RPN 76

RADIATION PROJECT PROGRESS REPORT NO. 5

Computer-Assisted Magnetic Field Design
Part II

Users' Notes for Program MAGNOL

by

Richard Y. Close

Electron Beams Branch, Plasma Physics Division

10 April 1968

COMPUTER-ASSISTED MAGNETIC FIELD DESIGN

Part II

USERS' NOTES FOR PROGRAM MAGCOIL

by

Richard C. Close

10 April 1963

I. INTRODUCTION

In Radiation Project Progress Report No. 5 of 7 March 1960, the background material for a computer program useful in magnetic field design was presented. This note gives the detailed instructions necessary for using the program. The program, called MAGCOIL, is written in Fortran IV for use on a CDC 3600. With minor modification it probably could be used on any machine with a Fortran IV compiler. In its present form, the program is made up of a main program and five subroutines. As time goes on, minor changes will probably be incorporated in the source deck. It is felt, however, that the program has evolved to the point that such changes will not greatly influence the instructions presented here.

The second section of this note deals with the initial preparation of the problem and the classification of mesh points. Section III lists all the variables including those that are dimensioned and/or placed in common. The deck structure and the input data is discussed in Section IV. Some guidance is given for the selection of the accelerating factor for the over-relaxation calculation and a sample problem input is displayed. Section V deals with the various program options and with the output formats. Again examples are given. The final section gives a listing of the complete program in its present form.

II. PROBLEM FORMULATION

The basic problem here is to solve the elliptic partial differential equation:

$$\frac{\partial^2 V}{\partial z^2} + r \frac{\partial}{\partial r}\left(\frac{1}{r}\frac{\partial V}{\partial r}\right) = 0 \quad . \tag{1}$$

The general solution of equation (1) can be obtained by separating variables. One finds that this solution is given as an infinite series of first order Bessel functions multiplied by exponential functions. This fact is not particularly useful in solving specific problems, but for program test one can construct exact solutions to compare with the numerical results generated by the program.

To prepare a problem for solution, the first procedure is to draw out the proposed electrode structure and to superimpose a suitable mesh. The nodes of the mesh can then be classified according to their position within the region of interest. Points along the edge of the region are

termed boundary points and must have their potential specified. This classification also applies to points that fall within or on the boundary of the electrode structure. Difference equations are not solved at these boundary points. Within the region one finds points at which the relaxation procedure is to be applied. These points can be termed regular or irregular depending upon whether their distances from adjoining mesh points are integral or fractional multiples of the mesh spacing. Finally, the points that lie along the axis of symmetry of the mesh (r=0) are termed Neumann points and a different set of finite difference equations is employed for their relaxation.

Once a given electrode system is drawn and a mesh superimposed, the classification of the various mesh points is easily accomplished. For purposes of the program, a code number (KD) is assigned to each mesh point. Boundary points are given KD=1, regular and irregular points given KD=2, and Neumann points identified by KD=3. The program automatically sets all points along the edges with KD=1 and potential equal to zero, all points within the mesh with KD=2 and all points along the axis of symmetry with KD=3. In the case of the KD=2 points, all distances to adjoining mesh points initially are set equal to the mesh spacing. After this procedure has taken place, the programmer can read in the modifications to the automatic setup which actually determines the specific problem under study. Here one specifies the number of the mesh point, its code, the distances to adjoining mesh points if KD=2 and the boundary value of the potential if KD=1.

As a simple example of this procedure consider Figure 1, which portrays a single electrode with a coarse 7x7 mesh superimposed. Points 1 through 7 are Neumann points (KD=3) and are set as such automatically

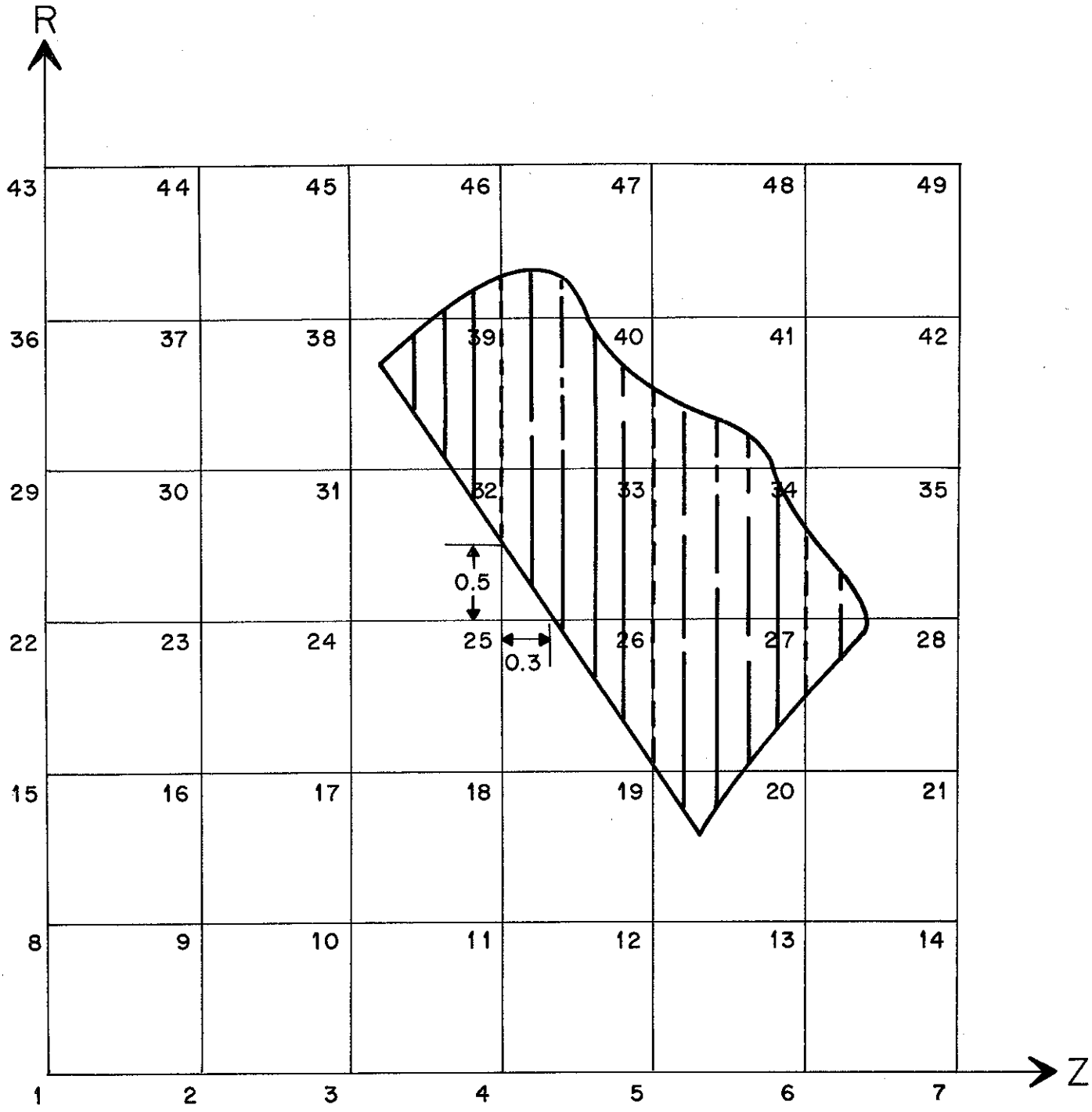
R
Z
43
44
45
46
47
48
49
36
37
38
39
40
41
42
29
30
31
32
33
34
35
0.5
22
23
24
25
26
27
28
0.3
15
16
17
18
19
20
21
8
9
10
11
12
13
14
1
2
3
4
5
6
7

Figure 1

by the program. Points around the edge of the mesh (8, 15, 22, 29, 36, 43, 44, 45, 46, 47, 48, 49, 42, 35, 28, 21, 14) are automatically set as boundary points (KD=1) with potential equal to zero. If this is not the correct boundary value of potential, the programmer can read in the appropriate values. Other boundary points which must be read in by the programmer specify the single electrode. These points (19, 26, 27, 32, 33, 39) are assigned KD=1 and the boundary value specified. In most runs the electrodes are assumed to be equipotentials and the boundary value of the potential is normalized to unity. All the rest of the mesh nodes are points at which the normal relaxation procedure must be followed. They are automatically set with KD=2 and integral mesh distances. To more faithfully represent the physical situation, the programmer may elect to read in fractional mesh spacings. For example mesh point 25 could be characterized by the four numbers 0.5, 0.5, 1.0, 1.0 (reading counter-clockwise starting on the right) which give its position relative to the boundary of the electrode in terms of mesh spacings. In an initial run with a very fine mesh, the procedure may not be necessary but as more precise information is desired the problem can be refined to include all so-called irregular points with KD=2. Once all points of the mesh have been classified the program proceeds with the solution of equation (1) by the method of successive over-relaxation (SOR).[1]

III. DEFINE VARIABLES USED IN THE PROGRAM

The following variables are placed in COMMON in the program:

(1. BETA)

The successive over-relaxation parameter. This is specified by the programmer and must be between 0 and 2 with a good initial value being about 1.5.

(2. DELTAZ, DELTAR)

These are the increments used in the relaxation procedure. They may be chosen so that one mesh distance corresponds to a definite physical dimension. In this sense, they are used to scale the problem to any convenient physical size.

(3. NR, NC)

These are the number of rows (NR) and the number of columns (NC) in the mesh being used.

(4. NPTA)

This is the number of points in the mesh, i.e., simply NR x NC.

(5. M)

This is a two-dimensional array which contains four values for each mesh point. These are the mesh spacings to adjoining mesh points. Initially all of these are set equal to unity but they can be changed by the programmer if a precise evaluation of a system is desired.

(6. C)

This is another two-dimensional array which contains five values for each mesh point. These are the coefficients of the difference equation that is to be solved at that point. Recall that at each mesh point (j) the difference equation to be solved appears as:

$$V_j = \sum_{i=1}^{4} C_{ji} V_i + C_{j5}$$

For this problem C_{j5} is always zero but if the right hand side of equation (1) were non-zero a value for C_{j5} could be specified.

(7. UMAT)

This is the value of the potential at each mesh point.

Other variables appearing in DIMENSION statements are:

(1. MR)

An array of integer values containing the mesh point numbers.

(2. KD)

An array of integer values containing the codes for each mesh point.
KD=1 signifies a boundary point.
KD=2 signifies a regular or irregular point.
KD=3 signifies a Neumann point.

(3. SIGT)

This is a 10-element array which contains the results of up to 10 different line integral paths used in computing $\oint \vec{H} \cdot d\vec{z}$.

(4. ANDEX)

Is an array which contain the values of the magnetic field index

$$n = -\frac{r}{H_z}\frac{\partial H_z}{\partial r}$$

along the median plane of the electrode system.

(5. CARD)

Is an array which is used to put an identifying title on a given problem. CARD can consist of any arrangement of 80 Hollerith characters on a single input card.

(6. HR)

Is an array which contains the value of the radial magnetic field at each mesh point.

(7. HZ)

Is an array which contains the value of the longitudinal magnetic field at each mesh point.

Other variables which are neither in COMMON or DIMENSION statements are as follows.

(1. NVECT)

Integer which controls the potential calculation. If NVECT is nonzero, the relaxation procedure is followed; otherwise potential values are read from cards. In this latter case, if KO1 is nonzero, it is a continuation job.

(2. NPUN)

Integer which controls punching of the vector potential on cards in anticipation of a continuation job or for input to another program. A nonzero value indicates punching desired.

(3. NFIELD)

Integer which controls the magnetic field calculation. If NFIELD is nonzero, the magnetic field components will be calculated.

(4. LINE)

Integer which controls the line integral calculation. Nonzero means to perform the calculation. If line integrals are desired one must also specify NFIELD as nonzero.

(5. INDEX)

Integer which controls the calculation of the field index. Nonzero value means to perform the calculation. Again if the field index is desired NFIELD must be nonzero.

(6. KD1)

Integer which signifies a continuation job. A nonzero value of KD1 demands that cards be read and hence that NWCP=0.

(7. NT2)

This is an integer which gives the number of iterations performed during the relaxation process.

(8. DMAX)

The maximum change in potential during a given cycle of the iteration.

(9. UMAX)

The potential existing at a mesh point where the maximum change DMAX has occurred.

(10. MMAX)

The number of the mesh point at which the change in potential is greatest during an iteration of the relaxation calculation.

(11. EWANT)

This is the accuracy demanded of the program during the calculation. If $\frac{DMAX}{UMAX}$ is less than EWANT after an iteration the problem is judged to be solved.

(12. NCURVE)

The number of different paths to be used during the line integral calculation. This number could be as high as 10 but in most cases is set equal to 6.

(13. IMT, NUMR)

These are the starting point mesh number and the width in mesh spacings of the rectangular line integral path. There are NCURVE sets of these numbers.

(14. IRESS)

This is the number of points to be read in which will give new boundary values or specify fractional mesh spacings.

The foregoing list contains all of the important variables used in the program. Of course, there are several other variables to be found in the Fortran source deck. A cursory examination of the program should make their use clear.

IV. DECK STRUCTURE, DATA SELECTION AND INPUT FORMATS

The main program, MAGCOIL, is to be used with several additional subroutines. Basically MAGCOIL reads the input data and by the use of various control sequences selects the proper subroutines for a given problem. The following is a list of the subroutines now used with MAGCOIL and a brief description of their functions.

(1. GENCO)

This subroutine calculates the C array which contains the coefficients necessary for the finite difference equations.

(2. ITRATE)

This subroutine actually performs the SOR calculation. It also calculates DMAX, UMAX, and RMAX which were defined earlier.

(3. PUNK)

This sub-routine will punch BCD cards if they are wanted.

(4. MAGFLD)

This is the main subroutine for magnetic field calculations. Given the potential values, this routine will calculate the field components, the line integral, the field index and several other parameters of interest.

(5. QUICKMAP)

This routine produces a topographical map of the various potential and field quantities. This routine was written by Mrs. Jeanne T. Ulrich of the Electron Beams Branch, NRL, and provides a quick, visual check of the computed values.

Several other routines have been included on occasion. For example, it is sometimes desired to find and plot equipotentials. Provision can be made for incorporation of routines that will utilize a Calcomp plotter for this purpose.

One parameter which must be specified by the programmer is the accelerating factor, β (BETA), for the SOR procedure. According to theory any value such that:

$$0 < \beta < 2$$

should provide for a convergent solution. In practice it is found that values less than unity are seldom of value because they often lengthen the running time and sometimes produce results that are less accurate than may be obtained with other values of β. If the value of β is not specified the program will use the value β=1.0 and there will be no acceleration. A good initial value of β has been found to be about 1.5. As an example of the influence of this parameter on the running time of a problem, the number of iterations versus the value of β for a test problem is shown in Figure 2. It is seen that a value of 1.5 gives a solution in about half the time compared with the β=1.0 case. The instability of the solution as $\beta \rightarrow 2.0$ is also clearly evident. For practical problems judicious choices of β (usually near 1.5) have resulted in even more startling reductions in running time. For simple regions it may be possible to calculate the optimum value of β[2] and there exist various empirical procedures[3] that give near optimum values of β. In this case, however, these procedures have not been incorporated in the program and the value of β is determined solely by the skill of the programmer.

Some ideas concerning the selection of a suitable mesh were given in Radiation Project Progress Report No. 3. Suffice it to say here that a mesh of about 1000 nodes (points) has been found to be adequate for most of the problems involved. The program is now written so that 1000 is the maximum number of points. This allows the entire problem to be solved in a single memory bank of the CDC 3800. This has certain advantages especially with regard to shorter running times. If a finer mesh is desired, one would simply increase the number of storage cells reserved in the memory for the various arrays and demand that the program be compiled for use of both memory banks. A factor of three or more in the total number of mesh points then seems possible.

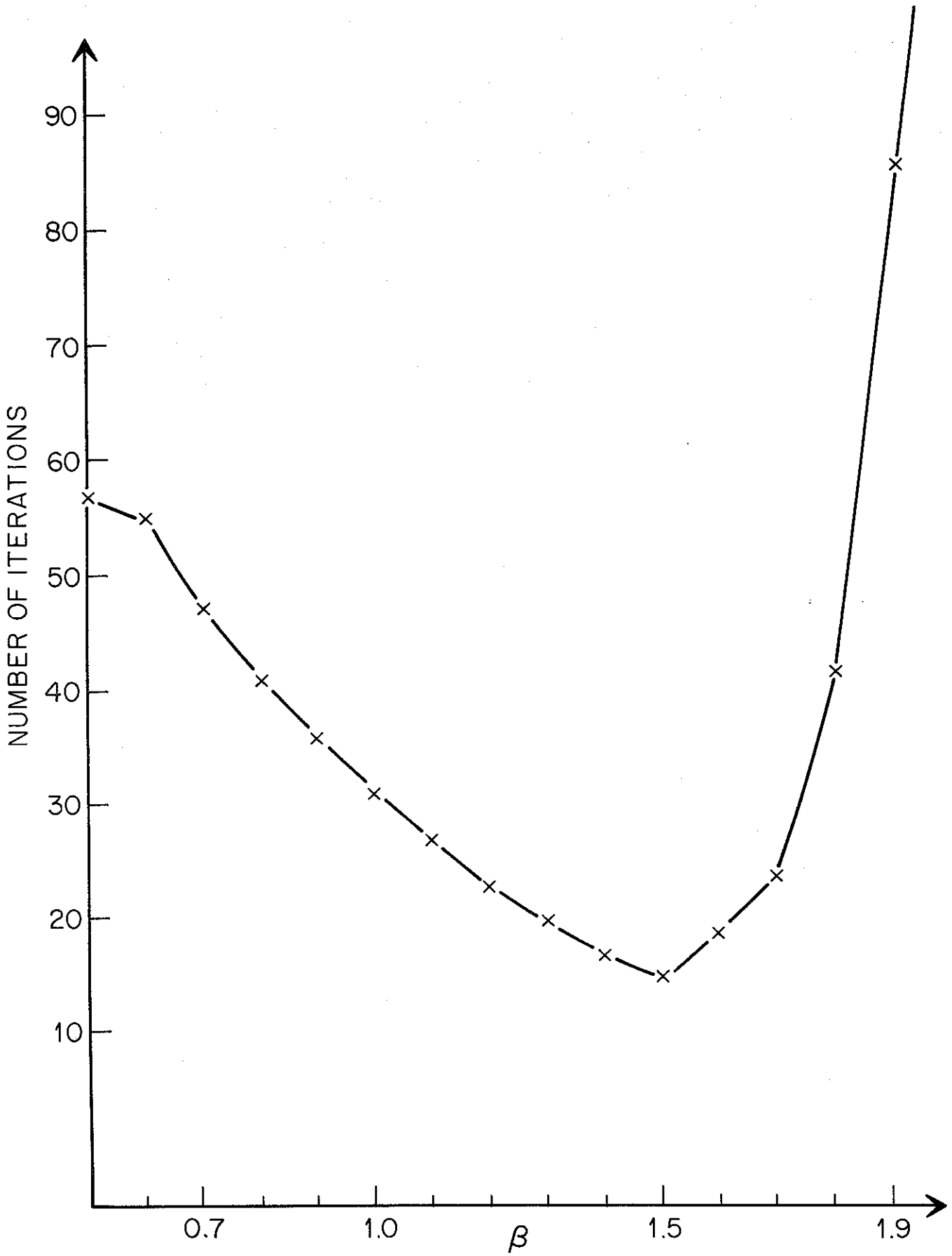

FIGURE 2

The formats for the input data are now detailed. The input data is normally punched on cards with various Fortran formats. The total number of cards varies from run to run and depends sensitively on the control integers, the accuracy desired and whether or not it is a continuation job.

The first card of the input data deck contains the CARD array written in A format. This card is used to provide an identifying title for the computation. The second card contains eight integers punched with an 8I 10 format. These are the control integers NVECT, NPUN, NFIELD, LINE, INDEX, KD1, KD2, and KD3. All of these have been defined except the last two which are not presently used by the program. The third card contains four floating point numbers and two fixed point numbers written in 4E 15.8, 2I 10 format. The four real variables are BETA, DELTAZ, DELTAR, and EGAM. The integers are NR and NC. If this is an initial calculation, the next (fourth) card carries the single integer IENTG which is simply the number of cards that follow. These cards contain the modification to the automatic mesh setup and specify boundary values and/or fractional mesh spacings. They are written in a 2I5, 5F 10.5 format. The first integer is the mesh number, the second is the code (KD) while the next four real variables are the mesh spacings. The last floating point number is the boundary value of potential. After IENTG of these cards have been read, the program carries out the relaxation procedure. If the job is a continuation job (NVECT=0, KD1=1), the values of

the potential are read five to a card in I5, 5E 15.8 format (here the integer is the mesh point number of the first potential value on the card) before one encounters the IRREG card. If the potentials already exist on cards they can be used for magnetic field calculations (NVECT=0, KD1=0) and the IRREG card and its subsequent input data are not needed.

At this point some remarks about the continuation feature may be in order. Most authorities agree that this feature is most useful and for that reason it was built into the program. Actual experience with the program indicates that at most a saving of 15% in running time can be achieved with the continuation feature as compared to doing the problem from scratch. With a machine as fast as the CDC 3800 it appears that the continuation feature is probably not worth the extra effort involved.

The next set of input data cards are concerned with the line integral calculation. The first card of this group is a single integer written with I15 format. This gives the value of NCURVE which is the number of cards which follow. Each of these subsequent cards contain two integers in 2I 15 format. The first integer is the mesh point number of the starting point of the line integral calculation. It is assumed that all of the paths start on the lower left of the region and follow a rectangular path along the mesh enclosing the electrodes to be studied. The second number on this card is the width of the integration path desired. The width (NUMR) is given in terms of mesh units.

The input data for an initial run of the sample problem shown in Figure 1 is now given. This of course is not a practical problem but it will illustrate most of the features involved. The sample data is shown on page 14a.

The first card gives the identification information and tells us that this is a sample problem. The second card has nonzero values for NVECT, NPUN, NFIELD, LINE, and INDEX. This indicates that the vector potential is to be calculated from scratch, the values are to be punched on cards for future use, and that the fields, the line integral and the field index are to be calculated. Note that KD1 is zero so this is not a continuation job. The third card gives the accelerating factor (β=1.5), the mesh increments (ΔR=Δz=0.1), the accuracy requirement (EWANT=0.001), the number of rows (NR=7), and the number of columns (NC=7). This value of EWANT lets the program stop the relaxation calculation when the maximum change in the potential at a mesh point is less than 0.1% of the value of the potential then existing at that point. The fourth card specifies that IRREG is 12 and this is followed by twelve additional cards specifying the problem. The edges of the mesh have been automatically set with boundary values of zero and the single electrode is assigned unit potential. The last group of input cards first gives NCURVE=2, followed by two cards which describe the two line integral paths desired. These are improper paths in that they intersect the electrode but nonetheless are useful for demonstration purposes.

V. OUTPUT FORMAT

The output of this program can take various forms depending on which control integers are used. For purposes of example the output generated by the sample problem (Figure 1) will be illustrated. Again, one should bear in mind that this is not a practical problem; but the principles involved will be the same in any case.

The first big block of output is the mesh information. After identifying the problem, the first line of numbers gives the values of BETA, DELTAR, DELTAZ, EWANT, NR, and NC. Then each mesh point is listed in order along with its code (KD), its distances to adjoining mesh points (in units of the mesh spacing), and the initial value of potential stored at that point. During the program boundary points (KD=1) retain this potential while all other points (KD=2,3) are subject to the relaxation.

After returning the input mesh information, the problem normally will enter the relaxation calculation. After each iteration the maximum change in the potential (DMAX), the value of the potential at that point (UMAX) and the location (MMAX) of that point are printed. When the accuracy criteria are met, the program prints a message and then outputs the values of the potential in a table written with a 6E 20.8 format. Another form of output follows where the potential is written such that the position of the number on the printed page corresponds to the position of the mesh point at which this value exists. This map output is very convenient for helping one visualize a given problem. Unfortunately to obtain such a map requires one to alter various output format

SAMPLE PROBLEM FOR RADIATION PROJECT PROGRESS REPORT

```
        1           1           1           1           1
1.5                 0.1                     0.1                 0.001                   7           7
      12
19      1                                                                           1.0
26      1                                                                           1.0
27      1                                                                           1.0
32      1                                                                           1.0
33      1                                                                           1.0
39      1                                                                           1.0
20      2   1.0         0.47        0.4         1.0
25      2   0.3         0.5         1.0         1.0
31      2   0.65        1.0         1.0         1.0
34      2   1.0         1.0         0.22        0.38
38      2   0.58        1.0         1.0         1.0
40      2   1.0         1.0         0.45        0.47
              2
              9                   4
             10                   3
```

statements in the program if different mesh arrangements are used. This point will be discussed briefly in the next section. Immediately after the map output, a topographical plot (from subroutine QUICKMAP) is printed.

The program then enters the magnetic field calculation and the outputs of tables and maps follow the same formats as the potential calculation. After the fields, the results of the line integral calculation are printed. The starting point (INT) and the width (NUMR) is given along with the various contributions along the Z dimension and the R dimension of the path. Finally the total value of the integral is given. This format is repeated for each path used. The average value of all the line integral calculations is printed and from this the current, inductance, and stored energy of the system under investigation are calculated. The final block of output is the field index along the median plane, which is simply printed in a column. The first value corresponding to $r=0$, the second to $r=\Delta r$, the third to $r=2\cdot\Delta r$, etc.

Along with the printed output, a deck of cards is also punched. These cards contain an integer and the values of the potential punched five to a card. The integer on each of these cards corresponds to the mesh point number of the first potential value on the card. This procedure makes sequencing the cards simpler and errors due to cards being out of order can be avoided.

The rest of this section presents the output obtained from a run of the sample problem drawn in Figure 1. The input data used was exhibited in the previous section. A listing of the punched card output for this problem is also included.

ELLIPTIC PARTIAL DIFFERENTIAL EQUATION

DATA READ IN

MESH INFORMATION

1.50000000+000 1.00000000-001 1.00000000-001 1.00000000-003 7 7

1	3	1.00000	1.00000	1.00000	1.00000	0.00000
2	3	1.00000	1.00000	1.00000	1.00000	0.00000
3	3	1.00000	1.00000	1.00000	1.00000	0.00000
4	3	1.00000	1.00000	1.00000	1.00000	0.00000
5	3	1.00000	1.00000	1.00000	1.00000	0.00000
6	3	1.00000	1.00000	1.00000	1.00000	0.00000
7	3	1.00000	1.00000	1.00000	1.00000	0.00000
8	1	1.00000	1.00000	1.00000	1.00000	0.00000
9	2	1.00000	1.00000	1.00000	1.00000	0.00000
10	2	1.00000	1.00000	1.00000	1.00000	0.00000
11	2	1.00000	1.00000	1.00000	1.00000	0.00000
12	2	1.00000	1.00000	1.00000	1.00000	0.00000
13	2	1.00000	1.00000	1.00000	1.00000	0.00000
14	1	1.00000	1.00000	1.00000	1.00000	0.00000
15	1	1.00000	1.00000	1.00000	1.00000	0.00000
16	2	1.00000	1.00000	1.00000	1.00000	0.00000
17	2	1.00000	1.00000	1.00000	1.00000	0.00000
18	2	1.00000	1.00000	1.00000	1.00000	0.00000
19	1	-0.00000	-0.00000	-0.00000	-0.00000	1.00000
20	2	1.00000	0.47000	0.40000	1.00000	-0.00000
21	1	1.00000	1.00000	1.00000	1.00000	0.00000
22	1	1.00000	1.00000	1.00000	1.00000	0.00000
23	2	1.00000	1.00000	1.00000	1.00000	0.00000
24	2	1.00000	1.00000	1.00000	1.00000	0.00000
25	2	0.30000	0.50000	1.00000	1.00000	-0.00000
26	1	-0.00000	-0.00000	-0.00000	-0.00000	1.00000
27	1	-0.00000	-0.00000	-0.00000	-0.00000	1.00000
28	1	1.00000	1.00000	1.00000	1.00000	0.00000
29	1	1.00000	1.00000	1.00000	1.00000	0.00000
30	2	1.00000	1.00000	1.00000	1.00000	0.00000
31	2	0.65000	1.00000	1.00000	1.00000	-0.00000
32	1	-0.00000	-0.00000	-0.00000	-0.00000	1.00000
33	1	-0.00000	-0.00000	-0.00000	-0.00000	1.00000
34	2	1.00000	1.00000	0.22000	0.38000	-0.00000
35	1	1.00000	1.00000	1.00000	1.00000	0.00000
36	1	1.00000	1.00000	1.00000	1.00000	0.00000
37	2	1.00000	1.00000	1.00000	1.00000	0.00000
38	2	0.58000	1.00000	1.00000	1.00000	-0.00000
39	1	-0.00000	-0.00000	-0.00000	-0.00000	1.00000
40	2	1.00000	1.00000	0.45000	0.47000	-0.00000
41	2	1.00000	1.00000	1.00000	1.00000	0.00000
42	1	1.00000	1.00000	1.00000	1.00000	0.00000
43	1	1.00000	1.00000	1.00000	1.00000	0.00000
44	1	1.00000	1.00000	1.00000	1.00000	0.00000
45	1	1.00000	1.00000	1.00000	1.00000	0.00000
46	1	1.00000	1.00000	1.00000	1.00000	0.00000

```
 47     1    1.00000    1.00000    1.00000    1.00000    0.00000
 48     1    1.00000    1.00000    1.00000    1.00000    0.00000
 49     1    1.00000    1.00000    1.00000    1.00000    0.00000

AFTER THE     1 ITERATION THE MAXIMUM CHANGE IS      1.19315267+000
WHERE THE POTENTIAL IS       1.19315267+000 AT MESH POINT    34


AFTER THE     2 ITERATION THE MAXIMUM CHANGE IS      7.57743218-001
WHERE THE POTENTIAL IS       1.28288115-001 AT MESH POINT    41


AFTER THE     3 ITERATION THE MAXIMUM CHANGE IS      4.26522054-001
WHERE THE POTENTIAL IS       5.54810169-001 AT MESH POINT    41


AFTER THE     4 ITERATION THE MAXIMUM CHANGE IS      2.48761380-001
WHERE THE POTENTIAL IS       2.00432863-001 AT MESH POINT    30


AFTER THE     5 ITERATION THE MAXIMUM CHANGE IS      1.56110712-001
WHERE THE POTENTIAL IS       8.39426941-002 AT MESH POINT    16


AFTER THE     6 ITERATION THE MAXIMUM CHANGE IS      6.44909019-002
WHERE THE POTENTIAL IS       2.18716314-001 AT MESH POINT    23


AFTER THE     7 ITERATION THE MAXIMUM CHANGE IS      4.84098504-002
WHERE THE POTENTIAL IS       2.89690222-001 AT MESH POINT    17


AFTER THE     8 ITERATION THE MAXIMUM CHANGE IS      1.94249610-002
WHERE THE POTENTIAL IS       2.70265261-001 AT MESH POINT    17


AFTER THE     9 ITERATION THE MAXIMUM CHANGE IS      1.33607763-002
WHERE THE POTENTIAL IS       2.04371622-001 AT MESH POINT    23


AFTER THE    10 ITERATION THE MAXIMUM CHANGE IS      6.19346616-003
WHERE THE POTENTIAL IS       2.10565088-001 AT MESH POINT    23


AFTER THE    11 ITERATION THE MAXIMUM CHANGE IS      2.14086093-003
WHERE THE POTENTIAL IS       2.73186518-001 AT MESH POINT    17


AFTER THE    12 ITERATION THE MAXIMUM CHANGE IS      1.19044243-003
WHERE THE POTENTIAL IS       3.39919961-002 AT MESH POINT     9


AFTER THE    13 ITERATION THE MAXIMUM CHANGE IS      5.70750914-004
```

```
WHERE THE P0TENTIAL IS      1.17791620-001 AT MESH P0INT   16


AFTER THE    14 ITERATI0N THE MAXIMUM CHANGE IS      2.41403919-004
WHERE THE P0TENTIAL IS      2.65775484-001 AT MESH P0INT   30


ACCURACY CRITERIA MET

0UTPUT RESULTS F0LL0W
```

P0TENTIAL VALUES AT EACH EACH MESH P0INT

0.00000000+000	0.00000000+000	0.00000000+000	0.00000000+000	0.00000000+000	0.00000000+000
0.00000000+000	0.00000000+000	3.35862437-002	7.53427048-002	1.31622171-001	1.91226925-001
1.33316886-001	0.00000000+000	0.00000000+000	1.17907146-001	2.72538453-001	5.19647931-001
1.00000000+000	6.84029212-001	0.00000000+000	0.00000000+000	2.09239747-001	4.78259230-001
8.55183993-001	1.00000000+000	1.00000000+000	0.00000000+000	0.00000000+000	2.65775484-001
6.36673297-001	1.00000000+000	1.00000000+000	8.34773813-001	0.00000000+000	0.00000000+000
2.17953303-001	5.79826240-001	1.00000000+000	7.65236799-001	4.20804209-001	0.00000000+000
0.00000000+000	0.00000000+000	0.00000000+000	0.00000000+000	0.00000000+000	0.00000000+000
0.00000000+000					

MAP OF REGION

0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	0.0336	0.1179	0.2092	0.2658	0.2180	0.0000
0.0000	0.0753	0.2725	0.4783	0.6367	0.5798	0.0000
0.0000	0.1316	0.5196	0.8552	1.0000	1.0000	0.0000
0.0000	0.1912	1.0000	1.0000	1.0000	0.7652	0.0000
0.0000	0.1333	0.6840	1.0000	0.8348	0.4208	0.0000
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

PEAK = 1.0000+000
DIFFERENCE BETWEEN LEVELS = 1.0000-001

```
0 0 0 0 0 0 0
0 2 5 * 7 4 0
0 2 6 * * 8 0
0 2 4 8 * * 0
0 1 2 5 * 6 0
0 0 0 1 1 1 0
0 0 0 0 0 0 0
```

MAGNETIC FIELD CALCULATION

RADIAL COMPONENT

0.00000000+000	0.00000000+000	0.00000000+000	0.00000000+000	0.00000000+000	0.00000000+000
0.00000000+000	0.00000000+000	-3.76713524+000	-4.90179637+000	-5.79421102+000	-8.47357496-002
9.56134626+000	0.00000000+000	0.00000000+000	-6.81346132+000	-1.00435196+001	-1.81865387+001
0.00000000+000	2.50000000+001	0.00000000+000	0.00000000+000	-7.97098716+000	-1.07657374+001
-8.69567950+000	0.00000000+000	0.00000000+000	0.00000000+000	0.00000000+000	-7.95841621+000
-9.17780645+000	0.00000000+000	0.00000000+000	1.25000000+001	0.00000000+000	0.00000000+000
-5.79826240+000	-7.82046697+000	0.00000000+000	5.79195791+000	7.65236799+000	0.00000000+000
0.00000000+000	0.00000000+000	0.00000000+000	0.00000000+000	0.00000000+000	0.00000000+000
0.00000000+000					

LONGITUDINAL COMPENENT

0.00000000+000	5.89535730+000	1.36269226+001	2.59823966+001	5.00000000+001	3.42014606+001
0.00000000+000	0.00000000+000	5.89535730+000	1.36269226+001	2.59823966+001	5.00000000+001
3.42014606+001	0.00000000+000	0.00000000+000	4.39133757+000	1.00729131+001	1.80890456+001
0.00000000+000	2.16670778+001	0.00000000+000	0.00000000+000	2.46447230+000	6.06891407+000
8.00586781+000	0.00000000+000	0.00000000+000	0.00000000+000	0.00000000+000	1.08919452-001
1.26958763+000	0.00000000+000	0.00000000+000	-7.23994739+000	0.00000000+000	0.00000000+000
-2.65775484+000	-6.36673297+000	0.00000000+000	-1.00000000+001	-8.34773813+000	0.00000000+000
0.00000000+000	0.00000000+000	0.00000000+000	0.00000000+000	0.00000000+000	0.00000000+000
0.00000000+000					

MAP 0F REGI0N RADIAL FIELD

0.0000	0.0000	0,0000	0.0000	0.0000	0.0000	0.0000
0.0000	-3.7671	-6,8135	-7.9710	-7.9584	-5.7983	0.0000
0.0000	-4.9018	-10,0435	-10.7657	-9,1778	-7.8205	0.0000
0.0000	-5.7942	-18,1865	-8.6957	0.0000	0.0000	0.0000
0.0000	-0.0847	0,0000	0.0000	0.0000	5.7920	0.0000
0.0000	9.5613	25,0000	0.0000	12.5000	7.6524	0.0000
0.0000	0.0000	0,0000	0.0000	0.0000	0.0000	0.0000

PEAK = 2.5000+001
DIFFERENCE BETWEEN LEVELS = 4.3187+000

```
4 4 4 4 4 4 4
4 2 2 4 5 5 4
4 2 2 4 4 7 4
4 2 1 2 4 4 4
4 2 1 0 4 * 4
4 3 3 2 4 6 4
4 4 4 4 4 4 4
```

MAP OF REGION LONGITUDINAL FIELD

0.0000	0.0000	0,0000	0.0000	0.0000	0.0000	0.0000
5.8954	5.8954	4,3913	2.4645	0.1089	-2.6578	0.0000
13.6269	13.6269	10,0729	6.0689	1.2696	-6.3667	0.0000
25.9824	25.9824	18,0890	8.0059	0.0000	0.0000	0.0000
50.0000	50.0000	0,0000	0.0000	0.0000	-10.0000	0.0000
34.2015	34.2015	21,6671	0.0000	-7.2399	-8.3477	0.0000
0.0000	0.0000	0,0000	0.0000	0.0000	-0.0000	0.0000

PEAK = 5.0000+001
DIFFERENCE BETWEEN LEVELS = 6.0000+000

```
1 1 1 1 1 1 1
1 1 0 1 0 0 1
1 1 1 1 1 0 1
1 2 2 3 1 1 1
1 2 3 4 1 5 1
1 2 3 5 * 7 1
1 2 3 5 * 7 1
```

LINE INTEGRAL CALCULATION

STARTING POINT 9

WIDTH 4

Z CONTRIBUTION TO INTEGRAL

SZ11= -2.00484090+000

SZ12= -8.96093192+000

SZ21= -5.50274648-001

SZ22= -1.63667330+000

SZ1= -1.09657728+001

SZ2= -2.18694795+000

R CONTRIBUTION TO INTEGRAL

SR1= -4.78269882-001

SR2= -2.27428647+000

SR= -5.50511270+000

TOTAL LINE INTEGRAL EQUALS -1.86578335+001

NEXT CASE FOLLOWS

LINE INTEGRAL CALCULATION

STARTING POINT 10

WIDTH 3

Z CONTRIBUTION TO INTEGRAL

SZ11= -3.18134613+000

SZ12= -2.59823966+000

SZ21= 6.34793816-002

SZ22= 0.00000000+000

SZ1= -5.77958579+000

SZ2= 6.34793816-002

R CONTRIBUTION TO INTEGRAL

SR1= -7.03980141-001

SR2= -2.08092571+000

SR= -5.56981170+000

TOTAL LINE INTEGRAL EQUALS -1.12859181+001

LINE INTEGRAL VALLES

-1.12859181+001 -1.86578335+001

AVERAGE VALUE 0F LINE INTEGRAL = -1.49718758+001

CURRENT = -1.19142402+000

INDUCTANCE = -5.27367688+000

ST0RED ENERGY =. -3,74296895+000

VI. PROGRAM MAGCOIL LISTING

A listing of a recent compilation of program MAGCOIL is presented. In this version the potential matrix (UMAT) is limited to 1000 locations. As stated previously, this number could probably be increased at least three times if both memory banks of the CDC 3800 were used. Format statement 410 in MAGCOIL is written for the sample 7X7 mesh used in this report for illustration. When finer meshes are used, this statement should be changed along with the calls to QUICKMAP which also depend on the mesh being used. If desired a variable format which is read in for each problem could be employed.

The running time of the problem is quite fast. For the complete compilation of the main program and five subroutines shown, the time was 46 sec. More startling is the fact that the execution time was a mere 5 sec. Of course, the sample problem is not quite representative of the type of problem usually solved. For normal runs where the number of mesh points is about 1000, execution times of less than one minute have been recorded.

FIELD INDEX ALONG MEDIAN PLANE

-0.00000000+000
1.51898055-001
9.93779838-001
3.38921014+000
0.00000000+000
0.00000000+000
0.00000000+000

********* LISTING OF POTENTIAL VALUE CARDS FOR SAMPLE PROBLEM **********

```
  1 0.00000000+000 0. 0   0  +000 0.00000000+000 0.00000000+000 0.00000000+000
  6 0.00000000+000 0.00000000+000 0.00000000+000 3.35862437-002 7.53427048-002
 11 1.31622171-001 1.91226925-001 1.33316886-001 0.00000000+000 0.00000000+000
 16 1.17907146-001 2.72538453-001 5.19647931-001 1.00000000+000 6.84029212-001
 21 0.00000000+000 0.00000000+000 2.09239747-001 4.78259230-001 8.55183993-001
 26 1.00000000+000 1.          +000 0.00000000+000 0.00000000+ 00 2.65775484-001
 31 6.36673297-001 1.00000000+000 1.00000000+000 8.34773813-001 0.00000000+000
 36 0.00000000+000 2.17953303-001 5.79826240-001 1.00000000+000 7.65236799-001
 41 4.20804209-001 0.00000000+000 0.00000000+000 0.00000000+000 0.00000000+000
 46 0.00000000+000 0. 0   0  +000 0.00000000+000 0.00000000+000 0.00000000+000
```

```
      PROGRAM MAGCOIL                                                    5
C                                                                        10
C     SOLVES ELLIPTIC DIFFERENTIAL EQUATIONS BY SOR                      15
C     IN THIS CASE THE EQUATION FOR THE VECTOR POTENTIAL IN SPHERICALLY   20
C     SYMMETRIC CYLINDRICAL COORDINATES IS SOLVED                        25
C                                                                        30
      DIMENSION CARD(10)                                                 35
      DIMENSION H(1000,4), C(1000,5), UMAT(1000), MR(1000), KD(1000)     40
      DIMENSION SIGT(10), ANDEX(50)                                      45
      COMMON BETA,DELTAZ,DELTAR,NR,NC,NPIA,H,C,UMAT                      50
C     JOB IDENTIFICATION                                                 55
C     THE IDENTIFICATION CAN TAKE ANY FORMAT ON ONE CARD                 60
      READ 260, (CARD(I),I=1,10)                                         65
      PRINT 270, (CARD(I),I=1,10)                                        70
C                                                                        75
C     OPTIONS FOR PROGRAMME                                              80
C                                                                        85
      READ 280, NVECT,NPUN,NFIELD,LINE,INDEX,KD1,KD2,KD3                 90
C                                                                        95
C     NVECT=1 CALCULATION OF VECTOR POTENTIAL                           100
C     NVECT=0 READ VECTOR POTENTIAL FROM CARDS                          105
C     NPUN=1  PUNCHES VECTOR POTENTIAL ON CARDS                         110
C     NPUN=0  NO PUNCHING NECESSARY                                     115
C     NFIELD=1 CALCULATES MAGNETIC FIELD COMPONENTS                     120
C     NFIELD=0 NO FIELDS NEEDED                                         125
C     LINE=1  COMPUTE LINE INTEGRAL                                     130
C     LINE=0  NO LINE INTEGRAL NEEDED                                   135
C     INDEX=1 COMPUTE FIELD INDEX                                       140
C     INDEX=0 NO COMPUTATION OF FIELD INDEX                             145
C     KD1 NONZERO INDICATES CONTINUATION JOB                            150
C     IN THIS CASE NVECT SHOULD BE ZERO                                 155
C     ALL PARTS OF THIS PROGRAM ARE NOT INDEPENDENT,THAT IS, IF NFIELD  160
C     EQUALS ZERO LINE INTEGRALS AND FIELD INDICIES WILL NOT BE         165
C     CALCULATED SINCE THE FIELDS ARE NOT AVAILABLE                     170
C                                                                       175
      READ 290, BETA,DELTAZ,DELTAR,EWANT,NR,NC                          180
      NPIA=NR*NC                                                        185
C     KODE 1=BOUNDARY,2=REGULAR,3=NEUMANN                               190
C     ENTERING DATA SIMPLIFIED METHOD                                   195
      DO 20 I=1,NPIA                                                    200
      MR(I)=I                                                           205
      KD(I)=2                                                           210
      H(I,1)=1.                                                         215
      H(I,2)=1.                                                         220
      H(I,3)=1.                                                         225
      H(I,4)=1.                                                         230
   20 UMAT(I)=0.                                                        235
C     SET NEUMAN POINTS ON AXIS                                         240
      DO 30 I=1,NC                                                      245
   30 KD(I)=3                                                           250
C     SET BOUNDARY POINTS ON EDGES                                      255
      NCC=NC+1                                                          260
      NCR=2*NC                                                          265
      IUP=NPIA-NC+1                                                     270
      DO 40 I=NCC,NPIA,NC                                               275
   40 KD(I)=1                                                           280
```

```
      DO 50 I=NCR,NPIA,NC                                                    285
   50 KD(I)=1                                                                290
      DO 60 I=IUP,NPIA                                                       295
   60 KD(I)=1                                                                300
      IF (NVECT) 70,170,70                                                   305
C     READ IN IRREGULAR POINTS AND BOUNDARY POINTS                           310
C     IRREG IS NUMBER OF POINTS READ IN FOR NON-STANDARD H OR BOUNDARY       315
C     ORDER IS NOT IMPORTANT                                                 320
   70 READ 300, IRREG                                                        325
      DO 80 I=1,IRREG                                                        330
   80 READ 310, ME,KD(ME),H(ME,1),H(ME,2),H(ME,3),H(ME,4),UMAT(ME)           335
      PRINT 320                                                              340
      PRINT 330, BETA,DELTAZ,DELTAR,EWANT,NR,NC                              345
      DO 90 I=1,NPIA                                                         350
   90 PRINT 340, MR(I),KD(I),H(I,1),H(I,2),H(I,3),H(I,4),UMAT(I)             355
      PRINT 350                                                              360
      DO 130 I=1,NPIA                                                        365
      IF (KD(I)-1) 100,130,110                                               370
  100 PRINT 360                                                              375
      RETURN                                                                 380
  110 IF (KD(I)-2) 100,120,120                                               385
  120 CALL GETCO (MR(I),KD(I),C(I,1),C(I,2),C(I,3),C(I,4),C(I,5))            390
  130 CONTINUE                                                               395
      NIT=0                                                                  400
  140 CALL ITRATE (DMAX,KD,UMAX,MMAX)                                        405
      NIT=NIT+1                                                              410
      PRINT 370, NIT,DMAX,UMAX,MMAX                                          415
      UMAX=ABSF(UMAX)                                                        420
      EMAX=DMAX/UMAX                                                         425
      IF (EMAX-EWANT) 150,150,140                                            430
  150 PRINT 380                                                              435
      PRINT 390, (UMAT(I),I=1,NPIA)                                          440
      PRINT 400                                                              445
      DO 160 J=1,NC                                                          450
  160 PRINT 410, (UMAT(I),I=J,NPIA,NC)                                       455
      CALL QUICKMAP(UMAT,7,7,1)                                              460
      GO TO 210                                                              465
C                                                                            470
  170 DO 190 I=1,NPIA,5                                                      475
      READ 420, MESH,UMAT(I),UMAT(I+1),UMAT(I+2),UMAT(I+3),UMAT(I+4)         480
      IF (MESH-I) 180,190,180                                                485
  180 PRINT 430                                                              490
      RETURN                                                                 495
  190 CONTINUE                                                               500
      IF (KD1) 200,210,200                                                   505
  200 PRINT 440                                                              510
      GO TO 70                                                               515
C     SECTION FOR PUNCHING JEANNI CARDS                                      520
  210 IF (NPUN) 220,230,220                                                  525
  220 CALL PUNK                                                              530
C     SECTION FOR DETERMINING MAGNETIC FIELD COMPONENTS                      535
  230 IF (NFIELD) 240,250,240                                                540
  240 CALL MAGFIELD (KD,LINE,INDEX)                                          545
  250 RETURN                                                                 550
C                                                                            555
  260 FORMAT (10A8)                                                          560
```

```
      270 FORMAT (1H1,10A8///)                                                  565
      280 FORMAT (8I10)                                                         570
      290 FORMAT (4E15.8,2I10)                                                  575
      300 FORMAT (I10)                                                          580
      310 FORMAT (2I5,5F10.5)                                                   585
      320 FORMAT (39H ELLIPTIC PARTIAL DIFFERENTIAL EQUATION////13H DATA REA    590
         1D IN//)                                                               595
      330 FORMAT (17H MESH INFORMATION//4E15.8,2I10//)                          600
      340 FORMAT (1H ,2I5,5F10.5)                                               605
      350 FORMAT (1H1)                                                          610
      360 FORMAT (31H MESH NODE HAS UNSPECIFIED CODE)                           615
      370 FORMAT (11H AFTER THE ,I5,32H ITERATION THE MAXIMUM CHANGE IS,E20.    620
         18/23H WHERE THE POTENTIAL IS,E20.8,14H AT MESH POINT,I5///)            625
      380 FORMAT (22H ACCURACY CRITERIA MET//22H OUTPUT RESULTS FOLLOW)          630
      390 FORMAT (41H1POTENTIAL VALUES AT EACH EACH MESH POINT///(6E20.8))      635
      400 FORMAT (14H1MAP OF REGION///)                                         640
      410 FORMAT(1H ,(7F10.4///))                                               645
      420 FORMAT (I5,5E15.8)                                                    650
      430 FORMAT (29H CARDS OUT OF ORDER TRY AGAIN///)                          655
      440 FORMAT (17H CONTINUATION JOB///)                                      660
          END                                                                   665
```

```
      SUBROUTINE GETC0(MESH,K0DE,C1,C2,C3,C4,C5)
C     CALCULATES DIFFERENCE C0EFFICIENTS F0R FLUX TYPE ELLIPTIC EQUATI0N
      DIMENSI0N H(1000,4)
      C0MM0N BETA,DELTAZ,DELTAR,NR,NC,NPIA,H
      IF(K0DE-2)1,2,3
    1 PRINT 103
  103 F0RMAT(14H ERR0R IN C0DE)
    2 AMEW=1./((DELTAR)**2*H(MESH,2)*H(MESH,4)*(H(MESH,2)+H(MESH,4)))
      ALAM=1./((DELTAZ)**2*H(MESH,1)*H(MESH,3)*(H(MESH,1)+H(MESH,3)))
      NR0=MESH/NC
      R=FL0ATF(NR0)*DELTAR
      AKAP=2.*AMEW*(H(MESH,4)+H(MESH,2))-(DELTAR/R)*AMEW*(H(MESH,4)**2-H
     A(MESH,2)**2)+2.*ALAM*(H(MESH,3)+H(MESH,1))
      C1  =2.*ALAM*H(MESH,3)/AKAP
      C2  =AMEW*H(MESH,4)/AKAP*(2.-(DELTAR/R)*H(MESH,4))
      C3  =2.*ALAM*H(MESH,1)/AKAP
      C4  =AMEW*H(MESH,2)/AKAP*(2.+(DELTAR/R)*H(MESH,2))
      C5=0.
   18 RETURN
    3 C1  =H(MESH,3)/(H(MESH,3)+H(MESH,1))
      C2=0.
      C3  =H(MESH,1)/(H(MESH,3)+H(MESH,1))
      C4=0.
      C5=0.
      G0 T0 18
      END
```

```
      SUBROUTINE ITRATE(DMAX,KD,UMAX,MMAX)
C     PERFORMS SOR TYPE CALCULATION AND FINDS LARGEST CHANGE AT A MESH
C     POINT
      DIMENSION KD(1000),C(1000,5),UMAT(1000),H(1000,4)
      COMMON BETA,DELTAZ,DELTAR,NR,NC,NPIA,H,C,UMAT
      IF(BETA)10,20,10
   20 BETA=1.
   10 CONTINUE
      DMAX=0.
      DO 1 I=1,NPIA
      IF(KD(I)-2)1,2,2
    2 K=I+NC
      L=I-NC
      RES=C(I,1)*UMAT(I+1)+C(I,2)*UMAT(K)+C(I,3)*UMAT(I-1)+C(I,4)*UMAT(L
     <)+C(I,5)-UMAT(I)
      DELTA=BETA*RES
      UMAT(I)=UMAT(I)+DELTA
      IF(ABSF(DELTA)-DMAX)1,3,3
    3 DMAX=ABSF(DELTA)
      UMAX=UMAT(I)
      MMAX=I
    1 CONTINUE
      RETURN
      END
```

```
      SUERCUTINE MAGFIELD(KD,LINE,INDEX)
C     C0MPLTES  THE MAGNETIC FIELD C0MP0NENTS FR0M VECT0R P0TENTIAL
C     AND IF SPECIFIED PERF0RMS THE LINE INTEGRAL 0F H*DL AND DETERMINES
C     THE FIELD INDEX AL0NG THE MEDIAN PLANE
      DIMENSI0N HR(1000),HZ(1000),KD(1000),ST0T(10),ANDEX(50),UMAT(1000)
     D,DUM(9000)
      C0MMEN BETA ,DELTAZ,DELTAR,NR,NC,NPIA,DUM,UMAT
  707 D0 40 I=1,NPIA
      IF(KD(I)-2)40,41,42
   41 NR0=   I /NC
      RR=DELTAR*FL0AT(NR0)
      HR(I)=-(1./RR)*0.5*(1./DELTAZ)*(UMAT(I+1)-UMAT(I-1))
      HZ(I)= (1./RR)*0.5*(1./DELTAR)*(UMAT(I+NC)-UMAT(I-NC))
      G0 T0 40
   42 HR(I)=0.
      HZ(I)=0.
C
C     THIS VALUE 0F HZ IS N0T C0RRECT AND LEADS T0 WR0NG INDICIES AT
C     P0INTS NEAR THE AXIS    THIS CAN BE FIXED EASILY IF NECESSARY
C
   40 C0NTINUE
C     TEMPERY EXPEDIENCY F0R HZ 0N THE AXIS
      D0 720 I=1,NC
  720 HZ(I)=HZ(I+NC)
C
      PRINT 99
   99 F0RMAT(27H1MAGNETIC FIELD CALCULATI0N/////)
      PRINT 1011,(HR(I),I=1,NPIA)
 1011 F0RMAT(17H RADIAL C0MP0NENT/////(6E20.8))
      PRINT 1012,(HZ(I),I=1,NPIA)
 1012 F0RMAT(23H1L0NGITUDINAL C0MP0NENT////(6E20.8))
      PRINT 591
  591 F0RMAT(27H1MAP 0F REGI0N RADIAL FIELD///)
      D0 99 J=1,NC
   99 PRINT158,(HR(I),I=J,NPIA,NC)
      CALL QUICKMAP(HR,7,7,1)
      PRINT 592
  592 F0RMAT(33H1MAP 0F REGI0N L0NGITUDINAL FIELD///)
      D0  9 J=1,NC
    9 PRINT158,(HZ(I),I=J,NPIA,NC)
  158 F0RMAT(1H ,(7F10.4///))
      CALL QUICKMAP(HZ,7,7,1)
C     SECTI0N F0R CALCULATING LINE INTEGRAL H*DL
  708 IF(LINE)709,710,709
  709 READ 411,NCURVE
  411 F0RMAT(I15)
      N0FIG=NCURVE
  409 READ 400,INT,NUMR,
  400 F0RMAT(2I15)
      PRINT 450,INT,NUMR
  450 F0RMAT(26H1LINE INTEGRAL CALCULATI0N//15H STARTING P0INT,I5//6H WI
     FDTH,I5///)
      LASTX=NUMR*NC+INT
      NR0W=INT/NC
      NC0L=INT-NR0W*NC
      NUMC=NC-2*NC0L
```

```
      LASTY1=INT+NUMC+1
      LASTY2=LASTX+NUMC+1
      SR1=.5*DELTAR*(HR(INT)+HR(LASTX))
      SR2=0.
      IR=INT+NC
      LR=LASTX-NC
      DO 401 I=IR,LR,NC
  401 SR2=SR2+DELTAR*HR(I)
      SR=2.*(SR1+SR2)
      SZ11=.5*DELTAZ*(HZ(INT)+HZ(LASTY1))
      IZ=INT+1
      LZ=LASTY1-1
      SZ12=0.
      DO 402 I=IZ,LZ
  402 SZ12=SZ12+HZ(I)*DELTAZ
      SZ11=-SZ11
      SZ12=-SZ12
      SZ1=SZ11+SZ12
      SZ21=.5*DELTAZ*(HZ(LASTX)+HZ(LASTY2))
      IZ=LASTX+1
      LZ=LASTY2-1
      SZ22=0.
      DO 403 I=IZ,LZ
  403 SZ22=SZ22+HZ(I)*DELTAZ
      SZ2=SZ21+SZ22
      STOT(NCURVE)=SR+SZ1+SZ2
      PRINT 404,SZ11,SZ12,SZ21,SZ22,SZ1,SZ2
  404 FORMAT(27H Z CONTRIBUTION TO INTEGRAL/// 6H SZ11=,E20.8// 6H SZ12=
     F,E20.8// 6H SZ21=,E20.8// 6H SZ22=,E20.8// 5H SZ1=,E20.8// 5H SZ2=
     F,E20.8//////)
      PRINT 405,SR1,SR2,SR
  405 FORMAT(27H R CONTRIBUTION TO INTEGRAL/// 5H SR1=,E20.8// 5H SR2=,E
     F20.8// 5H SR= ,E20.8//////)
      PRINT 406,STOT(NCURVE)
  406 FORMAT(27H TOTAL LINE INTEGRAL EQUALS,E20.8///)
      IF(NCURVE-1)710,710,408
  408 PRINT 410
  410 FORMAT(18H NEXT CASE FOLLOWS)
      NCURVE=NCURVE-1
      GO TO 409
C     SECTION FOR CALCULATING THE FIELD INDEX IN THE MEDIAN PLANE
  710 IF(INDEX)711,407,711
  711 SSUM=0.
      DO 500 I=1,NORIG
  500 SSUM=SSUM+STOT(I)
      AVSUM=SSUM/FLOATF(NORIG)
C     NOW CALCULATE INDUCTANCE
      CUR=1./12.56637061*AVSUM
      ANDUCT=6.28318531/CUR
C     STORED ENERGY
      ESTORE=3.141592654*CUR
      PRINT 501,(STOT(I),I=1,NORIG)
  501 FORMAT(21H1LINE INTEGRAL VALUES////(6E20.8))
      PRINT 502,AVSUM
  502 FORMAT(///34H AVERAGE VALUE OF LINE INTEGRAL = ,E20.8///)
      PRINT 503,CUR,ANDUCT,ESTORE
```

```
503 F0RMAT(11H CURRENT = ,E20.8///14H INDUCTANCE = ,E20.8///17H ST0RED
   F ENERGY =,E20.8/)
    NSTART=NC/2+1
    D0 504 I=NSTART,NPIA,NC
    NR0W=I/NC
    RVAL=DELTAR*NR0W
    QUOT=RVAL/HZ(I)
    DERIV=.5*1,/DELTAR*(HZ(I+NC)-HZ(I-NC))
    IC=NR0W+1
504 ANDEX(IC)=QUOT*DERIV
    PRINT 505,(ANDEX(I),I=1,NR)
505 F0RMAT(31H1FIELD INDEX AL0NG MEDIAN PLANE///(E20.8))
407 RETURN
    END
```

```
      SUBROUTINE QUICKMAP (F,NX,NY,IXROTATE)

C     QUICKMAP PRODUCES A PRINTOUT MAPPING OF A 2-DIMENSIONAL ARRAY, F(NX,NY),
C     WHERE THE MAXIMUM VALUE IS DENOTED BY *, VALUES ABOVE 90 PERCENT OF THIS
C     MAXIMUM BY 9, ABOVE 80 PERCENT BY 8,.... AND THOSE BELOW 10 PERCENTBY 0.
C     IXROTATE=1 WILL CAUSE THE ARRAY TO BE PRINTED WITH THE X-DIRECTION GOING
C     ACROSS THE PAGE, AND Y GOING UPWARDS.
C     IXROTATE=2 PRINTS X DOWN THE PAGE, AND Y ACROSS.
C     QUICKMAP IS LIMITED TO A MAXIMUM CROSSWISE DIMENSION OF 65.

      DIMENSION F(NX,NY), FROW(65)
  100 FORMAT (65(1XF1.0))
  101 FORMAT (/24H ERROR IN QUICKMAP.  NX= I4, 6H,  NY= I4/)
  102 FORMAT (/7H PEAK = E12.4 / 28H DIFFERENCE BETWEEN LEVELS = E12.4/)
      GO TO (11,12), IXROTATE
   11 IF (NX.LE.65) GO TO 1
      IXROTATE = 2
   12 IF (NY.GT.65) GO TO 9

    1 FMAX = FMIN = F(1,1)
      DO 2 I=1,NX
      DO 2 J=1,NY
      IF (F(I,J) - FMAX) 14,2,13
   13 FMAX = F(I,J)
      GO TO 2
   14 IF (F(I,J).LT.FMIN) FMIN=F(I,J)
    2 CONTINUE
      FSCALE = 10./(FMAX - FMIN)
      REGION = 1./FSCALE
      PRINT 102, FMAX, REGION
      GO TO (3,6), IXROTATE
    3 DO 5 IY=1,NY
      J = NY-IY+1
      DO 4 I=1,NX
    4 FROW(I) = INTF((F(I,J) - FMIN)*FSCALE)
    5 PRINT 100, (FROW(I), I=1,NX)
      RETURN
    6 DO 8 I=1,NX
      DO 7 J=1,NY
    7 FROW(J) = INTF((F(I,J) - FMIN)*FSCALE)
    8 PRINT 100, (FROW(J), J=1,NY)
      RETURN
    9 IF (NX.GT.65) GO TO 10
      IXROTATE = 1
      GO TO 1
   10 PRINT 101, NX, NY
      RETURN
      END
```

```
      SUBROUTINE PUNK
C     PUNCHES CARDS FOR FUTURE TRAJECTORY CALCULATIONS
      COMMON DUMMY,NPIA,AJUNK,UMAT
      DIMENSION DUMMY(5),AJUNK(9000),UMAT(1000)
  705 DO 492 I=1,NPIA,5
  492 PUNCH 593,I,UMAT(I),UMAT(I+1),UMAT(I+2),UMAT(I+3),UMAT(I+4)
  593 FORMAT(I5,5E15,8)
      RETURN
      END
```

REFERENCES

1. Forsythe, G. E. and Wasow, W. R., "Finite Difference Methods for Partial Differential Equations", Wiley, 1960.

2. Young, D. M. and Frank, T. G., "A Survey of Computer Methods for Solving Elliptic and Parabolic Partial Differential Equations", International Computation Centre Bulletin (Rome), Vol. 2, No. 1, January 1963, pp. 1-61.

3. Carré, B. A., "Determination of the Optimum Accelerating Factor for Successive Over-Relaxation", Computer Journal, Vol. 4, No. 1, April 1961, pp. 73-78.