LA-4676

DR-1837

MASTER

TWODIM, a Computer Code for Unfolding Diametral Gamma-Ray Scans on Reactor Fuel Elements

los alamos
scientific laboratory
of the University of California
LOS ALAMOS, NEW MEXICO 87544

UNITED STATES
ATOMIC ENERGY COMMISSION
CONTRACT W-7405-ENG. 36

Printed in the United States of America. Available from
National Technical Information Service
U. S. Department of Commerce
5285 Port Royal Road
Springfield, Virginia 22151
Price: Printed Copy $3.00; Microfiche $0.95

LA-4676
UC-80
ISSUED: May 1971

los alamos
scientific laboratory
of the University of California
LOS ALAMOS, NEW MEXICO 87544

TWODIM, a Computer Code for Unfolding Diametral Gamma-Ray Scans on Reactor Fuel Elements

by

B. K. Barnes*
J. R. Phillips

*Lowell Institute of Technology, Lowell, Mass.

TWODIM, A COMPUTER CODE FOR UNFOLDING DIAMETRAL GAMMA-RAY SCANS ON REACTOR FUEL ELEMENTS

by

B. K. Barnes and J. R. Phillips

ABSTRACT

A computer code has been written to accept one or more diametral scans taken at different angles around the outside of a fuel element and use this information to determine the distribution of the measured isotope over a cross section of the fuel element. The calculated distribution is plotted both as an isometric projection plot and as a density plot. Fission-product distributions calculated with this code were compared to actual distributions measured on thin cross sections of the fuel.

INTRODUCTION

The most common forms of gamma-ray scans made on reactor fuel elements are the axial and diametral scans. These are scans made from outside the fuel element along the major axis or across the major axis of the fuel element. For the axial scan, changes in the gamma-ray intensity are relatively simple to relate to variations in the amount of the particular isotope being considered. This is so because, in general, the detector is looking at practically a constant amount of fuel in each position as the fuel element is moved in front of the detector. For the diametral scan, on the other hand, variations in intensity are caused by changes in either the isotopic composition of the fuel, the amount of fuel being observed by the collimated detector, or the gamma-ray absorption. This makes interpretation of the diametral scans more difficult. The information desired is the distribution of the isotope over a cross section of the fuel element taken at the location of the diametral scan. While a single diametral scan yields insufficient information to determine this uniquely, an approximation to the distribution may be determined from a single scan. Two or more diametral scans made at different angles around the fuel element improve the approximation of the calculated distribution to the actual isotopic distribution.

PROCEDURE

Two techniques were considered for unfolding the diametral scans to two-dimensional distributions. The first was to subdivide the face of the fuel element cross section into an N x N matrix and assume a uniform intensity over each individual N x N square element in the face of the section. This would then be treated as a system of N^2 unknowns, and a sufficient number of scans would be made at different angles to yield a total of N^2 separate intensity measurements from outside the fuel element. Each measurement could then be written as a sum of the N^2 intensities across the fuel element, each intensity multiplied by its individual geometrical factor to take into account the collimation of the detector and the absorption of the gamma ray by the fuel element between the point of origin of the gamma rays and the detector. This technique would result in a system of N^2 equations in N^2 unknowns, which could be solved by matrix inversion or elimination techniques to determine the intensities of gamma-ray production and hence the isotope concentrations for each small area.

Although this appears to be the most straightforward technique, it was not used for several reasons. A typical diamtral scan across a fuel element may consist of 100 points. To preserve this accuracy across the sectional face of the fuel element would involve a 100 x 100 matrix or solving a set of 10,000 equations in 10,000 unknowns. Measurements of the intensity would have to be made at 10,000 points, for example, 100 scans of 100 points each, taken at different angles. Both the data acquisition and the data reduction would be very time consuming. The matrix and the scans must be carefully chosen to obtain a nonsingular set of equations for solution. In addition, the usual single diametral scan across a fuel element is not very suitable for calculation by this method. A 100-point scan would determine only a 10 x 10 matrix of values, which is too coarse a grid to be useful. In general, this method of solution would ignore the fact that the variations in intensity of interest are radial, with angular variations existing but not being as significant as the radial variations nor as rapidly changing.

The method acutally used for data unfolding was to subdivide the scan region into a series of concentric rings of the same width as the distance between scan points on the diametral scan and to determine the intensities for this geometry. Inverting large matrices was avoided by using a technique that determined intensities at points in these rings one at a time, rather than by simultaneous solution of a set of equations.

In general, the circular scan region is somewhat larger than the diameter of the fuel element being scanned, as the data taken starts off the fuel on one side of the element and goes past the edge of the fuel on the other side of the element. One requirement is that the scan region must be centered on the fuel element, that is, must extend the same distance left and right from the centerline of the fuel element. In the event that this centerline is not known, provision is made to attempt to select a centerline from a single scan by symmetrical considerations.

The details of a diametral scan consisting of eight scan points (rather than the typical 100) are indicated schematically in Fig. 1. The figure is a cross-section drawing with the diametral scan in the plane of the figure. The cross-hatched circle is the fuel element being scanned, which extends above and below the plane of the figure. A typical diameter for fuel elements scanned is 0.25 in.

The detector typically is collimated to view 0.004 in. along the scan direction (in the plane of the figure) and 0.5 in. perpendicular to the scan direction (perpendicular to the plane of the figure). The dashed circles indicate the scan region which would be taken and how it would be subdivided for analysis. The horizontal dashed lines indicate the position of the right edge of the collimator (looking at the fuel element through the collimator) for each scan point. Typically, these scan points are separated by 0.003 in.

The procedure for the reduction of the data is to determine the intensities of the various rings of the scan region, one ring at a time, starting at the outside and working toward the center. Two points are taken from each scan for each ring. For the example shown in Fig. 1, the points labeled 1 and 8 would be used to determine intensities at two points on the outside ring. This is done by using a Monte Carlo technique to evaluate the average absorption over the area of the ring that can be observed by the collimated detector. Note that for points 1 and 8, only areas within the outer ring can be viewed by the detector. A number of points within this observed area are randomly chosen, and the absorption of the gamma rays in traveling from each of these points to the edge of the fuel element is calculated. An average absorption factor is calculated. The area of the fuel element observed by the detector is determined by the same

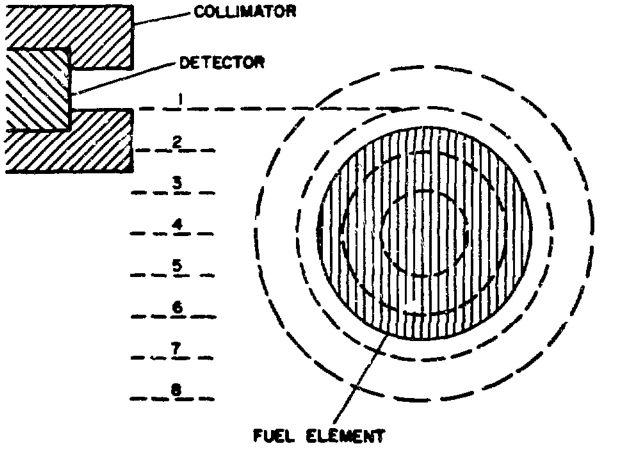

Fig. 1. A typical diametral scan, showing detector, fuel element, and regions used to analyze the data.

Monte Carlo technique. The observed intensity is corrected for absorption and divided by the area to get the intensity per unit area for the two points on the outside ring. These two points, together with any others determined by other scans taken at different angles around the element, are fitted as a function of angle with a spline function by a least squares technique. Thereafter, the intensity of the gamma-ray production around this outer ring is assumed to be known as a function of angle. The intensities in the second ring are calculated using this knowledge of the intensity per unit area from the first ring and points 2 and 7 from Fig. 1. In the case of the second ring and all subsequent rings, the Monte Carlo technique is used to evaluate the contribution from the known rings in the area viewed by the detector, as well as the gamma-ray absorption and the area of the unknown region. By the way the rings and scan points are chosen, there is, in each case, only one unknown region, together with one or more regions for which the intensity is known as a function of angle, thus allowing the intensity of the unknown region to be calculated. Additional intensities are again calculated from any other scans taken at different angles. Again, these points are fitted as a function of angle by the least squares spline fitting function. Thereafter, the intensities of the first and second rings are assumed to be known as a function of angle. This allows intensities to be calculated in the third ring. This procedure is continued until the intensities of all the rings are known as a function of angle. In general, each ring has 2N data points where N is the number of scans made at different angles around the fuel element. The case N = 1 (a single scan), however, is treated as a special case, and each ring is given four data points by a linear interpolation technique. As this procedure is followed, any error (statistical or systematic) in the determination of intensities in the first rings is carried through into the determination of the intensities on the inner rings. Thus, this technique leads to larger uncertainties for the intensities in the interior of the fuel element than on the exterior. As the intensities have larger uncertainties as one proceeds toward the interior of the fuel element, the least squares spline fitting is used as a smoothing function, with the degree of smoothing becoming greater as one proceeds toward the interior of the fuel element. This is accomplished by reducing the number of knots of the pline function fitted to the data that has the effect of reducing the free parameters in fitting the spline functions. The spline function fitting program uses the SMOOTH code package (Los Alamos Scientific Laboratory Library Subprogram E203A).*

After determining the intensities, the program allows the option of using the least squares spline function to resmooth the data in a radial direction from the center to the edge of the fuel element. If this option is selected, spline functions are fitted to sets of data points consisting of one data point from each ring, each set of points being along one radius. These fits are made to as many radii as there are data points in the rings (2N, in general). The smoothing is effected by fitting a spline function with nine-tenths as many knots as there are data points.

The programs used to plot the results have been described previously,[1] except that the main program (TWOP) was changed to a subroutine (with no arguments) so that it could be called by TWODIM to plot the results. Because TWOP requires a square matrix of data points, the gamma-ray intensity at each point of a 64 x 64 matrix is calculated from the spline function fits as a function of angle and radius. After calculating this square matrix of intensities, it is written on magnetic tape by subroutine TW, and TWOP is called to plot the data both as an isometric projection plot and as a density plot.

CODES

The computer code consists of the main program, TWODIM, and six subroutines (Appendix A). An additional eight subroutines are used to plot the results (Appendix B).

Program TWODIM

The main program, TWODIM, reads the input data and performs the calculations to unfold the scans

*Copies of all Los Alamos Scientific Laboratory Library Subprograms referenced in this report may be obtained by writing C Division, Los Alamos Scientific Laboratory.

to distributions. The intensities as a function of position for the selected gamma ray are taken from magnetic tape. A listing of this program, together with subroutines and typical data, is given in Appendix A. The required input data and formats for card input are given in Table I.

TABLE I

INPUT FOR TWODIM

Card Number	Variables	Format
1	ITITLE	8A10
2	ADS,RR,AMPS,AMOC	6F12.2
3	NSCAN,NPPSCN,NDPSCN	12I6
4 to (4+NSCAN-1)	REC,RECP	6F12.2

ITITLE is any title desired, up to 80 characters in length, and is printed with the results.

ADS is the gamma-ray adsorption coefficient for the material of the fuel element and for the particular gamma-ray energy being considered. This must be in units of $mils^{-1}$ (1/0.001 in.).

RR is the radius of the fuel element being scanned and must be in units of mils (0.001 in.). It is used in calculating the adsorption for the gamma rays in the fuel element.

AMPS is the number of mils per scan point, that is, the number of mils advanced between each scan point taken.

AMOC is the width of the collimator in mils and is measured in the plane of the scan.

NSCAN is the number of scans taken at different angles and must be greater than or equal to 1.

NPPSCN is the number of points per scan, that is, the number of steps taken in one scan across the fuel element. Scans at various angles around the fuel element must have the same number of points in each scan.

NDPSCN is the number of degrees between successive scans at different angles. The program assumes successive scans are made at regular angle increments. Thus, if NDPSCN=45 and NSCN=4, the program would assume scans were made at 0°, 45°, 90°, and 135°. This is illustrated in Fig. 2. NSCAN*NDPSCN must be less than 180°.

REC is the record number of the first record on magnetic tape that contains the first point of the scan.

RECP is the record number of the record on magnetic tape containing the last point of the same scan. If RECP=0, the records from REC to REC+NPPSCN-1 are taken to the records on magnetic tape containing the points (measured gamma-ray intensities) of the scan. This can be centered on the fuel element. If RECP≠0, the region from REC to RECP is taken to contain the scan, but it is not assumed that the region is centered on the fuel element. A scan center is chosen by the computer on the basis of symmetry, and the scan is taken to be the longest series of points which can be centered around this scan center and stay within the region between REC and RECP. The option to have the computer locate the center of the fuel element must be used with caution because the physical center of the fuel element is not necessarily the center of symmetry of the distribution of the gamma-ray-producing isotope.

The scan data points are taken from magnetic tape and are in the form of one record per scan point. Each record may contain several measured intensities for different gamma-ray energies. The format of the statement reading

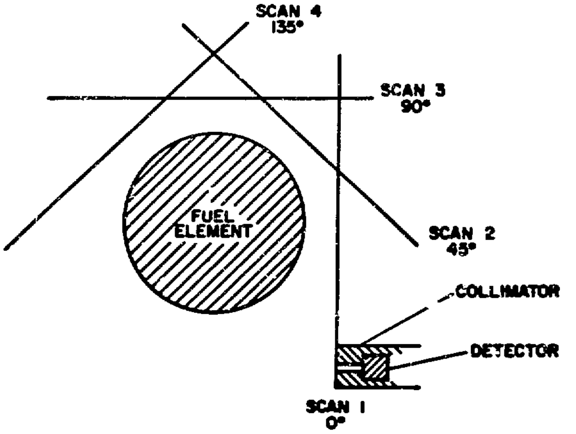

Fig. 2. A typical series of four scans with 45° of rotation between the scans. The lines represent the path of the collimated detector in making the scans. The detector is collimated to view a narrow portion of the fuel element.

the magnetic tape, together with the data list, is easily changed to be compatible with any data. This change is made by changing Card 85, a read statement in program TWODIM (Appendix A), and Card 464 (Format 81), the format used by the read statement. At present, two data tag numbers (ITAG1, ITAG2), the record number (IREC), and three gamma-ray intensities [SUM(1), SUM(2), SUM(3)] are read. The gamma-ray intensity to be used is determined by the value of the variable N, which is set by Card 63 (Appendix A). If N=1, SUM(1) is used, if N=2, SUM(2) is used, etc. The input data for the plot routines has been described.[1]

RESULTS

The program TWODIM was tested using three known isotopic distributions. These distributions were known from previous two-dimensional scans of a thin section of fuel that was excised from the fuel element. The data represented three distinct distributions in the form of density plots to allow the comparison of the results with the results obtained by ring unfolding. The known distributions were added along the X and Y axes to convert the known two-dimensional scans into two one-dimensional projections, simulating the scans obtained by diametral scanning at 0° and 90° rotation. These projections are identical to the data that would be obtained if the fuel element were diametrally scanned and if there were no absorption of the gamma-ray intensity. The projections used for listing the program are shown in Figs. 3 through 8. Figures 3 and 4 show the distribution of the ^{95}Nb isotope that would have resulted if the fuel element had been routinely scanned. The distributions of ^{106}Rh and ^{137}Cs are shown in Figs. 5 through 8, respectively. The program TWODIM unfolded concentric rings in the projections to generate a 4096-point matrix (64 x 64) which was plotted as b[illegible] isometric projections and density plots.

The results are shown in Figs. 9 through 14 and should be compared to the known distributions shown in Figs. 15a, b, and c. The density plots of the uniformly distributed ^{95}Nb isotope calculated from TWODIM (Fig. 9) and from the known

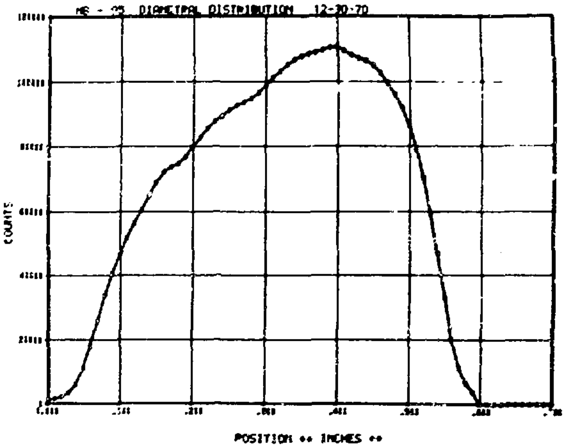

Fig. 3. ^{95}Nb diametral distribution taken at 0°.

results (Fig. 15a) are nearly identical. The most difficult of the three isotopes was the ^{106}Rh distribution that was concentrated in a ring offset f from the true center. The density plot from ring unfolding (Fig. 11) shows the same distribution as the known distribution (Fig. 15b).

The ^{137}Cs distribution Figs. 13 and 15c was chosen to show that, if the isotope were concentrated on the outside surface, the program would function normally when calculating the center region.

CONCLUSIONS

The comparison of the calculated isotopic distribution and the known distributions indicates that, in many cases, it is not necessary to section a fuel element to determine the radial distributions of fission products in the fuel matrix. To obtain

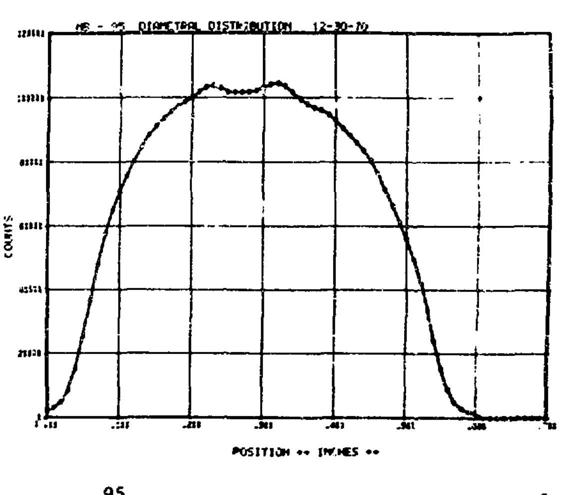

Fig. 4. ^{95}Nb diametral distribution taken at 90°.

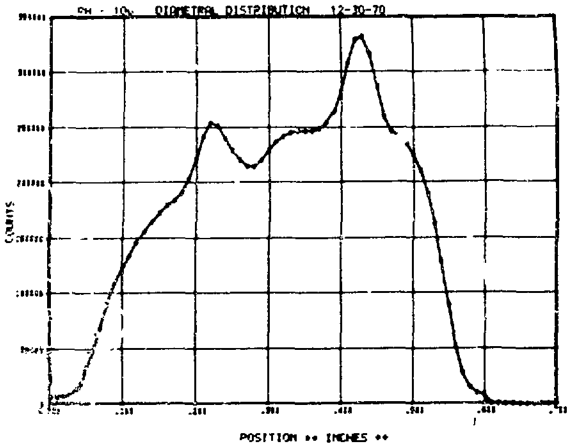

Fig. 5. ^{106}Rh diametral distribution taken at 0°.

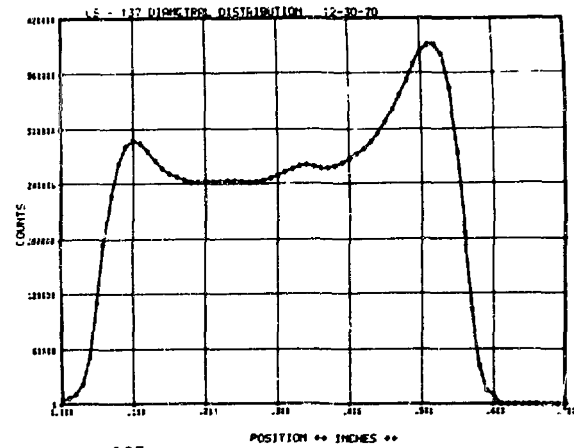

Fig. 7. ^{137}Cs diametral distribution taken at 0°.

the necessary data for TWODIM, a fuel element has only to be gamma scanned at one or more different rotations. The isotopic distributions are then calculated and plotted as isometric projections and density plots.

ACKNOWLEDGMENTS

The authors express their appreciation to W. M. Sanders and D. M. Holm for providing the test data for checking the program. We would also like to thank G. R. Waterbury and G. H. Mottaz for their assistance.

REFERENCE

1. B. K. Barnes and W. M. Sanders, "Computer Codes for Presentation, Interpolation, and Integration of Two-Parameter Data," LA-4326, Los Alamos Scientific Laboratory (1969).

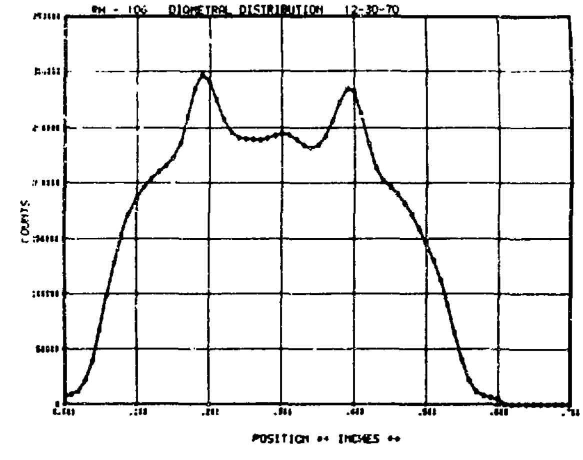

Fig. 6. ^{106}Rh diametral distribution taken at 90°.

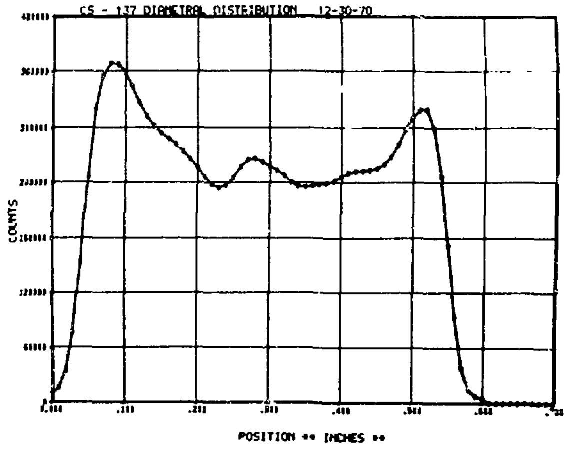

Fig. 8. ^{137}Cs diametral distribution taken at 90°.

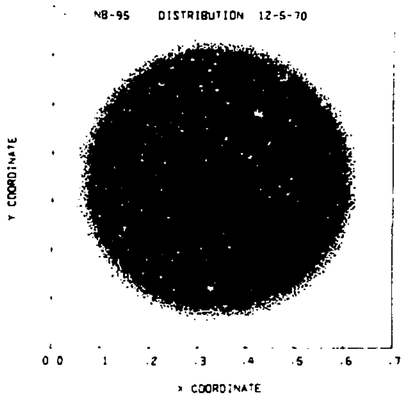

Fig. 9. Density plot of ^{95}Nb calculated from TWODIM.

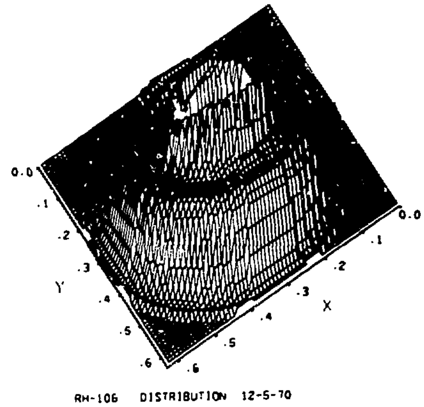

Fig. 12. Isometric projection of ^{106}Rh calculated from TWODIM.

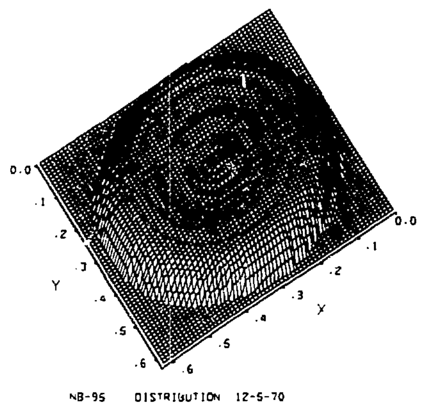

Fig. 10. Isometric projection of ^{95}Nb calculated from TWODIM.

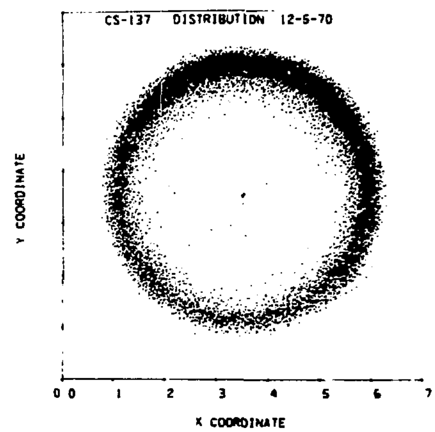

Fig. 13. Density plot of ^{137}Cs calculated from TWODIM.

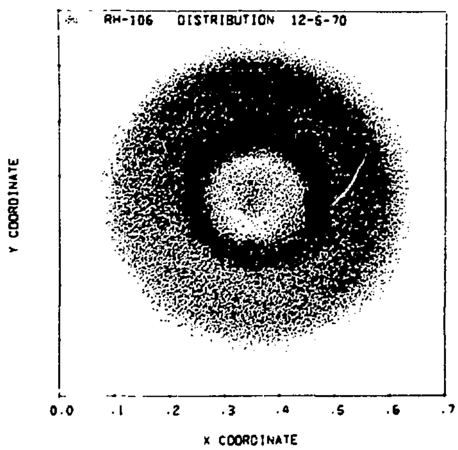

Fig. 11. Density plot of ^{106}Rh calculated from TWODIM.

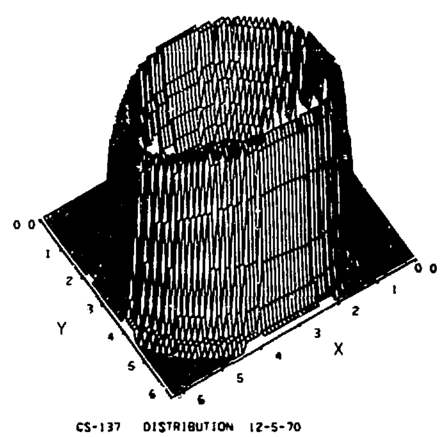

Fig. 14. Isometric projection of ^{137}Cs calculated from TWODIM.

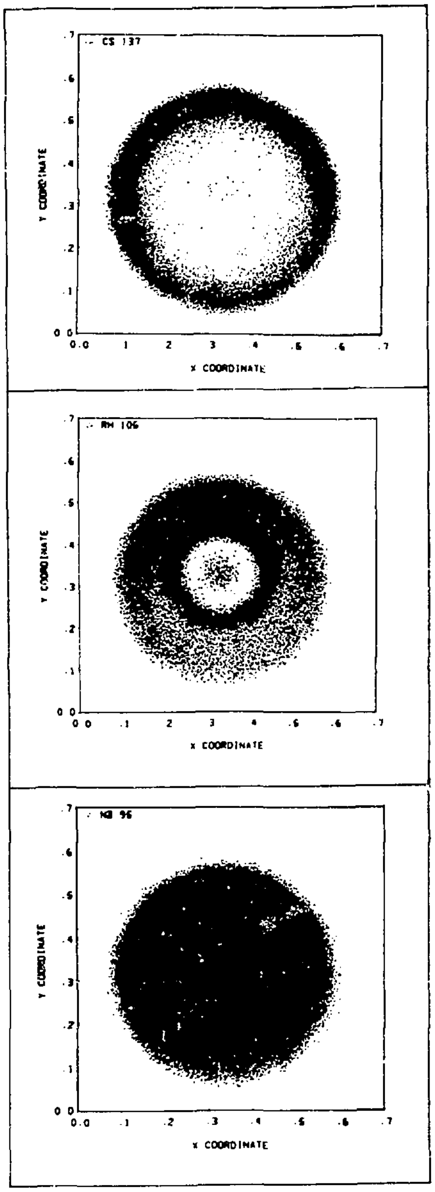

Fig. 15a,b,c. Density plots of the known distributions of ^{95}Nb, ^{106}Rh, and ^{137}Cs.

APPENDIX A

TWODIM AND SPLINE SUBROUTINES

```
      PROGRAM TWODIM (INPUT,OUTPUT,BGSB,TAPE7,TAPE2,FILM,TAPE10=INPUT,TA
     1PE9=OUTPUT,TAPE12=FILM)
      DIMENSION SCAN(10,110), RAND(12100), ANG(10,110), TITLE(8), SUM(3)
     1, A(10,110), W(10,110), NOPTF(110)
      COMMON NOPTF,ANG,A,W,NK
      COMMON /W/ M,TITLE,XS,DELX,YS,DELY,RAND
      CALL TW
C     0.002355 MILLS-1   ZR-NB-95   765KEV   85/15 UC/PUC + 10W/O M2C3
C     0.002837 MILLS-1   KH-106   622KEV   97 0/0 THEORECTICAL   (13.6G/CM3)
      DO 2 J=1,10
      DO 1 K=1,110
      A(J,K)=0.
      W(J,K)=0.
1     CONTINUE
2     CONTINUE
      DC=48000.
      DCSQ=DC*DC
C     DC IS DISTANCE FROM DETECTOR TO CENTER OF FUEL
C     (IN MILLS)
C      DISC BEING SCANNED IS CENTERED ON THE ORIGIN OF THE COORDINATE SY
      READ (10,63)TITLE
      IF (ENDFILE 10) 3,4
3     CALL EXIT
4     CONTINUE
      WRITE (9,64)
      WRITE (9,63)TITLE
C     ADS IS THE GAMMA RAY ADSORPTION COEFFICIENT
C     FOR THE GAMMA RAY BEING CONSIDERED IN
C     THE MATERIAL OF THE FUEL ELEMENT
C     THIS MUST BE IN (MILLS)**(-1)
      READ (10,67)ADS,RD,AMPS,AMOC
      RR=RD/2.
      WRITE (9,68)ADS,RR,AMPS,AMOC
      MPS=AMPS
C     MOC IS WIDTH OF COLLIMATOR IN MILLS
C     RR IS THE RADIUS (IN MILLS) OF THE DISC BEING SCANNED
      RR2=RR**2
C     MPS IS NUMBER OF MILLS BETWEEN SCAN POINTS
      READ (10,65)NSCAN,NPPSCN,NDPSCN
      IF (NSCAN.EQ.1) NDPSCN=90
C     NSCAN IS NUMBER OF SCANS AT DIFFERENT ANGLES
      WRITE (9,75)NSCAN,NPPSCN,NDPSCN
C     NPPSCN IS NUMBER OF POINTS PER SCAN
C     NDPSCN IS NUMBER OF DEGREES BETWEEN SCANS
      PTS=5000.0
C     PTS IS THE NUMBER OF POINTS USED IN MONTE CARLO CALCULATION
      PCT=(SQRT(PTS)/PTS)*100.
      WRITE (9,80)PTS,PCT
      WRITE (9,66)
C     THE FIRST SCAN IS ACROSS THE BOTTOM OF THE RESULTING TWOD MATRIX,
C     LOOKING UP AT THE MATRIX.  THE DATA POINTS FOR EACH SCAN
C     ARE ASSUMED TO BE FROM RIGHT TO LEFT WHEN LOOKING DOWN THE
C     COLLIMATOR.  SUCCESSIVE SCANS ARE ASSUMED TO RESULT FROM ROTATING
C     THE DETECTOR AROUND THE SAMPLE IN A COUNTERCLOCKWISE DIRECTION BY
C     AMOUNT OF NDPSCN DEGREES.
      ISCAN=1
C     ISCAN=1 MEANS THE SCAN WAS LEFT TO RIGHT (BACKWARDS) , SO THE
C     MATRIX IS LOADED IN REVERSE.  ISCAN0.NE. 1 MEANS THE SCAN WAS
C     RIGHT TO LEFT, AS THE PROGRAM ASSUMES, SO MATRIX IS LOADED NORMALL
C
C     WHEN THE PIT BUILDING SCANNER SCANS FROM LOW NUMBERS TO HIGH NUMBE
C     THE COLLIMATOR MOVES FROM LEFT TO RIGHT ACROSS THE SAMPLE (BACKWAR
5     CONTINUE
      N=2
      DO 16 I=1,NSCAN
```

```
      J=0
      REWIND 2
      READ (10,67)REC,RECP
C     REC IS FIRST RECORD OF SCAN
C     RECP IS LAST RECORD OF SCAN
C
C     IF ONLY ONE RECORD IS GIVEN, PROGRAM ASSUMES YOU HAVE LOCATED THE
C     CENTER OF THE ELEMENT BEING SCANNED AND CENTERED THE REGION GIVEN
C     BY THE STARTING POINT REC AND NPPSCN AROUND THIS CENTER OF THE
C     ELEMENT.  IF, HOWEVER TWO RECORD NUMBERS ARE GIVEN, THE PROGRAM TA
C     THESE TO BE THE FIRST AND LAST RECORDS OF THE SCAN AND ATTEMPTS TO
C     LOCATE THE CENTER OF THE FUEL ELEMENT ITSELF AND SET UP A REGION O
C     LENGTH NPPSCN AROUND THIS CENTER OF THE FUEL ELEMENT.
      IF (ENDFILE 10) 6,7
6     CALL EXIT
7     CONTINUE
      WRITE (9,67)REC,RECP
      IF (RECP.EQ.0.) GO TO 12
C     LOCATING CENTER OF SYMMETRY OF SCAN
      M=0
      READ (2,78)ITG1,ITG2,IREC,SUM(1),SUM(2),SUM(3)
      IF (IREC.GE.IFIX(REC)) REWIND 2
      BACKSPACE 2
      IREC=0
8     CONTINUE
      READ (2,78)ITG1,ITG2,IREC,SUM(1),SUM(2),SUM(3)
      IF (IREC.LT.IFIX(REC)) GO TO 8
      IF (IREC.GT.IFIX(RECP)) GO TO 9
      M=M+1
      RAND(M)=SUM(N)
      GO TO 8
9     CONTINUE
      ASUM=1.E+320
      M4=M/4
      M41=M4- 1
      M34=M*3/4
      DO 11 J=M4,M34
      BUM=0.
      DO 10 K=1,M41
      BUM=BUM+(RAND(J- K)- RAND(J+K))**2
10    CONTINUE
      IF (ASUM.LT.BUM) GO TO 11
      ASUM=BUM
      JCENT=J
11    CONTINUE
C     JCENT IS THE J VALUE FOR WHICH FOLDING THE SPECTRUM AROUND J
C     MINIMIZES THE DIFFERENCE BETWEEN THE TWO SIDES.
      REC=REC+JCENT- NPPSCN/2
      REWIND 2
12    CONTINUE
      WRITE (9,67)REC
      IRECS=- 1
      J=0
13    CONTINUE
      READ (2,79)ITG1,ITG2,IREC,SUM(1),SUM(2),SUM(3)
14    CONTINUE
      IRECS=IRECS+1
      IF (IREC.LT.IFIX(REC)) IRECS=IREC- 1
      IF (IREC.LT.IFIX(REC)) GO TO 13
      J=J+1
      K=J
      IF (ISCAN.EQ.1) K=NPPSCN- J+1
      L=K- 1
      IF (ISCAN.EQ.1) L=K+1
      IF (IREC.EQ.IRECS+1) GO TO 15
      SCAN(I,K)=SCAN(I,L)
      GO TO 14
15    CONTINUE
      N=2
      SCAN(I,K)=SUM(N)
      IF (IREC+1.LT.IFIX(REC)+NPPSCN) GO TO 13
      WRITE (9,81)(SCAN(I,K),K=1,J)
```

```
16    CONTINUE
C     SCAN(I,J) ARE INTENSITIES MEASURED FOR SCAN POINTS
      R=FLOAT(NPPSCN*MPS)/2.
      WRITE (9,74)R
C     R IS RADIUS OF SCAN REGION
      DA=2.*R*AMOC/PTS
C     DA IS THE DIFFERENTIAL AREA ASSOCIATED WITH EACH MONTE CARLO
C     POINT.  SUM(DA)=A
      KF=NPPSCN-1
      R2=R**2
      RK=R
C     RK IS THE CURRENT RADIUS OF THE KNOWN REGION.  OUTSIDE
C     RK, THE VALUES OF THE INTENSITIES CAN BE CALCULATED.  ORIGINALLY
C     THESE ARE TAKEN TO BE ZERO OUTSIDE THE SCAN REGION.
      NS2=2*NSCAN
      IF (NSCAN.EQ.1) NSCAN=2
      DO 17 J=1,NSCAN
      ANG(J)=(J-1)*NDPSCN
      ANG(NSCAN+J)=180.+(J-1)*NDPSCN
17    CONTINUE
      IF (NS2.EQ.2) NSCAN=1
      WRITE (9,69)
      IF (NSCAN.EQ.1) NS2=4
      WRITE (9,70)(ANG(J),J=1,NS2)
      IF (NSCAN.EQ.1) NS2=2
      WRITE (9,71)
      WRITE (9,72)
      NK=IFIX(FLOAT(NPPSCN)/2.+.5)
      DO 37 K=1,NK
C     FOR EACH SCAN ANGLE TWO POINTS ARE TAKEN INTO CONSIDERATION FOR
C     EACH PASS (INDEX K) OF THE PROGRAM.  THESE ARE ONE FOR THE UPPER
C     HALF OF THE DISC AND ONE FOR THE LOWER HALF OF THE DISC.
      YL=-R+FLOAT((K-1)*MPS)
      YU=YL+AMOC
C     THIS SECTION OF THE PROGRAM IS CONCERNED WITH THE LOWER HALF.
C     FOR Y, L STANDS FOR LOWER EDGE OF BAND, U FOR UPPER EDGE
      XL=-SQRT(R2-YU**2)
      IF (YU.GT.0.) XL=-R
C     FOR X, L STANDS FOR THE LEFT EDGE OF THE BAND (AT ITS WIDEST POINT
      FRACT=-XL/R
      DELX=-2.*XL
C     DELX IS THE WIDTH OF THE BAND
      NP=PTS*FRACT
      NR=2*NP
      DO 26 J=1,NSCAN
      CALL RANVECT (RAND,NR)
      AK=0.
      AD=0.
      DO 23 L=1,NP
      X=XL+DELX*RAND(L)
      Y=YL+AMOC*RAND(2*L)
      RC=SQRT(X**2+Y**2)
C     RC IS CURRENT RADIUS
      IF (RC.GT.R) GO TO 23
      D=DC+X
C     D IS DISTANCE FROM POINT TO DETECTOR
      IF (Y**2.LT.RR2) GO TO 18
      DX=0.
      GO TO 19
18    CONTINUE
      XS=-SQRT(RR2-Y**2)
      DX=X-XS
19    CONTINUE
C     DX IS THE DISTANCE OF THE POINT BEING CONSIDERED
C     FROM THE EDGE OF THE DISC BEING SCANNED
      GAM=EXP(-ADS*DX)*D**2/DCSQ
      IF (RC.LT.RK) GO TO 22
      IF (K.LE.1) WRITE (9,76)
      THTA=57.2958*ATAN(Y/X)
      IF (X.LT.0.) THTA=180.+THTA
      IF (X.GT.0..AND.Y.LT.0.) THTA=360.+THTA
```

```
      THTA=THTA+FLOAT(J- 1)*NDPSCN
      THTA=THTA+90.
20    IF (THTA.LT.360.) GO TO 21
      THTA=THTA- 360.
      GO TO 20
21    CONTINUE
      KR=(R- RC)/AMPS+1.
      IF (KR.GT.NK) KR=NK
      APT=7.
      CALL SPL (THTA,KR,APT)
      AK=AK+APT*GAM*DA
      GO TO 23
22    AD=AD+GAM*DA
23    CONTINUE
      A(J,K)=(SCAN(J,K)- AK)/AD
C     A=A1*E**(- ADS*DEL1)+A2*E**(- ADS*DEL2)
C     THUS A1=(A- A2*E(- ADS*DEL2))/E**(- ADS*DEL1)
C     OR, A1=(A- AK)/AD
      IF (A(J,K).GT.0) GO TO 25
      A(J,K)=0.
      IF (K.LE.1) GO TO 24
      CALL SPL (ANG(J,1),K- 1,APT)
      IF (APT.LT.0.) APT=0.
      A(J,K)=A(J,K)+APT
24    IF (J.LE.1) GO TO 25
      A(J,K)=A(J,K)+A(J- 1,K)
      A(J,K)=A(J,K)/2.
25    CONTINUE
26    CONTINUE
C     THIS SECTION OF THE PROGRAM IS CONCERNED WITH THE UPPER HALF.
      KU=NPPSCN+1- K
      YU=R- FLOAT((K- 1)*MPS)
      J2=NSCAN+1
      DO 35 J=J2,NS2
      CALL RANVECT (RAND,NR)
      AK=0.
      AD=0.
      DO 32 L=1,NP
      X=XL+DELX*RAND(2*L)
      Y=YU- AMOC*RAND(L)
      RC=SQRT(X**2+Y**2)
      IF (RC.GT.R) GO TO 32
      D=DC+X
      IF (Y**2.LT.RR2) GO TO 27
      DX=0.
      GO TO 28
27    CONTINUE
      XS=- SQRT(RR2- Y**2)
      DX=X- XS
28    CONTINUE
      GAM=EXP(- ADS*DX)*D**2/DCSQ
      IF (RC.LT.RK) GO TO 31
      IF (K.LE.1) WRITE (9,77)
      THTA=57.2958*ATAN(Y/X)
      IF (X.LT.0.) THTA=180.+THTA
      IF (X.GT.0..AND.Y.LT.0.) THTA=360.+THTA
      THTA=THTA+FLOAT(J- 1)*NDPSCN+180.
      THTA=THTA+90.
29    IF (THTA.LT.360.) GO TO 30
      THTA=THTA- 360.
      GO TO 29
30    CONTINUE
      KR=(R- RC)/AMPS+1.
      IF (KR.GT.NK) KR=NK
      APT=8.
      CALL SPL (THTA,KR,APT)
      AK=AK+APT*GAM*DA
      GO TO 32
31    AD=AD+GAM*DA
32    CONTINUE
      NJ=J- NSCAN
```

```
      A(J,K)=(SCAN(NJ,KU)- AK)/AD                                        TWOD 279
      IF (A(J,K).GT.0) GO TO 34                                          TWOD 280
      A(J,K)=0.                                                          TWOD 281
      IF (K.LE.1) GO TO 33                                               TWOD 282
      CALL SPL (ANG(J,1),K- 1,APT)                                       TWOD 283
      IF (APT.LT.0.) APT=0.                                              TWOD 284
      A(J,K)=A(J,K)+APT                                                  TWOD 285
33    IF (J.LE.1) GO TO 34                                               TWOD 286
      A(J,K)=A(J,K)+A(J- 1,K)                                            TWOD 287
      A(J,K)=A(J,K)/2.                                                   TWOD 288
34    CONTINUE                                                           TWOD 289
35    CONTINUE                                                           TWOD 290
      RK=YU- AMPS                                                        TWOD 291
      IF (NSCAN.NE.1) GO TO 36                                           TWOD 292
      A(3,K)=A(2,K)                                                      TWOD 293
      A(2,K)=(A(1,K)+A(3,K))/2.                                          TWOD 294
      A(4,K)=A(2,K)                                                      TWOD 295
      NS2=4                                                              TWOD 296
36    CONTINUE                                                           TWOD 297
      RADL=R- FLOAT(MPS*(K- 1))                                          TWOD 298
      RADU=RADL- FLOAT(MPS)                                              TWOD 299
      IF (RADU.LT.0) RADU=0.                                             TWOD 300
      WRITE (9,73)(RADL,RADU,(A(J,K),J=1,NS2))                           TWOD 301
      CALL SPLF (NS2,K)                                                  TWOD 302
      IF (NSCAN.EQ.1) NS2=2                                              TWOD 303
37    CONTINUE                                                           TWOD 304
      IRF=2                                                              TWOD 305
      IF (IRF.LT.1) GO TO 39                                             TWOD 306
      WRITE (9,82)                                                       TWOD 307
      WRITE (9,83)                                                       TWOD 308
      DO 38 K=1,NK                                                       TWOD 309
      RADL=R- FLOAT(MPS*(K- 1))                                          TWOD 310
      RADU=RADL- FLOAT(MPS)                                              TWOD 311
      IF (RADU.LT.0) RADU=0.                                             TWOD 312
      NF=NOPTF(K)                                                        TWOD 313
      WRITE (9,73)(RADL,RADU,(A(J,K),J=1,NF))                            TWOD 314
38    CONTINUE                                                           TWOD 315
39    CONTINUE                                                           TWOD 316
      IF (NS  AN.EQ.1) NS2=4                                             TWOD 317
      IF (NS2.EQ.4) NSCAN=2                                              TWOD 318
      IF (IRF.LT.1) GO TO 54                                             TWOD 319
      DO 43 J=1,NS2                                                      TWOD 320
      DO 42 K=1,NK                                                       TWOD 321
      CALL SPL (ANG(J,1),K,RAND(K))                                      TWOD 322
      IF (RAND(K).GT.0.) GO TO 41                                        TWOD 323
      RAND(K)=0.                                                         TWOD 324
      IF (K.LE.1) GO TO 40                                               TWOD 325
      CALL SPL (ANG(J,1),K- 1,APT)                                       TWOD 326
      IF (APT.LT.0.) APT=0.                                              TWOD 327
      RAND(K)=RAND(K)+APT/3.                                             TWOD 328
40    IF (J.LE.1) GO TO 41                                               TWOD 329
      CALL SPL (ANG(J- 1,1),K,APT)                                       TWOD 330
      IF (APT.LT.0.) APT=0.                                              TWOD 331
      RAND(K)=RAND(K)+APT/3.                                             TWOD 332
41    CONTINUE                                                           TWOD 333
42    CONTINUE                                                           TWOD 334
      CALL SPLFR (NK,RAND(1),SCAN(1,1),J)                                TWOD 335
43    CONTINUE                                                           TWOD 336
      DO 45 J=1,NS2                                                      TWOD 337
      DO 44 K=1,NK                                                       TWOD 338
      A(J,K)=SCAN(J,K)                                                   TWOD 339
C     RAND(K)=R- FLOAT(MPS*(K- 1)- 0.5*FLOAT(MPS)                        TWOD 340
C     RAND(K+110)=SCAN(J,K)                                              TWOD 341
44    CONTINUE                                                           TWOD 342
C     CALL SPLOT(1,NK,RAND(1),RAND(111),1,1)                             TWOD 343
45    CONTINUE                                                           TWOD 344
      WRITE (9,82)                                                       TWOD 345
      WRITE (9,84)                                                       TWOD 346
      DO 46 K=1,NK                                                       TWOD 347
      RADL=R- FLOAT(MPS*(K- 1))                                          TWOD 348
      RADU=RADL- FLOAT(MPS)                                              TWOD 349
      IF (RADU.LT.0) RADU=0.                                             TWOD 350
```

```
      WRITE (9,73)(RADL,RADU,(A(J,K),J=1,NS2))
      CALL SPLF (NS2,K)
46    CONTINUE
      WRITE (9,82)
      WRITE (9,85)
      DO 53 K=1,NK
      RADL=R-FLOAT(MPS*(K-1))
      RADU=RADL-FLOAT(MPS)
      IF (RADU.LT.0) RADU=0.
      NF=NOPTF(K)
      SUM1=0.0
      SUM2=0.0
      ICHECK=0
      DO 47 IJ=1,NF
      IF (A(IJ,K).LE.0.0) ICHECK=1
47    CONTINUE
      IF (ICHECK.EQ.0) GO TO 52
      DO 48 JJ=1,NF
      SUM1=SUM1+A(JJ,K)/NF
48    CONTINUE
      DO 49 KJ=1,NF
      A(KJ,K)=SUM1
49    CONTINUE
      IF (K.EQ.NK) GO TO 52
      DO 50 LJ=1,NF
      SUM2=SUM2+A(LJ,K+1)/NF
50    CONTINUE
      SUM2=SUM1/2.0+SUM2/2.0
      DO 51 KJ=1,NF
      A(KJ,K+1)=SUM2
51    CONTINUE
52    CONTINUE
      WRITE (9,73)(RADL,RADU,(A(J,K),J=1,NF))
53    CONTINUE
      WRITE (9,82)
54    CONTINUE
      M=64
      DEL=R/FLOAT(M/2)
      DO 61 J=1,M
      X=-R+(J-1)*DEL
      DO 60 K=1,M
      Y=R-(K-1)*DEL
      RC=SQRT(X**2+Y**2)
      JK=(J-1)*M+K
      IF (RC.LT.R) GO TO 55
      RAND(JK)=0.
      GO TO 60
55    CONTINUE
      KR=(R-RC)/AMPS+1.
      IF (KR.GT.NK) KR=NK
      IF (X.NE.0.) GO TO 56
      IF (Y.GT.0.) THTA=90.
      IF (Y.LE.0.) THTA=270.
      GO TO 57
56    CONTINUE
      THTA=57.2958*ATAN(Y/X)
      IF (X.LT.0.) THTA=180.+THTA
      IF (X.GT.0..AND.Y.LT.0.) THTA=360.+THTA
57    CONTINUE
      RAND(JK)=9.
      CALL SPL (THTA,KR,RAND(JK))
      IF (RAND(JK).GT.0.) GO TO 59
      RAND(JK)=0.
      IF (K.LE.1) GO TO 58
      JK1=(J-1)*M+K-1
      RAND(JK)=RAND(JK)+RAND(JK1)/3.
58    IF (J.LE.1) GO TO 59
      JK1=(J-2)*M+K
      RAND(JK)=RAND(JK)+RAND(JK1)/3.
59    CONTINUE
      ILOG=0
```

```
      IF (ILOG.LT.1) GO TO 60
      RAND(JK)=10.*RAND(JK)
      IF (RAND(JK).LE.1) RAND(JK)=0.
      IF (RAND(JK).GT.1) RAND(JK)=ALOG10(RAND(JK))
60    CONTINUE
61    CONTINUE
      N=M/16
      IF (N.LT.1) N=1
      WRITE (9,86)(J,J=1,M,N)
      MN=M*N
      DO 62 K=1,M,N
      KK=K+(M-1)*M
      WRITE (9,87)K,(RAND(J),J=K,KK,MN)
      WRITE (9,88)
62    CONTINUE
      DELX=DEL
      DELY=DEL
      XS=0.
      YS=0.
      CALL W2P
      CALL TWOP
      GO TO 5
C
C
63    FORMAT (8A10)
64    FORMAT (1H1)
65    FORMAT (12I6)
66    FORMAT (*0SCANPOINTS*)
67    FORMAT (6F12.2)
68    FORMAT (*0GAMMAADSORPTIONCOEFFICIENTIS*,1X,F6.5,*/MILL*/*0RADIUSOF
     1DISCBEINGSCANNEDIS*,F5.0,*MILLS*/*0SCANSTEPINMILLSIS*,F4.0/*0WIDTH
     2OFCOLLIMATORINMILLSIS*,F4.0)
69    FORMAT (*0INTENSITYVERSUSANGLEANDRADIUS*)
70    FORMAT (*0ANGLE*,9X,17F7.1)
71    FORMAT (*0RADIUS*,40X,*INTENSITIES*)
72    FORMAT (*0(MILLS)*)
73    FORMAT (1H0,F5.1,1X,*T0*,1X,F5.1,3X,17F7.2)
74    FORMAT (*0RADIUSOFTHESCANREGIONIS*,F6.1,1X,*MILLS.*)
75    FORMAT (*0NUMBEROFSCANSIS*,I3/*0NUMBEROFPOINTSPERSCANIS*,I4,/*0NUM
     1BEROFDEGREESBETWEENSCANSIS*I3)
76    FORMAT (*0KISEQUALTO1,YETINKNOWNRWGIONLOWERHALF*)
77    FORMAT (*0KISEQUALTO1,YETINKNOWNRWGIONUPPERHALF*)
78    FORMAT (3I6,3F12.2)
79    FORMAT (3I6,3F12.2)
80    FORMAT (*0NUMBEROFMONTECARLOPOINTSUSED(NP)IS*,F6.0/10X,*(SQRT(NP)/
     1NP)X100IS*,F5.2)
81    FORMAT (1X,16F8.0)
82    FORMAT (1H0)
83    FORMAT (*INTENSITIESAFTERCIRCULARSPLINESMOOTHING*)
84    FORMAT (*INTENSITIESAFTERRADIALSPLINESMOOTHING*)
85    FORMAT (*INTENSITIESAFTERRE-SMOOTHINGCIRCULARLY*)
86    FORMAT (*1CALCULATEDSQUAREMATRIX*/2X,16I8)
87    FORMAT (1X,I2,F7.2,15(1X,F7.2))
88    FORMAT (//)
      END


C       ***************************************************************


      SUBROUTINE SPL (Y,K,A)
C     Y IS ANGLE, K IS BAND OR REGION, A IS VALUE AT THAT ANGLE IN THAT
      DIMENSION AA(10,110), W(10,110), ANG(10,110), M(110)
      COMMON M,ANG,AA,W
      N=M(K)
      IF (Y.LT.0.001) Y=.001
      CALL SEARCH (Y,ANG(1,K),N,I,MFLAG)
      IF (I.EQ.N) I=N-1
      IF (MFLAG.LT.2) GO TO 1
```

```
      WRITE (9,2)
      WRITE (9,3)A
      WRITE (9,4)I,N,K
      WRITE (9,3)Y
      WRITE (9,4)(M(J),J=1,K)
      WRITE (9,5)(ANG(J,K),J=1,N)
      WRITE (9,5)AA(I,K),AA(I+1,K),W(I,K),W(I+1,K)
1     CONTINUE
      FLK=ANG(I+1,K)- ANG(I,K)
      A=(W(I,K)*(ANG(I+1,K)- Y)**3+W(I+1,K)*(Y- ANG(I,K))**3)/(6.*FLK)+(AA
     1(I+1,K)/FLK- W(I+1,K)*FLK/6.)*(Y- ANG(I,K))+(AA(I,K)/FLK- FLK*W(I,K)/
     26.)*(ANG(I+1,K)- Y)
      RETURN
C
2     FORMAT ('0ZZZZZZZZZZSPL')
3     FORMAT (8F10.5)
4     FORMAT (12I10)
5     FORMAT (12F10.3)
      END


C     ****************************************************************


      SUBROUTINE SPLF (NSCAN,K)
      DIMENSION A(10,110), ANG(10,110), W(10,110), IOP(2), B(324), F(32)
     1, N(110), WT(16), THTA(16)
      COMMON N,ANG,A,W,NK
      NP=NSCAN+1
      ANG(NP)=360.
      A(NP,K)=A(1,K)
      W(NP,K)=W(1,K)
      IOP(1)=4
      IOP(2)=4
C     THIS ESTABLISHES PERODIC DATA
      M=FLOAT(NP)*FLOAT(NK- K+1)/FLOAT(NK)
      IF (M.LE.4) M=4
      IF (K.LE.1) GO TO 3
      AVG=0.
      L=K- 1
      NL=N(L)
      APT1=1.
      CALL SPL (ANG(NP- 1,1),L,APT1)
      APT2=2.
      CALL SPL (ANG(1,1),L,APT2)
      APT3=3.
      CALL SPL (ANG(2,1),L,APT3)
      AVG=(APT1+APT2+APT3)/3.
      WT(1)=AVG/(AVG+(A(1,K)- AVG)**2)
      WT(NP)=WT(1)
      NP1=NP- 1
      DO 1 J=2,NP1
      APT1=4.
      CALL SPL (ANG(J- 1,1),L,APT1)
      APT2=5.
      CALL SPL (ANG(J,1),L,APT2)
      APT3=6.
      CALL SPL (ANG(J+1,1),L,APT3)
      AVG=(APT1+APT2+APT3)/3.
      WT(J)=AVG/(AVG+(A(J,K)- AVG)**2)
1     CONTINUE
      DEL=360./FLOAT(M- 2)
      THTAZ=DEL*RANDOM(L)
      MP=M- 1
      THTA(1)=0.
      THTA(M)=360.
      DO 2 J=2,MP
      THTA(J)=THTAZ+FLOAT(J- 2)*DEL
2     CONTINUE
      GO TO 5
```

```
3     DO 4 J=1,NP
      WT(J)=1.
      THTA(J)=ANG(J,1)
4     CONTINUE
5     CONTINUE
      M=2*M
      CALL SMOOTH (NP,M,IOP,ANG,THTA,A(1,K),WT,F,B)
      M=M/2
      N(K)=M
      DO 6 J=1,M
      ANG(J,K)=THTA(J)
      A(J,K)=F(J+M)
      W(J,K)=F(J)
6     CONTINUE
      RETURN
C
      END


C     ***************************************************************


      SUBROUTINE TW
      REWIND 2
      IREC=0
      DUMMY=0.
      IDMY=0
      DO 1 J=1,122
      READ (10,2)DATA
      IREC=IREC+1
      WRITE (2,3)IDMY,IDMY,IREC,DUMMY,DATA,DUMMY
      WRITE (9,3)IDMY,IDMY,IREC,DUMMY,DATA,DUMMY
1     CONTINUE
      REWIND 2
      RETURN
C
C
2     FORMAT (F12.2)
3     FORMAT (3I6,3F12.2)
      END


C     ***************************************************************


      SUBROUTINE SPLFR (NK,RAND,SCAN,L)
      DIMENSION IOP(2), SCAN(10,110), RAND(1), TAB(3)
      IOP(1)=5
      IOP(2)=5
      KN=NK-NK/10
      IF (KN.LT.3) KN=3
      R=FLOAT(NK)/FLOAT(KN)
      DO 1 J=1,NK
      IF (RAND(J).LT.0.) RAND(J)=0.
      RAND(J+110)=SQRT(RAND(J))
      RAND(J+220)=J
1     CONTINUE
      DO 2 J=1,KN
      RAND(J+330)=FLOAT(J)*R
2     CONTINUE
      KN=2*KN
      CALL SMOOTH (NK,KN,IOP,RAND(221),RAND(331),RAND(1),RAND(111),RAND(
     1431),RAND(531))
      KN=KN/2
      DO 4 J=1,NK
      X=J
      CALL SPL1D2 (KN,RAND(331),RAND(KN+431),RAND(431),1,X,TAB)
```

```
      SCAN(L,J)=TAB(1)
      IF (SCAN(L,J).GE.0.) GO TO 3
      SCAN(L,J)=0.
      IF (J.EQ.1) GO TO 3
      SCAN(L,J)=SCAN(L,J-1)/2.
3     CONTINUE
4     CONTINUE
      RETURN
      END

C     ****************************************************************

      SUBROUTINE W2P
      DIMENSION TITLE(8), XM(64), YM(64), Z(64,64)
      COMMON /W/ L,TITLE,XS,DELX,YS,DELY,Z
      REWIND 7
      WRITE (7,3)(TITLE(J),J=1,8)
      WRITE (7,4)L
C     L = SIZE OF SQUARE MATRIX.
      XS=XS/1000.0
      DELX=DELX/1000.0
      YS=YS/1000.0
      DELY=DELY/1000.0
      WRITE (7,5)XS,DELX,YS,DELY
      DO 1 I=1,L
      XM(I)=XS+FLOAT(I-1)*DELX
      YM(I)=YS+FLOAT(I-1)*DELY
1     CONTINUE
      DO 2 J=1,L
      WRITE (7,6)J
      WRITE (7,7)(Z(I,J),I=1,L)
2     CONTINUE
      END FILE 7
      END FILE 7
      END FILE 7
      RETURN
C
C
3     FORMAT (8A10)
4     FORMAT (4I6)
5     FORMAT (5F12.3)
6     FORMAT (*COLUMNMATRIX*,I3)
7     FORMAT (1H 8F16.4)
      END

C     ****************************************************************
```

APPENDIX B

PLOT SUBROUTINES

```
      SUBROUTINE TWOP
C     TWO DIMENSIONAL GAMMA-RAY SCAN PLOT PROGRAM
C       THIS PROGRAM USES RESULTS TAPE FROM SQMAT
      DIMENSION X(4096), Y(4096), ATITLE(14), ZC(32), XPLT(64), YPLT(64)
     1, Z(4096), NOLR(11), NOLL(11), FL(4)
      DATA ASTER/10H*         /
      DATA AASTER/10H**        /
      DATA BLACK/10HCOLOR      /
      DATA (ATITLE(I),I=9,14)/10H          ,10H          ,10HX         ,
     110H          ,10HY          ,10H          /
      READ (10,15)BW
      REWIND 7
      WRITE (9,11)
      NTITLE=80
      NXLBL=12
      NYLBL=12
      FL(1)=.79538
      FL(3)=120.0
      FL(4)=120.0
      FL(2)=1.17797
      NF=1
      NPASS=1
      IADV=1
      ICZ=1
      ICP=ICZ+1
      ICOLOR=ICZ
      READ (10,10)
      WRITE (9,10)
      WRITE (9,13)
1     READ (10,12)(ATITLE(J),J=1,8)
      WRITE (9,12)(ATITLE(J),J=1,8)
      IF (ATITLE(1).EQ.ASTER) GO TO 8
      IF (ATITLE(1).EQ.AASTER) RETURN
      WRITE (9,13)
      CALL DCARE (NZX,NZY,Z,X,Y,ATITLE)
      WRITE (9,14)NZX,NZY
C     NZX IS THE NUMBER OF X POINTS TO PLOT
C     NZY IS THE NUMBER OF Y POINTS TO PLOT
      DO 2 J=1,NZY
      YPLT(J)=Y((J-1)*NZX+1)-Y(1)
2     CONTINUE
      DO 3 J=1,NZX
      XPLT(J)=X(J)-X(1)
3     CONTINUE
      NOPTS=NZX*NZY
      ZMN=Z(1)
      ZMX=Z(1)
      DO 4 J=2,NOPTS
      IF (Z(J).LT.0.) Z(J)=Z(J-1)
      ZMN=AMIN1(ZMN,Z(J))
      ZMX=AMAX1(ZMX,Z(J))
4     CONTINUE
      IF (ICOLOR.EQ.ICZ) ZMXSAV=ZMX
      ICOLOR=ICOLOR+1
      IF (ICOLOR.GT.3) ICOLOR=1
      IF (NPASS.NE.1) GO TO 7
      ICOLOR=1
      CALL EMPTY
      CALL ADV (2)
      ICOLP=3
      CALL COLOR (ICOLP)
      IF (BW.NE.BLACK) GO TO 5
      CALL EXL
      CALL SWEEP (1,1,1022,1022)
```

```
      CALL EXH
 5    CONTINUE
      CALL PLS3D (Z,XPLT,YPLT,NZX,NZY,FL,ATITLE,ICOLOR)
      CALL EMPTY
      CALL ADV (2)
      CALL COLOR (ICOLP)
      IF (BW.NE.BLACK) GO TO 6
      CALL EXL
      CALL SWEEP (120,50,980,910)
      CALL EXH
 6    CONTINUE
 7    CONTINUE
      NZ=20
      NC=NZ*IFIX(ZMX/ZMXSAV)
      IF (NC.LE.4) NC=5
      IF (ICOLOR.EQ.ICP.AND.NPASS.NE.1) CALL ADV (2)
      CALL CDEN (XPLT,YPLT,Z,NZX,NZY,NC,ZMN,ZMX,ATITLE,NTITLE,12HX COORD
     1INATE,NXLBL,12HY COORDINATE,NYLBL,ICOLOR,IADV)
C     ICOLOR CONTROLLS COLOR  0=WHITE  1=RED  2=GREEN  3=BLUE
C     IADV=0 MEANS DO NOT ADVANCE FILM AFTER DENSITY PLOT
      GO TO 1
 8    CONTINUE
      IF (NPASS.NE.1) GO TO 9
      ICOLOR=ICZ
      NPASS=NPASS+1
      REWIND 7
      IADV=0
      GO TO 1
 9    CONTINUE
      CALL EXIT
C
C
C
 10   FORMAT (72H
     1                    )
 11   FORMAT (1H1)
 12   FORMAT (8A10)
 13   FORMAT (1H0)
 14   FORMAT (30H0NUMBER OF X POINTS PLOTTED IS,I4,32H.  NUMBER OF Y POI
     1NTS PLOTTED IS,I4,1H.)
 15   FORMAT (A10)
      END


C         **************************************************************


      SUBROUTINE DCARE (NZX,NZY,ZZ,XX,YY,ATITLE)
      DIMENSION TITLE(8), ZZ(1), XX(1), YY(1), ATITLE(1)
      DATA BLANK/10H          /
      READ (7,14)TITLE
      IF (ENDFILE 7) 11,1
 1    WRITE (9,14)TITLE
      IF (ATITLE(1).NE.BLANK) GO TO 3
      DO 2 J=1,8
      ATITLE(J)=TITLE(J)
 2    CONTINUE
 3    CONTINUE
      READ (7,15)L
      IF (ENDFILE 7) 11,4
 4    CONTINUE
      NZX=L
      NZY=L
      READ (7,16)XS,DELX,YS,DELY
      IF (ENDFILE 7) 11,5
 5    CONTINUE
      WRITE (9,16)XS,DELX,YS,DELY
      DO 10 J=1,NZY
      READ (7,17)ICOL
```

```
      IF (ENDFILE 7) 11,6
6     CONTINUE
      I=(J-1)*NZX+1
      IP=J*NZX
      READ (7,18)(ZZ(K),K=I,IP)
      IF (ENDFILE 7) 11,7
7     CONTINUE
      DO 8 K=I,IP
      IF (ZZ(K).LT.0.) ZZ(K)=0.
8     CONTINUE
      DO 9 K=1,NZX
      I=(J-1)*NZX+K
      XX(I)=XS+FLOAT(K-1)*DELX
      YY(I)=YS+FLOAT(J-1)*DELY
9     CONTINUE
10    CONTINUE
      RETURN
11    READ (7,14)TITLE
      IF (ENDFILE 7) 13,12
12    CONTINUE
      WRITE (9,14)TITLE
      GO TO 1
13    WRITE (9.19)
      CALL EXIT
C     51 CONTINUE
      CALL EXIT
C
C
C
14    FORMAT (8A10)
15    FORMAT (4I6)
16    FORMAT (4F12.3)
17    FORMAT (15X,I3)
18    FORMAT (1X,8F16.4)
19    FORMAT (6H TWOEF)
      END


C     ****************************************************************


      SUBROUTINE CDEN (X,Y,Z,NOX,NOY,NC,ZMN,ZMX,ITITLE,NTITLE,XLABEL,NXL
     1BL,YLABEL,NYLBL,ICOLOR,IADV)
      DIMENSION X(1), Y(1), Z(1,1)
      DIMENSION IX(64), IY(64)
      DIMENSION XX(900), YY(900)
      CALL COLOR (0)
      DMPX=NOX
      DMPY=NOY
      XMIN=X(1)
      XMAX=X(1)
      DO 1 I=1,NOX
      IF (X(I).LT.XMIN) XMIN=X(I)
      IF (X(I).GT.XMAX) XMAX=X(I)
1     CONTINUE

      YMIN=Y(1)
      YMAX=Y(1)
      DO 2 I=1,NOY
      IF (Y(I).LT.YMIN) YMIN=Y(I)
      IF (Y(I).GT.YMAX) YMAX=Y(I)
2     CONTINUE
      XMX=XMAX
      XMN=XMIN
      YMX=YMAX
      YMN=YMIN
      CALL ASCL (5,XMN,XMX,MAJX,MINX,KKX)
      CALL ASCL (5,YMN,YMX,MAJY,MINY,KKY)
      FACT=860./AMAX1(DMPX,DMPY)
      IXR=120.+IFIX(FACT*DMPX)
```

```
      IXL=120.
      IYB=50.+IFIX(FACT*DMPY)
      IYT=50.
      CALL DGA (IXL,IXR,IYT,IYB,XMN,XMX,YMX,YMN)
      IF (IADV.EQ.0.AND.ICOLOR.NE.2) GO TO 3
      CALL FRAME (IXL,IXR,IYT,IYB)
      CALL SBLIN (MAJX,KKX)
      CALL SLLIN (MAJY,KKY)
      INCX=120+(IXR-120-NXLBL*18)/2
      CALL DLCH (INCX,IYB+90,NXLBL,XLABEL,2)
      INCY=IYB-(IYB-50-NYLBL*18)/2
      CALL DLCV (10,INCY,NYLBL,YLABEL,2)
      INCZ=120+(IXR-120-NTITLE*8)/2
3     CONTINUE
      ITYP=60+(ICOLOR-1)*29
      IF (IADV.NE.0) ITYP=60
      CALL DLCH (170,ITYP,NTITLE,ITITLE,2)
      CALL COLOR (ICOLOR)
      A=FLOAT(NC)
      B=A/2.
      NTPTS=0
      DO 5 K=1,5
      DO 4 J=1,5
      NTPTS=NTPTS+1
      JX=130+J*6+IFIX((1.-2.*RANDOM(A))*3.)
      JY=ITYP+K*4+IFIX((1.-2.*RANDOM(B))*2.)-2
      CALL PLT (JX,JY,42)
4     CONTINUE
5     CONTINUE
      DELX=X(2)-X(1)
      DELY=Y(2)-Y(1)
      NPTS=NOX*NOY
      DO 10 J=1,NPTS
      FN=Z(J)/ZMX*FLOAT(NC)
      N=IFIX(FN)
      IF (N.EQ.0) GO TO 10
      NTPTS=0
      JY=J/NOX+1
      JX=J-(JY-1)*NOX
      IF (JX.EQ.0) JY=JY-1
      IF (JX.EQ.0) JX=NOX
6     FNS=SQRT(DELY/DELX*FN)
      NS=IFIX(FNS)
      IF (NS.EQ.0) NS=1
      NR=N/NS
      IF (NR.EQ.0) NR=1
      DDELX=DELX/FLOAT(NR)
      DDELY=DELY/FLOAT(NS)
      XC=X(JX)-DELX/2.-DDELX/2.
      YC=Y(JY)-DELY/2.-DDELY/2.
      DO 8 K=1,NS
      YC=YC+DDELY
      DO 7 L=1,NR
      NTPTS=NTPTS+1
      A=FLOAT(K*L)
      B=FLOAT(X*L+1)
      XX(NTPTS)=XC+FLOAT(L)*DDELX+(1.-2.*RANDOM(A))*DDELX/2.
      YY(NTPTS)=YC+(1.-2.*RANDOM(B))*DDELY/2.
7     CONTINUE
8     CONTINUE
      N=N-NS*NR
      FN=FLOAT(N)
      IF (N.GT.0) GO TO 6
      DO 9 I=1,NTPTS
      YY(I)=YY(I)+0.030
      XX(I)=XX(I)+0.030
9     CONTINUE
      CALL PLOT (NTPTS,XX,1,YY,1,42,0)
10    CONTINUE
      IF (IADV.NE.0) CALL ADV (1)
      RETURN
C
```

```
      END

C     ************************************************************

      SUBROUTINE SLLIN (NNY,NK)
      COMMON /CJE07/ IXL,IXR,IYT,IYB,XL,XR,YT,YB
      DIMENSION FMT(14), OUT(2)
      DATA (FMT(K),K=1,14)/6H(F6.0),8H(0PF7.1),6H(F8.2),6H(F9.3),7H(F10.
     14),7H(F11.5),7H(F12.6),8H(1PE7.0),8H(1PE8.1),8H(1PE9.2),9H(1PE10.3
     2),9H(1PE11.4),9H(1PE12.5),9H(1PE13.6)/
      IF (NK.GT.6) GO TO 1
      K=MIN0(6,MAX0(0,NK))+1
      NC=K+1
      GO TO 2
    1 K=MIN0(16,MAX0(10,NK))-2
      NC=K-1
    2 A=FMT(K)
      ENCODE (20,A,OUT)YB
      IXT=IXL-18*NC-9
      IYK=IYB-11
      CALL DLCH (IXT,IYK,NC,OUT,2)
      CALL TSP (IXL,IYB,1,1H+)
      IF (NNY.LE.0) RETURN
      NY=MIN0(128,NNY)
      IYC=IYB
      DY=(YT-YB)/NY
      DDY=FLOAT(IYT-IYB)/NY
      DO 3 I=1,NY
      YC=YB+I*DY
      IYC=IYB+I*DDY
      IYD=IYC-6
      ENCODE (20,A,OUT)YC
      CALL DLCH (IXT,IYD,NC,OUT,2)
    3 CALL TSP (IXL,IYC,1,1H+)
      RETURN
      END

C     ************************************************************

      SUBROUTINE SBLIN (NNX,NK)
      COMMON /CJE07/ IXL,IXR,IYT,IYB,XL,XR,YT,YB
      DIMENSION FMT(12), OUT(2)
      DATA (FMT(K),K=1,12)/2H(F,1H ,1H.,1H ,1H),8H(1PE7.0),8H(1PE8.1),8H
     1(1PE9.2),9H(1PE10.3),9H(1PE11.4),9H(1PE12.5),9H(1PE13.6)/
      IY=IYB
      IYDEL=22
      GO TO 1
      ENTRY STLIN
      IY=IYT
      IYDEL=-22
    1 IF (NK.GT.9) GO TO 2
      NC=MAX0(INT(ALOG10(AMAX1(ABS(XL),ABS(XR)))+.00001)+1,1)
      IF (MIN0(XL,XR).LT.0) NC=NC+1
      IF (NK.GT.0) NC=NC+1
      NC=NC+NK
      ENCODE (10,5,FMT(2))NC
      ENCODE (10,5,FMT(4))NK
      K=1
      GO TO 3
    2 K=MIN0(16,MAX0(10,NK))-4
      NC=K+1
    3 ENCODE (20,FMT(K),OUT)XL
      CALL TSP (IXL,IY,1,1H+)
      IXTT=IXL-9*NC+9
```

```
      IYC=IY+IYDEL
      CALL DLCH (IXTT,IYC,NC,OUT,2)
      IF (NNX.LE.0) RETURN
      NX=MIN0(NNX,128)
      IXC=IXL
      DDX=FLOAT(IXR-IXL)/NX
      DX=(XR-XL)/NX
      DO 4 I=1,NX
      XC=XL+I*DX
      IXT=IXTT+I*DDX-9
      IXC=IXL+I*DDX
      ENCODE (20,FMT(K),OUT)XC
      CALL TSP (IXC,IY,1,1H+)
4     CALL DLCH (IXT,IYC,NC,OUT,2)
      RETURN
C
C
C
5     FORMAT (I4)
      END


C         **************************************************************


      SUBROUTINE PLS3D (F,X,Y,IM,JM,FL,WR,ICOLOR)
      DIMENSION F(1), X(1), Y(1), AA(64), AB(64), RA(64), RB(64), ER(64)
     1, WR(1), FL(1), T(4), YMX(901), XMX(901), YMZ(901), OUT(2)
      CALL COLOR (0)
      YT=SINF(FL(1))*FL(3)
      XT=COSF(FL(1))*FL(3)
      YP=SINF(FL(2))*FL(4)
      XP=COSF(FL(2))*FL(4)
      YTB=YT*X(IM)
      XTB=XT*X(IM)
      YPB=YP*Y(JM)
      XPB=XP*Y(JM)
      XA=XTB+XPB
      EA=0.
      EB=1000.
      NX=901
      DX=XA/900
      DO 1 J=1,NX
      YMZ(J)=0.
      XMX(J)=FLOAT(J-1)*DX
1     CONTINUE
      DO 2 I=1,IM
      L=I
      DO 2 J=1,JM
      E=F(L)-X(I)*YT-Y(J)*YP
      EA=MAX1F(EA,E)
      EB=MIN1F(EB,E)
2     L=L+IM
      YC=YTB+YPB
      IF (EB) 3,5,5
3     DIF=YC+EB
      IF (DIF) 4,5,5
4     YB=-DIF
      GO TO 6
5     YB=0.
6     YA=YC+YB+EA
      T(1)=0.
      T(2)=XA
      T(3)=0.
      T(4)=YA
      CALL DGA (63,963,0,900,0.,XA,YA,0.)
      YD=YB+YTB
      IA=IM+1
      NSCALE=10
```

```
      IS=IM/2
      DO 11 L=1,IM
      I=IA-L
      NSCALE=NSCALE+1
      AB(I)=XPB+XTB-XT*X(L)
      RB(I)=YD-YT*X(L)
      IF (L.NE.IS) GO TO 7
      SAB=AB(I)
      SRB=RB(I)
7     CONTINUE
      IF (NSCALE.LT.10) GO TO 8
      CALL CONVRT (AB(I),IAB,0,XA,63,963)
      CALL CONVRT (RB(I),IRB,0,YA,900,0)
      IAB=IAB+2
      IRB=IRB+2
      CALL TSP (IAB,IRB,1,1H*)
      IRB=IRB+9
      ENCODE (20,31,OUT)X(L)
      CALL DLCH (IAB,IRB,3,OUT,2)
      NSCALE=0
8     CONTINUE
      NR=AB(I)/DX+1.
      IF (L.EQ.1) NS=NR
      IF (L.EQ.1) YMZ(NR)=RB(I)
      IF (NR.EQ.NS) GO TO 10
      NMIN=AMIN0(NS,NR)
      NMAX=AMAX0(NS,NR)
      YSAVE=YMZ(NS)
      DO 9 K=NMIN,NMAX
      YMZ(K)=YSAVE+FLOAT(K-NS)/FLOAT(NR-NS)*(RB(I)-YSAVE)
9     CONTINUE
      NS=NR
10    CONTINUE
11    CONTINUE
      CALL PLOT (IM,AB,1,RB,1,32,1)
      YE=YB+YPB
      NSCALE=10
      IS=JM/2
      DO 16 J=1,JM
      NSCALE=NSCALE+1
      AA(J)=XP*Y(J)
      RA(J)=YE-YP*Y(J)
      IF (J.NE.IS) GO TO 12
      SAA=AA(J)
      SRA=RA(J)
12    CONTINUE
      IF (NSCALE.LT.10) GO TO 13
      CALL CONVRT (AA(J),JAA,0,XA,63,963)
      CALL CONVRT (RA(J),JRA,0,YA,900,0)
      JRA=JRA+2
      JAA=JAA-2
      CALL TSP (JAA,JRA,1,1H*)
      JAA=JAA-4*18
      IF (JAA.LT.0) JAA=0
      ENCODE (20,31,OUT)Y(J)
      CALL DLCH (JAA,JRA,3,OUT,2)
      NSCALE=0
13    CONTINUE
      NR=AA(J)/DX+1.
      IF (J.EQ.1) NS=NR
      IF (J.EQ.1) YMZ(NR)=RA(J)
      IF (NR.EQ.NS) GO TO 15
      NMIN=AMIN0(NS,NR)
      NMAX=AMAX0(NS,NR)
      YSAVE=YMZ(NS)
      DO 14 K=NMIN,NMAX
      YMZ(K)=YSAVE+FLOAT(K-NS)/FLOAT(NR-NS)*(RA(J)-YSAVE)
14    CONTINUE
      NS=NR
15    CONTINUE
16    CONTINUE
```

```
      CALL PLOT (JM,AA,1,RA,1,32,1)
      ZH=.05*EA
      YF=YC+YB
      CALL COLOR (ICOLOR)
      DO 17 J=1,NX
      YMX(J)=YMZ(J)
17    CONTINUE
      DO 23 IJ=1,IM
      I=IM-IJ+1
      N=0
      DO 22 JJ=1,JM
      J=JM-JJ+1
      N=N+1
      AA(N)=XTB-X(I)*XT+Y(J)*XP
      L=(J-1)*IM+I
      RA(N)=YF-X(I)*YT-Y(J)*YP+F(L)
      NR=AA(N)/DX+1.
      IF (RA(N).GE.YMX(NR)) GO TO 18
      N=N-1
      IF (N.LE.1) N=0
      IF (N.EQ.0) GO TO 22
      CALL PLOT (N,AA,1,RA,1,42,1)
      N=0
      GO TO 22
18    CONTINUE
      IF (JJ.NE.1) GO TO 19
      NS=NR
      YMX(NS)=RA(N)
      GO TO 21
19    CONTINUE
      IF (N.LE.1) GO TO 21
      NMIN=AMIN0(NS,NR)
      NMAX=AMAX0(NS,NR)
      YSAVE=YMX(NS)
      DO 20 K=NMIN,NMAX
      YMX(K)=YSAVE+FLOAT(K-NS)/FLOAT(NR-NS)*(RA(N)-YSAVE)
20    CONTINUE
21    CONTINUE
      NS=NR
22    CONTINUE
      IF (N.LE.1) GO TO 23
      CALL PLOT (N,AA,1,RA,1,42,1)
23    CONTINUE
      DO 24 J=1,NX
      YMX(J)=YMZ(J)
24    CONTINUE
      DO 30 JJ=1,JM
      J=JM-JJ+1
      N=0
      DO 29 IJ=1,IM
      I=IM-IJ+1
      N=N+1
      AA(N)=XTB-X(I)*XT+Y(J)*XP
      L=(J-1)*IM+I
      RA(N)=YF-X(I)*YT-Y(J)*YP+F(L)
      NR=AA(N)/DX+1.
      IF (RA(N).GE.YMX(NR)) GO TO 25
      N=N-1
      IF (N.EQ.1) N=0
      IF (N.EQ.0) GO TO 29
      CALL PLOT (N,AA,1,RA,1,42,1)
      N=0
      GO TO 29
25    CONTINUE
      IF (IJ.NE.1) GO TO 26
      NS=NR
      YMX(NS)=RA(N)
      GO TO 28
26    CONTINUE
      IF (N.LE.1) GO TO 28
      NMIN=AMIN0(NS,NR)
```

```
      NMAX=AMAX0(NS,NR)
      YSAVE=YMX(NS)
      DO 27 K=NMIN,NMAX
      YMX(K)=YSAVE+FLOAT(K- NS)/FLOAT(NR- NS)*(RA(N)- YSAVE)
27    CONTINUE
28    CONTINUE
      NS=NR
29    CONTINUE
      IF (N.LE.1) GO TO 30
      CALL PLOT (N,AA,1,RA,1,42,1)
30    CONTINUE
      CALL CONVRT (SAB,IAB,0,XA,63,963)
      CALL CONVRT (SAA,JAA,0,XA,63,963)
      CALL CONVRT (SRB,IRB,0,YA,900,0)
      CALL CONVRT (SRA,JRA,0,YA,900,0)
      IAB=IAB+2*18+2*30
      JAA=JAA- 3*18- 2*30
      IRB=IRB+36
      JRA=JRA+36
      CALL COLOR (0)
      CALL DLCH (JAA,JRA,20,WR(13),4)
      CALL DLCH (IAB,IRB,20,WR(11),4)
      CALL DLCH (100,960,100,WR,2)
      CALL ADV (1)
      RETURN
C
C
C
31    FORMAT (F3.1)
      END


C     ***************************************************************


      SUBROUTINE SWEEP (IX1,IY1,IX2,IY2)
*
*
*
      DIMENSION WORD(8)
      DATA WORD/8*10H * * * * */,XONE,YONE,XTWO,YTWO,XDIFF,YDIFF/10H
     1    X1,10H         Y1,10H         X2,10H         Y2,10H X1 AND X2,10H
     2 Y1 AND Y2/
*
      IERROR=0
*     CHECK FOR ARGUMENT LESS THAN ZERO
      IF (IX1.GE.0) GO TO 1
      IERROR=IERROR+1
      PRINT 21, XONE
1     IF (IX2.GE.0) GO TO 2
      IERROR=IERROR+1
      PRINT 21, XTWO
2     IF (IY1.GE.0) GO TO 3
      IERROR=IERROR+1
      PRINT 21, YONE
3     IF (IY2.GE.0) GO TO 4
      IERROR=IERROR+1
      PRINT 21, YTWO
*     CHECK FOR ARGUMENT GREATER THAN 1023
4     IF (IX1.LE.1023) GO TO 5
      IERROR=IERROR+1
      PRINT 22, XONE
5     IF (IX2.LE.1023) GO TO 6
      IERROR=IERROR+1
      PRINT 22, XTWO
6     IF (IY1.LE.1023) GO TO 7
      IERROR=IERROR+1
      PRINT 22, YONE
7     IF (IY2.LE.1023) GO TO 8
      IERROR=IERROR+1
```

```
      PRINT 22, YTWO
*     CHECK FOR DIFFERENCE GREATER THAN 63
8     IF (IABS(IX1-IX2).GE.64) GO TO 9
      IERROR=IERROR+1
      PRINT 23, XDIFF
9     IF (IABS(IY1-IY2).GE.64) GO TO 10
      IERROR=IERROR+1
      PRINT 23, YDIFF
*     RETURN TO CALLER IF ANY ERRORS HAVE BEEN DETECTED
10    IF (IERROR.EQ.0) GO TO 11
      PRINT 20
      RETURN
*
11    IF (IY1.GT.IY2) GO TO 13
      DO 12 I=IY1,IY2
12    CALL GXA (IX1,IX2,I)
      GO TO 15
13    DO 14 I=IY2,IY1
14    CALL GXA (IX1,IX2,I)
*
15    IF (IX1.GT.IX2) GO TO 17
      DO 16 I=IX1,IX2
16    CALL GYA (IY1,IY2,I)
      GO TO 19
17    DO 18 I=IX2,IX1
18    CALL GYA (IY1,IY2,I)
*
19    RETURN
C
20    FORMAT (//10X,43H  * ARGUMENT ERROR(S) - EXECUTION DELETED *)
21    FORMAT (10X,A10,10H NEGATIVE )
22    FORMAT (10X,A10,21H GREATER THAN 1023    )
23    FORMAT (10X,19H DIFFERENCE BETWEEN,A10,13H LESS THAN 64)
      END
```

```
C          ****************************************************************
```

```
        IDENT  COLOR
        ENTRY  COLOR
        EXT    BS4020
COLOR   BSSZ   1
        SA1    B1                    PICK UP ARG
        MX6    58
        BX6    -X6*X1                0,1,2,3 INSURED
        SB2    X6
        BX6    -X6*X1                0,1,2,3 INSURED
        SB2    X6
        SA1    WHITE+B2
        RJ     BS4020                INTO BUFFER
        JP     COLOR
WHITE   DATA   40000000000B          WHITE
        DATA   41000000000B          RED
        DATA   42000000000B          GREEN
        DATA   43000000000B          BLUE
        END
```

CM/tb:355(50)