Revised by POTS

FEB 02 1989

SLAC-PEP-NOTE--326

DE89 006333

A USERS GUIDE TO POISSON*

R. A. Early

Stanford Linear Accelerator Center
Stanford University, Stanford, California 94305

I. INTRODUCTION

This writeup describes the POISSON program package in operation at SLAC. The program can be used to solve two-dimensional or cylindrical magnetostatic or electrostatic problems. The program uses an irregular triangular mesh of up to 6600 points in the finite difference approximation to Poisson's equation. Symmetry properties of a problem are taken advantage of so that usually, only part of a given problem has to be set up. For example: only a 45 degree segment of a symmetrical quadrupole is solved in order to obtain a complete solution. Successive over-relaxation (SOR) is used to find values of the vector potential at the mesh points from the boundary conditions and input currents. Non-linear regions (iron) are treated in a quasi-linear manner by holding the permeabilities constant during a relaxation cycle, computing local fields at the end of a cycle, performing a table lookup and under-correcting the permeability to keep the problem stable.

Although this is a production program and has been thoroughly debugged, users should exercise caution in its use. Output errors can be generated, especially by new

* Work supported by the Department of Energy, contract DE-AC03-76SF00515.

MASTER

REPRODUCTION OF THIS DOCUMENT IS UNLIMITED

users, due to the great flexibility of the program. Ten different symmetry conditions can be imposed, as well as specification of boundary conditions at both external and special internal regions. Improper use of symmetry codes will lead to errors in results for fields which may be large and will lead to gross errors in the harmonic analysis with forbidden harmonics appearing.

The program consists of four job steps: LATTICE, POISSON, TRIPLOTT and MINT. The last two are optional. LATTICE, as its name implies, reads most of the input data and generates an irregular triangular finite difference mesh. POISSON solves for the vector potential and fields at mesh points. TRIPLOTT produces plots of the mesh and field lines. MINT generates a harmonic analysis of the fields at specified points. The results of the first two steps are written on a scratch file to be read by succeeding job steps.

II. LATTICE

The POISSON package utilizes the concept of regions. An outer boundary for the problem, called the universe, must be defined as the first region. Succeeding regions must lie within the universe and their properties (material, current, etc.) override previously defined regions having the same mesh points.

Regions are usually closed, defining a certain area of the problem, but occasionally, line or point regions may be useful. These are allowed, but must be specified after area

regions. Closed regions are indicated when the first and last points in the region input data are identical.

Physical boundaries of a problem must be input in two sets of coordinate systems: mesh coordinates (K,L), and physical (x,y) coordinates. The mesh coordinate system is illustrated in Figure 1. Lines may be specified by their logical end points only when these points lie on the same "logical line". Logical lines are those of constant K, constant L, or whose end points can be connected by a 60 degree slanting line IN THE LOGICAL DIAGRAM. The lines may actually have a slope other than 0, 60, -60, 90, or -90 degrees in the physical system. For example, suppose we want to specify a symmetry line for a symmetric sextupole problem. The line could be specified by end points which are actually 60 degrees slope in the logical diagram even though the real line has a slope of 30 degrees. This results in a distorted mesh, and it is better to use the example shown on Figure 1 and labeled "30 degree line". The drawback is that every logical point on the line must now be specified instead of only the two end points.

Note that unlike programs which employ a rectangular mesh, the POISSON package does not interpolate across region boundaries. Such boundaries always lie on mesh points and in the case of a curved boundary, every mesh point must be specified in the input data stream.

Since POISSON is most often used for magnetostatic problems, the description of input data below will apply to

such problems. Electrostatic problems will be discussed in Section IIIc.

Input data for the program is read from card images by NAMELIST. At SLAC, it is most convenient to run the program from WYLBUR, and utilize an EXEC file to add the necessary JCL and NAMELIST information. Although the user does not have to be bothered with this detail, he should be aware of the fact that NAMELIST is being used in order to determine the cause of certain error messages.

A sample H-magnet problem is shown in Figure 2 and an example of an input data stream for the EXEC file is given in Figure 4. Figure 5 shows the same data after processing by the EXEC file. The most recent version of this EXEC can be found on WYL.RL.RAE.LIB#PEXEC. This version produces plots on the Versatek plotter. If plots are desired on the Tektronic 4013 scope terminals, use WYL.RL.RAE.LIB#POIEXEC.

The first card of the input data contains the word LATTICE. This is a flag for the EXEC file to bring into the active file the JCL for the first job step. Parameters for the job are set by the PARM array, which is of dimension 125. Only a few of these must be set by the user. Most are used internally by the program. The second card of the input stream is a comment card. The first four characters are placed in the PARM array as an identifier for the problem and are printed out on the title pages of the final results.

Next come the members of the PARM array which may be set by the user. Those which MUST be set are underlined.

- PARM(2) = the number of regions in the problem.
- PARM(3) = LMAX, the maximum value of the vertical mesh coordinate.
- PARM(4) = KMAX, the maximum value of the horizontal mesh coordinate.

In order to avoid exceeding the array dimensions in the program, $(KMAX + 2) * (LMAX + 2)$ must be less than, or equal to 2502 for the small version of POISSON, or less than or equal to 6600 for the larger version. (See Section VI on the EXEC file).

- PARM(6) = MODE, = -2 for air solution only (infinite permeability, or an electrostatic problem).
= -1 for fixed permeability (or fixed dielectric constant).
= 0 for variable permeability (air and iron solution) DEFAULT is -2.
- PARM(7) = Stacking factor. DEFAULT is 1.0.
- PARM(8) = Desired field in gauss. DEFAULT is 1.E15, which means current is not adjusted and fields due to input currents are obtained.
- PARM(9) = Conversion factor to centimeters. 2.54 for input in inches. 0.1 for input in millimeters. DEFAULT is 1.0 for input in cm.
- PARM(10) = Gamma value to be used in fixed permeability run. Gamma = 1./permeability. DEFAULT is .004.
For electrostatics, Gamma = 1./permittivity.
- PARM(19) = 0 for two-dimensional problems.
1 for cylindrical problems.
DEFAULT is 0.
- PARM(20) = The number of special potential points to be read in. (See Section III on INPUTA).

The following parameters set boundary conditions around the perimeter of the universe. 0 specifies a parallel flux boundary (Dirichlet). 1 specifies a normal (Neumann) boundary.

PARM(21) = Upper boundary. DEFAULT is 0.

PARM(22) = Lower boundary. DEFAULT is 0.

PARM(23) = Right boundary. DEFAULT is 0.

PARM(24) = Left boundary. DEFAULT is 0.

NOTE: BOUNDARY CONDITIONS MUST BE SET IN THE LATTICE STEP ONLY! They cannot be re-set later in a POISSON job step.

PARM(25) = This is a special flag which should be set to 1 when the problem is an H-magnet with only mid-plane symmetry. DEFAULT is 0. H-magnet problems fail to converge otherwise. Should not be set unless the upper, left, and right boundaries are also the boundaries of iron regions.

PARM(26) = Boundary potential. Parallel flux boundaries are never relaxed and the value of the potential there remains zero unless reset by this parameter. Normal flux boundaries ARE relaxed, so this parameter is meaningless for them. This parameter applies only to the four boundaries of the universe.

PARM(30) = Maximum number of iterations in the relaxation process. DEFAULT is 3000. Certain non-linear problems require more than 3000 iterations, but generally, if the problem hasn't converged by 3000, it never will.

PARM(31) = Print frequency for convergence information during relaxation. DEFAULT is 20.

PARM(37) = Power of z in a problem which has been transformed by: $W = Z^{**N}$ (see Section IIb).

PARM(38) = X coordinate of origin to be used in EDIT.

PARM(39) = Y coordinate of origin to be used in EDIT.

(EDIT refers to the part of POISSON which calculates B from the vector potential A)

PARM(40) = X coordinate of desired field (PARM(8)) DEFAULT is 0.

PARM(41) = Y coordinate of desired field. DEFAULT is 0.

At least one of the two parameters above should be reset when the field is zero at the origin as in quadrupoles and sextupoles.

The following four parameters set the limits of the printout in the EDIT step.

PARM(42) = KMIN, smallest value of K to be printed out, DEFAULT is 1.

PARM(43) = KTOP, largest value of K to be printed out, DEFAULT is KMAX.

PARM(44) = LMIN, smallest value of L to be printed out, DEFAULT is 1.

PARM(45) = LTOP, largest value of L to be printed out, DEFAULT is 1.

PARM(46) = ITYPE, symmetry code (see Section IIIa).

1. No symmetry.
2. Midplane symmetry (example: a C-magnet).
3. Unsymmetric quad (example: Collins quad).
4. Symmetric quad (45 degree segment).
5. Skew quad (rotated unsymmetric quad).
6. Sextupole symmetry (symmetric H-magnet).
7. Symmetric sextupole (30 degree segment).
8. Octupole (90 degree).
9. Symmetric octupole (22.5 degree).
10. Same as 6 but rotated 90 degrees.

PARM(66) = Current factor. Used internally to adjust current for desired field. DEFAULT is 1.0.
SET TO 0.0 FOR ELECTROSTATIC PROBLEMS.

See Section X for a discussion of the following relaxation and convergence factors.

PARM(75) = Starting over-relaxation factor in air. DEFAULT is 1.90.

PARM(76) = Starting over-relaxation factor on interface. DEFAULT is 1.90.

PARM(77) = Constant over-relaxation factor in iron. DEFAULT is 1.00.

Convergence is sometimes faster if the over-relaxation factors in air and on the interface are started at higher values. They must be less than 2.0 for stability. The over-relaxation factor in iron must be less than or equal to 1.0 for stability, since it is a non-linear medium.

PARM(78) = Under-relaxation factor for reluctivity.
DEFAULT is 0.08.

PARM(80) = Reluctivity recalculation frequency.
DEFAULT is 1, meaning that reluctivity
is recalculated every relaxation cycle.

Severe convergence problems may sometimes be overcome by changing the two previous parameters. The user should be sure that the final reluctivities (printed out as GAMMA in the edit) and the local B values correspond to the iron table in use, since convergence can always be forced by super-linearizing the problem.

PARM(82) = When non-zero, XYEDIT is skipped in iron regions and field values are returned as zero. This can save considerable time in some problems.
DEFAULT is 0.

PARM(84) = Lattice convergence criterion. DEFAULT is 1.E-10.

PARM(85) = Convergence criterion in air. DEFAULT is 1.E-11. This is good enough for an initial check run, but should be reset at least as low as 1.E-13 or even 1.E-16 if possible.

PARM(86) = Convergence criterion in iron. DEFAULT is 1.E-11. Can be reset to 1.E-13 or lower, depending on the problem.

PARM(125) = R0 for a conformally transformed problem.
(See Section IIb.) Equal to the distance from the center of the original magnet to the poleface.
(The bore radius.)

Referring to Figure 4 again, we see that after certain elements in the PARM array have been set, region data is read.

The REGION card contains five data elements in the following order:

- (1) The region number. (Regions are numbered consecutively. The first region is always the universe for the problem.)

- (2) The material code. 0 for special regions which are ignored in the relaxation.
- 1 for air or coil regions.
- 3 for Utah Plate Steel (used in the SLAC switch-yard bend magnets).

Any number greater than 1 is treated as magnetic material. Materials 2 and 4 are special tables used for PEP. Users should use material 3 or create their own B-H table.

- (3) Current in amperes which is distributed uniformly throughout the region. DO NOT PUT A REGION WITH CURRENT ON A PARALLEL FLUX BOUNDARY!
- (4) Current density in the region. Either (3) or (4) must be zero. You cannot enter both current and current density in the same region.
- (5) Zoning code. The lattice may be weighted towards equilateral or right triangles. There is usually trouble if all of the regions are zoned greater than 0.

0 for equal weight triangles.

1 for equalateral triangles.

2 for right triangles.

Zero zoning usually is satisfactory. When running a quad in 90 degrees, the user may prefer right triangles for better symmetry. Generally, the relaxation is more accurate with equalateral triangles, but it is best to use this zoning only in the air regions to avoid convergence problems in the lattice step.

If the material code is 0, a special BOUNDARY card must follow the REGION card. This card consists of the word BOUNDARY followed by -1 or 0 and sets the boundary conditions for the 0 material region.

BOUNDARY -1 for a parallel flux boundary

BOUNDARY 0 for a normal flux boundary

An example of the use of 0 material is in solving a 45 degree segment of a quadrupole. The universe must be square logically, and this leaves us with a triangular region which we wish to ignore. The material code 0 is assigned to this region and it is ignored in the relaxation and mesh generation. BOUNDARY -1 gives us a parallel flux boundary. (See Figure 9.)

This feature is of use in electrostatic problems since the potential assigned to all of the points in such a region is the same as the current specified on the REGION card. Figures 5 and 6 show input data and a plot of potential lines for one quadrant of the PEP electrostatic separation plates.

After the REGION card comes the most difficult and troublesome part of setting up input data. The mesh coordinates (K,L) and the corresponding physical coordinates (x,y) are read in.

The problem is laid out on isometric paper (Figure 3) in which every interior mesh point is a common vertex for six triangles. Each horizontal line of mesh points is called a row and is numbered with the index L as shown, running from $L = 1$ to $L = LMAX$ are the top of the mesh. Within each row, mesh points are numbered with the index K, running from $K = 1$ to $K = KMAX$ at the end. Note that the convention is made that the origin $L = 1$ and $K = 1$ is a point at which three mesh lines meet. Also, by convention, lines of constant K begin always with the sense indicated by the arrow (that is, they are zig-zag with the sense shown).

Associated with each mesh point are variables such as x and y coordinates and functions which are dependent variables for the differential equations to be solved, like the permeability. .

If a region is square, only the corner points need be read in. Boundary points are found by LATTICE by interpolation. Interior points are found by a pseudo-equipotential method in which the x, y coordinates are the weighted average of those for the six nearest neighbors. This is done by successive-over-relaxation.

Because boundary points are determined by interpolation along logical lines, points specified along common boundaries of adjacent regions must be specified in both regions.

The midplane symmetry plane is always the x axis with $L = 0$ and $y = 0$.

When cylindrical symmetry is used, the y axis becomes the z axis and the x axis becomes the radius r . The midplane symmetry plane lies along $x (r)$.

Referring again to Figure 6, note that each mesh point has associated with it two triangles on the right called upper and lower triangles. If the physical shape (x, y) differs radically from the logical shape (K, L) the mesh becomes severely distorted and triangles may become inverted or reversed after the mesh relaxation. This condition is called a negative triangle and is flagged as such by LATTICE. The POISSON step may fail if there is even one negative triangle. The "SEP SHOULD ALWAYS OBTAIN A PLOT OF THE MESH to help correct such distortions.

The procedure then is to first draw a logical diagram of the problem on isometric paper, then generate a series of card images containing the mesh coordinates (K,L) and their corresponding physical (x,y) . The coordinates may be input either clockwise or counter-clockwise, and the first and last points must be the same for closed regions. Line and point regions are allowed, but must come after closed regions. Current filaments may be input for point regions, but current cannot be input to line regions.

The mesh plot and flux plot for the problem given in Figure 2 are shown in Figures 6 and 7.

Program LATTICE superposes the regions in the order they are given in the input. Thus, all or part of a region which has one been specified may be respecified later in the input. This can cause trouble, since one may easily force mesh lines to cross one another in impossible ways producing triangles of negative or zero area. This problem can be avoided by twice recording points that belong in the boundary of two regions with a common border and by keeping the logical picture of the problem close to the physical one.

Since the number of mesh points available is limited, the user should use a higher density of points in the regions which are most important. This means exaggerating the size of such regions in the logical diagram. This requires judgement based on experience.

The region input data are placed on card images in the following order:

- a. K = index for horizontal mesh line.
- b. L = index for vertical mesh line.
- c. X = x coordinate for this point.
- d. Y = y coordinate for this point.

When LATTICE executes successfully, an irregular triangular mesh is generated, and a scratch file is written on disk to be used as input for succeeding steps of the program.

When setting up a problem, one should do just the LATTICE and PLOT steps alone until the mesh is successfully generated. The following procedure is recommended when running from a remote terminal:

First, RUN HOLD, then FETCH the output from LATTICE.

Next, LIST 'GENERATION' - If the run is successful, "GENERATION COMPLETED" will be printed out. If not, follow the steps below.

LIST 'LOGICAL' - If an input error has been made, successive points may not lie on the same logical line. The line printed and the next line will give the last correct point. LATTICE is aborted and no plot is generated.

LIST 'TRIANGLE' - Input errors or a badly strained mesh may cause negative triangles to be generated. The logical coordinates of such points are printed out and a plot can be obtained of the mesh.

Negative triangles are the most common and annoying problem encountered when using LATTICE. They can occur when the real geometry of the problem is too different from the geometry drawn on the logical diagram. Most often, the user

loses track of the relationship between distances on the logical diagram and the actual physical distances involved. This can cause a severely distorted or strained mesh.

Triangles which are not equilateral do not result in as good a solution as triangles which are equilateral, and a large number of oblique triangles can cause convergence convergence failure in the POISSON step.

The best way to prevent such problems is to break up complicated regions into several smaller regions of simple geometry. Line regions are sometimes useful in relieving strains.

The LATTICE and corresponding TRIPLLOT steps usually require less than 30 seconds on the SLAC computer system, but large meshes of over 2500 points may require 1 to 2 minutes.

III. POISSON

When LATTICE is running successfully, POISSON is normally run as a second job step of the problem. The data read and generated by LATTICE is written on a scratch disk file and is read by POISSON in either a second job step, or a separate job run later in the day. The procedure for this is outlined in the section on the EXEC file (Section xx). Referring to Figure 4, we see that after the input data for LATTICE, is the key word POISSON which is a flag telling the EXEC file to bring into the active file the JCL and NAMELIST cards for this job step. The POISSON card is followed by the DUMP card which tells the program which part of the scratch file is to be read in. If DUMP = 0, the results of the LATTICE step

are read in. Members of the PARM array may be set or reset in this job step as shown in Figure 4. Normally the cards look like this:

```
POISSON
DUMP = 0
PARM(6) = -2
DUMP = 2
PARM(6) = 0
```

Which means that LATTICE is followed by a POISSON run with infinite permeability, which is followed by a POISSON run which uses these results as input data for a variable permeability run.

```
POISSON
XYEDIT
FLAG, X0, Y0, Nx, Ny, Dx, Dy
DUMP = 0
PARM(6) = -2
DUMP = 2
PARM(6) = 0
```

The XYEDIT and succeeding card are optional and are used to get field values at "off mesh" points, i.e., anywhere.

All values are floating point and have the following meaning:

FLAG = 1	for edit line(s)
X0, Y0	starting coordinates
Nx, Ny	number of x, y points
Dx, Dy	x, y step size

Or:

```
POISSON
XYEDIT
FLAG, X0, Y0, NTH, NY, R, DTH
```

FLAG = 0 for an edit circle

X0, Y0 are coordinates of center of circle

NTH is number of angular points

NY is set to 1

R is radius of circle

DTH is angular step size, degrees.

The line starts at X0, Y0 and produces Nx points.

The next line starts at Y0+Dy and produces Nx more points separated by Dx. Finally a grid Nx by Ny is produced.

The circle starts at theta = 0. degrees and produces Nx points separated by Dy in angle.

The XYEDIT cards may be omitted, but if they are used, they must follow the POISSON card.

```
POISSON
INPUTA
3, 5, 1000.
4, 5, 1200.
etc.
DUMP = 0
PARM(6) = -2
DUMP = 2
PARM(6) = 0
```

The INPUTA statement allows the user to input special values of the vector potential A at specified (K,L) mesh points. This is pointless unless these points lie on a boundary which has been flagged as Dirichlet. This means either an external boundary for which the appropriate PARM has been set to 0, or the boundary of an internal region which has material code 0 and a BOUNDARY -1 card. The reason

for this is that such points are not relaxed and remain at their initial (usually 0) value. Such points do contribute to the other points in the problem however.

The number of such points to be read in is specified by PARM(20).

If both XYEDIT and INPUTA are specified, XYEDIT must come first, followed by INPUTA. They are always the first cards to follow POISSON.

The data on the scratch file is labeled by sequence numbers which run 0,2,3,4, etc. LATTICE writes the file labeled 0, the file labeled 1 is used internally and does not exist in the scratch file. POISSON writes files 2,3,4, etc. at the conclusion of each job step.

Variable permeability runs should usually be preceded by an infinite permeability run unless there is very little iron in the problem. This is because convergence usually fails when the entire non-linear problem is attempt first.

A FLUX PLOT SHOULD ALWAYS BE OBTAINED TO CHECK FOR INPUT ERRORS.

IIIa. Symmetry Codes for POISSON

POISSON is capable of solving a large variety of two-dimensional problems in magneto and electrostatics. Boundary conditions for the logical mesh are determined by:

PARM(21) upper boundary, 0 for parallel, 1 for normal flux
PARM(22) lower boundary
PARM(23) right boundary
PARM(24) left boundary.

These flags affect both the relaxation process to find the vector potential A, and the part of the code which finds the fields Bx, By, and Btotal from A.

The fields are determined by taking partial derivatives of a least squares fit to the vector potential near a given mesh point. A is fitted to a series of the form:

$$A = \text{SUMMATION}(AN \cdot R^{*N} \cdot \cos(N \cdot \text{THETA}) + BN \cdot R^{*N} \cdot \sin(N \cdot \text{THETA}))$$

The symmetry code, PARM(46), determines which values of AN and BN are allowed in the fit. This does not affect the relaxation, but has a marked effect on the fields in the printout and in the allowed harmonics of an harmonic analysis by the program MINT.

Some examples are given below:

TYPE	SYMMETRY	EXAMPLE	PARMS 21,22,23,24
1	NONE	AN ENTIRE C-MAGNET, 360 DEG	0 0 0 0
2	MIDPLANE	TOP HALF OF A C-MAGNET, 180 DEG OR TOP HALF OF AN H-MAGNET	0 1 0 0
3	ASYM QUAD	COLLINS QUADRUPOLE, 90 DEG	0 1 0 1
4	SYM QUAD	SYMMETRIC QUAD SOLVED IN 45 DEG	0 1 0 0
5	SKEW QUAD	QUAD WITH POLES ALONG AXES NOT SYMMETRIC, 90 DEG	0 0 0 0
6	SEXTUPOLE	AN H-MAGNET, OR A SEXTUPOLE IN A STEERING MODE, 90 DEG	0 1 0 0
7	SYM SEXT	SYMMETRIC SEXTUPOLE, 30 DEG	0 1 0 0
8	OCTUPOLE	GENERAL OCTUPOLE, 90 DEG	0 1 0 1
9	SYM OCT	SYMMETRIC OCTUPOLE, 22.5 DEG	0 1 0 0
10	ROT SEXT	SEXTUPOLE ROTATED 90 DEG	0 0 0 1

The allowed harmonics for the above codes are:

TYPE	AN'S	BN'S
1	1,2,3,...	1,2,3,...
2	1,2,3,...	NONE
3	2,4,6,...	NONE
4	2,6,10,...	NONE
5	NONE	2,4,6,...
6	1,3,5,...	NONE
7	3,9,15,...	NONE
8	2,4,6,...	NONE
9	4,12,20,...	NONE
10	NONE	1,3,5,...

Generally, there are convergence problems in the relaxation process if all of the available symmetry features of a problem are not used. For example, symmetric quadrupoles do not converge well when solved as 90 degree problems instead of 45 degree problems.

When solving H-magnets in midplane symmetry, set PARM(25) TO 1, this helps convergence.

Cylindrical problems require that PARM(19) = 1 and the symmetry type is either 1 for no symmetry (other than cylindrical) or 2 for midplane symmetry.

TYPE	SYMMETRY	EXAMPLE	PARMS 21,22,23,24
1	NONE	A tapered solinoid	0 0 0 0
2	MIDPLANE	Upper and lower parts have reflection symmetry	0 1 0 0

Often, there is some question as to which symmetry code to use in a given problem. The user should consider which harmonics should be allowed and chose the corresponding TYPE.

IIIb. Transformed Problems

POISSON has the capability of solving problems which have been conformally transformed by the transformation:

$$W = (R0/N) * (Z/R0)**N \quad (Z = X + iY, \quad W = U + iV)$$

for a 2N pole magnet, where R0 is the distance from the center of the original magnet to the nearest iron (i.e., the bore radius).

The purpose of such a transformation is to transform multipoles into dipoles in order to obtain greater accuracy in the fields and harmonics obtained. An example is shown in Figure 8 for a quadrupole (N = 2).

For a dipole magnet, B is uniform in the region of interest and the vector potential A varies linearly with x. For an N-pole magnet, A varies as $x**N$ near the center, and the finite difference approximation used in POISSON loses accuracy. Thus, Halbach (Nuclear Instruments and Methods, 64 (1968), 278-284) has suggested solving the transformed problem instead.

Users who wish to use this technique must transform their magnet's coordinates according to the proper values of N and R0, set PARM(37) = N, and PARM(125) = R0. The appropriate boundary conditions and symmetry codes for the geometry of the transformed problem must also be set.

The main drawbacks to this method are:

- (1) The mesh will be severely distorted in regions outside the good field region.
- (2) Lines which were straight in the original geometry are now curved and this means more labor for the user in inputting them.
- (3) Experience at SLAC has shown that at least for a .1% magnet, there is no significant improvement in accuracy.
- (4) Because of (1), you will probably not get an iron solution.

IIIc. Zero Material Code and Electrostatic Problems

Zero material codes are of use in multipole problems where $N \geq 1$ (quad and higher) and in electrostatic problems. Regions of zero material can be point, line, or closed regions. A region card with material code zero must be followed by a BOUNDARY card which has the format:

BOUNDARY - 1

or

BOUNDARY 0

The potential within the region is assigned the value of the current on the REGION card and results in an equipotential (parallel flux) surface for BOUNDARY -1. For BOUNDARY 0, the boundary points of the region are allowed to relax, which results in a normal or Neumann boundary for the potential lines. In either case, interior points are ignored in the relaxation and plotting.

Consider the problem shown in Figure 9, one of the PEP cell quads. The universe must be logically rectangular, yet we are interested in a wedge shaped problem. Regions are input as usual; only a special triangular region is defined in the upper left hand corner so that those points will be ignored in the calculation. BOUNDARY -1 sets a parallel flux boundary. The same procedure is used for higher order multipoles.

An electrostatic problem (the PEP electrostatic plates) is shown in Figure 10. Again, the universe is square and the upper right hand corner of the problem is defined as zero material with a parallel (constant potential) boundary whose current (potential) is zero. We only need to solve one quarter of this problem since it has sextupole symmetry.

The plate itself is defined as zero material with BOUNDARY -1 (constant potential) and a current on the REGION card of 5500. This produces a constant potential on the plate and a nearly uniform field at the center of the problem. The left boundary is set Neumann and symmetry code 10 is used. The problem could have been solved by rotating it 90 degrees and using symmetry code 6.

NOTE: PARM(66) must be set to 0. in electrostatic problems. This does not affect the input potentials, but serves as an internal flag to tell POISSON it is solving an electrostatic problem.

One must be careful here, because while the potentials, and their harmonics, are correct, the fields are reversed in X and Y in the printout of the problem. This is because POISSON takes the curl of A to get fields and field harmonics. For electrostatics, we want the gradient. The user must use By for Ex and Bx for Ey in the output. Similarly, the phase of the field harmonics must be corrected 90 degrees.

IV. TRIPLOT

TRIPLOT reads the disk file generated by LATTICE and POISSON and generates plots of the mesh and flux plots (see Figures 6 and 7). Examples of input data are shown in Figures 4 and 5. Any dump may be read and plotted. Again, the exec file converts the input data to NAMELIST format. The meaning of the input parameters is specified below:

DUMP - Dump number to be read.
XMIN - minimum value of x to be plotted.
XMAX - maximum value of x to be plotted.
YMIN - minimum value of y to be plotted.
YMAX - maximum value of y to be plotted.
SCALE - scaling factor for plot (see below).
TSENT - set to 1 for mesh plot, 0 for flux plot.
NPHI - set to 0 for mesh plot, set to desired number of flux lines for a flux plot.

Regardless of which plotting device is being used, TRIPLOT thinks it's generating a plot on a 10 by 10 inch Calcomp plotter. The rule for setting SCALE is the following:

$SCALE * PARM(9) * (XMAX - XMIN) < \text{or} = 10$

$SCALE * PARM(9) * (YMAX - YMIN) < \text{or} = 10$

V. MINT

The last card which may be added to the input data for the exec file PEXEC is the HARMONIC card and has the following format:

HARMONIC

This card brings in the JCL for MINT and it's harmonic analysis. The input data will be discussed in the following section on the exec file. Briefly, MINT samples the vector potential along an edit circle similar to that used in XYEDIT and performs a harmonic analysis. Only allowed harmonics are listed in the printout. Again, the user is urged to be sure he is using the right symmetry code. MINT operates in a separate job step and must read the dump files written by POISSON in the previous job step.

Vector potential harmonics are printed out as well as field harmonics obtained by taking the curl of the potential. For this reason, the user must be careful when solving an electrostatic problem. Field harmonics will not have the correct phase, but must be adjusted 90 degrees.

VI. THE EXEC FILE

The exec file WYL.RL.RAE.LIB(PEXEC) saves the user the task of writing JCL and worrying about NAMELIST parameters. Input data as shown in Figure 4 must be in your active file. The prompts are as follows:

TIME, C/R for 15 sec

The response should correspond to TIME = on a job card, (,30) for a 30 second job, or 4 for a 4 minute one. Do not put in any blanks. LATTICE plus TRIPlot generally run in less than 30 seconds, but can require up to 2 minutes in a very large mesh. Electrostatic, or infinite permeability runs generally run in two minutes total time. Variable permeability runs always require something like 4 minutes, but often require as much as 40 minutes in problems where the steel is saturating. Long run times may be caused by a badly distorted mesh.

RUN HOLD? YES, OR C/R FOR RUN

The response should be Y or YES to run hold. No blanks.

L FOR 6600 MESH, OR C/R FOR 2500

There are two mesh sized available, 2500 points, or 6600 points. The response should be L or C/R.

PLOT, VERSATEK, P, S, OR C/R

To get a campus Calcomp plot tape written, respond P. To get a Versatek plot generated, respond S. C/R for no plot.

If you have included a HARMONIC card, you will get the following:

HIGHEST HARMONIC

Respond with the maximum value of N that you want, or C/R to skip the harmonic analysis entirely.

DUMP NO

Respond with the desired dump for analysis.

CENTER OF CIRCLE X0

When doing multipoles this will be zero, but occasionally an harmonic analysis may be required of a C-magnet where the good field region is not located at the origin.

CIRCLE RADIUS

This refers to the radius of the harmonic analysis circle. DO NOT PUT THE CIRCLE INSIDE THE IRON! It may touch surface points of iron, but must not touch interior points.

NUM POINTS ON CIRCLE

The number of points in the analysis. This probably shouldn't be much larger than the number of mesh points in the region of the circle circumference.

ANGULAR RANGE (DEG)

The angular range in degrees of the analysing circle.

NORM RADIUS

The output of MINT is normalized such that:

$$B = \text{summation } (AN * (R/R0)^{(N-1)} * \cos(N*THETA))$$

for example.

This means that the AN's and BN's in the printout have a value that corresponds to the norm radius.

DSNAME

Respond with a name of up to 7 characters. The scratch files will have the name WYL.RL.RAE.DSNAME. For this reason, users are encouraged to get their own copies of PEXEC and PJCL in their own libraries and change 'WYL.RL.RAE' to their own group and initials.

VOLUME? OR C/R FOR WORK03

Normally, a scratch volume is used for the POISSON scratch files, but it is occasionally useful to specify a permant WYLBUR volume. An example might be that you are putting a 40 minute job in on a Friday night and want to run your plot or harmonic analysis on the following Monday afternoon.

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

ALLOWED LOGICAL LINES
Only First and Last
Points Need to be Specified

Fig. 1

Fig. 2

Fig. 3

LATTICE

SANDY #SANDY, STANDARD TEST PROBLEM
 PARM(2)=4, PARM(3)=22, PARM(4)=26, PARM(6)=-2,
 PARM(9)=2.54, PARM(22)=1, PARM(40)=0.0, PARM(45)=22,
 PARM(46)=6, PARM(41)=0.0, PARM(43)=26, PARM(31)=20
 PARM(85)=1.0E-13, PARM(86)=1.0E-13
 PARM(66)=.9579065, PARM(110)=0

REGION 1, 1, 0.0, 0.0, 0

1, 1, 0.0, 0.0,
 11, 1, 2.125, 0.0,
 12, 1, 2.25, 0.0,
 26, 1, 9.0, 0.0,
 26, 22, 9.0, 5.375,
 1, 22, 0.0, 5.375,
 1, 6, 0.0, 0.875,
 1, 1, 0.0, 0.0,

REGION 2, 1, -28300.0, 0.0, 0

12, 1, 2.25, 0.0,
 19, 1, 5.75, 0.0,
 19, 13, 5.75, 2.375,
 12, 13, 2.25, 2.375
 12, 6, 2.25, 0.875,
 12, 1, 2.25, 0.0,

REGION 3, 3, 0.0, 0.0, 0

1, 6, 0.0, 0.875,
 8, 6, 1.4375, 0.875,
 9, 6, 1.625, 0.6959,
 10, 6, 1.8125, 0.6959,
 11, 6, 2.0, 0.6959,
 11, 14, 2.0, 2.625,
 1, 14, 0.0, 2.625,
 1, 6, 0.0, 0.875,

REGION 4, 3, 0.0, 0.0, 0

1, 14, 0.0, 2.625,
 11, 14, 2.0, 2.625,
 12, 14, 2.25, 2.625,
 20, 14, 6.25, 2.625,
 20, 1, 6.25, 0.0,
 26, 1, 9.0, 0.0,
 26, 22, 9.0, 5.375,
 1, 22, 0.0, 5.375,
 1, 14, 0.0, 2.625,

POISSON

XYEDIT

1.,0., 0., 10.,4., 1.00, 1.0,

DUMP=0

PARM(110)=1

DUMP=2

PARM(6)=0

PARM(66)=1.124893

DUMP=3

PARM(8)=16100.

PARM(85)=1.E-14

Fig. 4

Figure 4 (Continued)

PLOT

DUMP=0

XMIN= 0.000, XMAX=9.000, YMIN=0.000, YMAX=5.375,
SCALE=0.394, TSENT=1, NPHI=0

DUMP=2

XMIN= 0.000, XMAX=9.000, YMIN=0.000, YMAX=5.375,
SCALE=0.394, TSENT=0, NPHI=19

DUMP=3

XMIN= 0.000, XMAX=9.000, YMIN=0.000, YMAX=5.375,
SCALE=0.394, TSENT=0, NPHI=19


```

// JOB ,TIME 2
//SCR EXEC PGM=IEFBRI4
//SCR DD DSN=WYL.RL.RAE.SANDY,UNIT=DISK,VOL=SER=WORK03,
// DISP=(MOD,DELETE)
//SCR DD DSN=WYL.RL.RAE.SANDY2,UNIT=DISK,VOL=SER=WORK03,
// DISP=(MOD,DELETE)
// EXEC LINKGO,REGION.GO=320K
//LKED.LIB DD DSN=WYL.RL.PUB.POISSON,UNIT=DISK,DISP=SHR
//LKED.SYSIN DD *
    INCLUDE LIB(LATTICE)
//GO.FT35F001 DD DSN=WYL.RL.RAE.SANDY,UNIT=DISK,VOL=SER=WORK03,
// DISP=(,KEEP),
// DCB=(RECFM=VS,LRECL=3604,BLKSIZE=3608),SPACE=(CYL,(1,1))
//GO.SYSIN DD *
SANDY #SANDY, STANDARD TEST PROBLEM
&PRBINP
    PARM(2)=4, PARM(3)=22, PARM(4)=26, PARM(6)=-2,
    PARM(9)=2.54, PARM(22)=1, PARM(40)= 0.0,PARM(45)=22,
    PARM(46)=6,PARM(41)=0.0,PARM(43)=26,PARM(31)=20
    PARM(85)=1.0E-13,PARM(86)=1.0E-13
    PARM(66)=.9979065,PARM(110)=0

&END
&REGCRD CONST= 1, 1, 0.0, 0.0, 0
&END
&DEFINE TEMP=
    1, 1, 0.0, 0.0,
    11, 1, 2.125, 0.0,
    12, 1, 2.25, 0.0,
    26, 1, 9.0, 0.0,
    26, 22, 9.0, 5.375,
    1, 22, 0.0, 5.375,
    1, 6, 0.0, 0.875,
    1, 1, 0.0, 0.0,
&END
&REGCRD CONST= 2, 1, -28300.0, 0.0, 0
&END
&DEFINE TEMP=
    12, 1, 2.25, 0.0,
    19, 1, 5.75, 0.0,
    19, 13, 5.75, 2.375,
    12, 13, 2.25, 2.375,
    12, 6, 2.25, 0.875,
    12, 1, 2.25, 0.0,
&END
&REGCRD CONST= 3, 3, 0.0, 0.0, 0
&END

```

Fig. 5

Figure 5 (Continued)

```

&DEFINE TEMP=
  1, 6, 0.0, 0.875,
  8, 6, 1.4375, 0.875,
  9, 6, 1.625, 0.6959,
  10, 6, 1.8125, 0.6959,
  11, 6, 2.0, 0.6959,
  11, 14, 2.0, 2.625,
  1, 14, 0.0, 2.625,
  1, 6, 0.0, 0.875,
&END
&REGCRD CONST= 4, 3, 0.0, 0.0, 0
&END
&DEFINE TEMP=
  1, 14, 0.0, 2.625,
  11, 14, 2.0, 2.625,
  12, 14, 2.25, 2.625,
  20, 14, 6.25, 2.625,
  20, 1, 6.25, 0.0,
  26, 1, 9.0, 0.0,
  26, 22, 9.0, 5.375,
  1, 22, 0.0, 5.375,
  1, 14, 0.0, 2.625,
&END
// EXEC LINKGO,REGION,GO=448K
//LKED.LIB DD DSN=WYL.RL.PUB.POISSON,UNIT=DISK,DISP=SHR
//LKED.SYSIN DD *
  INCLUDE LIB(POISSON)
//GO.FT35F001 DD DSN=WYL.RL.RAE.SANDY,UNIT=DISK,VOL=SER=WORK03,
//  DISP=SHR,DCB=(RECFM=VS,LRECL=3604,BLKSIZE=3608,BUFNO=1)
//GO.FT37F001 DD DSN=WYL.RL.RAE.SANDY2,UNIT=DISK,VOL=SER=WORK03,
//  DISP=(,KEEP),
//  DCB=(RECFM=VBS,LRECL=44,BLKSIZE=880),SPACE=(TRK,(5,5),RLSE)
//GO.SYSIN DD *
&XYEDIT XYTEMP=
  1.,0., 0., 10.,4., 1.00, 1.0,
&END
&INPUTA TEMP=0.
&END
&NDMP NUMDMP =0
&END
&PRBINP
  PARM(110)=1
&END
&NDMP NUMDMP =2
&END
&PRBINP
  PARM(6)=0
  PARM(66)=1.124893
&END
&NDMP NUMDMP =3
&END
&PRBINP
  PARM(8)=16100.
  PARM(85)=1.E-14
&END
&NDMP NUMDMP=-1 &END

```

Figure 5 (Continued)

```

/** PLOT POISSON OUTPUT ON VERSATEK ~P
// EXEC LINKGO,REGION.GO=448K,LKEDLB1='WYL.CG.RCB.UGFTNLIS',
//      GOSL1='WYL.CG.RCB.UGRUNLIB'
//LKED.LIBCAL DD DSN=WYL.RL.RAE.LOADLIB,DISP=SHR
//LKED.LIB DD DSN=WYL.RL.PUB.POISSON,UNIT=DISK,DISP=SHR
//LKED.SYSIN DD *
//      INCLUDE LIB(TRIPLOT1) ~P
//      INCLUDE LIBCAL(PLOTS1) ~P
//GO.FT35F001 DD DSN=WYL.RL.RAE.SANDY,DISP=SHR,UNIT=DISK,
//      VOL=SER=WORK03,
//      DCB=(RECFM=VS,LRECL=3604,BLKSIZE=3608)
//GO.UGDEVICE DD SYSOUT=P,DCB=(RECFM=FBA,LRECL=134,BLKSIZE=1340) ~P
//GO.SYSIN DD *
&PLOADAT
IDUMP=0
      XPMIN= 0.000, XPMAX=9.000, YPMIN=0.000, YPMAX=5.375,
      SCALE=0.394, TSENT=1, NPHI=0
&END
&PLOADAT
IDUMP=2
      XPMIN= 0.000, XPMAX=9.000, YPMIN=0.000, YPMAX=5.375,
      SCALE=0.394, TSENT=0, NPHI=19
&END
&PLOADAT
IDUMP=3
      XPMIN= 0.000, XPMAX=9.000, YPMIN=0.000, YPMAX=5.375,
      SCALE=0.394, TSENT=0, NPHI=19
&END
&PLOADAT IDUMP=-1 &END

```

SANDY

12-79

3732A4

Fig. 6

SANDY

Fig. 7

TRANSFORMED QUAD
From [NIM, 64, (1968), 278-284]

Fig. 8

Fig. 9

Fig. 10