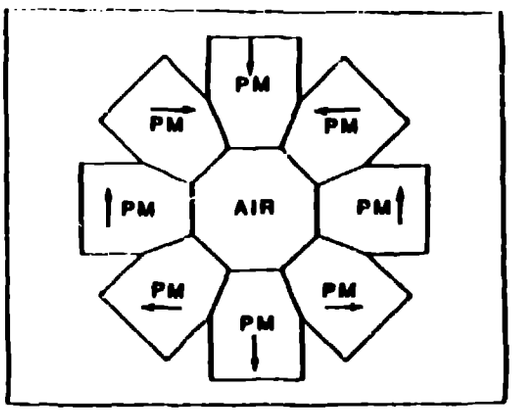
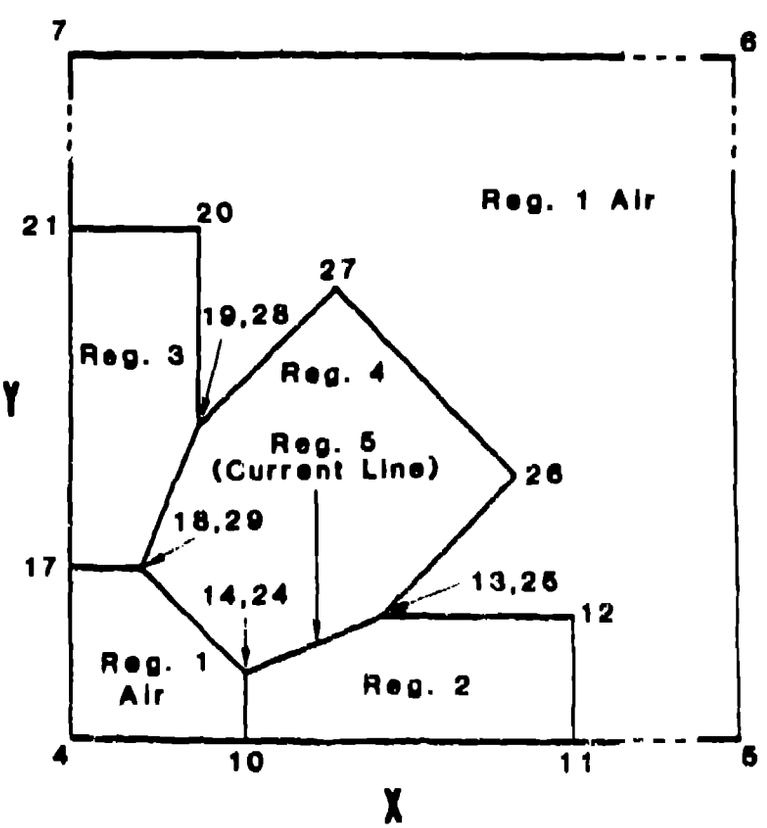


Figure 10.5-1: Vertical cross section of the ring dipole permanent magnet.

Below is the AUTOMESH input file DIPM, and the geometric configuration which corresponds to the upper right-hand quadrant of Fig. 10.5-1.

DIPM

```
10 kg ring dipole permanet magnet 9/5/86
$reg nreg=5,dx=.1,dy=.1,xmax=12.0,ymax=12.0,npoint=5,
     xreg1=7.3,yreg1=7.38
$po x= 0.00    , y= 0.00 $
$po x=12.00    , y= 0.00 $
$po x=12.00    , y=12.00 $
$po x= 0.00    , y=12.00 $
$po x= 0.00    , y= 0.00 $
$reg mat=6,npoint=6 $
$po x= 2.5000 , y= 0.00 $
$po x= 7.3000 , y= 0.00 $
$po x= 7.3000 , y= 1.8285 $
$po x= 4.5000 , y= 1.8285 $
$po x= 2.5000 , y= 1.00 $
$po x= 2.5000 , y= 0.00 $
$reg mat=7,npoint=6 $
$po x= 0.00    , y= 2.5000 $
$po x= 1.00    , y= 2.5000 $
$po x= 1.8285 ,y= 4.5000 $
$po x= 1.8285 ,y= 7.3000 $
$po x= 0.00    , y= 7.3000 $
$po x= 0.00    , y= 2.5000 $
$reg mat=8,npoint=7 $
$po x= 2.5000 , y= 1.00    $
$po x= 4.5000 , y= 1.8285 $
$po x= 6.47    , y= 3.82    $
$po x= 3.82    , y= 6.47    $
$po x= 1.8285 , y= 4.5     $
$po x= 1.00    , y= 2.5000 $
$po x= 2.5000 , y= 1.00    $
$reg mat=1, cur=20 ,npoint=2 $
$po x= 1.5000 , y= 1.00  $
$po x= 4.5000 , y= 1.8285$
```



Line 3: XREG1 = 7.3 – the mesh size will double in the x direction
YREG1 = 7.3 starting after x = 7.3 and in the y direction after y = 7.3 (see Fig. 10.5-3).

Lines 9, 16, 23: MAT > 5 – defines these regions as permanent magnetic material with the permeability functions input by a straight line B(H).

Lines 31–33: – added region defining a current line region that MUST be given to initialize PANDIRA. The value of current (CUR) and the location of current line region are immaterial.

The procedural file, CDIPM, and the input file, PDIPM, for PANDIRA are listed below. Given these three files - DIPM, CDIPM and PDIPM the user types:

`cosmos 1 = cdipm` for CRAY

`@CDIPM` for VAX

Line no.	COSMOS file, cdipm	VAX file, CDIPM.COM
1	*automesh	$RUN AUTOMESH
2	dipm	DIPM
3	*lattice	$RUN LATTICE
4	tape73	TAPE73
5	*81 0 *101 1 S	*81 0 *101 1 S
6	*pandira	$RUN PANDIRA
7	pdipm	PDIPM
8	*/	$EXIT

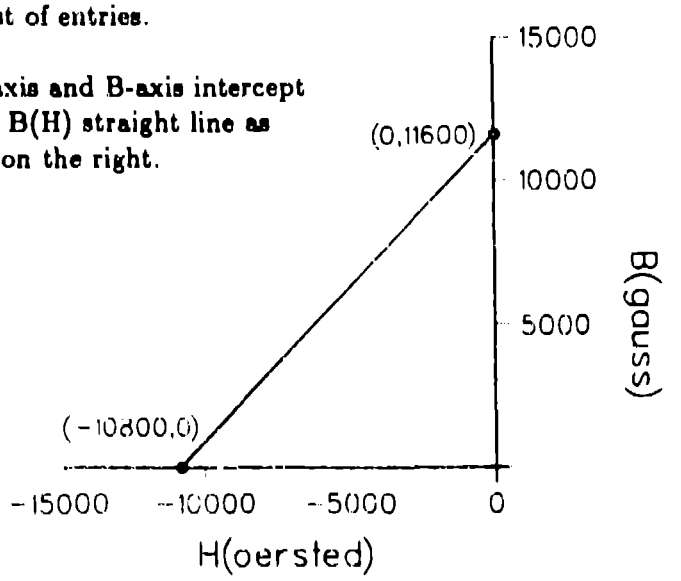
where:

Line 5: CON(81)= 0 – for PANDIRA runs this MUST be set to zero in LATTICE.
CON(101) = 1 – permanent magnet problem.

PDIPM

```
 1  0 dump
 2  *18 3 *6 0 s
 3  6 1  -1
 4  90  1 0 s
 5  -10800. 11600.
 6  7 1. -1
 7  -90. 1.0 s
 8  -10800. 11600.
 9  8 1  -1
10  180. 1.0 s
11  -10800. 11600.
12  -1
```

Line 2:	CON(18) = 3	- no. of straight line B(H) inputs.
	CON(6) = 0	- must be set to zero if CON(18) $\neq$ 0.
Line 3:	CON(18) $\neq$ 0	- indicates optional input:
	6	- material code for which this input B(H) applies.
	1.	- stacking factor.
	-1	- must be a negative no. (value immaterial).
Line 4:	90.	- the direction angle (in deg.) of the "easy axis" relative to the horizontal axis in the counter-clockwise direction.
	1.0	- the $\gamma(= 1/\mu)$ value perpendicular to the "easy axis".
	s	- skip rest of entries.
Line 5:	-10800. 11600.	- the H-axis and B-axis intercept for the B(H) straight line as shown on the right.



Line 6-12: same as 3-5, except for material code 7 "easy axis", -90° (downward), and for material code 8, 180° (to the left).

(See Sec. 5.4.1-C - Permeability/Permittivity Function Input for more detail.)

Fig. 10.5-2 shows parts of the output from file OUTPAN that PANDIRA generates. First part lists the input B(H) curves (note that values not input such as ϕ, have default values). The second part lists the fields on axis. As can be seen the value of $|B| = bt$ = 10.010879 kgauss at origin which is close to the measured field.

```
 material no.  6,  stack=1.000,  fixgam= 0.004

  aniso =  90.000       gamper =   1.000       x0 =   0.000     y0 =   0.000     phi =1111.000

  hcept =  -0.108e+05    bcept =   0.116e+05
1
 material no.  7,  stack=1.000,  fixgam= 0.004

  aniso = 270.000       gamper =   1.000       x0 =   0.000     y0 =   0.000     phi =1111.000

  hcept =  -0.108e+05    bcept =   0.116e+05
1
 material no.  8,  stack=1.000,  fixgam= 0.004

  aniso = 180.000       gamper =   1.000       x0 =   0.000     y0 =   0.000     phi =1111.000

  hcept =  -0.108e+05    bcept =   0.116e+05

   .                                  .                                  .

   .                                  .                                  .

   .                                  .                                  .

least squares edit of problem , cycle     2

midplane symmetry type

xjfact=  1.000000                              bx         by          bt         dyb/dy       dby/dx
 k   l    s(vector)      x         y         (gauss)   (gauss)     (gauss)    (gauss/cm)   (gauss/cm)     efit
 1   1  0.000000e+00   0.00000   0.00000    0.000   -10010.879   10010.879   0.0000e+00   8.6914e-04   9.4e-06
 2   1  1.001088e+03   0.10000   0.00000    0.000   -10010.885   10010.885   0.0000e+00  -1.3170e-01  -1.0e-06
 3   1  2.002178e+03   0.20000   0.00000    0.000   -10010.907   10010.907   0.0000e+00  -3.1266e-01   8.2e-06
 4   1  3.003270e+03   0.30000   0.00000    0.000   -10010.951   10010.951   0.0000e+00  -5.9554e-01   1.5e-05
 5   1  4.004369e+03   0.40000   0.00000    0.000   -10011.030   10011.030   0.0000e+00  -1.0231e+00   2.0e-05
 6   1  5.005478e+03   0.50000   0.00000    0.000   -10011.161   10011.161   0.0000e+00  -1.6266e+00   2.5e-05
 7   1  6.006603e+03   0.60000   0.00000    0.000   -10011.361   10011.361   0.0000e+00  -2.4119e+00   2.7e-05
 8   1  7.007753e+03   0.70000   0.00000    0.000   -10011.647   10011.647   0.0000e+00  -3.3398e+00   2.2e-05
 9   1  8.008936e+03   0.80000   0.00000    0.000   -10012.030   10012.030   0.0000e+00  -4.2971e+00   6.9e-06
10   1  9.010162e+03   0.90000   0.00000    0.000   -10012.501   10012.501   0.0000e+00  -5.057.e+00  -2.6e-05
11   1  1.001144e+04   1.00000   0.00000    0.000   -10013.023   10013.023   0.0000e+00  -5.2280e+00  -8.6e-05
12   1  1.101277e+04   1.10000   0.00000    0.000   -10013.510   10013.510   0.0000e+00  -4.1854e+00  -1.9e-04
13   1  1.201413e+04   1.20000   0.00000    0.000   -10013.797   10013.797   0.0000e+00  -9.9271e-01  -3.4e-04
14   1  1.301551e+04   1.30000   0.00000    0.000   -10013.607   10013.607   0.0000e+00   5.6961e+00   5.8e-04
15   1  1.401682e+04   1.40000   0.00000    0.000   -10012.502   10012.502   0.0000e+00   1.7731e+01  -9.3e-04
16   1  1.501796e+04   1.50000   0.00000    0.000   -10009.832   10009.832   0.0000e+00   3.7615e+01  -1.4e-03
17   1  1.601871e+04   1.60000   0.00000    0.000   -10004.663   10004.663   0.0000e+00   6.9517e+01  -2.1e-03
18   1  1.701878e+04   1.70000   0.00000    0.000    9995.704    9995.704   0.0000e+00   1.1437e+02  -3.2e-03
19   1  1.801788e+04   1.80000   0.00000    0.000    9981.243    9981.243   0.0000e+00   1.7974e+02   4.7e-03
20   1  1.901476e+04   1.90000   0.00000    0.000    9959.089    9959.089   0.0000e+00   2.6968e+02   7.1e-03
21   1  2.000913e+04   2.00000   0.00000    0.000   -9926.575   9926.575   0.0000e+00   3.8853e+02   1.1e-02
22   1  2.099961e+04   2.10000   0.00000    0.000   -9880.635   9880.635   0.0000e+00   5.4020e+02   1.9e-02
23   1  2.198467e+04   2.20000   0.00000    0.000   -9818.016   9818.016   0.0000e+00   7.2843e+02   3.8e-02
24   1  2.296245e+04   2.30000   0.00000    0.000   -9734.980   9734.980   0.0000e+00   9.4621e+02   9.3e-02
25   1  2.393079e+04   2.40000   0.00000    0.000   -9633.113   9633.113   0.0000e+00   1.1558e+03   4.0e-01
```

Figure 10.5 2: A portion of listing from file OUTPAN showing the input B(H) parameters and fields on axis for the ring dipole permanent magnet.

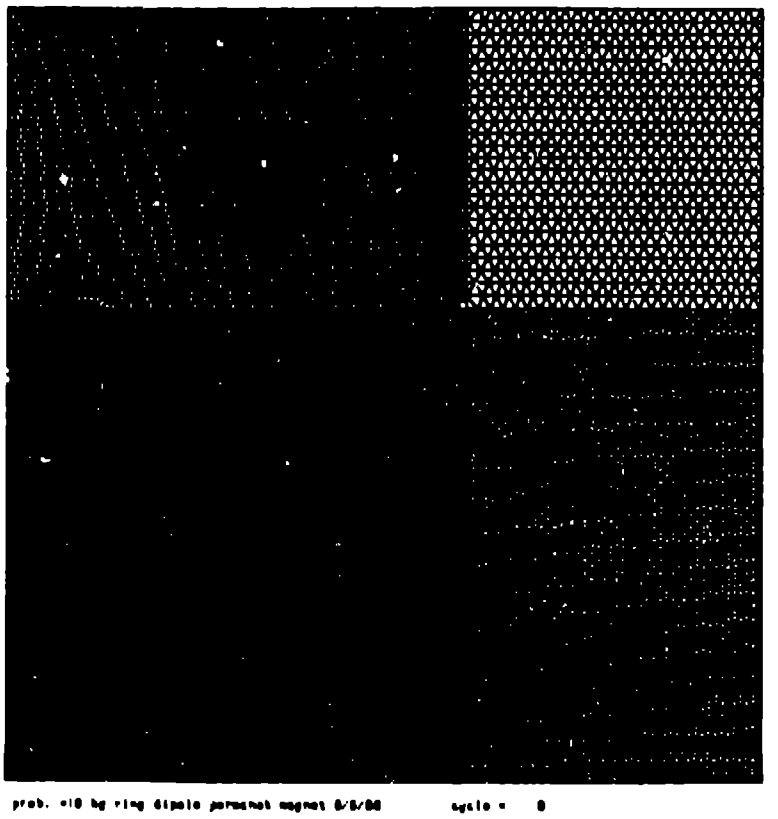


Figure 10.5 3: Mesh plot from TEKPLOT for the ring dipole permanent magnet.

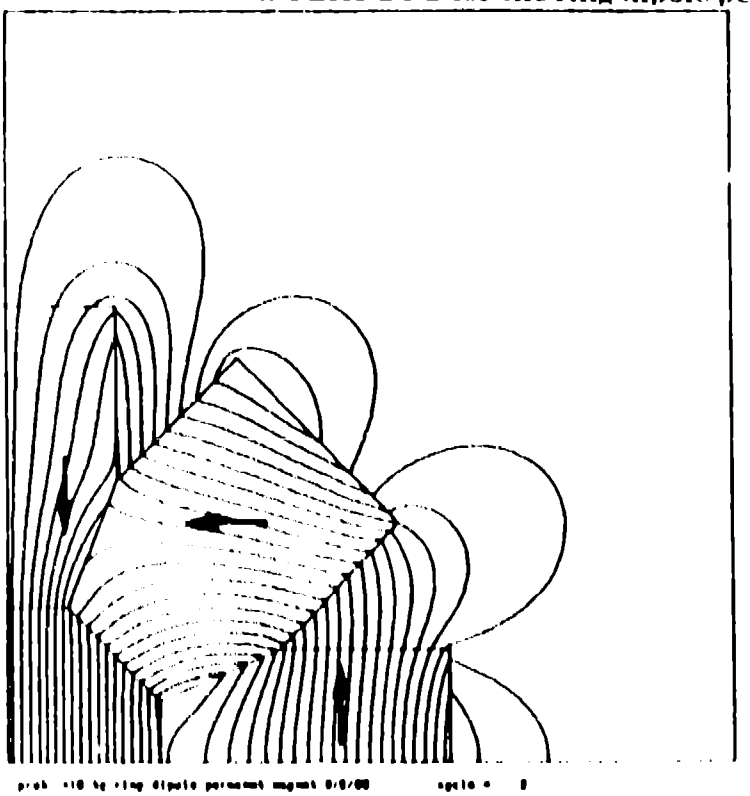


Figure 10.5 4: Field lines TEKPLOT for the ring dipole permanent magnet. Arrows inserted to show direction of the "easy axis."

10.6 PANDIRA -- Cylindrical Permanent Magnet Problem

Given four cylinders of specified dimensions and composed of permanent magnetic (PM) material, we find that a geometric arrangement of the cylinders as shown in Fig. 10.6 produced the desired maximum B_r field in the plane z = 0 of ~ 270 gauss.

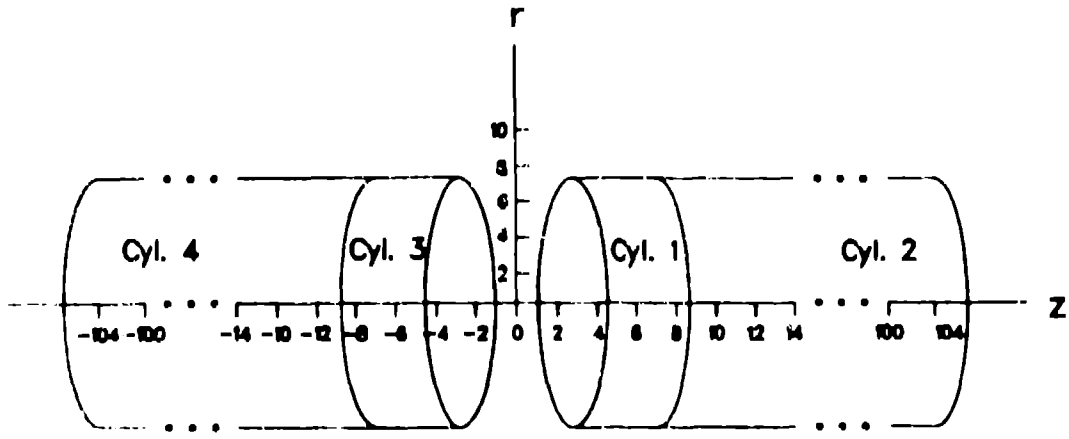


Fig. 10.6-1 Four permanent magnetic cylinders

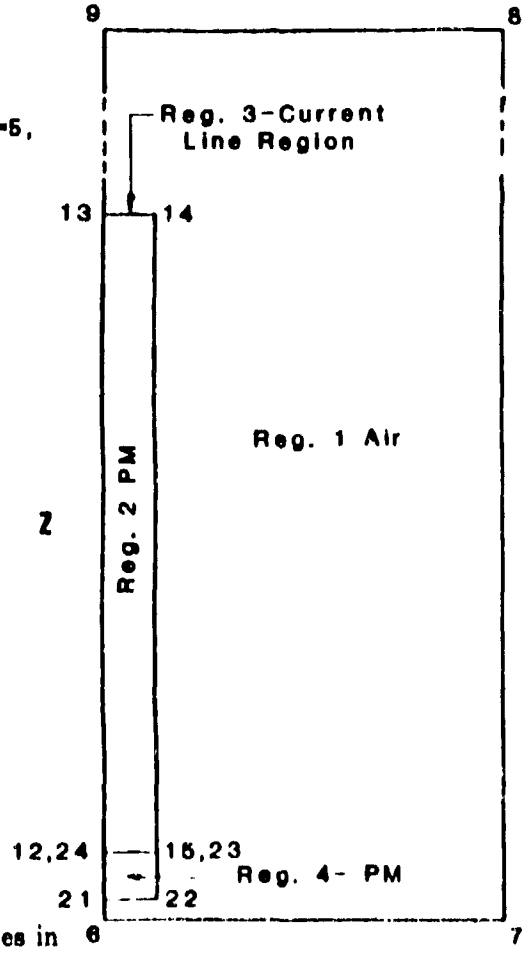
The PANDIRA calculation is done using the following options:

- cylindrical coordinates
- variable mesh in both r and z directions
- input of B(H) curve

Here again, taking advantage of the symmetry we only need compute one-quarter of the geometry. Below is the AUTOMESH input file CYPM, corresponding to the geometric configuration on the right. It should be noted that x ⟶ r and y ⟶ z.

CYPM

```
 1  cylindrical permanent magnet 9/4/86
 2  $reg nreg=4,dx=0.25,xmax=60.,ymax=150.,npoint=5,
 3  xreg1=7.,xreg2=20.,yreg1=12.,yreg2=15.,
 4  kreg1=8,kreg2=71,kmax=101,
 5  lreg1=17,lreg2=27,lmax=47$
 6  $po y= 0.00, x= 0.00$
 7  $po y=150.0, x= 0.00$
 8  $po y=150.0, x=60.00$
 9  $po y= 0.00, x=60.00$
10  $po y= 0.00, x= 0.00$
11  $reg mat=6,npoint=5$
12  $po y=10.20, x= 0.00$
13  $po y=103.0, x= 0.00$
14  $po y=103.0, x= 7.62$
15  $po y=10.20, x= 7.62$
16  $po y=10.20, x= 0.00$
17  $reg mat=1,cur=100.,npoint=2$
18  $po y=103.0, x= 0.00$
19  $po y=103.0, x= 7.62$
20  $reg mat=7,npoint=5$
21  $po x= 0 00, y= 3.00$
22  $po x= 7.62, y= 3.00$
23  $po x= 7.62, y=10.20$
24  $po x= 0.00, y=10.20$
25  $po x= 0.00, y= 3.00$
```



Lines 3–6: – variable mesh – 3 different mesh sizes in the horizontal and vertical dimension:

dx = (7.-0.)/(8-1) = 1.	$0 \leq x \leq 7.$
= (20.-7.)/(71-8) = 0.2063	$7. < x \leq 20.$
=(60.-20.)/(101-71) = 1.3333	$20. < x < 60.$
dy = (12.-0.)/(17-1) = 0.75	$0 \leq y \leq 12.$
= (15.-12.)(27-17) = 0.3	$12. < y \leq 15.$
= (150.-15.)(47-27) = 6.75	$15. < y < 150.$

(See Figs. 10.6-4 and 10.6-5.) The user can verify that AUTOMESH calculated the correct mesh size by checking the output file, OUTAUT.

Lines 11, 20: MAT > 5 – defines these regions as permanent magnetic material with the permeability functions will be input by a straight line B(H).

Lines 17-19: – added region defining a current region that must be given to initialize PANDIRA. The value of the current (CUR) and the location of the current line region are immaterial.

The procedural file, CCYPM, and the input file, PCYPM, for PANDIRA are listed below. Given these three files - CYPM, CCYPM, and PCYPM - the user types:

`cosmos i = ccypm` for CRAY

`@CCYPM` for VAX

Line No.	COSMOS File, ccypm	VAX File, CCYPM.COM
1	*automesh	$RUN AUTOMESH
2	cypm	CYPM
3	*lattice	$RUN LATTICE
4	tape73	TAPE73
5	*22 0 *81 0 s	*22 0 *81 0 S
6	*pandira	$RUN PANDIRA
7	pcypm	PCYPM
8	/*	$EXIT

where:

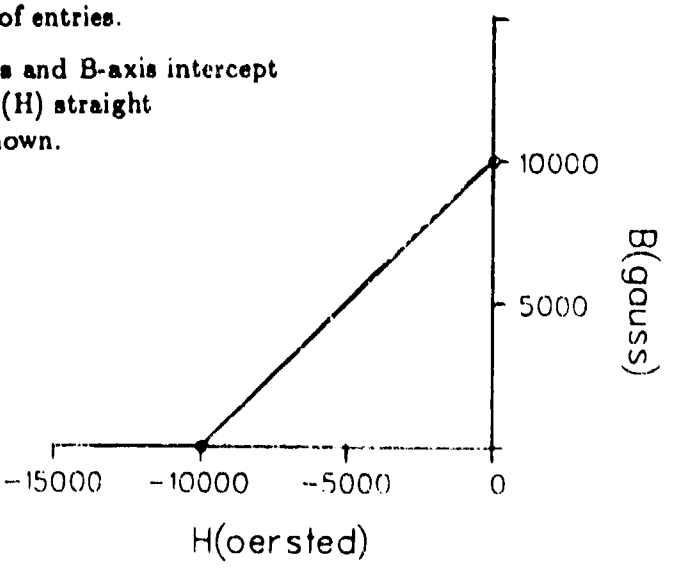
Line 5: CON(22) = 0 - Boundary conditions (if changed) MUST be set in LATTICE.
CON(81) = 0 - PANDIRA run. This MUST be set to zero in LATTICE.

PCYPM

```
1  0 dump
2  *19 1 *101 1 *6 0 *18 2 *46 1 s
3  6 1.0 -1 mat stack type
4  -90.0 1.0 s
5  -10000. 10000. hcept bcept
6  7 1.0 -1 mat stack type
7  90.0 1.0 s
8  -10000. 10000. hcept bcept
9  -1 dump
```

(Note that for identification, we chose to put comments after all values are given for each line)

Line 2:	CON(19) = 1	- cylindrical coordinates.
	CON(101) = 1	- permanent magnet problem.
	CON(6) = 0	- must be set to zero if CON(18) $\neq$ 0.
	CON(18) = 2	- no. of straight line B(H) input.
	CON(46) = 1	- cylindrical, ITYPE, no symmetry. Specified boundary condition in LATTICE (CON(22) = 0) that field lines are parallel to the r-axis.
Line 3:	if CON(18) $\neq$ 0	- indicates optional input:
	6	- material codes for which this input B(H) applies.
	1.0	- stacking factor.
	-1	- must be a negative no. (value immaterial).
Line 4:	-90.	- the direction angle (in deg.) of the "easy axis" relative to the horizontal axis in the clockwise (negative angle) direction.
	1.0	- the $\gamma(= 1/\mu)$ value perpendicular to the "easy axis".
	s	- skip rest of entries.
	-10000. 10000.	- the H-axis and B-axis intercept for the B(H) straight line as shown.



Line 6-8: same as 3-5, except for material code 7 "easy axis," 90 (upward).

(See Sec. 5.4.1-C - Permeability/Permittivity Function Input - for more detail.)

Fig. 10.6-2 lists the part of the output from file OUTPAN that shows maximum B_r = 270.434 gauss, at coordinates (21.29032, 0.0).

```
least squares edit of problem , cycle     2
none   symmetry type
xjfact=  1.000000
                                          br        bz        bt        dbz/dr
 k l   ra(vector)     r         z        (gauss)   (gauss)   (gauss)   (gauss/cm)   n=r/bz*bz/dr, rafit
 1 1  0.00000e+00  0.00000   0.00000      0.000    -4.415     4.415   0.0000e+00   0.0000e+00  -2.7e-01
 2 1  0.00000e+00  1.00000   0.00000   -244.739     0.052   244.739  -3.7999e-02  -7.3009e-01   1.3e-01
 3 1  0.00000e+00  2.00000   0.00000   -506.336     0.475   506.336   1.8513e+00   7.7902e+00   1.5e+00
 4 1  0.00000e+00  3.00000   0.00000   -794.332     1.934   794.334   2.0009e+00   3.1044e+00   2.7e+00
   .                                                  .                          .
   .                                                  .                          .
   .                                                  .                          .

64 1  0.00000e+00 18.74194  0.00000    251.031    -0.009   251.031   8.3795e+00  -1.7871e+04   3.0e+00
65 1  0.00000e+00 18.95161  0.00000    254.733    -0.006   254.733   7.5910e+00  -2.3421e+04   2.8e+00
66 1  0.00000e+00 19.16129  0.00000    257.971    -0.004   257.971   6.8714e+00  -3.1289e+04   2.6e+00
67 1  0.00000e+00 19.37097  0.00000    260.771    -0.003   260.771   6.2248e+00  -3.5673e+04   2.3e+00
68 1  0.00000e+00 19.58065  0.00000    263.151    -0.006   263.151   5.6651e+00  -1.9881e+04   2.2e+00
69 1  0.00000e+00 19.79032  0.00000    265.272    -0.480   265.273   4.9275e+00  -2.0303e+02   1.9e+00
70 1  0.00000e+00 20.00000  0.00000    267.259    -1.409   267.263   3.3662e+00  -4.7797e+01   3.5e+00
71 1  0.00000e+00 21.29032  0.00000    270.434     0.026   270.434   5.4518e-01   4.4816e+02   6.3e+00
72 1  0.00000e+00 22.58065  0.00000    266.984     0.053   266.984   1.0474e-01   4.4480e+01   1.4e+00
73 1  0.00000e+00 23.87097  0.00000    258.638     0.073   258.638  -1.1245e-01  -3.6552e+01  -2.0e+00
74 1  0.00000e+00 25.16129  0.00000    247.335     0.069   247.335  -2.4661e-01  -9.1016e+01  -4.4e+00
75 1  0.00000e+00 26.45161  0.00000    234.388     0.059   234.388  -3.2235e-01  -1.4512e+02  -6.0e+00
76 1  0.00000e+00 27.74194  0.00000    220.673     0.048   220.673  -3.[illegible]e-01  -2.0533e+02  -6.9e+00
```

Figure 10.6 2: A portion of the output of OUTPAN showing maximum B_r = 270.434 gauss for the cylindrical permanent magnet problem.

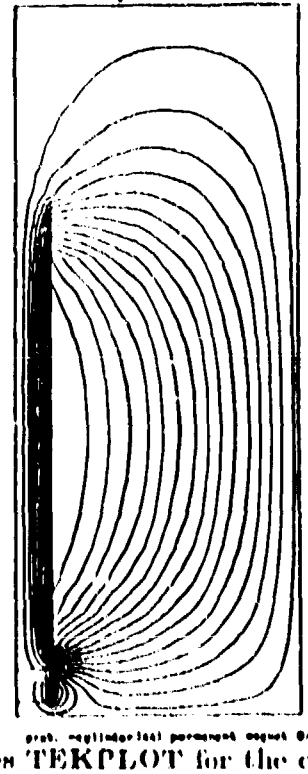


Figure 10.6 3: Field lines TEKPLOT for the cylindrical permanent magnet problem.

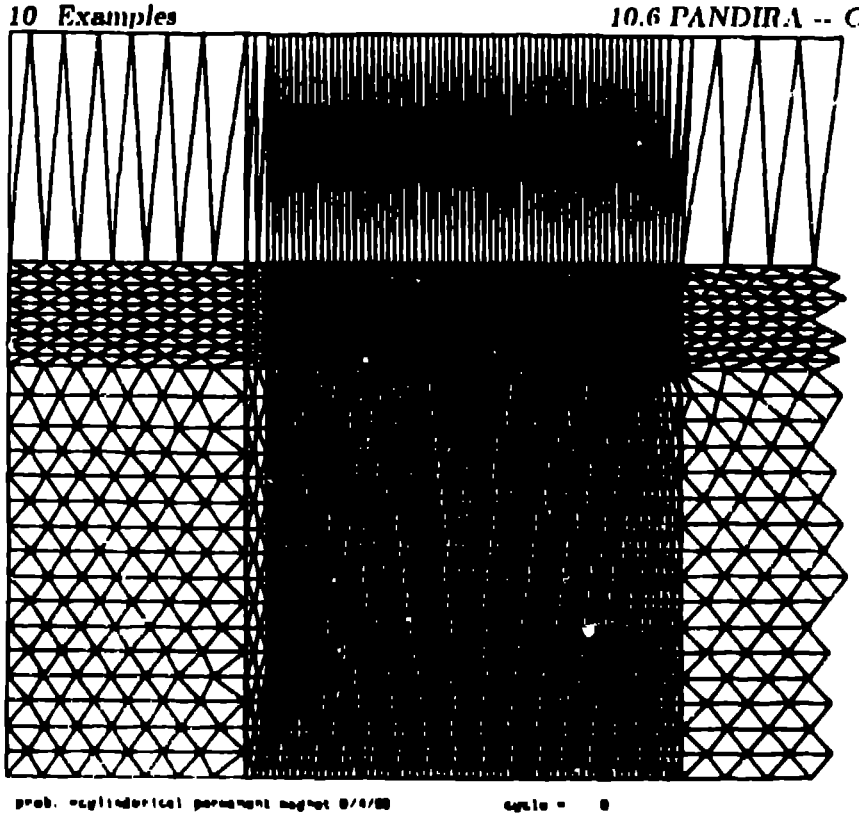


Figure 10.6–4: Mesh plot from TEKPLOT for XMAX = YMAX = 25. for the cylindrical permanent magnet problem.

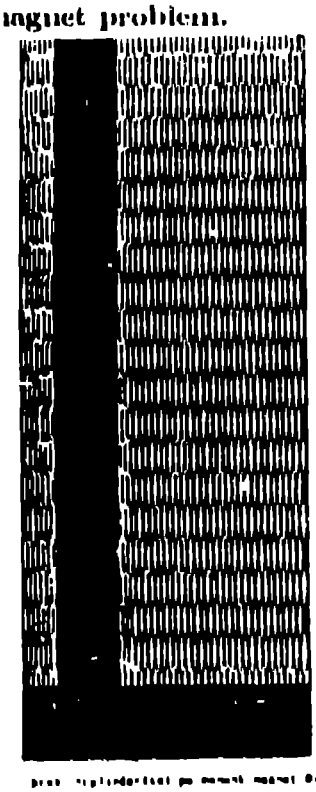


Figure 10.6–5: Mesh plot from TEKPLOT for full dimension of cylindrical permanent magnet problem.

10.7 POISSON -- Vector Potential Problem

The problem here is to calculate the effect a 2 mm-thick metal enclosure has in reducing the earth's magnetic fields inside the region. This calculation is done using:

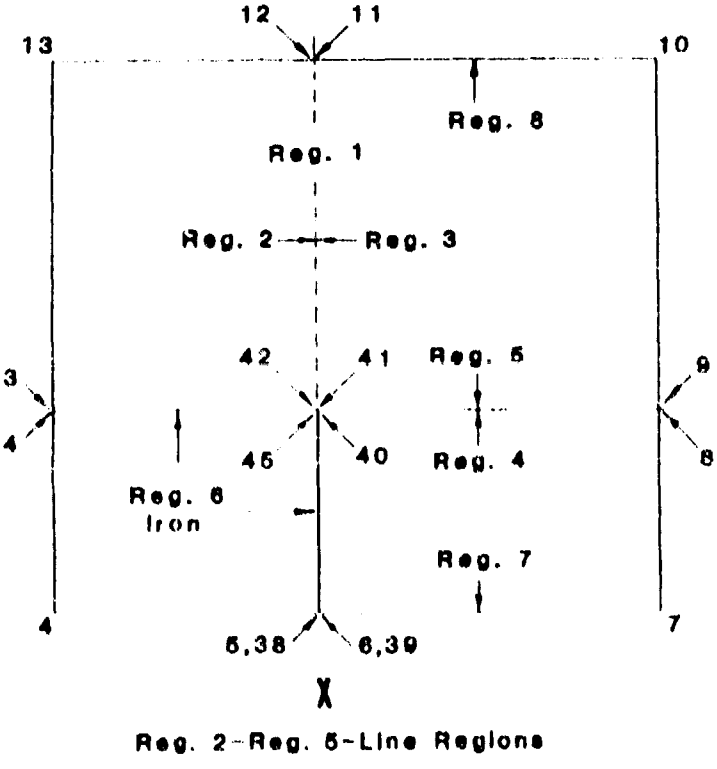
- input file to LATTICE directly
- input fixed vector potential on boundaries
- a constant fixed permeability value

Because the metal region is very small in comparison to the overall dimensions, we ran into problems using AUTOMESH. Also since additional variations of this problem will require more than the maximum of three mesh sizes, in each direction, (that AUTOMESH allows), we opt to input the problem directly into LATTICE.

Again we take advantage of the symmetry and use only one-quarter of the geometry. The input file, VECP, with the corresponding configuration is given below.

VECP

```
 1   vector potential problem 9/11/86
 2  *2  8 *21   0 0 1 1 *46 1 *9 1.0000 skip
 3   1  1        0.0000       0.0000  0  0 region
 4    1   1     0.0000     0.0000
 5   36   1   175.2000     0.0000
 6   40   1   175.4000     0.0000
 7   85   1   400.4000     0.0000
 8   85  26   400.4000   129.5000
 9   85  30   400.4000   129.7000
10   85  75   400.4000   354.7000
11   40  75   175.4000   354.7000
12   36  75   175.2000   354.7000
13    1  75     0.0000   354.7000
14    1  30     0.0000   129.7000
15    1  26     0.0000   129.5000
16    1   1     0.0000     0.0000 coun
17   2  1        0.0000       0.0000  0  1 region
18   36   1   175.2000     0.0000
19   36  26   175.2000   129.5000
20   36  30   175.2000   129.7000
21   36  75   175.2000   354.7000 coun
22   3  1        0.0000       0.0000  0  1 region
23   40   1   175.4000     0.0000
24   40  26   175.4000   129.5000
25   40  30   175.4000   129.7000
26   40  75   175.4000   354.7000 coun
27   4  1        0.0000       0.0000  0  1 region
28    1  26     0.0000   129.5000
29   36  26   175.2000   129.5000
30   40  26   175.4000   129.5000
31   85  26   400.4000   129.5000 coun
32   5  1        0.0000       0.0000  0  1 region
33    1  30     0.0000   129.7000
34   36  30   175.2000   129.7000
```



Reg. 2-Reg. 5-Line Regions

```
35   40  30  175.4000  129.7000
36   86  30  400.4000  129.7000 coun
37   6  2      0.0000      0.0000  0  1 region
38   36   1  175.2000    0.0000
39   40   1  175.4000    0.0000
40   40  26  175.4000  129.5000
41   40  30  175.4000  129.7000
42   36  30  175.2000  129.7000
43    1  30    0.0000  129.7000
44    1  26    0.0000  129.5000
45   36  26  175.2000  129.5000
46   36   1  175.2000    0.0000 coun
47   7  1      10.0000      0.0000  0 -1 region
48    1   1    0.0000    0.0000
49   36   1  175.2000    0.0000
50   40   1  175.4000    0.0000
51   86   1  400.4000    0.0000 coun
52   8  1     188.0000      0.0000  0 -1 region
53    1  76    0.0000  354.7000
54   36  76  175.2000  354.7000
55   40  76  175.4000  354.7000
56   86  76  400.4000  354.7000 coun
```

Line 1: Title line, with blank in column 1 for POISSON/PANDIRA/MIRT problem.

Line 2: CON variables that have been set by AUTOMESH.
See Table 3-1 CON Variables for LATTICE for more detail.

CON(2)	=	8	– NREG – total number of regions.
CON(21)	=	0	– boundary conditions set up for upper, lower,
CON(22)	=	0	right and left boundaries, respectively, of the
CON(23)	=	1	rectangular region of the problem.
CON(24)	=	1	
CON(46)	=	1	– symmetry type is none
CON(9)	=	1.000	– CONV – coordinates are in centimeters.
skip			– the "s" designates end of CON entries; any comments may follow "s".

Line 3: This is the region entry line. LATTICE expects six entries. The variable names are identical to REG NAMELIST variables, so refer to TABLE 2-1 REG NAMELIST VARIABLES for more detail.

IREG	=	1	– the region number.
MAT	=	1	– the material number for this region. MAT = 1 – air region, when CUR = 0.
CUR	=	0.0000	– the total current, if a coil region.
DEN	=	0.0000	– the current density, if a coil region.
ITRI	=	0	– the type of triangle for the mesh. ITRI = 0 – equal weight, equilateral triangle.

IBOUND = 0 – the special region boundary indicator.
IBOUND = 0 – Dirichlet boundary for this region.
region – a comment. No "s" is required since the maximum number, (6), of entries is given.

Lines 4–15: Each line lists the horizontal and vertical mesh numbers and the corresponding coordinates. (*i.e.*, line 5: horizontal mesh number = 36 for horizontal coordinate = 175.2 cm., vertical mesh number = 1 for vertical coordinate = 0. cm.; this determines a mesh size of $dx = 175.2/(36 - 1) = 5.006$ cm. line 6: $dx = (175.4 - 175.2)/(40 - 36) = .05$ cm. for x from 175.2 cm. to 175.4 cm.) Notice that we specify points that are not needed to define region 1, but will be used to define subsequent regions. This assures that the given coordinate points are assigned mesh points and thus avoids LATTICE problems in subsequent regions.

Line 16: Mesh points and coordinates identical to line 4, to form a closed region.
coun – the "c" designates both end of entries and to count and store number of boundary point entries for this region.

Line 17–36: Are four line regions used to make: $dx = .05$ for $x = 175.2$ to 175.4 and $dy = .05$ for $y = 129.5$ to 129.7.

Line 37: Sixth region line entry.
IREG = 6 – region number 6.
MAT = 2 – iron region.
⋮
IBOUND = 1 – Neumann boundary for this region.

Lines 38–46: Lists mesh point numbers and their corresponding coordinates for region 6 as described in lines 4–16.

Line 47: Region 7 defines a line region with a fixed potential (IBOUND = -1) value (CUR) = 10. V for the lower boundary line

Lines 48–85: Lists mesh point numbers and their corresponding coordinates for region 7 as described in lines 4–16.

Line 52: Region 8 defines a line region with a fixed potential (IBOUND = -1) value (CUR) = 188. V for the upper boundary line.

Lines 53–56: Lists mesh point numbers and their corresponding coordinates for region 8 as described in lines 4–16.

First, we want to calculate the magnetic field lines without the 2 mm enclosure. In fact, we adjusted the vector potential values so that the program would generate fields of 0.5 gauss which correspond to the earth's magnetic fields at this geographic location. The input file, VECM, is identical to VECP except region 6 is omitted (lines 37–56) and line 2 of VECM has *2 7 (instead of 8 of VECP since now one region is omitted). The file VECM, input to LATTICE, and the procedural file are listed.

VECM

```
 1  vector potential problem 9/11/86
 2  *2  7 *21   0 0 1 1 *46 1 *9 1.0000 skip
 3   1  1        0.0000       0.0000  0  0 region
 4    1   1     0.0000     0.0000
 5   36   1   175.2000     0.0000
 6   40   1   175.4000     0.0000
 7   85   1   400.4000     0.0000
 8   85  26   400.4000   129.5000
 9   85  30   400.4000   129.7000
10   85  75   400.4000   354.7000
11   40  75   175.4000   354.7000
12   36  75   175.2000   354.7000
13    1  75     0.0000   354.7000
14    1  30     0.0000   129.7000
15    1  26     0.0000   129.5000
16    1   1     0.0000     0.0000 coun
17   2  1        0.0000       0.0000  0  1 region
18   36   1   175.2000     0.0000
19   36  26   175.2000   129.5000
20   36  30   175.2000   129.7000
21   36  75   175.2000   354.7000 coun
22   3  1        0.0000       0.0000  0  1 region
23   40   1   175.4000     0.0000
24   40  26   175.4000   129.5000
25   40  30   175.4000   129.7000
26   40  75   175.4000   354.7000 coun
27   4  1        0.0000       0.0000  0  1 region
28    1  26     0.0000   129.5000
29   36  26   175.2000   129.5000
30   40  26   175.4000   129.5000
31   85  26   400.4000   129.5000 coun
32   5  1        0.0000       0.0000  0  1 region
33    1  30     0.0000   129.7000
34   36  30   175.2000   129.7000
35   40  30   175.4000   129.7000
36   85  30   400.4000   129.7000 coun
37   7  1       10.0000       0.0000  0  1 region
38    1   1     0.0000     0.0000
39   36   1   175.2000     0.0000
40   40   1   175.4000     0.0000
41   85   1   400.4000     0.0000 coun
42   8  1     188.0000       0.0000  0  1 region
```

```
43    1  75     0.0000  354.7000
44   36  75   175.2000  354.7000
45   40  75   175.4000  354.7000
46   85  75   400.4000  354.7000 coun
```

COSMOS File, cvecm

```
1  *lattice
2  vecm
3  s
4  *poisson
5  tty
6  0
7  s
8  -1
9  */
```

VAX File, CVECM.COM

```
1  $RUN LATTICE
2  VECM
3  s
4  $RUN POISSON
5  TTY
6  0
7  S
8  -1
9  $EXIT
```

To execute, using the above two files — VECM and CVECM — the user types:

cosmos i = cvecm	for CRAY
@CVECM	for VAX

Fig 10.7-1 lists a few lines of the output from OUTPOI showing the fields on axis to be .503 gauss and uniform everywhere as seen by TEKPLOT plot of field lines in Fig. 10.7-5.

```
 least squares edit of problem , cycle  3270

 none      symmetry type
0 tored energy =  0.0000e+00 joules / meter or radian

 xjfact= 1.000000                                  bx       by       bt       dby/dy       dby/dx
0 k l   a(v  )        x         y          (gauss) (gauss) (gauss) (gauss/cm)  (gauss/cm)     afit
0 1 1 1.000000e+01   0.00000   0.00000    0.502   0.000   0.502   -1.5571e-08  1.9681e-08  7.1e-08
0 2 1 1.000000e+01   5.00871   0.00000    0.502   0.000   0.502   -2.4097e-08  4.3716e-09 -6.1e-09
0 3 1 1.000000e+01  10.01143   0.00000    0.502   0.000   0.502   -1.0534e-08 -5.7178e-11  3.3e-08
0 4 1 1.000000e+01  15.01714   0.00000    0.502   0.000   0.502   -2.1011e-08 -6.5921e-11  3.4e-08
0 5 1 1.000000e+01  20.02286   0.00000    0.502   0.000   0.502   -2.3697e-08 -5.5462e-11  3.5e-08
0 6 1 1.000000e+01  25.02857   0.00000    0.502   0.000   0.502   -2.6266e-08 -4.9839e-11  3.6e-08
0 7 1 1.000000e+01  30.03429   0.00000    0.502   0.000   0.502   -2.8660e-08 -4.7322e-11  3.6e-08
0 8 1 1.000000e+01  35.04000   0.00000    0.502   0.000   0.502   -3.0901e-08 -4.6446e-11  3.7e-08
```

Figure 10.7.1: Portion of output from OUTPOI for vector potential problem. Adjusted vector potential on boundaries to produce uniform fields of ~ .5 gauss

To run our original problem we use input file, VECP, that was listed and described initially, with the procedural file, CVECP, listed below.

COSMOS, File cvecp

```
1  *lattice
2  vecp
3  s
4  *poisson
5  tty
6  0
7   *6 -1 *10 .00005 s
8  -1
9  */
```

VAX File, CVECP.COM

```
1  $RUN LATTICE
2  VECP
3  S
4  $RUN POISSON
5  TTY
6  0
7  *6 -1 *10 .00005 S
8  -1
9  $EXIT
```

Line 7: CON(6) = -1 – the permeability (μ) for region 2 is finite, constant, and defined by $\gamma = 1.0/\mu$ in CON(10).

CON(10) = .00005 – γ value which means $\mu = 20000$.

To execute with the two files, VECP and CVECP user types:

```
cosmos i = cvecp        for CRAY
@CVECP                  for VAX
```

Figure 10.7-2, which lists a portion of the fields on axis, shows that the fields inside the region have decreased by a factor of ~14.

The mesh plots of TEKPLOT, Figs. 10.7-3 and 10.7-4, show the full geometry mesh and the mesh at the upper right-hand corner of the metal, respectively. Since the mesh size in the metal is 100 times smaller than outside, the mesh inside can not be seen in full geometry mesh plot. To get a plot of the mesh at the upper corner of the metal we run TEKPLOT with:

```
XMIN = 170.     XMAX = 175.4
YMIN = 124.     YMAX = 129.7
```

```
lleast squares edit of problem, cycle  6150

 none      symmetry type
Ostored energy =  0.0000e+00 joules / meter or radian
```

xjfact= 1.000000					bx	by	bt	dby/dy	dby/dx	
k	l	a(vector)	x	y	(gauss)	(gauss)	(gauss)	(gauss/cm)	(gauss/cm)	afit
1	1	1.000000e+01	0.00000	0.00000	0.036	0.000	0.036	-1.2034e-06	-1.4357e-06	-4.0e-06
2	1	1.000000e+01	5.00571	0.00000	0.036	0.000	0.036	-1.8152e-06	-3.2701e-08	1.4e-07
3	1	1.000000e+01	10.01143	0.00000	0.036	0.000	0.036	-3.7494e-06	-1.3198e-09	-8.0e-09
4	1	1.000000e+01	15.01714	0.00000	0.036	0.000	0.036	-5.6814e-06	-6.9283e-10	6.1e-09
5	1	1.000000e+01	20.02286	0.00000	0.036	0.000	0.036	-7.6450e-06	-5.3219e-10	6.5e-09
6	1	1.000000e+01	25.02857	0.00000	0.036	0.000	0.036	-9.6619e-06	-5.2011e-10	6.0e-09
7	1	1.000000e+01	30.03429	0.00000	0.036	0.000	0.036	-1.1748e-05	-5.3785e-10	5.7e-09
8	1	1.000000e+01	35.04000	0.00000	0.036	0.000	0.036	-1.3918e-05	-5.6262e-10	5.5e-09
9	1	1.000000e+01	40.04571	0.00000	0.036	0.000	0.036	-1.6187e-05	-5.9013e-10	5.5e-09
10	1	1.000000e+01	45.05143	0.00000	0.036	0.000	0.036	-1.8567e-05	-6.1895e-10	5.6e-09
11	1	1.000000e+01	50.05714	0.00000	0.036	0.000	0.036	-2.1073e-05	-6.4828e-10	5.8e-09
12	1	1.000000e+01	55.06286	0.00000	0.036	0.000	0.036	-2.3718e-05	-6.7698e-10	6.0e-09
13	1	1.000000e+01	60.06857	0.00000	0.036	0.000	0.036	-2.6517e-05	-7.0414e-10	6.2e-09
14	1	1.000000e+01	65.07429	0.00000	0.037	0.000	0.037	-2.9481e-05	-7.2856e-10	6.4e-09
15	1	1.000000e+01	70.08000	0.00000	0.037	0.000	0.037	-3.2624e-05	-7.4925e-10	6.7e-09
16	1	1.000000e+01	75.08571	0.00000	0.037	0.000	0.037	-3.5958e-05	-7.6456e-10	6.9e-09
17	1	1.000000e+01	80.09143	0.00000	0.037	0.000	0.037	-3.9493e-05	-7.7375e-10	7.2e-09
18	1	1.000000e+01	85.09714	0.00000	0.037	0.000	0.037	-4.3240e-05	-7.7500e-10	7.4e-09
19	1	1.000000e+01	90.10286	0.00000	0.037	0.000	0.037	-4.7206e-05	-7.6740e-10	7.5e-09
20	1	1.000000e+01	95.10857	0.00000	0.038	0.000	0.038	-5.1400e-05	-7.4968e-10	7.6e-09
21	1	1.000000e+01	100.11429	0.00000	0.038	0.000	0.038	-5.5827e-05	-7.2078e-10	7.6e-09
22	1	1.000000e+01	105.12000	0.00000	0.038	0.000	0.038	-6.0489e-05	-6.8023e-10	7.4e-09
23	1	1.000000e+01	110.12571	0.00000	0.039	0.000	0.039	-6.5386e-05	-6.2748e-10	7.1e-09
24	1	1.000000e+01	115.13143	0.00000	0.039	0.000	0.039	-7.0517e-05	-5.6311e-10	6.6e-09
25	1	1.000000e+01	120.13714	0.00000	0.039	0.000	0.039	-7.5876e-05	-4.8835e-10	5.7e-09
26	1	1.000000e+01	125.14286	0.00000	0.040	0.000	0.040	-8.1454e-05	-4.0672e-10	4.3e-09
27	1	1.000000e+01	130.14857	0.00000	0.040	0.000	0.040	-8.7238e-05	-3.2617e-10	2.1e-09
28	1	1.000000e+01	135.15429	0.00000	0.041	0.000	0.041	-9.3213e-05	-2.6736e-10	-1.3e-09
29	1	1.000000e+01	140.16000	0.00000	0.041	0.000	0.041	-9.9360e-05	-2.9174e-10	-7.0e-09
30	1	1.000000e+01	145.16571	0.00000	0.042	0.000	0.042	-1.0566e-04	-5.6969e-10	-1.8e-08
31	1	1.000000e+01	150.17143	0.00000	0.042	0.000	0.042	-1.1210e-04	-1.9936e-09	-4.0e-08
32	1	1.000000e+01	155.17714	0.00000	0.043	0.000	0.043	-1.1869e-04	-7.3876e-09	1.1e-07
33	1	1.000000e+01	160.18286	0.00000	0.043	0.000	0.043	-1.2540e-04	-2.2517e-08	-4.6e-07
34	1	1.000000e+01	165.18857	0.00000	0.044	0.000	0.044	-1.3052e-04	-8.3920e-08	-4.3e-06
35	1	1.000000e+01	170.19429	0.00000	0.045	0.000	0.045	-1.3342e-04	-1.6752e-06	-1.5e-05
40	1	1.000000e+01	175.40000	0.00000	0.624	0.000	0.624	-1.0173e-04	-5.1211e-05	1.4e-04
41	1	1.000000e+01	180.40000	0.00000	0.625	0.000	0.625	-8.0886e-05	-1.4813e-06	7.5e-06
42	1	1.000000e+01	185.40000	0.00000	0.625	0.000	0.625	-1.3768e-05	-1.7271e-09	7.9e-07
43	1	1.000000e+01	190.40000	0.00000	0.626	0.000	0.626	-5.0236e-05	-1.9035e-08	1.1e-07
44	1	1.000000e+01	195.40000	0.00000	0.624	0.000	0.624	-1.1136e-04	-2.4427e-08	7.1e-08
45	1	1.000000e+01	200.40000	0.00000	0.624	0.000	0.624	-1.6906e-04	-2.3206e-08	8.8e-08
46	1	1.000000e+01	205.40000	0.00000	0.623	0.000	0.623	-2.2276e-04	-2.0801e-08	1.0e-07
47	1	1.000000e+01	210.40000	0.00000	0.621	0.000	0.621	-2.7199e-04	-1.7980e-08	1.2e-07
48	1	1.000000e+01	215.40000	0.00000	0.620	0.000	0.620	-3.1636e-04	-1.5104e-08	1.2e-07
49	1	1.000000e+01	220.40000	0.00000	0.618	0.000	0.618	-3.5564e-04	-1.2309e-08	1.3e-07
50	1	1.000000e+01	225.40000	0.00000	0.616	0.000	0.616	-3.8969e-04	-9.6907e-09	1.3e-07
51	1	1.000000e+01	230.40000	0.00000	0.614	0.000	0.614	-4.1852e-04	-7.3122e-09	1.3e-07

Figure 10.7.2: Portion of output of file OUTPOI for vector potential problem with 2 mm enclosure. Listing shows fields inside region to be < .04g.

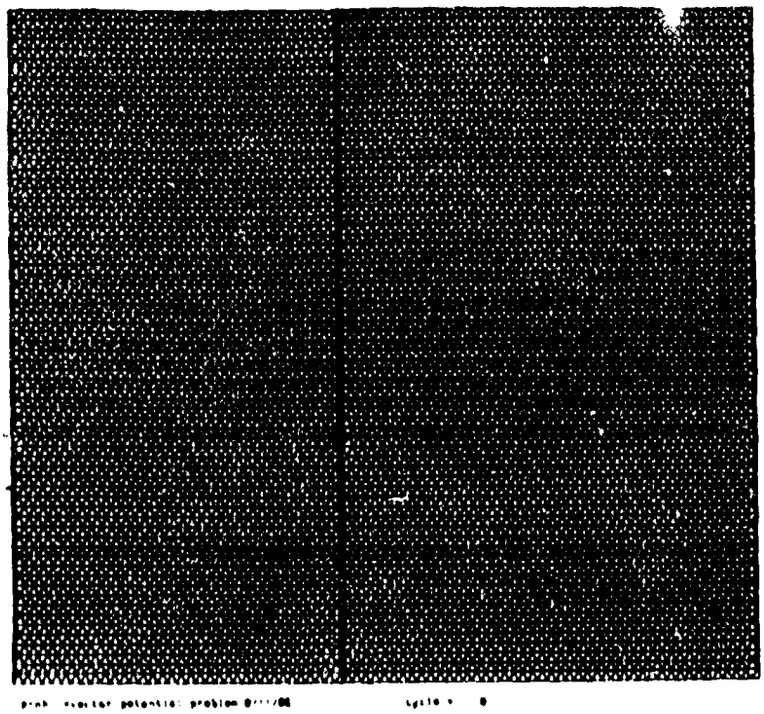


Figure 10.7.3: Mesh plot of TEKPLOT for the vector potential problem with enclosure. The "dot" is the mesh blown up below.

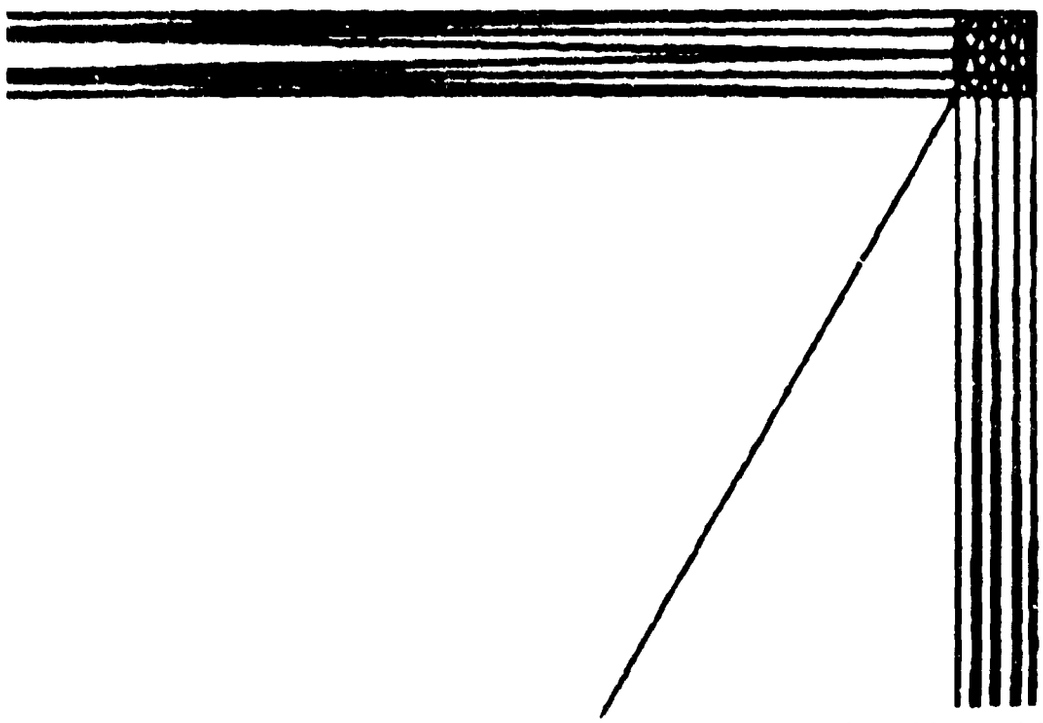


Figure 10.7.4: Mesh plot of TEKPLOT for vector potential problem with enclosure zeroed in on (170., 124.) to (175.4, 129.7).

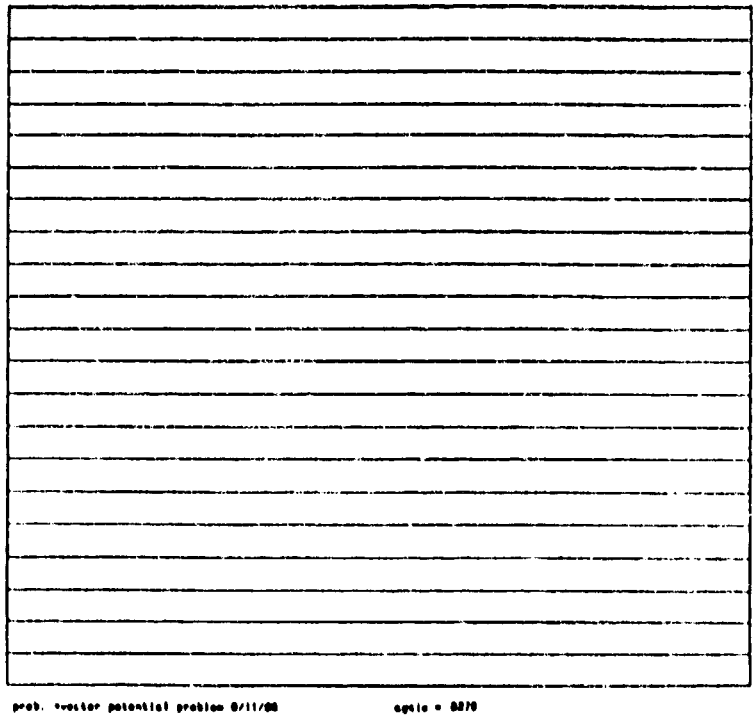


Figure 10.7.5: Field lines TEKPLOT for the vector potential problem showing uniform field lines.

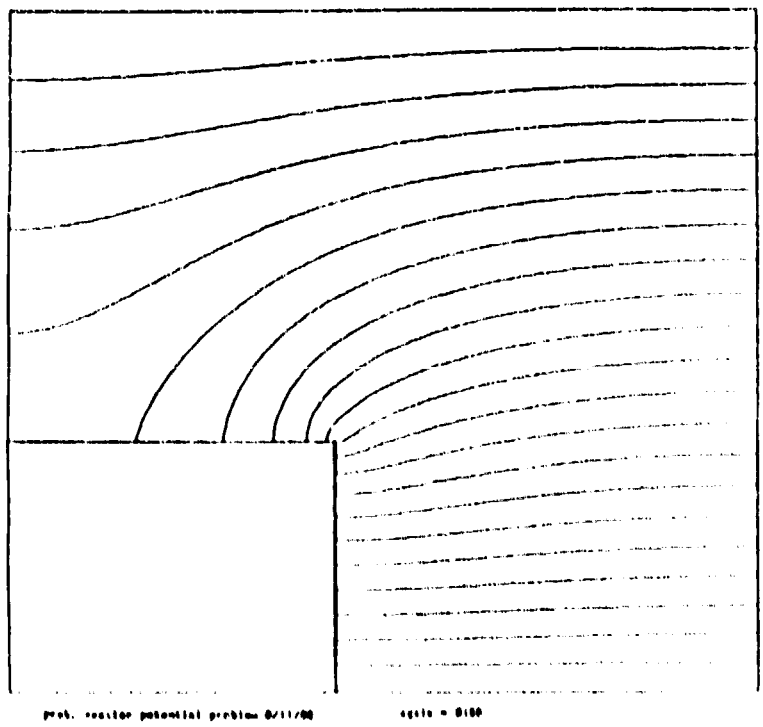


Figure 10.7.6: Field lines TEKPLOT for the vector potential problem with 2 mm-thick metal enclosure.

10.8 POISSON -- Electrostatic Problem

The electric fields are calculated for an electrostatic problem consisting of two plates at fixed potential inside an infinite pipe as shown in Fig. 10.8-1.

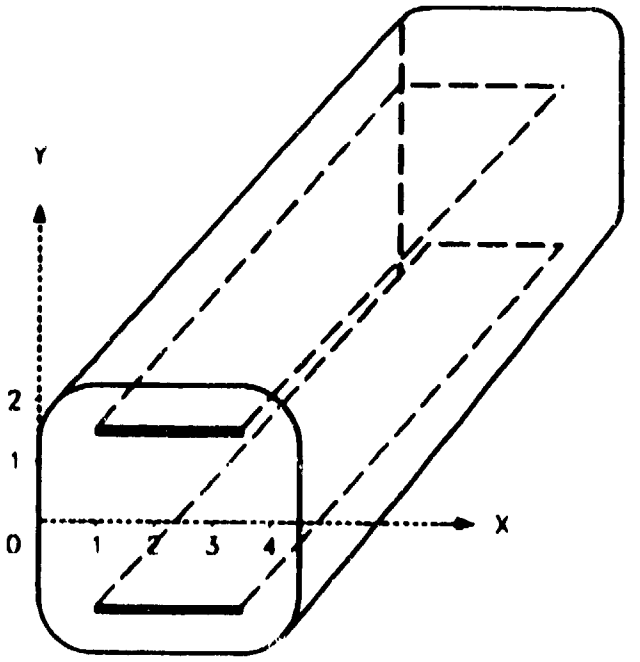
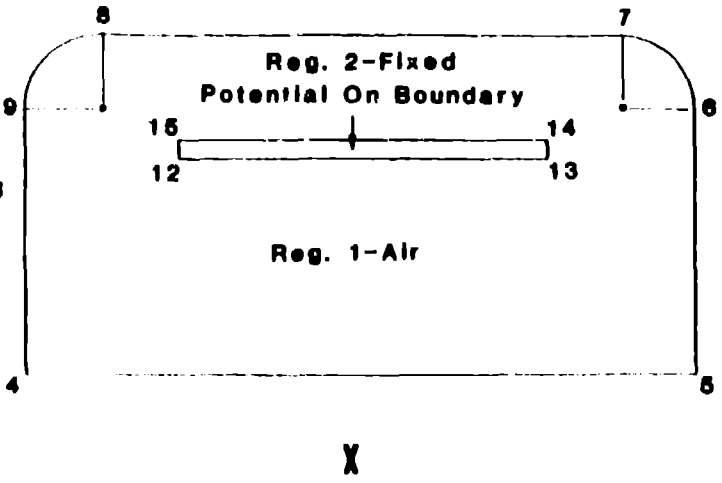


Figure 10.8-1: Two plates inside infinite pipe.

Because of the symmetry, we need only calculate one-quarter of the geometry but we here opt to do one-half of the configuration. A vertical cross section of the upper half of Fig. 10.8-1 with the corresponding AUTOMESH input file, ELEC, is given below.

```
 1 electrostatic--infinite pipe, 2 plates 9/20/86
 2  $reg nreg=2,dx=.0625,dy=.0625,xmax=5.,ymax=2.5,
 3       npoint=7$
 4  $po x=0.0000, y=0.0000$
 5  $po x=4.5000, y=0.0000$
 6  $po x=4.5000, y=1.7500$
 7  $po nt=2,x0=4.0,y0=1.75,r=0.5,theta=90.$
 8  $po x=0.5000, y=2.2500$
 9  $po nt=2,x0=0.5,y0=1.75,r=0.5,theta=180.$
10  $po x=0.0000, y=0.0000$
11  $reg mat=0,cur=500.,ibound=-1,npoint=5$
12  $po x=1.0000, y=1.4375$
13  $po x=3.5000, y=1.4375$
14  $po x=3.5000, y=1.5625$
15  $po x=1.0000, y=1.5625$
16  $po x=1.0000, y=1.4375$
```



Lines 4–12: inputs the geometry of region 1 which is given by straight lines and circular arcs as shown in the figure.

Line 13: region entries for region 2.
MAT = 0 – all points inside the region are omitted (See Fig. 10.8-3.) Points on the boundary are set according to CUR.
IBOUND = -1 – indicates fixed potential whose value is given by CUR.
CUR = 500. – fixed potential of 500. volts.

Lines 12-16: inputs geometry of region 2.

Given the above file, ELEC, and the procedural file, CELEC, which is listed below, the user types:

cosmos i=celec for CRAY

@CELEC for VAX

COSMOS File, celec

```
 1  *automesh
 2  elec
 3  *lattice
 4  tape73
 5   *9 2.54 *22 0 *66 0.
 6   *45 34 1 8
 7  *poisson
 8  tty
 9  0
10  8
11  -1
12  */
```

VAX File, CELEC.COM

```
$RUN AUTOMESH
ELEC
$RUN LATTICE
TAPE73
*9 2.54 *22 0 *66 0.
*45 34 1 8
$RUN POISSON
TTY
0
8
-1
$EXIT
```

Line: 4–10: Con array entries:

CON(9) = 2.54 – input coordinates are in inches.
CON(22) = 0 – set lower boundary for potential lines parallel
(The other 3 boundaries have default values set = 0.).

CON(66) = 0. – indicates an electrostatic problem.

CON(45) = 34 – prints electric field values up to KMAX = 34
(default, prints only values on x-axis).

CON(46) = 1 – no symmetry.

Fig. 10.8–2 shows part of the output from file OUTPOI that lists the fields on x-axis. Note that headings are "ex," "ey," and "et" which indicate an electrostatic computation.

```
 least squares edit of problem , cycle   160

 none      symmetry type
 k    l     v(scalar)        x          y          ex(v/cm)     ey(v/cm)     et(v/cm)     vfit
 1    1   0.000000e+00    0.00000    0.00000      -0.002         0.002        0.002     -8.4e-06
 2    1   0.000000e+00    0.06250    0.00000      -0.001        -5.777        5.777      1.2e-04
 3    1   0.000000e+00    0.12500    0.00000      -0.001       -11.548       11.548      5.5e-05
 4    1   0.000000e+00    0.18750    0.00000      -0.001       -17.312       17.312      1.6e-05
 5    1   0.000000e+00    0.25000    0.00000      -0.001       -23.063       23.063     -2.3e-06
 6    1   0.000000e+00    0.31250    0.00000      -0.001       -28.793       28.793      1.6e-06
 7    1   0.000000e+00    0.37500    0.00000      -0.002       -34.494       34.494     -1.0e-06
 8    1   0.000000e+00    0.43750    0.00000      -0.002       -40.154       40.154     -1.2e-05
 9    1   0.000000e+00    0.50000    0.00000      -0.002       -45.761       45.761      2.4e-06
10    1   0.000000e+00    0.56250    0.00000      -0.003       -51.300       51.300     -1.0e-05
11    1   0.000000e+00    0.62500    0.00000      -0.003       -56.752       56.752     -1.1e-05
12    1   0.000000e+00    0.68750    0.00000      -0.003       -62.099       62.099      7.8e-08
13    1   0.000000e+00    0.75000    0.00000      -0.003       -67.321       67.321     -4.5e-06
14    1   0.000000e+00    0.81250    0.00000      -0.003       -72.396       72.396     -3.5e-06
15    1   0.000000e+00    0.87500    0.00000      -0.003       -77.304       77.304     -1.7e-06
16    1   0.000000e+00    0.93750    0.00000      -0.003       -82.023       82.023     -4.4e-06
17    1   0.000000e+00    1.00000    0.00000      -0.002       -86.535       86.535     -1.8e-06
18    1   0.000000e+00    1.06250    0.00000      -0.002       -90.824       90.824     -9.5e-08
19    1   0.000000e+00    1.12500    0.00000      -0.001       -94.874       94.874     -1.9e-06
20    1   0.000000e+00    1.18750    0.00000      -0.001       -98.676       98.676      9.6e-07
21    1   0.000000e+00    1.25000    0.00000       0.000      -102.220      102.220     -2.6e-07
22    1   0.000000e+00    1.31250    0.00000       0.000      -105.504      105.504      5.2e-07
23    1   0.000000e+00    1.37500    0.00000       0.000      -108.525      108.525      7.6e-07
24    1   0.000000e+00    1.43750    0.00000       0.001      -111.286      111.286     -4.9e-07
25    1   0.000000e+00    1.50000    0.00000       0.001      -113.790      113.790      4.6e-06
26    1   0.000000e+00    1.56250    0.00000       0.001      -116.045      116.045     -5.1e-07
27    1   0.000000e+00    1.62500    0.00000       0.001      -118.058      118.058      1.9e-06
28    1   0.000000e+00    1.68750    0.00000       0.001      -119.839      119.839      2.9e-06
29    1   0.000000e+00    1.75000    0.00000       0.001      -121.398      121.398     -3.5e-06
30    1   0.000000e+00    1.81250    0.00000       0.001      -122.745      122.745      4.0e-07
31    1   0.000000e+00    1.87500    0.00000       0.001      -123.889      123.889      8.7e-07
32    1   0.000000e+00    1.93750    0.00000       0.001      -124.840      124.840      1.3e-06
33    1   0.000000e+00    2.00000    0.00000       0.001      -125.606      125.606     -3.9e-07
34    1   0.000000e+00    2.06250    0.00000       0.001      -126.194      126.194     -1.6e-06
35    1   0.000000e+00    2.12500    0.00000       0.000      -126.609      126.609      1.1e-06
36    1   0.000000e+00    2.18750    0.00000       0.000      -126.857      126.857     -2.1e-06
37    1   0.000000e+00    2.25000    0.00000       0.000      -126.938      126.938     -1.3e-06
38    1   0.000000e+00    2.31250    0.00000       0.000      -126.855      126.855     -4.7e-07
39    1   0.000000e+00    2.37500    0.00000       0.000      -126.605      126.605     -1.7e-06
40    1   0.000000e+00    2.43750    0.00000      -0.001      -126.188      126.188      1.7e-06
41    1   0.000000e+00    2.50000    0.00000      -0.001      -125.598      125.598     -2.2e-06
42    1   0.000000e+00    2.56250    0.00000      -0.001      -124.830      124.830     -1.2e-06
43    1   0.000000e+00    2.62500    0.00000      -0.001      -123.877      123.877     -3.5e-06
44    1   0.000000e+00    2.68750    0.00000      -0.001      -122.731      122.731     -2.1e-06
45    1   0.000000e+00    2.75000    0.00000      -0.001      -121.382      121.382     -1.5e-06
46    1   0.000000e+00    2.81250    0.00000      -0.001      -119.821      119.821     -3.0e-06
```

Figure 10.8 2: Part of the output of file OUTPOI for electrostatic problem.

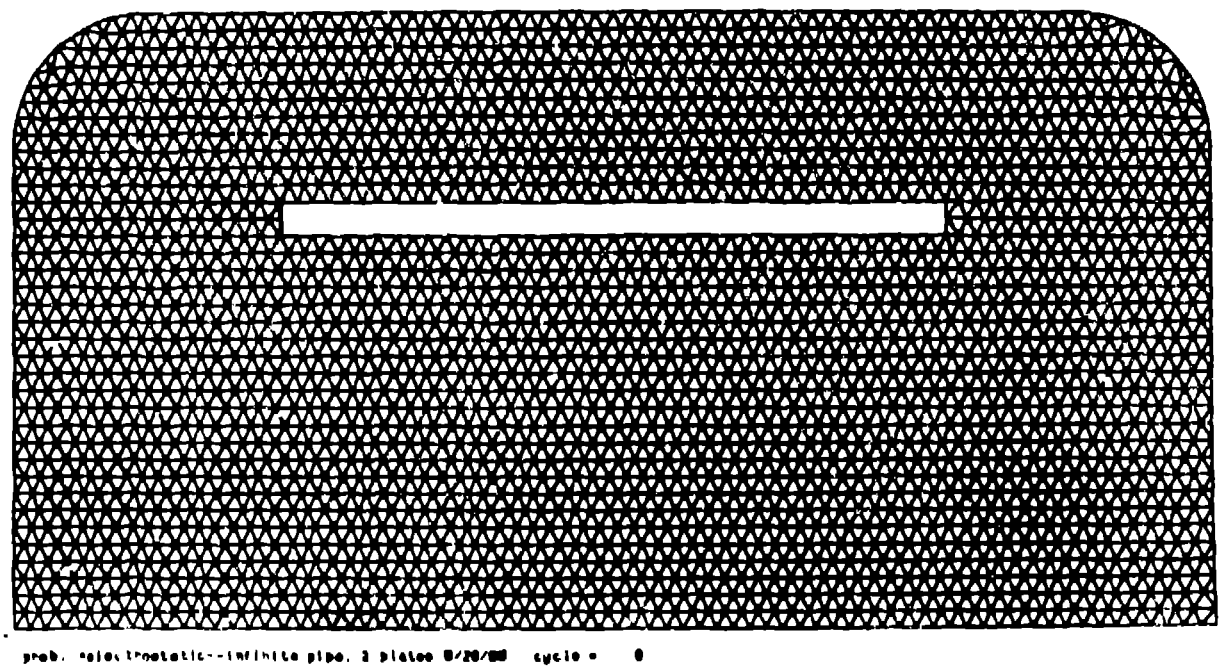


Figure 10.8-3: Mesh plot of TEKPLOT for the electrostatic problem showing that points inside plate region are omitted.

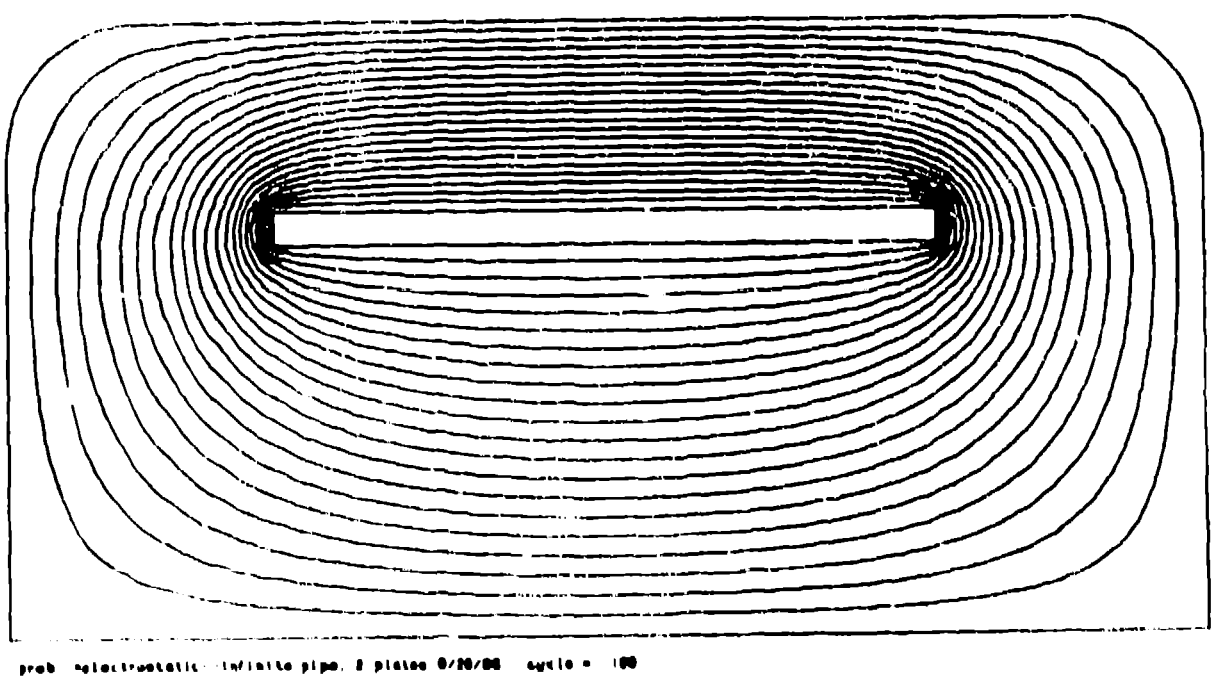


Figure 10.8-4: Equipotential line plot of TEKPLOT for the electrostatic problem. The electric field lines are perpendicular to these potential lines.

10.9 POISSON -- Septum Magnet

Fig. 10.9–1 shows a vertical cross-section picture of a septum magnet that is used for beam extraction.

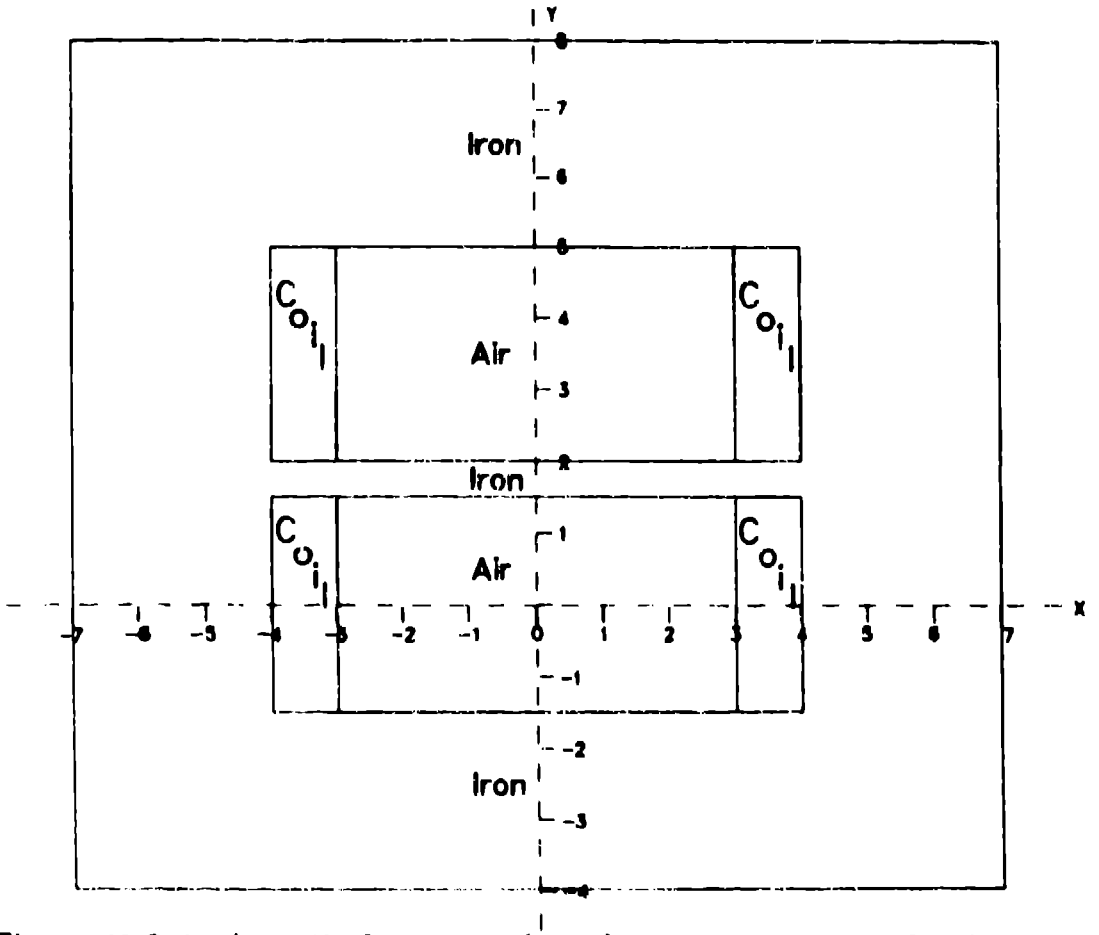


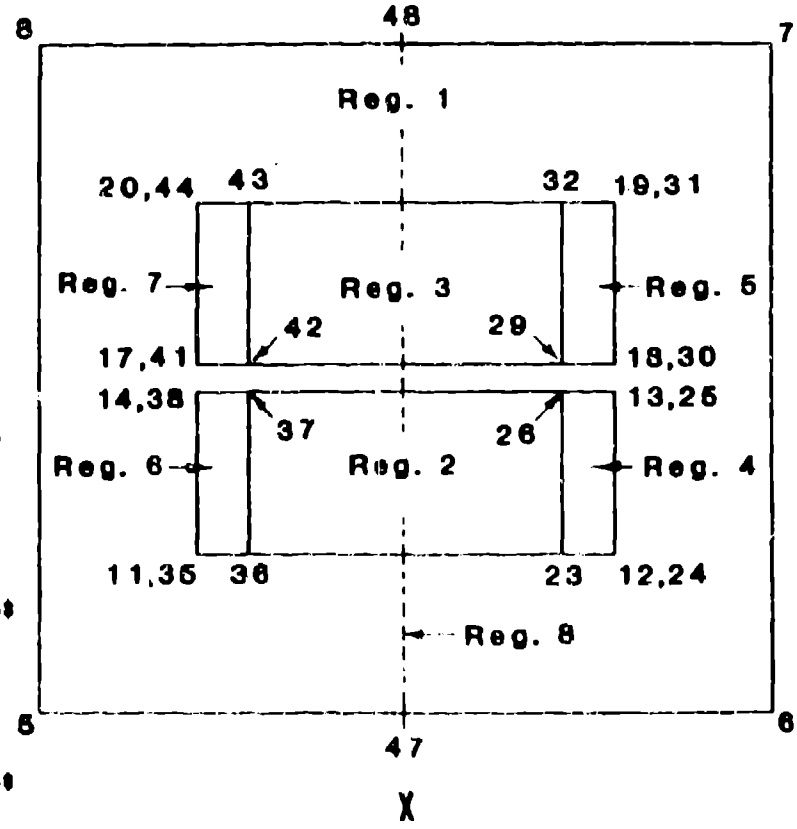
Figure 10.9–1: A vertical cross-section of a septum magnet showing iron, coil, and air regions.

The magnetic fields and their gradients are calculated for this magnet utilizing the folowing options:

- negative coordinates
- full geometry input
- off-center harmonic analysis

POISSON performs harmonic analysis only at the origin. Since we need to do the harmonic analysis at the center of the lower air gap, we are forced to put the origin there. Also because POISSON has no harmonic symmetry type for this configuration we must use the full geometry. Below is the AUTOMESH input file, SEPT, corresponding to the figure on the right.

```
 1 septum magnet 11/17/86
 2 $reg nreg=8, dx=.25,xmax= 7.,ymax= 8.0,
 3   npoint=5,ymin=-4.5,
 4    mat=2,xmin=-7.$
 5 $po x=-7.00, y=-4.50$
 6 $po x= 7.00, y=-4.50$
 7 $po x= 7.00, y= 8.00$
 8 $po x=-7.00, y= 8.00$
 9 $po x=-7.00, y=-4.50$2
10  $reg mat=1,npoint=5$
11  $po x=-4.00, y=-1.50$
12  $po x= 4.00, y=-1.50$
13  $po x= 4.00, y= 1.50$
14  $po x=-4.00, y= 1.50$
15  $po x=-4.00, y=-1.50$
16  $reg mat=1,npoint=5$
17  $po x=-4.00, y= 2.00$
18  $po x= 4.00, y= 2.00$
19  $po x= 4.00, y= 5.00$
20  $po x=-4.00, y= 5.00$
21  $po x=-4.00, y= 2.00$
22  $reg mat=1,cur=11937.,npoint=5$
23  $po x= 3.00, y=-1.50$
24  $po x= 4.00, y=-1.50$
25  $po x= 4.00, y= 1.50$
26  $po x= 3.00, y= 1.50$
27  $po x= 3.00, y=-1.50$
28  $reg mat=1,cur=-11937.,npoint=5$
29  $po x= 3.00, y= 2.00$
30  $po x= 4.00, y= 2.00$
31  $po x= 4.00, y= 5.00$
32  $po x= 3.00, y= 5.00$
33  $po x= 3.00, y= 2.00$
34  $reg mat=1,cur=-11937.,npoint=5$
35  $po x=-4.00, y=-1.50$
36  $po x=-3.00, y=-1.50$
37  $po x=-3.00, y= 1.50$
38  $po x=-4.00, y= 1.50$
39  $po x=-4.00, y=-1.50$
40  $reg mat=1,cur=11937.,npoint=5$
41  $po x=-4.00, y= 2.00$
42  $po x=-3.00, y= 2.00$
43  $po x=-3.00, y= 5.00$
44  $po x=-4.00, y= 5.00$
45  $po x=-4.00, y= 2.00$
46  $reg mat=1,ibound=0,npoint=2$
47  $po x= 0.00, y=-4.5 $
48  $po x= 0.00, y= 8.00$
```



Line 46-48: – A line region was added to set boundary conditions (IBOUND=0) to Dirichlet in order to assure that fields are parallel at the center. (Without this extra line region, POISSON has trouble converging.)

The procedural file, CSEPT, is given below.

Line No.	COSMOS File, csept	Line No.	VAX File, CSEPT.COM
1	*automesh	1	$RUN AUTOMESH
2	sept	2	SEPT
3	*lattice	3	$RUN LATTICE
4	tape73	4	TAPE73
5	*6 0 *22 0 *46 1 *110 14	5	*6 0 *22 0 *46 1 *110 14
6	37 1. 360. 1. *45 59 s	6	37 1. 360. 1. *45 59 S
7	*poisson	7	$RUN POISSON
8	tty	8	TTY
9	0	9	0
10	s	10	S
11	-1	11	-1
12	*/	12	$EXIT

Line 6–7: CON array entries.

CON(6) = 0	– use internal permeability table for region 2.
CON(22) = 0	– set lower boundary to be a Dirichlet boundary -- vector potential lines parallel. (The other 3 boundaries have default value = 0.)
CON(46) = 1	– no symmetry.
CON(110) = 14	– harmonic analysis parameters: as described in Table 5–1.9,
CON(111) = 37	here we request: 14 coefficients, 37 points on the arc of
CON(112) = 1.	a circle starting at 0° (default CON(115) up to 360°
CON(113) = 360.	with radius = 1.0 = CON(112) and normalization
CON(114) = 1.	radius = 1.0 = CON(114)).
CON(45) = 59	– prints fields and gradients in non-iron regions for all horizontal mesh (default CON(43)) up to mesh 59 vertical. (Notice that default value of CON(45) = 1 will not print any values for that is an iron region.)

Given the two files, SEPT and CSEPT, the user types:

cosmos i=csept	for CRAY
@CSEPT	for VAX

Fig. 10.9–2 shows the harmonic listings from file OUTPOI. The first part lists the POISSON calculated vector potential for the 37 angle values (only 10 are shown). The second part lists the coefficients for the vector potential and for the fields. As seen, for the "an" coefficients the odd number are the main contributors while for "bn" the even. This agrees with the theory of harmonic analyses – see Reference Manual Sec. B.13.3.

As a check, we calculated the vector potential $A(x,y)$ using the series expansion

$$A(x,y) = \sum_{n=0}^{14} (a_n + ib_n)(z/r)^n \tag{10.1}$$

where: a_n, b_n the computed vector potential coefficients "an" and "bn" (we computed a_0, b_0)

z - $x + iy$

r - 1.

Fig. 10.9–3 shows the vector potential calulated by POISSON ("vec.pot") and by the above series ("ver.ser") and the ratio of vec.ser/vec.pot ("ratio"). As seen, agreement is quite good.

```
harmonic analysis
integration radius =   1.00000
table for interpolated points
       n     angle   x coord   y coord   kf   lf       vec.pot.

       1    0.0000    1.0000    0.0000   33   22    2.94610e+03
       2   10.0000    0.9848    0.1736   33   23    2.86234e+03
       3   20.0000    0.9397    0.3420   33   24    2.69510e+03
       4   30.0000    0.8660    0.5000   33   24    2.45440e+03
       5   40.0000    0.7660    0.6428   32   25    2.15010e+03
       6   50.0000    0.6428    0.7660   32   26    1.79111e+03
       7   60.0000    0.5000    0.8660   31   26    1.38600e+03
       8   70.0000    0.3420    0.9397   31   26    9.44348e+02
       9   80.0000    0.1736    0.9848   30   26    4.78092e+02
      10   90.0000    0.0000    1.0000   29   27    3.18333e-02
                                   .
                                   .
                                   .
      36  350.0000    0.9848   -0.1736   33   21    2.93813e+03
      37  360.0000    1.0000    0.0000   33   22    2.94610e+03
table for vector potential coefficients
normalization radius =   1.00000
       a(r,y) = re( sum (an + i bn) * (z/r)**n )
        n         an             bn          abs(cn)
        1     2.9390e+03     1.1516e-02     2.9390e+03
        2     1.3716e-02    -9.7117e+01     9.7117e+01
        3     6.0210e+00     3.8391e-03     6.0210e+00
        4     6.9942e-04    -6.6583e+00     6.6583e+00
        5     8.9323e-01    -6.2872e-03     8.9326e-01
        6    -3.8325e-03    -6.5357e-01     6.5358e-01
        7     1.4967e-01     3.6255e-03     1.4972e-01
        8     3.7719e-03     2.2181e-01     2.2184e-01
        9    -1.8840e-02     5.2332e-03     1.9554e-02
       10    -1.8893e-04    -7.9247e-02     7.9247e-02
       11     2.1414e-03    -1.1381e-02     1.1581e-02
       12    -5.3281e-03     3.7245e-02     3.7624e-02
       13     2.1864e-03     9.4120e-03     9.6626e-03
       14     1.2243e-02    -2.1294e-02     2.4563e-02
table for field coefficients
normalization radius =   1.00000
       (bx - i by) = 1 + sum n*(an + i bn)/r * (z/r)**(n-1)
        n      n(an)/r        n(bn)/r      abs(n(cn)/r)
        1     2.9390e+03     1.1516e-02     2.9390e+03
        2     2.7432e-02    -1.9423e+02     1.9423e+02
        3     1.8063e+01     1.1517e-02     1.8063e+01
                                   .
                                   .
                                   .
       11     2.3556e-02     1.2619e-01     1.2739e-01
       12    -6.3937e-02     4.4094e-01     4.5149e-01
       13     2.8423e-02     1.2236e-01     1.2561e-01
       14     1.7140e-01    -2.9811e-01     3.4386e-01
```

Figure 10.9 2: Harmonic analysis output from file OUTPOI for the septum magnet problem.

```
1  a0=  5.6329900e-02  b0=  0.0000000e+00
  integration radius=   1.00000 normalization radius=   1.00000
```

j	x coord	y coord	vec.pot.	vec.ser.	ratio
1	1.0000	0.0000	2.9460989e+03	2.9461070e+03	1.0000027e+00
2	0.9848	0.1736	2.8623398e+03	2.9301284e+03	1.0264778e+00
3	0.9397	0.3420	2.6951035e+03	2.8340213e+03	1.0515445e+00
4	0.8660	0.5000	2.4543964e+03	2.6344015e+03	1.0733399e+00
5	0.7660	0.6428	2.1500967e+03	2.3451042e+03	1.0906976e+00
6	0.6428	0.7660	1.7911098e+03	1.9765518e+03	1.1036347e+00
7	0.5000	0.8660	1.3859965e+03	1.5421040e+03	1.1126320e+00
8	0.3420	0.9397	9.4434637e+02	1.0572322e+03	1.1196363e+00
9	0.1736	0.9848	4.7809195e+02	5.3777268e+02	1.1248311e+00
10	0.0000	1.0000	3.1833234e-02	9.7446528e-03	3.0611570e-01

Figure 10.9-3: Vector potential calculated by **POISSON** (vec.pot) and by the series (vec.ser.) for septum magnet.

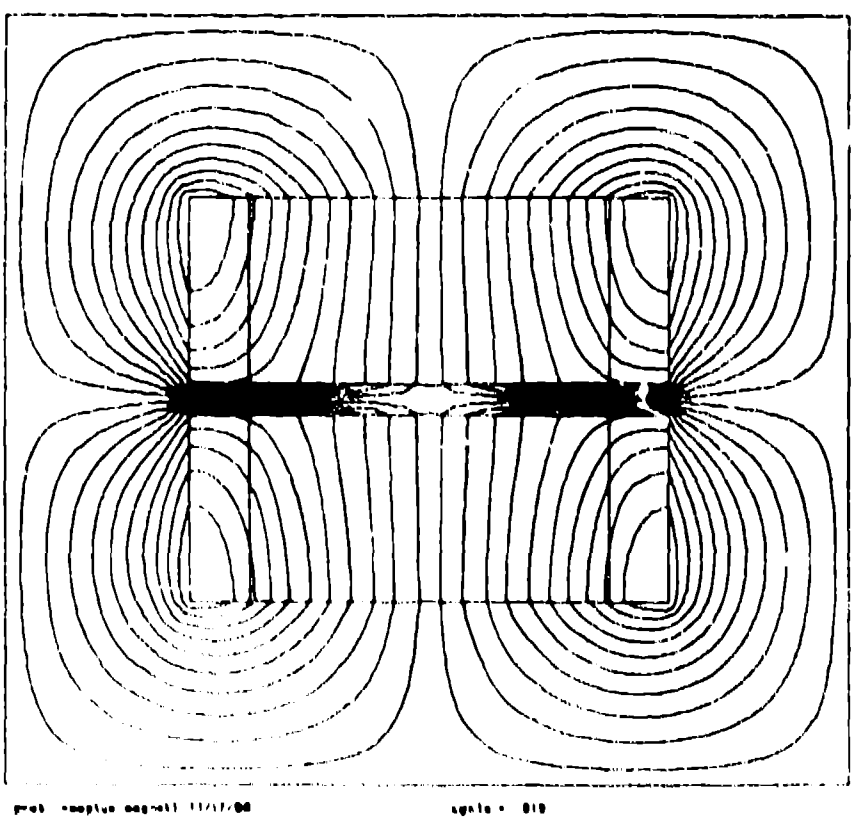


Figure 10.9-4: Field lines **TEKPLOT** for the full geometry of the septum magnet problem.

10.10 POISSON -- Dipole Magnet, Center Field Calculations

We are interested in computing the magnetic fields at the center of a dipole magnet and the fringe fields at the end. Since POISSON is a two-dimensional program, this problem is solved by two separate POISSON runs using:

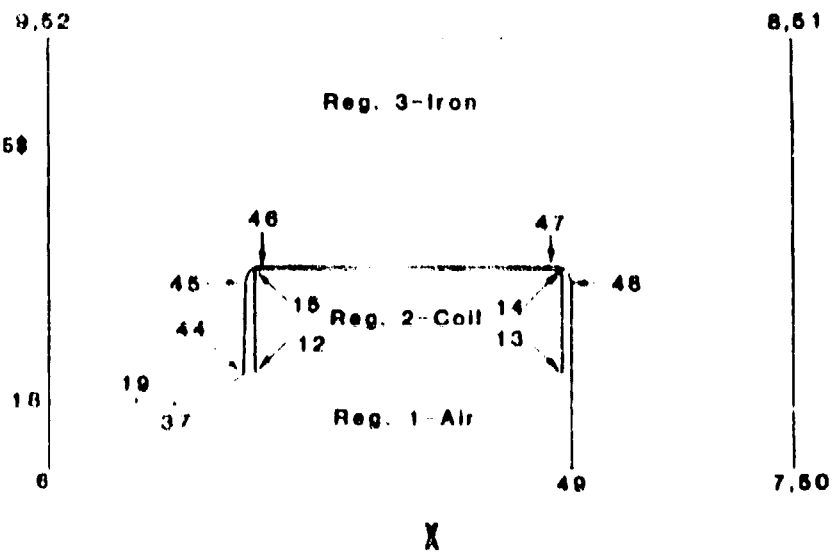
- a front vertical cross-section geometry
- a side view cross-section geometry with a return yoke

In this section we calculate the standard type geometry – that is, a front vertical cross-sectional geometry with infinite length. In Sec. 10.11 we compute the end effects of the magnet.

The pole of this dipole magnet was "trimmed" by the program MIRT to produce uniform field in a circular region of 4 cm radius at the center of the magnet. We incorporated the pole geometry generated by MIRT into the input file for AUTOMESH. Below is the AUTOMESH input file, DPFR corresponding to the geometrical configuration on the right.

DPFR

```
 1  dipole front view 10/2/86
 2  $reg nreg=3, dx= 30,dy=.50,xreg1=16.8,xreg2=43.,
 3    npoint=5, xmax=57.15,ymax=32.4,
 4    lreg1=33,yreg1=17.,lmax=42,
 5    kreg1=43,kreg2=73,kmax=82$
 6  $po x= 0.0000, y= 0.0000$
 7  $po x=57.1500, y= 0.0000$
 8  $po x=57.1500, y=32.4000$
 9  $po x= 0.0000, y=32.4000$
10  $po x= 0.0000, y= 0.0000$
11  $reg mat=1, cur=-51000.0, npoint=5$
12  $po x=15.6800, y= 7.3500$
13  $po x=39.3800, y= 7.3500$
14  $po x=39.3800, y=15.0600$
15  $po x=15.6800, y=15.0600$
16  $po x=15.6800, y= 7.3500$
17  $reg mat=3, cur=0., npoint=36$
18  $po x= 0.0000, y= 5.2400$
19  $po x= 2.0232, y= 5.2400$
20  $po x= 2.4279, y= 5.2401$
21  $po x= 2.8326, y= 5.2403$
22  $po x= 3.2371, y= 5.2407$
23  $po x= 3.6418, y= 5.2412$
24  $po x= 4.0464, y= 5.2419$
25  $po x= 4.4511, y= 5.2392$
26  $po x= 4.8557, y= 5.2369$
27  $po x= 5.2604, y= 5.2309$
28  $po x= 5.6650, y= 5.2307$
29  $po x= 6.0696, y= 5.2416$
30  $po x= 6.4743, y= 5.2536$
31  $po x= 6.8789, y= 5.2668$
32  $po x= 7.2836, y= 5.3144$
33  $po x= 7.6882, y= 5.2699$
```



```
34  $po x= 8.0929, y= 5.2233$
35  $po x= 8.4975, y= 5.1145$
36  $po x= 8.9021, y= 5.0129$
37  $po x= 9.3068, y= 4.9679$
38  $po x= 9.7114, y= 4.9793$
39  $po x=10.1161, y= 4.9850$
40  $po x=10.5207, y= 5.0350$
41  $po x=10.9254, y= 5.1000$
42  $po x=11.3300, y= 5.2400$
43  $po nt=2,x0=10.5,y0=9.3,x=2.7067,y=-3.1424,new=1$
44  $po x= 14.7800, y= 7.2500$
45  $po x=14.9900, y=14.0600$
46  $po nt=2, x0=16.26,y0=14.06,x=0.,y=1.27,new=1$
47  $po x=38.8000, y=15.3300$
48  $po nt=2, x0=38.8,y0=14.06,x=1.27,y=0.,new=1$
49  $po x=40.0700, y= 0.0000$
50  $po x=57.1500, y= 0.0000$
51  $po x=57.1500, y=32.4000$
52  $po x= 0.0000, y=32.4000$
53  $po x= 0.0000, y= 5.2400$
```

In line number 17, MAT=3 designates that the material code for region 2 will be a user-defined permeability/permittivity function. In this case it will be an input permeability table.

The procedural file CDPFR, designates that POISSON will be executed with input file, PDPFR. Given the files DPFR, CDPFR and PDPFR, the user types:

cosmos i = cdpfr	for CRAY
@CDPFR	for VAX

Line No.	COSMOS File, cdpfr	VAX File, CDPFR.COM
1	*automesh	$RUN AUTOMESH
2	dpfr	DPFR
3	*lattice	$RUN LATTICE
4	tape73	TAPE73
5	s	S
6	*poisson	$RUN POISSON
7	pdpfr	PDPFR
8	*/	$EXIT

PDPFR

```
 1  0
 2  *18 1 *6 0 s
 3  3 1.0 1 s
 4   0.0000000e+00   0.0017513135
 5   0.1142000e+04   0.0017513135
 6   0.2953000e+04   0.0010159504
 7   0.5114000e+04   0.0007821666
 8   0.8476000e+04   0.0007078644
 9   0.9667000e+04   0.0007241130
10   0.1057800e+05   0.0007562580
11   0.1131900e+05   0.0007951022
12   0.1194000e+05   0.0008375209
13   0.1245100e+05   0.0008834703
14   0.1291200e+05   0.0009293680
15   0.1331300e+05   0.0009764671
16   0.1365400e+05   0.0010253255
17   0.1393500e+05   0.0010764263
18   0.1421600e+05   0.0011254924
19   0.1444700e+05   0.0011767475
20   0.1461800e+05   0.0012313603
21   0.1478900e+05   0.0012846885
22   0.1502000e+05   0.0013315579
23   0.1513100e+05   0.0013879251
24   0.1525200e+05   0.0014423770
25   0.1542300e+05   0.0014912019
26   0.1559400e+05   0.0015389351
27   0.1570600e+05   0.0015918497
28   0.1618000e+05   0.0018542555
29   0.1684000e+05   0.0023752969
30   0.1715000e+05   0.0029154519
31   0.1736000e+05   0.0034566194
32   0.1762000e+05   0.0039729837
33   0.1783000e+05   0.0044863167
34   0.1820000e+05   0.0054945055
35   0.1895000e+05   0.0079176564
36   0.1950000e+05   0.0102564103
37   0.2020000e+05   0.0148588410
38   0.2065000e+05   0.0193798450
39   0.2095000e+05   0.0238663484
40   0.2160000e+05   0.0370370370
41   0.2190000e+05   0.0456621005
42   0.2300000e+05   0.0869565217 count
43  -1
```

where:

Line 1: 0 - to read dump number (NUM) = 0 on TAPE35 generated by LATTICE. This is the first required POISSON entry.

Line 2: CON array is the second required POISSON entry where:

- CON(18) = 1 - one permeability table to be read in.
- CON(6) = 1 - must be set to zero if CON(18) $\neq$ 0.
- s - designates end of CON array entries.

Line 3:	CON(18) $\neq 0$	– indicates optional input:
	3	– material code for which input permeability applies.
	1.0	– stacking factor.
	1	– input table given as (B, γ).
Lines 4–42:	the values of (B, γ) with last value having a "c" (count) to indicate end of table. (see Sec. 5.4.1-B for more detail.)	
Line 43:	–1	– designates the end of POISSON execution.

Fig. 10.10–1 lists the part of the output from POISSON that shows the field at the center on axis to be uniform at $\sim$ 12 kgauss.

```
least squares edit of problem , cycle  1010

midplane symmetry type
stored energy =  6.1349e+03 joules / meter or radian

xjfact=  1.000000                                  bx       by       bt          dby/dy     dby/dx
 k  l    a(vector)      x         y               (gauss) (gauss)  (gauss)     (gauss/cm) (gauss/cm)  afit
 1  1   0.000000e+00  0.00000   0.00000   0.000 12001.230 12001.230  0.0000e+00 -1.3106e-04 -8.8e-05
 2  1  -4.800495e+03  0.40000   0.00000   0.000 12001.254 12001.254  0.0000e+00  1.2050e-01  1.6e-04
 3  1  -9.601009e+03  0.80000   0.00000   0.000 12001.327 12001.327  0.0000e+00  2.4853e-01  1.1e-04
 4  1  -1.440156e+04  1.20000   0.00000   0.000 12001.455 12001.455  0.0000e+00  3.9474e-01  8.6e-05
 5  1  -1.920218e+04  1.60000   0.00000   0.000 12001.647 12001.647  0.0000e+00  5.6759e-01  6.9e-05
 6  1  -2.400289e+04  2.00000   0.00000   0.000 12001.914 12001.914  0.0000e+00  7.7553e-01  5.6e-05
 7  1  -2.880373e+04  2.40000   0.00000   0.000 12002.273 12002.273  0.0000e+00  1.0267e+00  4.4e-05
 8  1  -3.360472e+04  2.80000   0.00000   0.000 12002.742 12002.742  0.0000e+00  1.3285e+00  3.6e-05
 9  1  -3.840594e+04  3.20000   0.00000   0.000 12003.343 12003.343  0.0000e+00  1.6864e+00  4.1e-05
10  1  -4.320742e+04  3.60000   0.00000   0.000 12004.098 12004.098  0.0000e+00  2.1020e+00  8.7e-05
11  1  -4.800924e+04  4.00000   0.00000   0.000 12005.031 12005.031  0.0000e+00  2.5662e+00  2.4e-04
12  1  -5.281147e+04  4.40000   0.00000   0.000 12006.155 12006.155  0.0000e+00  3.0459e+00  6.4e-04
13  1  -5.761419e+04  4.80000   0.00000   0.000 12007.465 12007.465  0.0000e+00  3.4587e+00  1.5e-03
```

Figure 10.10 1: A portion of the output of OUTPOI showing the field values at the center of the dipole magnet.

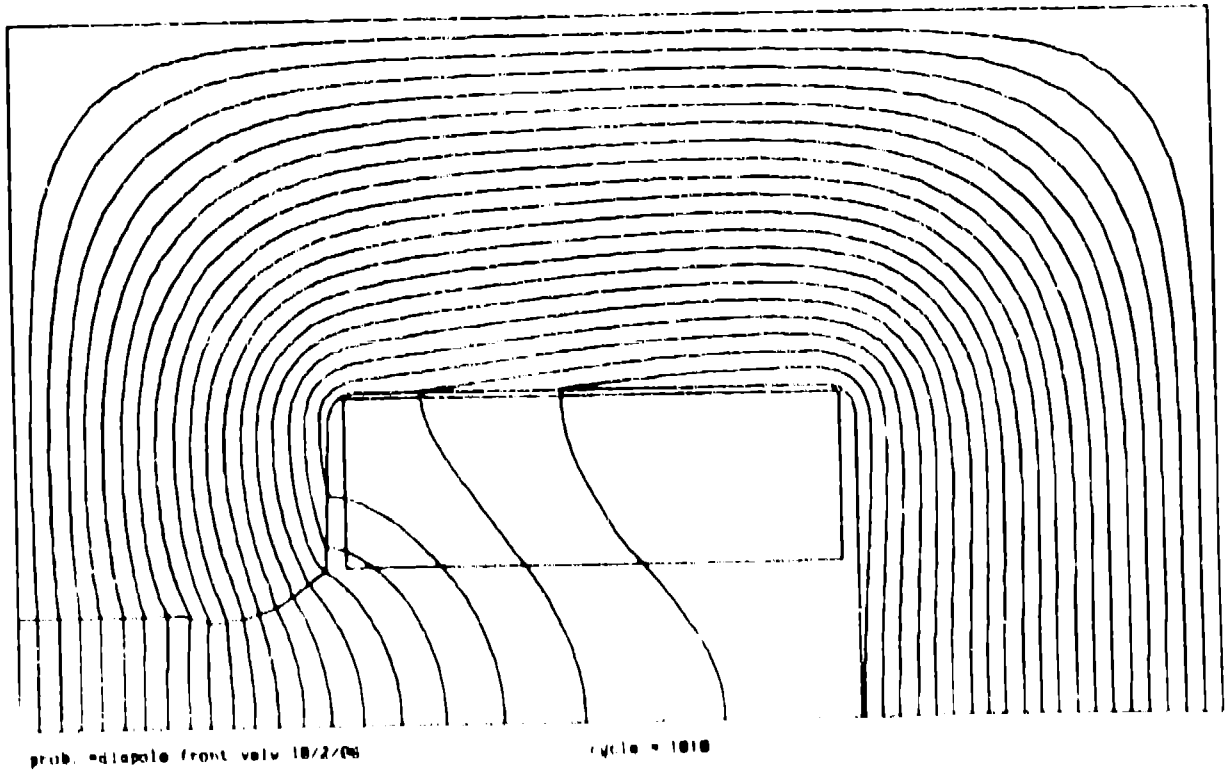


Figure 10.10 2: Field lines TEKPLOT for the dipole center magnet solution.

10.11 POISSON -- Dipole Magnet, End Field Calculation

In this section we compute the fringe fields at the end of the dipole magnet that was solved in Sec. 10.10. We now take a vertical cross-section of the side (length view) of the magnet as our input geometry and add a return leg of infinite permeability. In addition we must:

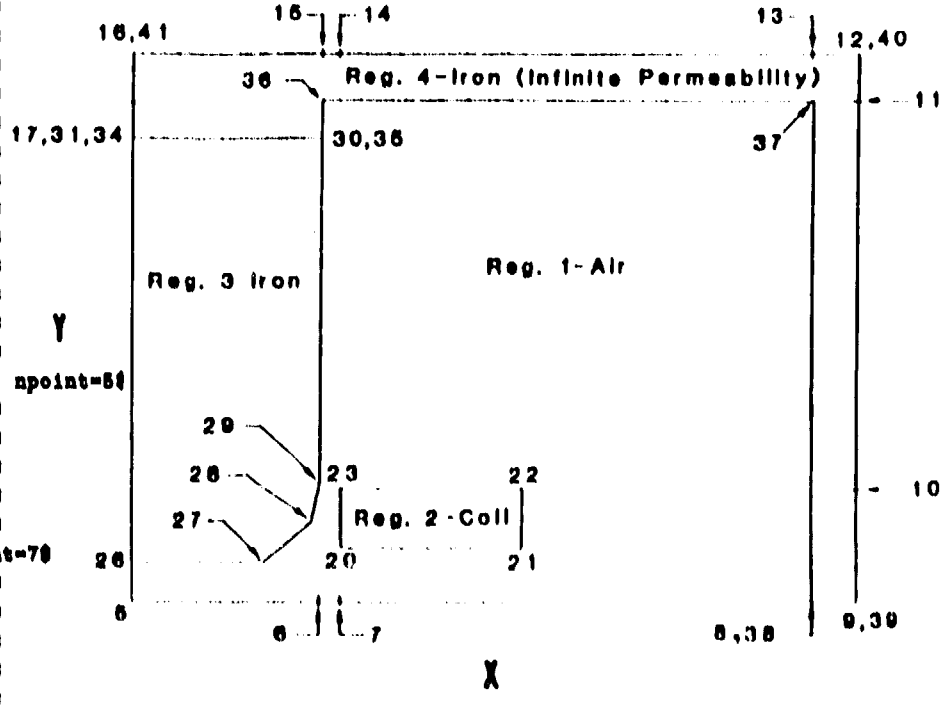
- adjust the length of the magnet to generate the same fields at the center as was obtained in the run in Sec. 10.10.
- put the return leg far enough so that this addition won't affect the fields at the area of interest.

We set up and modified the input file to AUTOMESH to meet these two conditions. First we choose the end of the magnet to be at coordinate X = 0., so we could adjust the length by changing only the initial value. We found that a length of 25 cm (initial value set to X = -25.), produced the same uniform field of ~ 12 kgauss as was obtained in run of Sec. 10.10. (see Figs. 10.10-1 and 10.11-1.)

Second, we adjusted the location and width of the return leg until the field at X = ~ 10. cm. changed by only one gauss. The final input file, DPED, to AUTOMESH and the corresponding geometrical configuration is given below.

DPED

```
 1  dipole side view-end fields 10/8/86
 2  $reg nreg=4, xmin=-25.000,xmax=71.,ymin=0.,ymax=71.,npoint=14,
 3    xreg1=26.707,kreg1=79,xreg2=65.,kreg2=101,kmax=105,
 4    yreg1=18.,lreg1=37, lmax=51,dx=.3,dy=.3$
 5  $po x=-25.000, y= 0.0000$
 6  $po x= 0.0000, y= 0.0000$
 7  $po x= 2.6710, y= 0.0000$
 8  $po x=65.0000, y= 0.0000$
 9  $po x=71.0000, y= 0.0000$
10  $po x=71.0000, y=15.0600$
11  $po x=71.0000, y=65.0000$
12  $po x=71.0000, y=71.0000$
13  $po x=65.0000, y=71.0000$
14  $po x= 2.6710, y=71.0000$
15  $po x= 0.0000, y=71.0000$
16  $po x=-25.000, y=71.0000$
17  $po x=-25.000, y=65.0000$
18  $po x=-25.000, y= 0.0000$
19  $reg mat=1, cur=-51000.0, npoint=5$
20  $po x= 2.6710, y= 7.3500$
21  $po x=26.7070, y= 7.3500$
22  $po x=26.7070, y=15.0600$
23  $po x= 2.6710, y=15.0600$
24  $po x= 2.6710, y= 7.3500$
25  $reg mat=3, cur=0., npoint=7$
26  $po x=-25.000, y= 5.2390$
27  $po x=-7.5710, y= 5.2390$
28  $po x=-1.1880, y=10.6240$
29  $po x= 0.0000, y=15.6480$
30  $po x= 0.0000, y=60.0000$
```



```
31  $po x=-25.000, y=60.0000$
32  $po x=-25.000, y= 5.2390$
33  $reg mat=4, npoint=9$
34  $po x=-25.000, y=60.0000$
35  $po x= 0.0000, y=60.0000$
36  $po x= 0.0000, y=65.0000$
37  $po x=65.0000, y=65.0000$
38  $po x=65.0000, y= 0.0000$
39  $po x=71.0000, y= 0.0000$
40  $po x=71.0000, y=71.0000$
41  $po x=-25.000, y=71.0000$
42  $po x=-25.000, y=60.0000$
```

MAT = 3 in line number 25 and MAT = 4 in line number 33 designate that the material code for region 3 and 4 will be user-defined permeability/permittivity functions. In this case it will be two input permeability tables.

The procedural file, CDPED, given below deignates that POISSON will be executed with an input file, PDPED. Given the files DPED, CDPED, and PDPED, the user types:

Line No.	COSMOS File, cdped	VAX File, CDPED.COM
1	*automesh	$RUN AUTOMESH
2	dped	DPED
3	*lattice	$RUN LATTICE
4	tape73	TAPE73
5	s	S
6	*poisson	$RUN POISSON
7	pdped	PDPED
8	*/	$EXIT

PDPED

```
 1  0
 2  *18 2 *6 0 *8 12000.  8
 3  3 1.0 1  8
 4   0.0000000e+00   0.0017513135
 5   0.1142000e+04   0.0017513135
 6   0.2953000e+04   0.0010159504
 7   0.5114000e+04   0.0007821666
 8   0.8476000e+04   0.0007078644
 9   0.9667000e+04   0.0007241130
10   0.1057800e+05   0.0007562580
11   0.1131900e+05   0.0007951022
12   0.1194000e+05   0.0008375209
13   0.1245100e+05   0.0008834703
14   0.1291200e+05   0.0009293680
15   0.1331300e+05   0.0009764671
16   0.1365400e+05   0.0010253265
17   0.1393500e+05   0.0010764263
18   0.1421600e+05   0.0011254924
19   0.1444700e+05   0.0011767475
20   0.1461800e+05   0.0012313603
21   0.1478900e+05   0.0012846865
22   0.1502000e+05   0.0013315579
23   0.1513100e+05   0.0013879251
24   0.1525200e+05   0.0014423770
25   0.1542300e+05   0.0014912019
26   0.1559400e+05   0.0015389351
27   0.1570500e+05   0.0015918497
28   0.1618000e+05   0.0018542555
29   0.1684000e+05   0.0023752969
30   0.1715000e+05   0.0029154519
31   0.1736000e+05   0.0034566194
32   0.1762000e+05   0.0039729837
33   0.1783000e+05   0.0044863167
34   0.1820000e+05   0.0054945055
35   0.1895000e+05   0.0079176564
36   0.1950000e+05   0.0102564103
37   0.2020000e+05   0.0148588410
38   0.2065000e+05   0.0193798450
39   0.2095000e+05   0.0238663484
40   0.2160000e+05   0.0370370370
41   0.2190000e+05   0.0456621005
42   0.2300000e+05   0.0869565217 count
43  4 1.0 1
44   0.0000000e+00   0.0000100000
45   1.0000000e+02   0.0000100000
46   1.0000000e+03   0.0000100000
47   1.0000000e+05   0.0000100000
48   1.0000000e+07   0.0000100000 coun
49  -1
```

where:

Line 1:	0	- to read dump number (NUM) = 0 on TAPE35 generated by LATTICE. This is the first required POISSON entry.
Line 2:	CON array is the second required POISSON entry where:	
	CON(18) = 2	- two permeability tables to be read in.
	CON(6) = 0	- must be set to zero if CON(18) $\neq$ 0.
	CON(8) = 12000.	- the current factor XJFACT is to be adjusted to produce a field of 12000 gauss at mesh point (0, 0). (the default values of CON(40) and CON(41).)
	s	- designates the end of CON array entries.
Line 3:	CON(18) $\neq$ 0	- indicates optional input:
	3	- material code for which input permeability applies.
	1.0	- stacking factor.
	1	- input table given as (B, γ).
Lines 4–42:	the values of (B, γ) with last value having a "c" (count) to indicate end of table. (see Sec. 5.4.1-B for more detail.)	
Line 43:	similar to line 3, except the input permeability table which follows applies to the region with material code (MAT) = 4.	
Lines 44–48:	the values of (B, γ) with last value having a "c" (count) to indicate end of table. Here all γ 's are equal to 10^{-5} which implies $\mu = 10^5$ for a constant infinite permeability. Note that if CON(18) > 0, then the user must supply input permeability tables for ALL material codes (MAT) for $2 \leq MAT \leq 5$. (If CON(18) = 0, can define infinite permeability for MAT = 2 by CON(6) = -2 (default value.)	
Line 43:	-1	- designates the end of POISSON execution.

Figure 10.11-1 shows that the fields are uniform at the center and start dropping as they aproach the end (X = 0., Y = 0.) of magnet. At $\sim$ 10 cm. beyond the magnet the fields are $\sim$ 20% of the peak field.

```
least squares edit of problem , cycle  2240

midplane symmetry type
stored energy =  8.2475e+03 joules / meter or radian

xjfact=  0.992465                                bx        by        bt       dby/dy       dby/dx
 k  l     a(vector)        x         y          (gauss)   (gauss)   (gauss)   (gauss/cm)   (gauss/cm)     afit
 1  1   0.000000e+00  -25.00000 0.00000  0.000  12000.957 12000.957  0.0000e+00   1.1812e-04 -1.9e-05
 2  1   8.108752e+03  -24.32432 0.00000  0.000  12000.944 12000.944  0.0000e+00  -3.8900e-02  3.2e-04
 3  1  -1.621749e+04  -23.64865 0.00000  0.000  12000.905 12000.905  0.0000e+00  -8.2996e-02  6.9e-04
 4  1  -2.432618e+04  -22.97297 0.00000  0.000  12000.832 12000.832  0.0000e+00  -1.3807e-01  9.9e-04
 5  1  -3.243482e+04  -22.29730 0.00000  0.000  12000.718 12000.718  0.0000e+00  -2.1125e-01  1.3e-03
 6  1  -4.054335e+04  -21.62162 0.00000  0.000  12000.545 12000.545  0.0000e+00  -3.1301e-01  1.5e-03
 7  1  -4.865175e+04  -20.94595 0.00000  0.000  12000.290 12000.290  0.0000e+00  -4.5850e-01  1.8e-03
 8  1  -5.675994e+04  -20.27027 0.00000  0.000  11999.917 11999.917  0.0000e+00  -6.7009e-01  2.0e-03
      .                                           .                                .
      .                                           .                                .
      .                                           .                                .
26  1  -2.006754e+05   -8.10811 0.00000  0.000  10967.989 10967.989  0.0000e+00  -4.4540e+02 -3.2e-02
27  1  -2.079779e+05   -7.43243 0.00000  0.000  10638.430 10638.430  0.0000e+00  -5.2841e+02 -1.1e-02
28  1  -2.150398e+05   -6.75676 0.00000  0.000  10256.960 10256.960  0.0000e+00  -5.9771e+02  8.3e-03
29  1  -2.218295e+05   -6.08108 0.00000  0.000   9836.328  9836.328  0.0000e+00  -6.4664e+02  2.1e-02
30  1  -2.283260e+05   -5.40541 0.00000  0.000   9388.284  9388.284  0.0000e+00  -6.7301e+02  2.2e-02
31  1  -2.345141e+05   -4.72973 0.00000  0.000   8930.579  8930.579  0.0000e+00  -6.7882e+02  1.4e-02
32  1  -2.403939e+05   -4.05405 0.00000  0.000   8474.666  8474.666  0.0000e+00  -6.6852e+02  5.1e-04
33  1  -2.459689e+05   -3.37838 0.00000  0.000   8029.686  8029.686  0.0000e+00  -6.4721e+02 -1.6e-02
34  1  -2.512487e+05   -2.70270 0.00000  0.000   7601.496  7601.496  0.0000e+00  -6.1939e+02 -3.3e-02
35  1  -2.562458e+05   -2.02703 0.00000  0.000   7193.277  7193.277  0.0000e+00  -5.8848e+02 -5.1e-02
36  1  -2.609742e+05   -1.35135 0.00000  0.000   6806.280  6806.280  0.0000e+00  -5.5674e+02 -7.6e-02
37  1  -2.654483e+05   -0.67568 0.00000  0.000   6440.505  6440.505  0.0000e+00  -5.2567e+02 -1.2e-01
38  1  -2.696822e+05    0.00000 0.00000  0.000   6095.270  6095.270  0.0000e+00  -4.9026e+02 -4.3e-02
39  1  -2.736437e+05    0.66776 0.00000  0.000   5773.110  5773.110  0.0000e+00  -4.6881e+02  2.2e-02
40  1  -2.773961e+05    1.33550 0.00000  0.000   5468.700  5468.700  0.0000e+00  -4.4311e+02 -4.0e-02
41  1  -2.809508e+05    2.00325 0.00000  0.000   5180.676  5180.676  0.0000e+00  -4.1947e+02  1.0e-01
42  1  -2.843183e+05    2.67100 0.00000  0.000   4907.936  4907.936  0.0000e+00  -3.9776e+02  3.0e-02
43  1  -2.874239e+05    3.32062 0.00000  0.000   4656.100  4656.100  0.0000e+00  -3.7810e+02  1.7e-01
44  1  -2.903703e+05    3.97024 0.00000  0.000   4416.780  4416.780  0.0000e+00  -3.5968e+02  1.2e-01
45  1  -2.931650e+05    4.61986 0.00000  0.000   4188.999  4188.999  0.0000e+00  -3.4239e+02  8.6e-02
46  1  -2.958152e+05    5.28949 0.00000  0.000   3972.007  3972.007  0.0000e+00  -3.2626e+02  7.1e-02
47  1  -2.983278e+05    5.91911 0.00000  0.000   3765.141  3765.141  0.0000e+00  -3.1114e+02  6.2e-02
48  1  -3.007091e+05    6.56873 0.00000  0.000   3567.786  3567.786  0.0000e+00  -2.9693e+02  5.6e-02
49  1  -3.029652e+05    7.21835 0.00000  0.000   3379.374  3379.374  0.0000e+00  -2.8357e+02  5.1e-02
50  1  -3.051016e+05    7.86797 0.00000  0.000   3199.377  3199.377  0.0000e+00  -2.7100e+02  4.8e-02
51  1  -3.071237e+05    8.51759 0.00000  0.000   3027.299  3027.299  0.0000e+00  -2.5917e+02  4.6e-02
52  1  -3.090364e+05    9.16722 0.00000  0.000   2862.670  2862.670  0.0000e+00  -2.4805e+02  4.4e-02
53  1  -3.108445e+05    9.81684 0.00000  0.000   2705.043  2705.043  0.0000e+00  -2.3761e+02  4.2e-02
54  1  -3.125523e+05   10.46646 0.00000  0.000   2553.996  2553.996  0.0000e+00  -2.2779e+02  4.1e-02
```

Figure 10.11 1: A portion of output from OUTPOI showing the fields inside and outside ($X = 0$) the dipole magnet, end field calculation.

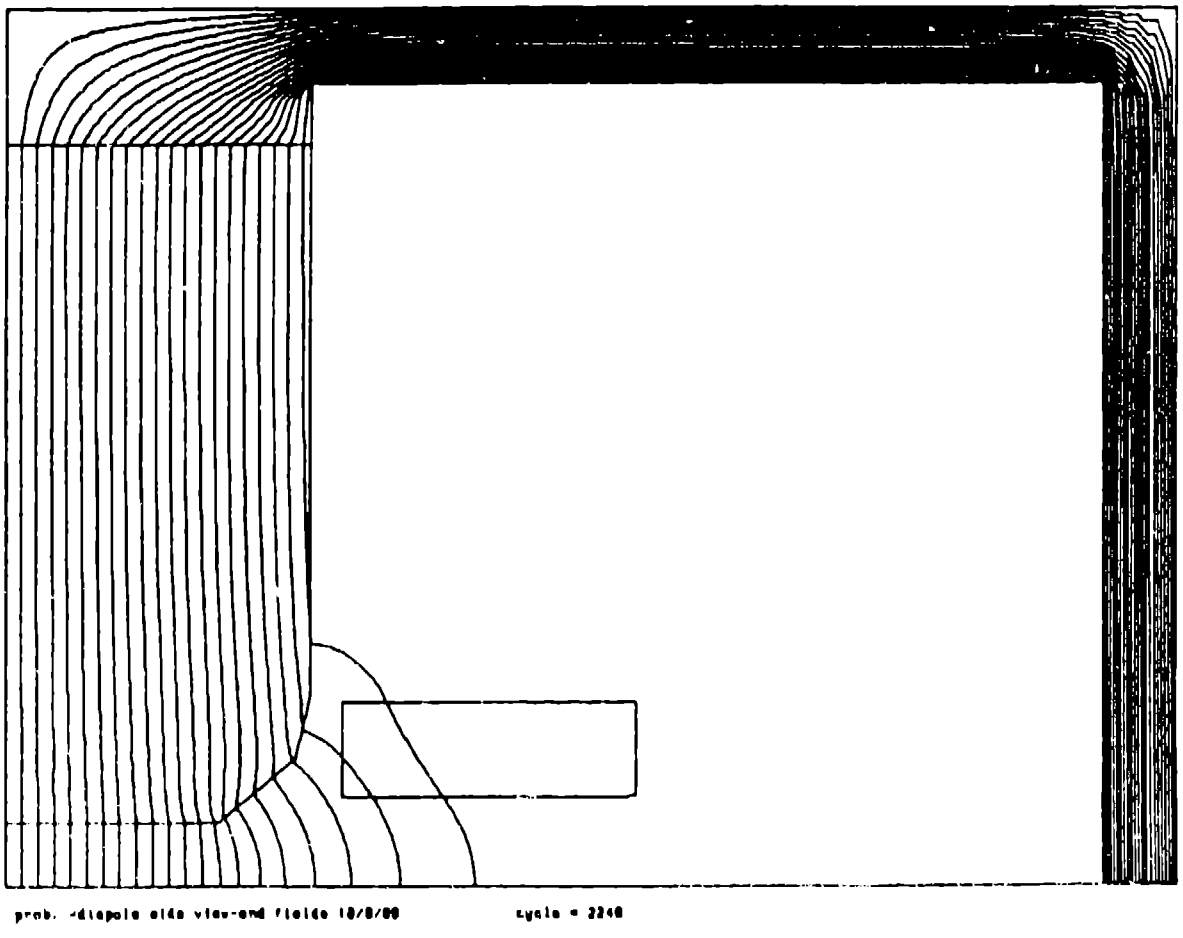


Figure 10.11–2: Field lines TEKPLOT for the dipole magnet, end field solutions.

Appendix A

Access to the Codes

Contents A-1

APPENDIX A — ACCESS TO THE POISSON/SUPERFISH GROUP CODES

A1. IN GENERAL

There are several versions of the POISSON/SUPERFISH Group Codes at various locations around the world. As a service to the user community, Los Alamos National Laboratory's Group AT-6 has undertaken the maintenance and distribution of a "standard" version of these codes.

A1.1 Copyright Notice

We have received financial support from two offices in the Department of Energy for this purpose. Therefore, to protect the interests and liabilities of the United States Government and the University of California, we make the following statement:

> Copyright 1965. The Regents of the University of California.
>
> This software was produced under a U.S. Government contract (W-7405-ENG-36) by Los Alamos National Laboratory, which is operated by the University of California for the U.S. Department of Energy. The U.S. Government is licensed to use, reproduce, and distribute this software. Permission is granted to the public to copy and use this software without charge, provided that this notice and any statement of authorship are reproduced on all copies. Neither the Government nor the University makes any warranty, express or implied, or assumes any liability or responsibility for the use of this software.

A2.1 Supported Versions

The POISSON/SUPERFISH Group Codes are written in Fortran 77 using the Tektronix graphics package PLOT10. If this package is not supported on the user's system, the conversion to another package is an easy job. One only has to change the nine calls to PLOT10 (listed and defined at the beginning of the source file TEKSO) to the equivalent calls in the user's supported package.

There are two supported versions of the codes: a VAX/VMS version and a CRAY version. Both versions are contained in the same files; lines of code which only pertain to a specific version are labeled, facilitating the transition from one version to another:

The VAX/VMS version, which is in upper-case, is shown with numbered columns:

```
 2 4 6 8(1)2 4 6 8(2)2 4 6 8(3)2 4 6 8(4)2 4 6 8(5)2 4 6 8(6)2 4 6 8(7)2 4 6 8(8)
CRAY      PARAMETER (IMX=101)                                          CRAY
     PARAMETER (IMX=71)                                              VAX
```

The CRAY version, which is in lower-case, is shown with numbered columns:

```
 2 4 6 8(1)2 4 6 8(2)2 4 6 8(3)2 4 6 8(4)2 4 6 8(5)2 4 6 8(6)2 4 6 8(7)2 4 6 8(8)
     parameter (imx=101)                                              cray
cvax      parameter (imx=71)                                             vax
```

Fig. A-1 Map of the codes on the CFS (MASS)

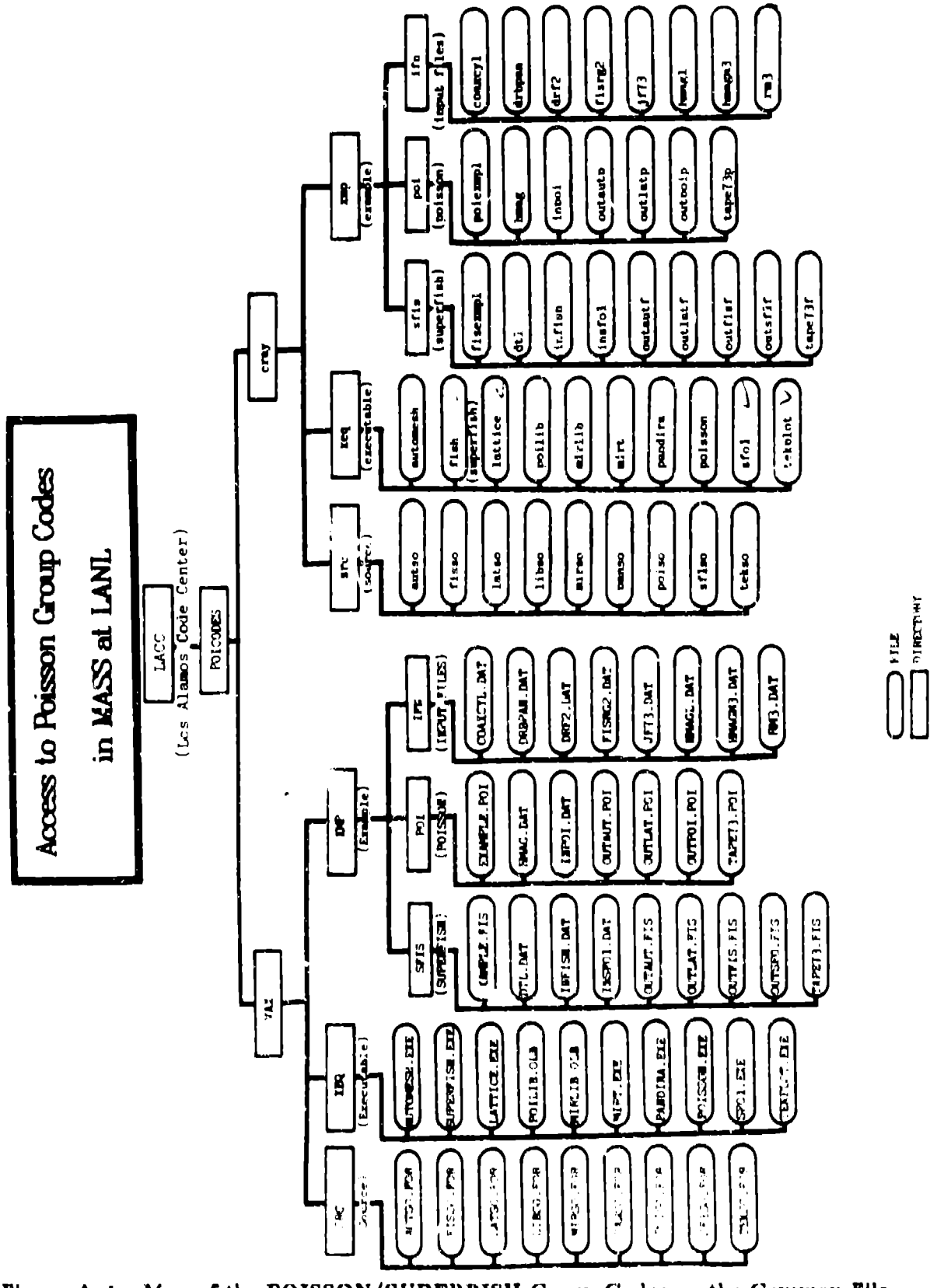


Figure A-1: Map of the POISSON/SUPERFISH Group Codes on the Common File System. Files are accessed using the utility MASS.

A2. AT LOS ALAMOS NATIONAL LABORATORY

A2.1 On the Common File System (CFS)

Source code, executable code, and example files for both the VAX/VMS and CRAY versions are on the CFS. Figure A-1 is a map of the available files and where these files are located.

MASS is a utility available on all machines connected to the Integrated Computer Network (ICN) to communicate with the CFS. Information on the use of this utility is available from the Computer Information Center (CIC) upon request. Therefore, we will not go into detail but we will present some information from their "Quick-Reference", CIC document number 809, or their "MASS Primer", CIC document number 389, to facilitate the access of these files.

To use this utility, a user simply enters the name of the utility. If no parameters are entered, the user is given a ? as a prompt.

```
mass
?
```

In this context, there are two verbs the user needs to know: `list` and `get`. Given a pathname or location as a parameter, `list` provides information on specified files and directories while `get` provides copies of specified files in the user's local file space.

A pathname to a directory or file beginning with a slash fully defines the node's location, beginning with the root directory and naming all the subdirectories encountered in reaching the node.

A pathname that does not begin with a slash is appended to the existing primary path directive, which is set by the system to `/usernumber` and may be defaulted to other settings by the user. Once the primary path directive has been set by the `default` command, this setting remains fixed throughout the current execution of the MASS utility until reset by a new `default` comand.

For example, a user can reach the directory containing the executable files by entering either of the following.

```
? list dir=/lacc/poicodes/cray/xeq
```

or

```
? default dir=/lacc/poicodes/cray
? list dir=xeq
```

If `list` is not given a parameter, the default directory is assumed.

To access an expanded title or to determine the date the file/directory was last updated, enter the following.

```
? list dir=xeq lo=g automesh
? list dir=xeq lo=g
```

To extract a file(s) from MASS, the get command is used:

```
? get dir=/lacc/poicodes/cray/xeq automesh lattice tekplot poisson
? get dir=/lacc/poicodes/cray/xmp/poi hmag
```

or

```
? default dir=/lacc/poicodes/cray
? get dir=xeq automesh lattice tekplot poisson
? get dir=xmp/poi hmag
```

or

```
mass get /lacc/poicodes/cray/xeq/automesh
mass get /lacc/poicodes/cray/xeq/lattice
mass get /lacc/poicodes/cray/xeq/tekplot
mass get /lacc/poicodes/cray/xeq/poisson
mass get /lacc/poicodes/cray/xmp/poi/hmag
```

To end the execution of the utility MASS, enter end at the prompt.

A2.2 On the Data Analysis Center's VAX Cluster on XNET

The source codes, executable codes, and sample files are available on the Data Analysis Center's VAX Cluster, Meson Physics Facility. The machines in this VAX cluster are MPX0 and MPX1, nodes 140 and 141, respectively, on XNET. Any node on XNET can be accessed by any other node (machine) on the network. It is not necessary to be validated on MPX0 and MPX1, however, validation forms are available at the Data Analysis Center's bulletin board, TA 53, MPF 24, MS H810. Files may be copied to the user's location and executable files can be run from their locations on MPX0 and MPX1. The files on machines MPX0 and MPX1 are located in the following directory.

```
ATOO$DISK:[ATOHKS.VAXFILES]
```

If the user is on MPX0 or MPX1, to copy the source file AUTSO.FOR from the ATOHKS.VAXFILES directory to the directory the user is currently in, enter the following.

```
$ COPY ATOO$DISK:[ATOHKS.VAXFILES]AUTSO.FOR *
```

If the user is on another XNET node, to copy the source file AUTSO.FOR from the ATOHKS.VAXFILES directory to the directory the user is currently in, enter the following:

```
$ COPY MPX0::ATOO$DISK:[ATOHKS.VAXFILES]AUTSO.FOR *
```

To copy all the files from the ATOHKS.VAXFILES directory to the directory the user is in, enter the following.

```
$ COPY ATOO$DISK:[ATOHKS.VAXFILES]*.* *
```

or

```
$ COPY MPX0::ATOO$DISK:[ATOHKS.VAXFILES]*.* *
```

Copying files into your space is not required; all files, except input files, can be used from the user's current directory. The following are two examples.

```
$ RUN ATOO$DISK:[AT6HKS.VAXFILES]AUTOMESH
```

or

```
$ RUN MFXO::ATOO$DISK:[AT6HKS.VAXFILES]AUTOMESH
```

A.3 OUTSIDE OF LOS ALAMOS NATIONAL LABORATORY

A.3.1 In General

The complete set of POISSON/SUPERFISH Group Codes source files plus sample input and output for both a magnet problem (POISSON) and a radio-frequency cavity problem (SUPERFISH) are available to individuals as well as institutions upon request. Please let us know which version (VAX/VMS or CRAY) is desired and your primary area of interest.

A3.2 On Networks

We are located on ARPANET which is available through BITNET, UUCP, as well as other networks. Our address is the following.

```
hks@lanl.arpa
```

A3.3 On Magnetic Tapes

We request that the user send us a tape (a seven inch reel is sufficient) and information that would facilitate installation of the codes on the target system. In each of the following subsections, we have roughly outlined the type of information that might be helpful. Your system manager should be able to provide this information if there are any questions.

A.3.3.1 VA[illegible] Machines

Tapes are avi[illegible] with the utilities COPY and BACKUP. Please specify which is preferable.

Tapes made with [illegible]ity COPY are: 9 track, 1600 bits per inch, 80 ASCII characters per line, and 512 bytes per block. To copy files from tape to disk, first mount the tape on the tape-reader unit, then enter the following.

```
$ ALLOCATE UnitName
$ MOUNT UnitName: Label
$ COPY UnitName:[ ]*.* *
```

The format for tapes made with the utility BACKUP are always the same. To copy files from tape to disk, first mount the tape on the tape-reader unit, then enter the following.

```
$ ALLOCATE UnitName
$ MOUNT/FOREIGN UnitName: Label
$ BACKUP UnitName:SaveSetName DiskName:[DirectoryName]/OWNER_UIC=ORIGINAL
```

A3.3.2 Other Machines

Please supply the following information:

1. Target Machine and Operating System

2. Choice of 1:
   - unlabeled, 9 track, 1600 bits per inch
     80 ASCII characters per record
     48 records per block
   - unlabeled, 9 track, 1600 bits per inch
     80 EBCDIC characters per record
     48 records per block

Appendix B

Program Construction

Contents B-1

B1. VAX/VMS Version

- The order in which the files are linked is important. These codes use 2 libraries that contain common subroutines used by the programs. The libraries must be created and in your local file space before an object file which calls it is linked.

  POILIB — This library is created from the object file LIBSO. It contains subroutines used by all the programs except AUTOMESH.

  MIRLIB — This library is created from the object files LIBSO and POISO. It contains subroutines used by the program MIRT.

- To run an executable file, enter RUN followed by the name of the file.

```
RUN AUTOMESH
```

To Create:	The Process:	
POILIB	FORT LIBR/CREATE	LIBSO POILIB LIBSO
AUTOMESH	FORT LINK/EXEC=AUTOMESH	AUTSO AUTSO
LATTICE	FORT LINK/EXEC=LATTICE	LATSO LATSO, POILIB/LIB
PANDIRA	FORT LINK/EXEC=PANDIRA	PANSO PANSO, POILIB/LIB
POISSON	FORT LINK/EXEC=POISSON	POISO POISO, POILIB/LIB
MIRLIB	LIBR/CREATE	MIRLIB POISO, LIBSO
MIRT	FORT LINK/EXEC=MIRT	MIRSO MIRSO, MIRLIB/LIB
SFO1	FORT LINK/EXEC=SFO1	SF1SO SF1SO, POILIB/LIB
SUPERFISH	FORT LINK/EXEC=SUPERFISH	FISSO FISSO, POILIB/LIB
TEKPLOT	FORT LINK/EXEC TEKPLOT	TEKSO TEKSO, POILIB/LIB, USER$OLB:PLOT10/LIB

B2. Cray Version

- The order in which the files are compiled is important. These codes use two libraries that contain common subroutines used by the programs. The libraries must be created and in your local file space before a binary file which calls it is loaded.

  POILIB — This library is created from the source file LIBSO. It contains subroutines used by all the programs except AUTOMESH.

  MIRLIB — This library is created from the source files LIBSO and POISO. It contains subroutines used by the program MIRT.

SOURCE FILE	BINARY FILE	EXECUTABLE FILE
autso		automesh
fisso		fish
latso		lattice
libso	poilib	
mirso		mirt
panso		pandira
poiso	mirlib	poisson
sflso		sfol
tekso		tekplot

- The first 2 lines of all source files contain input data for the Cray job control language utility XEQ, which produces binary and executable code.

- All files are compiled/loaded in the same manner.

```
xeq autso
```

The argument of autso to the utility XEQ produces the executable code AUTOMESH. To run AUTOMESH, enter the following.

```
automesh
```

Appendix C

Complete List of CON Variables for POISSON/PANDIRA/MIRT

Contents C-1

Complete tables of the 125 CON array variables, both in numerical and alphabetical order, for the POISSON/PANDIRA/MIRT programs are given in this section. The default values in these tables correspond to the values assigned by the execution program if there were no user input changes of the CON array in LATTICE.

There are a number of variables which are used internally by the program and over which the user has no control. These variables are designated by ⊗ preceding the element. In addition, those elements that must be changed in LATTICE, if they are to have an effect on the problem, are preceeded by a †.

TABLE C-1
NUMERIC ORDER OF CON VARIABLES
FOR POISSON/PANDIRA/MIRT

where:

⊗ used internally

† must be changed in LATTICE

	Number	Name	Default	Description
	CON(1)	KPROB	0	Set by LATTICE to differentiate type of problem. KPROB = 0 – POISSON/PANDIRA/MIRT. = 1 – SUPERFISH.
†	CON (2)	NREG	None	Total number of regions for the problem.
	CON(3)	LMAX	None	Total number of mesh points in the vertical
	CON(4)	KMAX	None	(LMAX) and horizontal (KMAX) direction. Input or computed either by AUTOMESH or by LATTICE.
⊗	CON(5)	IMAX	None	IMAX = KMAX + 2 computed by LATTICE.
	CON(6)	MODE	−2	The permeability code in iron. MODE = −2 – μ-infinite in iron. = −1 – μ-finite, constant and defined by CON(10) = FIXGAM, ($\gamma = 1.0/\mu$). = 0 – Option a function of REG NAMELIST parameter MAT. (See Table 2-1.) MAT = 2 – μ-finite and defined by internal table (very low-carbon steel) or user-defined constant permeability/permittivity or user-defined stacking (fill) factor. 3 ≤ MAT ≤ 5 – μ-finite and defined by table1 - table3. or user-defined constant permeability/permittivity or user-defined stacking (fill) factor. 6 ≤ MAT ≤ 11 permanent magnet material with a user-defined $B(H)$ function.

Appendix C — Complete List of CON Variables

	Number	Name	Default	Description
	CON(7)	STACK	1.0	Stacking or fill factor for iron regions using MAT = 2 (Table 2-1).
	CON(8)	BDES	1.0E+15	The value of the field, |B| = BDES at mesh location [KBZERO = CON(40), LBZERO = CON(41)]. If BDES ≠ 1.0E+15, the current factor, XJFACT = CON(66) will be adjusted so that |B| = BDES within a tolerance XJTOL = CON(67) (see example, Sec. 10.2).
†	CON(9)	CONV	1.0	Conversion factor for coordinate units. CONV = (no. of centimeters) per (unit), e.g.: CONV = 1.0 - Centimeters. = 0.1 - Millimeters. = 2.54 - Inches.
	CON(10)	FIXGAM	.004	The value of γ (= $1.0/\mu$) used in a μ-finite but constant solution. [CON(6) = MODE = −1] (See example, Sec. 10.7.) Also used to initialize γ for μ-finite and variable [CON(6) = MODE = 0].
⊛	CON(11)	NAIR	0	Total number of mesh points in the air (NAIR) and
⊛	CON(12)	NFE	0	iron (NFE) regions. Computed in LATTICE.
⊛	CON(13)	NINTER	0	Total number of interface points. An interface point is a point whose nearest neighbors are a combination of air and iron points.
	CON(14)		None	Not used.
⊛	CON(15)	NPINP	None	Total number of points in the problem NPINP = NAIR + NFE + NINTER + NBND + NSPL.
⊛	CON(16)	NBND	0	Total number of Dirichlet boundary points computed in LATTICE.
⊛	CON(17)	NSPL	0	The number of points held at special fixed potential computed in LATTICE and used in POISSON/ PANDIRA/MIRT.

Appendix C — Complete List of CON Variables

	Number	Name	Default	Description
	CON (18)	NPERM	0	The number of permeability functions to be read in as data. NPERM ≠ 0 – must set CON(6) = MODE = 0 and the first optional input data listing the permeability/permittivity functions and/or tables must follow the CON array entry (see Sec. 5.4.1).
	CON(19)	ICYLIN	0	Coordinate system indicator ICYLIN = 0 – Cartesian (x, y) coordinates = 1 – cylindrical (r, z) coordinates (horizontal → x → r) (vertical → y → z)
	CON(20)	INPUTA	0	The number of special fixed potential values to be read in as data by POISSON/PANDIRA. INPUTA > 0 – second optional input data listing mesh points and fixed potential values must be given (see Sec. 5.4.2).
†	CON(21)	NBSUP	0	Indicator for boundary conditions on the UPper,
†	CON(22)	NBSLO	1	LOwer, RighT and LeFt boundaries of the rectan-
†	CON(23)	NBSRT	0	gular region defining the boundary.
†	CON(24)	NBSLF	0	0 – Indicates Dirichlet boundary conditions, which means magnetic field lines are PARALLEL to the boundary line. 1 – Indicates Neumann boundary conditions, which means magnetic field lines are PERPENDICULAR to the boundary line.
⊛	CON(25)	NAMAX	0	The number of elements in the GTV and GTL arrays.
⊛	CON(26)	NWMAX	0	The number of points for recalculating couplings
⊛	CON(27)	NGMAX	0	(NWMAX) and gammas (NGMAX).

	Number	Name	Default	Description
⊗	CON(28)	NGSAM	0	The number of points for recalculating gammas.
	CON(29)	LIMTIM	0	The amount of time remaining in the run. At present deactivated by a comment.
	CON(30)	MAXCY		Maximum number of iteration cycles.
			100000	POISSON. (If not converging, decrease MAXCY and rerun to get a dump).
			20	PANDIRA. (If problem terminates before converging, increase MAXCY and continue from current dump).
	CON(31)	IPRFQ	0	An indicator for the cycle iteration print frequency for POISSON only. IPRFQ = 0 – POISSON determines frequency print. IPRFQ > 0 – Prints every IPRFQ cycles. IPRFQ must be a multiple of IVERG = CON(87).
†	CON(32)	IPRINT	0	Indicator for additional printout IPRINT = −1 – LATTICE input only. = 0 – no additional printout. = 1 – print the vector potential array. = 2 – print the |B| in iron regions. = 4 – print the B_x, B_y in iron regions. = sum – a combination of any of the above three options (i.e. IPRINT = 7 = 1 + 2 + 4 will give all three options).
	CON(33)		None	Not used.
	CON(34)	INACT	−1	An indicator used in interactive POISSON/PANDIRA run to allow user interaction. INACT −1 – no interaction 1 program stops at each iteration cycle, queries the user and proceeds according to typed value: GO continues to next iteration. IN inquires for new CON values before proceeding to next iteration. NO run terminates and results are written on TAPE35 and on OUTPOI/OUTPAN.

Appendix C — Complete List of CON Variables

Number	Name	Default	Description
CON(35)	NODMP	0	Indicator to write TAPE35 dump at completion of POISSON/PANDIRA run. NODMP = 1 – do not write dump. = 0 – write dump.
CON(36)	IRNDMP	0	An indicator used in MIRT only. IRNDMP = 0 – uses one dump of TAPE35. = 1 – uses two dumps of TAPE35 when continuing a simultaneous μ-finite/μ-infinite optimization.
† CON(37)	MAP	1	For POISSON/MIRT: A parameter in the conformal transformation [$\omega = z$ ** MAP/[MAP * RZERO ** (MAP-1)] where: RZERO = CON(125) MAP ≠ 1 – the current density is adjusted to conform to the transformed geometry in all closed regions. MAP = 1 – no current density adjustment. Note: if no current density adjustment is wanted (user has input the correct density for the transformed geometry), MAP should not be input until execution of POISSON/PANDIRA.
CON(38)	XORG	0.0	The real part of z_n used to specify the origin in the polynomial expansion for vector potential $A(x,y) = Re[\sum c_n(z - z_n)^n]$, the derivatives of which give the field and gradient. XORG = 0.0 for cylindrical coordinates. NOTE: For programs PRIOR to 11/10/86 if XMIN ≠ 0., user MUST set XORG = XMIN for correct field calculation. XMIN is a REG NAMELIST parameter, (Sec. 2..2.2), and not the XMIN of CON(54).

Number	Name	Default	Description
CON(39)	YORG	0.0	The imaginary part of z_o described in CON(38). NOTE: For programs PRIOR to 11/10/86 if YMIN $\neq$ 0., user MUST set YORG = YMIN for correct field calculation. YMIN is a REG NAMELIST parameter, (Sec. 2.2.2), and not the YMIN of CON(55).
CON(40)	KBZERO	1	The vertical and horizontal mesh coordinates specifying the location of BDES [CON(8)] for adjusting the current factor.
CON(41)	LBZERO	1	
CON(42)	KMIN	1	The mesh point limits of the region in which the fields and gradients are to be calculated and written on the files OUTPOI/OUTPAN for non-iron regions only. [Use CON(32) for IRON regions]. Default value writes fields and gradients at all mesh points on horizontal axis (L = 1). To get values for all geometry set LTOP to value of LMAX [KMAX, LMAX values listed as CON(3), CON(4) in files OUTLAT and OUTPOI/ OUTPAN.] (See example, Sec. 10.2.)
CON(43)	KTOP	KMAX	
CON(44)	LMIN	1	
CON(45)	LTOP	1	

Number	Name	Default	Description
CON(46)	ITYPE	2	A code specifying the problem symmetry. For Cartesian symmetry: ITYPE = 1 - no symmetry. ITYPE = 2 - midplane symmetry. ITYPE = 3 - elliptic aperture quadrupole. ITYPE = 4 - symmetric quadrupole. ITYPE = 5 - skew elliptic aperture quadrupole. ITYPE = 6 - symmetric "H" magnet or elliptical aperture sextupole. ITYPE = 7 - symmetric sextupole. ITYPE = 8 - elliptic aperture octupole. ITYPE = 9 - symmetric octupole. For all of the above symmetry codes, except ITYPE = 1 or = 5, field lines are perpendicular to the x-axis. For ITYPE = 5, the x-axis is a field line. For cylindrical symmetry: ITYPE = 1 - no symmetry. ITYPE = 2 - midplane symmetry. vector problems—field lines perpendicular to r-axis. scalar problems—potential (v) lines perpendicular to r-axis. ITYPE = 3 - midplane symmetry for scalar problems only r-axis is a v=constant line. NOTE: If in doubt as to the type of symmetry, use ITYPE = 1 or = 2 and set boundary conditions by CON(21) - CON(24), (see Secs. 2.2.5 and Table 3-1). For further detail on problem symmetry, consult POISSON/PANDIRA Reference Manual, (Sec. B.5.3.2).
CON(47)	W2ND	0.125	The weight factor for the second nearest neighbors used in determining the c_n in the polynomial expansion for the vector potential $A(x,y) \quad Re[\sum c_n(z \quad z_0)^n]$.

	Number	Name	Default	Description
	CON(48)	ISECND	1	Indicator for use of first or second neighbors in determining the c_n above ISECND = 1 - first and second = 0 - first neighbors only. (Use this option if a problem has trouble converging.)
	CON(49)	NFIL	0	The number of current filaments to be read in as data by POISSON/PANDIRA. NFIL > 0 - third optional input data listing mesh points and current filaments must be given (see Sec. 5.4.3).
	CON(50)	IHDL	100000	The number of cycles between making a quasi-integral $H \cdot dl$ calculation around the Dirichlet boundary during POISSON iteration. Increasing IHDL sometimes speeds the convergence, particularly for nonsymmetric "H" magnets.
⊗	CON(51)	NPONTS	0	LATTICE - the no. of points in the relaxation mesh in POISSON/PANDIRA. - N = NAIR[CON(11)] + NINTER[CON(13)] NPONTS = N if MODE[CON(6)] ≤ −2 NPONTS = N + NFE [CON(12)] if MODE[CON(6)] < −2.
	CON(52)	OMEGA0 or OMEGA	.001	An overrelaxation parameter used only in LATTICE and POISSON.
⊗	CON(53)	IRMAX	25	Used in LATTICE for optimization of the overrelaxation factors.
	CON(54)	XMIN	0.0	The vertical and horizontal limits of the region in which the fields and the gradients are calculated for the computed values of (x,y) or (r,z), (not necessarily on a mesh point) and written on file OUTPOI or OUTPAN for AIR regions only. The coordinates are computed by starting from XMIN, YMIN and incrementing by DX, DY where: DX (XMAX-XMIN)/(KTOP-1) [KTOP CON(43)] DY (YMAX-YMIN)/(LTOP-1) [LTOP CON(45)] up to XMAX, YMAX (see example, Sec. 10.2).
	CON(55)	XMAX	0.0	
	CON(56)	YMIN	0.0	
	CON(57)	YMAX	0.0	
	CON(58)	IBOUT	0	Presently not used. Deactivated by a comment.

	Number	Name	Default	Description
⊗	CON(59)	PI	3.141592 ...	π given to machine accuracy.
⊗	CON(60)	SPOSG	0.0	The total positive (SPOSG), negative (SNEGG),
⊗	CON(61)	SNEGG	0.0	and total (STOTG) current at generation.
⊗	CON(62)	STOTG	0.0	
⊗	CON(63)	SPOSA	0.0	The total positive (SPOSA), negative (SNEGA)
⊗	CON(64)	SNEGA	0.0	and total (STOTG) current at solution.
⊗	CON(65)	STOTA	0.0	
⊗	CON(66)	XJFACT	1.0	The factor by which all currents and current densities (except current filaments) will be scaled. If BDES = CON(8) is input, then current will be adjusted. (See example, Sec. 10.2.) XJFACT = 0. - a scalar potential problem (no current). (See example, Sec. 10.5.)
	CON(67)	XJTOL	1.0E-4	The tolerance on the determination of XJFACT = CON(66) for BDES = CON(8).
	CON(68)	AFACT	1.0	MIRT – the factor by which scalar potentials are scaled when XJFACT = CON(66) = 0.0.
⊗	CON(69)	RATIO	0.0	The ratio |BZERO|/(XJFACT) for air region solution in POISSON/PANDIRA.
†	CON(70)	ICAL	0	Indicator for type of formula to use for calculating current density. ICAL = 0 – Use normal area formula. ICAL = 1 – Use angle formula for calculating the current associated with a point or when accurate fields near coil boundaries are needed.
⊗	CON(71)	NEGAT	0	An indicator for a zero or negative area triangle in mesh generation in LATTICE. NEGAT = 0 – no negative or zero areas. ≠ 0 – negative or zero area. Outputs a diagnostic message.
⊗	CON(72)	SNOLDA	0.0	The old sum of $(delta)^2$ for air (SNOLDA) and
⊗	CON(73)	SNOLDI		iron (SNOLDI) points.

	Number	Name	Default	Description
	CON(74)	RHOPT1	1.9	See CON(75) = RHOAIR.
	CON(75)	RHOAIR	1.9	The over-relaxation factor in POISSON for air and interface points and for iron points with a constant, but finite permeability. RHOAIR = RHOPT1 – optimizes RHOAIR during iteration. RHOAIR ≠ RHOPT1 – RHOAIR does not optimize; uses value assigned.
⊗	CON(76)	RHOM1	None	RHOM1 = RHOGAM –1. [RHOGAM = CON(78)].
	CON(77)	RHOFE	1.0	The over-relaxation factor in POISSON for iron points with a finite variable permeability.
	CON(78)	RHOGAM	0.08	The under-relaxation factor in POISSON for gamma (= 1./permeability) for finite variable permeability.
†	CON(79)	RHOXY	1.6	The starting over-relaxation factor for the irregular mesh generation.
	CON(80)	ISKIP	1	The number of cycles between recalculating the γ during a finite, variable permeability solution.
†	CON(81)	NOTE	1	An indicator for determining the order in which points are relaxed. NOTE = 0 – the order is air points, interface points, then iron points. MUST be used for PANDIRA. (See PANDIRA examples, Secs. 10.4 – 10.6.) NOTE = 1 – the order is (air + interface) points, then iron points.
⊗	CON(82)	BMAX	None	PANDIRA – maximum value of B.
⊗	CON(83)	IABORT	0	An indicator of an error in LATTICE and POISSON/PANDIRA/MIRT. IABORT = 0 no errors. IABORT = 1 error or errors. Program outputs diagnostic messages and aborts.

	Number	Name	Default	Description
†	CON(84)	EPSO	1.0E −5	The convergence criterion for mesh generation. If program has trouble converging, increasing EPSO might help.
	CON(85)	EPSILA	5.0E −7	The convergence criterion for the potential solution of air and interface points and for iron points with a finite, but constant permeability.
	CON(86)	EPSILI	5.0E −7	The convergence criterion for the potential solution of iron points with finite, variable permeability. NOTE: For problem to converge, both values printed under columns: "residual-air" "residual-iron" in file OUTPOI and the terminal must be less than EPSILA and EPSILI respectively. If printed values are near EPSILA/EPSILI and solution is not converging, increasing EPSILA/ EPSILI will force program to converge, with less accuracy.
	CON(87)	IVERG	10	The number of cycles between convergence test. The default value of 10 should not be altered if the option to optimize the over-relaxation factor CON(75) = RHOAIR is used.
⊗	CON(88)	RESIDA	1.0	POISSON – the residual of the air (RESIDA)
⊗	CON(89)	RESIDI	1.0	and iron (RESIDI) at each IVERG cycle. [IVERG = CON(87)].
⊗	CON(90)	ICYCLE	0	The present iteration cycle number. Used in LATTICE and POISSON/PANDIRA/MIRT.
⊗	CON(91)	NUMDMP	0	Current dump number for writing to TAPE35.
	CON(92) ⋮ CON(99)			This set of eight words stores the title of the problem which was read by LATTICE.

	Number	Name	Default	Description
	CON(100)	ITERM	0	A print indicator used in POISSON/PANDIRA for cylindrical problems when XJFACT[CON(66)] $\neq 0$. ITERM = 0 – prints (to OUTPOI/OUTPAN) dbzdr, xm, afit ITERM $\neq$ 0 – no print of above quantities
	CON(101)	IPERM	0	Indicator for permanent problem in PANDIRA only. IPERM = 0 – not permanent magnet problem. IPERM = 1 – a permanent magnet problem. The vector potential is initialized by either a current region or by current filaments [CON(49)] which the user MUST input. (See PANDIRA examples, Secs. 10.5 - 10.6.)
⊗	CON(102)	IAMASK	$600\ 000\ 000_8$	An octal word used as a mask to isolate bits in certain index words. IMASK, IFILT – used in LATTICE and POISSON/PANDIRA/MIRT. ISCAT, IDIRT – used in LATTICE.
⊗	CON(103)	ISCAT	$200\ 000\ 000_8$	
⊗	CON(104)	IFILT	$400\ 000\ 000_8$	
⊗	CON(105)	IDIRT	$100\ 000_8$	
⊗	CON(106)	ETAAIR	1.0	The rate of convergence in air (ETAAIR) and iron (ETAFE) regions in the current cycle.
⊗	CON(107)	ETAFE	1.0	
	CON(108)	AROTAT		Presently not used.
		ICYSEN	0	An indicator for output in POISSON. ICYSEN = 0 – no print of boundary integrals. ICYSEN $\neq$ 0 – print boundary integrals.
⊗	CON(109)	ITOT	None	The total number of mesh points in the problem. ITOT = (KMAX + 2)•(LMAX + 2).
	CON(110)	NTERM	0	The number of coefficients to be obtained in the harmonic analyses of the potential. $0 \leq$ NTERM ≤ 14. See harmonic analysis examples in Secs. 10.3 and 10.9. (For complete discussion of harmonic analysis, refer to POISSON/SUPERFISH Reference Manual.)

	Number	Name	Default	Description
	CON(111)	NPTC	0	The number of equidistant points on the arc of a circle with its center at the origin, at which points the vector potential is to be interpolated. Fourier analysis of the vector potential at these points yields the harmonic coefficients. NPTC should be approximately equal to the number of mesh points adjacent to the arc. $0 \leq \text{NPTC} \leq 101$.
	CON(112)	RINT		The radius of the arc of a circle at which the vector potential is to be calculated for harmonic analysis. RINT should be less than the radius to nearest singularity (pole or coil) by at least one mesh space.
	CON(113)	ANGLE		The final angle, in degrees, that defines the arc of the circle that is given in CON(112) = RINT.
	CON(114)	RNORM		The aperture radius or other normalization radius used in the harmonic analysis.
	CON(115)	ANGLZ		The initial angle, in degrees, that defines the arc of the circle with radius RINT = CON(112). Both ANGLE and ANGLZ are measured from the x-axis
	CON(116)	MASK37	37_8	Octal words used to isolate bits in certain index words. MASK37 – used in LATTICE and POISSON/ PANDIRA, MIRT. MASK5 – used in POISSON/PANDIRA/MIRT.
	CON(117)	MASK5	77777_8	
⊗	CON(118)	MAXDIM	None	The maximum number of mesh points allowed. ITOT = CON(109) ≤ MAXDIM. (MAXDIM is computed internally and depends on MXDIM. User may change parameter MXDIM.)
⊗	CON(119)	NWDIM		NWDIM = MAXDIM/2. User can change by changing parameter MXDIM and recompiling.
⊗	CON(120)	MASKC1	377_8	Octal words used as masks to isolate bits in certain index words.
⊗	CON(121)	MASKC2	$177\,400_8$	
⊗	CON(122)	TSTART	None	The starting time (wall clock) for execution of a run.

	Number	Name	Default	Description
†	CON(123)	TNEGC	0.0	A parameter used in conformal transformation. Input the total negative current in original geometry. LATTICE stores the negative transformed currents.
†	CON(124)	TPOSC	0.0	A parameter used in conformal transformation. Input the total positive current in original geometry. LATTICE stores the positive transformed currents.
†	CON(125)	RZERO	1.0	The scaling factor of the conformal transformation. $w = z$ ** MAP/[MAP * RZERO ** (MAP-1)] where: [MAP=CON(37)] and normally, RZERO = aperture radius.

TABLE C-2
ALPHABETIC ORDER OF CON VARIABLES FOR POISSON/PANDIRA/MIRT

Name	Number	Name	Number	Name	Number
AFACT	CON(68)	KBZERO	CON(40)	NSPL	CON(17)
ANGLE	CON(113)	KMAX	CON(4)	NTERM	CON(110)
ANGLZ	CON(115)	KMIN	CON(42)	NUMDMP	CON(91)
AROTAT	CON(108)	KPROB	CON(1)	NWDIM	CON(119)
BDES	CON(8)	KTOP	CON(43)	NWMAX	CON(26)
BMAX	CON(82)	LBZERO	CON(41)	OMEGA	CON(52)
CONV	CON(9)	LIMTIM	CON(29)	OMEGA0	CON(52)
EPSILA	CON(85)	LMAX	CON(3)	PI	CON(59)
EPSILI	CON(86)	LMIN	CON(44)	RATIO	CON(69)
EPSO	CON(84)	LTOP	CON(45)	RESIDA	CON(88)
ETAAIR	CON(106)	MAP	CON(37)	RESIDI	CON(80)
ETAFE	CON(107)	MASK37	CON(116)	RHOAIR	CON(75)
FIXGAM	CON(10)	MASK 5	CON(117)	RHOFE	CON(77)
IABORT	CON(83)	MASKC1	CON(120)	RHOGAM	CON(78)
IAMASK	CON(102)	MASKC2	CON(121)	RHOM1	CON(76)
IBOUT	CON(58)	MAXCY	CON(30)	RHOPT1	CON(74)
ICAL	CON(70)	MAXDIM	CON(118)	RHOXY	CON(79)
ICYCLE	CON(90)	MODE	CON(6)	RINT	CON(112)
ICYLIN	CON(19)	NAIR	CON(11)	RNORM	CON(114)
ICYSEN	CON(108)	NAMAX	CON(25)	RZERO	CON(125)
IDIRT	CON(105)	NBND	CON(16)	SNEGA	CON(64)
IFILT	CON(104)	NBSLF	CON(24)	SNEGG	CON(61)
IHDL	CON(50)	NBSLO	CON(22)	SNOLDA	CON(72)
IMASK	CON(102)	NBSRT	CON(23)	SNOLDI	CON(73)
IMAX	CON(5)	NBSUP	CON(21)	SPOSA	CON(63)
INACT	CON(34)	NEGAT	CON(71)	SPOSG	CON(60)
INPUTA	CON(20)	NFE	CON(12)	STACK	CON(7)
IPERM	CON(101)	NFIL	CON(49)	STOTA	CON(65)
IPRFO	CON(31)	NGMAX	CON(27)	STOTG	CON(63)
IPRINT	CON(32)	NGSAM	CON(28)	TNEGC	CON(123)
IRMAX	CON(53)	NINTER	CON(13)	TPOSC	CON(124)
IRNDMP	CON(36)	NODMP	CON(35)	TSTART	CON(122)
ISCAT	CON(103)	NOTE	CON(81)	W2ND	CON(47)
ISECND	CON(48)	NPERM	CON(18)	XJFACT	CON(66)
ISKIP	CON(80)	NPINP	CON(15)	XJTOL	CON(67)
ITERM	CON(100)	NPONTS	CON(51)	XMAX	CON(55)
ITOT	CON(109)	NPTC	CON(111)	XMIN	CON(54)
ITYPE	CON(46)	NREG	CON(2)	YMAX	CON(57)
IVERG	CON(87)			YMIN	CON(56)

Index

Each entry is followed by the number of the chapter (letter for Appendices A, B and C), a dash, and the page number(s) where the entry occurs. Entry with "(CON)" following designates a CON array parameter; entry with "(REG)" or "(PO)" following designates a REG or PO — FORTRAN Namelist parameter.

*U.S. Government Printing Office: 1995 671-046/02040