UNITED STATES DEPARTMENT OF THE INTERIOR

GEOLOGICAL SURVEY

MAGPOLY: A modification of a three-dimensional magnetic modelling program

by

Richard H. Godson

Open-File Report 83-345

1983

Table of Contents

Abstract

The three-dimensional magnetic modelling program of Donald Plouff (Plouff, 1975) has been made operational on a Honeywell Multics 68/80 computer. Modifications were made to input and output procedures to make the program compatible with U.S. Geological Survey (USGS) standard geophysical input and output formats.

Introduction

This program calculates the magnetic effect of three-dimensional models that consist of elements that form polygons in plan view (Plouff, 1976). The elements of the models are bounded by two horizontal planes and a series of intersecting vertical planes. The calculations use an exact formula rather than using a numerical integration approach of previous Talwani (1965) programs. Thus, values can be obtained on or inside the model. A least-squares comparison between the calculated anomalies and an observed field can be used to determine the best total magnetization vector (amplitude, declination and inclination) and also the best magnetic susceptibility for the assemblage of polygonal prisms.

Description

Model input to the program consists of x, y, and z coordinates of polygon corners. A maximum of 50 polygons can be specified and the total number of corners for all polygons cannot exceed 900. Parameter input such as plotting variables, magnetic susceptibility and inclination, and so forth are entered from a command file in namelist format. Fieldpoint locations and optionally observed values at these locations are read in from a disk file. This file can be user-formatted random data (see Appendix B) or it can be in USGS geophysical grid format (see Appendix A). If the height of the fieldpoint locations of a grid file are not on a level plane, a second USGS grid-format file containing the elevations of the fieldpoint locations can be read.

Output from the program consists of printed information about the bodies and fieldpoint coordinates and values. Contoured output may be directed to a line printer or terminal, or grid-format files can be created that can be used as direct input to a contouring program.

Program Usage

Parameters and body coordinate data are obtained from an ASCII segment on disk created by the user prior to running the program. This segment is read by Fortran unit 9; attachment and detachment of the segment are accomplished within the program. The program parameters are read from the first part of the segment in namelist format. The segment therefore must start with the characters "&parms" followed by any number of namelist variables explained below and followed with the character "&". After the above character, the body coordinate information explained below is read from the segment.

Fieldpoint information in either a user-specified format or in a USGS geophysics grid format is read from a separate segment created on disk by the user before running the program. The maximum number of fieldpoints allowed is 1,000. This segment is read by Fortran unit 10 and is attached and detached within in the program. With user-formatted input, the height of each fieldpoint location is specified; this is not the case with grid files. In order to allow this flexibility with grid files, a second file that contains the heights of the fieldpoint locations may be read in from disk. This file must have the same header information as the location file, that is, it must be registered. This optional segment is read by Fortran unit 11 and is attached and detached by the program.

Following are explanations of the namelist variables, grouped under usage subheadings.

File information

<u>ifile</u> - the name of the file that contains the fieldpoint information. It must be enclosed in quotes with a maximum length of 50 characters.

<u>ifile2</u> - the name of a grid file that contains the heights of the fieldpoint locations. It must be enclosed in quotes with a maximum length of 50 characters. This file is not necessary if the parameter "height" below is used.

<u>name</u> - a character variable that states the type of data that will be read from "ifile". Default is the characters "gridded", which means that the fieldpoint data will be read as a standard USGS geophysics grid file. Any other characters (maximum of 8) enclosed in quotes will assumed a user-formatted file.

<u>height</u> - the height of the fieldpoint grid above the same datum that is used to reference the body heights, for example, sea level. Positive values are upward. See parameter "unix" below for specification units. This parameter is used only when name = "gridded" and ifile2 is not specified. Default is zero.

<u>ifmt</u> - the format to be used when reading a user-formatted file for fieldpoint data. Four variables are read for each fieldpoint location in the following order: x coordinate, y coordinate, observed value, and height. The parameter must be enclosed in quotes, for example, "(4F10.2)". A maximum of 80 characters can be used. Default is "(v)" for list directed input.

Plotting information and scaling

I. iplotr - a number that determines which plotter device is to be used. The number 9, which is the default, directs all plotting to an output file on disk, which is named with the characters "print." as a prefix to the command file name. One of the numbers listed below will direct the plot of the body corners and fieldpoint locations to that device.

0 = Calcomp 7900

1 = Tektronix 4010

2 = Hewlett-Packard 7202

3 = Hewlett-Packard 7203

4 = Tektronix 4014 low resolution

5 = Tektronix 4014 high resolution

6 = General vector output, for example, electrostatic printer/plotter

ibody - a nonzero number will create a plot of body corners and fieldpoint locations. Default is zero.

iobs - a nonzero number and iplotr = 9 will create printer contours for gridded input fieldpoint data. Default is zero.

icalc - a nonzero number and iplotr = 9 will create printer contours of the calculated data. A nonzero number will create a standard USGS geophysical grid file with the characters "calc." as a prefix to the command file name, which can be contoured later by a contouring program. Default is zero. [This parameter is applicable only with gridded input fieldpoint data.]

ires - a nonzero number and iplotr = 9 will create printer contours of the residual data values (observed-calculated). A nonzero number will create a standard USGS geophysics grid file with the characters "res." as a prefix to the command file name, which can be contoured later by a contouring program. Default is zero. [This parameter is applicable only with gridded input fieldpoint data.]

idplot - 40 characters of identification that is printed at the top of printed output. This parameter must be enclosed in quotes and the default is blanks.

dc - the contour interval to be used if printer plots are requested. Default is zero.

xscale - the scaling factor to be used only when plotting body coordinates on a plotter device other than a printer. It is specified in coordinate units per inch, for example, 62500, if inches are the units of measurement (see "unix" below). Default is zero.

unix - the units of measurement in the x and y directions. It is a four character variable which must be enclosed in quotes. Options available are "feet", "mile", "kilf", "kilm" and "metr". Default is "kilm".

uniz - the units of measurement in the z direction. It has the same options as "unix" above. Default is "feet".

naxcol - a number that controls the horizontal width of printer plots. If later printing is desired on a data terminal, then 79 should be used. Default is 130 for plotting on the full width of a line printer.

Magnetization information

earth - amplitude of Earth's total field in gammas. A positive value of "earth" indicates that the exact formula will be used for calculating the true anomalous total field.

$T = \quad (lH+X)^2 + (mH+Y)^2 + (nH+Z)^2 - H$, where H = "earth" and X, Y, and Z are the components of the anomaly along the three orthogonal axes and l, m, and n are the direction cosines of the Earth's field. For the most commonly used option, if "earth" is less than zero (that is, "earth" is the negative of the amplitude of the Earth's field), an approximation to the above formula will be used for determining the anomalous field: $T Z lX + mY + nZ$. Use of this approximation implies that the direction and magnitude of remanent magnetization of the model has a negligible effect on locally changing the apparent direction of the Earth's field. The value of "earth" can be zero for the rare case in which the user wishes to isolate the effect of remanent magnetization yet assuming a negligible external magnetic field: this option must not be used if the magnetic susceptibility within the model is specified because the induced magnetization could not be calculated.

dec - the declination of the Earth's field. It is measured in degrees positive clockwise from the positive y axis as viewed from above. Default is zero.

fincl - the inclination of the Earth's field. It is measured in degrees positive downward from a horizontal plane. Default is zero.

datum - a number to be added to the calculated anomaly to convert to the observed anomaly values. This is nearly always zero, which is the default.

lsqs - a number that determines what type of fieldpoint (observed) values will be read or what type of least-squares comparison will be made between the calculated and observed data.

0 = no fieldpoint (observed) values will be used. It must be kept in mind that it is always necessary to read fieldpoint locations in order to define the positions at which the model values will be calculated.

1 = y component of magnetic anomaly.

2 = x component of magnetic anomaly.

3 = vertical component of magnetic anomaly (some ground surveys).

4 = total horizontal component of magnetic anomaly.

5 = total magnetic anomaly (the usual observed aeromagnetic anomaly).

6 = a best total magnetization vector (amplitude, declination and inclination) will be determined by a least-square comparison between the observed and calculated values.

7 = a best susceptibility, assuming no remanent magnetization, will be determined by a least-squares comparison between the observed and calculated values.

Body information

Following the namelist variables in the command segment is parameter information for each body of the model. A total of 50 bodies is permitted and each body has the following format (see Examples of Program Execution):

Eight parameter values precede the body coordinates. The values are read from one line in v format. The order of the values are as follows. 1) The number of corners of the body. 2) A print switch for body information. A zero means no additional printout. A positive number will print the contribution of this body at each field point. A negative number, used for debugging, will print the values of key parameters for each polygonal side of the body (integrals V_1-V_6). 3) The volume magnetic susceptibility for this body in e.m.u. times 100,000. This can be zero only if the following value is nonzero or "lsqs" = 6 or 7. 4) The remanent or total volume magnetization in e.m.u. times 100,000. This can be zero only if the preceding value is nonzero or "lsqs" =6 or 7. 5) The declination of remanent or total magnetization in degrees, measured positive clockwise from the direction of the y axis. 6) The inclination of remanent or total magnetization in degrees, meaxured positive downward from the horizontal plane. 7) The height of the top of body. The positive direction is upward. 8) The height of the bottom of the body and again the positive direction is upward.

The body coordinate corners follow the above parameter line in x and y pairs. The progression of corners is clockwise as viewed from above. The first pair is not repeated at the end of the list and the total number of pairs must equal the first number of the preceding parameter line. A hole may be removed from a large body by treating the hole as a separate body with the same magnetization contrast as the parent body but with input progressing counterclockwise from corner to corner.

After the program finishes computations using the above parameter and model information, it will try to read another parameter line from the command segment. This optional parameter line will permit calculation of magnetic components using a different set of magnetization parameters for the previous model without recalculating the six time-consuming volume integrals. However, the parameters are the same for all bodies of the model. Any number of these optional parameter lines can be placed in the command segment. If this option is used, a line containing eight zeros must precede the first parameter line. The parameter line consists of six values read in a v format. The order of these values are as follows. 1) The plot switch "icalc" (see above under Namelist variables). A word of caution: if this is a nonzero number and "iplotr" is not 9--the grid file created will supercede the standard calculated file created from the previous model. 2) The amplitude of the Earth's total field in gammas (see "Earth" above under namelist variables). The declination and inclination of the earth's field remain the same as defined before. 3) The volume magnetic susceptibility of the model in e.m.u. times 100,000. This can be zero only if the following value is nonzero. 4) The remanent or total volume magnetization of the model in e.m.u. times 100,000. This can be zero only if the preceding value is nonzero. 5) The declination of magnetization of the model, measured in degrees clockwise from the positive y axis. 6) The inclination of magnetization of the model in degrees, measured positive downward from the horizontal plane.

Program Execution

To run the program type the characters: >online>Reg>Pgms>magpoly. The program will then type the statement: "enter command file name:". The user then types in the name of the command file that he has previously created on disk and program execution continues. If no error messages are generated and the program terminates normally, a message is printed that states that the output is in a file that is named with the characters "print." as a prefix to the command file name. The user then can observe the results of the program by printing this file, which resides in the user's working directory on disk.

If a plot of body corners and fieldpoint locations is requested on any of the offline plotting devices, then the program will be interrupted by the plotting system to ask the user questions.

References

Evenden, G. J., and Wahl, R. P., 1977, Basic plotting system: U.S. Geological Survey unpublished documentation.

Plouff, Donald, 1975, Derivation of formulas and Fortran programs to compute magnetic anomalies of prisms: U.S. Geological Survey Report, 112 p. available from National Technical Information Service No. PB-243-525, U.S. Department of Commerce, Springfield, VA 22161.

Plouff, Donald, 1976, Gravity and magnetic fields of polygonal prisms and application to magnetic terrain corrections: Geophysics, v. 41, no. 4, p. 727-741.

Talwani, Manik, 1965, Computation with help of a digital computer of magnetic anomalies caused by bodies of arbitrary shape: Geophysics, v. 30, no. 5, p. 797-817.

Appendix A

Command Segment Examples

1. The following example will calculate a best susceptibility between the observed and calculated values and a plot of the body corners and contours of the observed, calculated and residual values will be placed in a disk file.

```
&parms unix="mile",dec=17.9,fincl=62.9,earth=-52000,
ibody=1,icalc=1,ires=1,iobs=1,lsqs=7,ifile="magpoly.grid",
height=4500,dc=20,idplot="magpoly test 1",&
36 0 0 0 17.9 62.9 2000 1200
1.14 .46 1.01 .71 .84 1.07 .46 .6 .54 1.04 .05 .62 -.05 1.07 .46 1.56
.05 1.96 .47 1.89
.26 2.16 .52 2.77 .3 3.09 .49 3.92 .83 3.91 .76 3.63 1.3 3.43 1.83 4.39      BODY 1
2.31 4.39 2.42 3.9 2.88 3.57 3.01 3.23 2.84 3.06 3.2 2.7 3.27 2.36
3.1 1.82 3.6 1.52 3.65 1.13 4.03 .77 3.92 .52 3.61 1.06 2.95 1.12
2.31 .68 2.77 .5 2.47 .03 2.17 .46
11 0 0 0 17.9 62.9 2400 2000
1.2 1.21 .54 2.29 .8 2.36 .51 3.34 1.51 3.25 1.95 4.12 2.14 4.06 2.71 3.49  BODY 2
2.61 3.01 3.13 2.54 2.93 1.24
4 0 0 0 17.9 62.9 2400 2000
.4 1.26 .56 1.43 .72 1.34 .6 1.19                                             BODY 3
12 0 0 0 17.9 62.9 2800 2400
2.51 1.48 1.4 1.49 .98 2.04 1. 2.51 1.46 2.97 2. 3.93 2.21 3.86 2.6 3.47     BODY 4
2.43 2.96 3.03 2.54 2.8 1.82 2.58 1.77
13 0 0 0 17.9 62.9 3200 2800
1.52 1.08 1.27 1.94 1.6 2.64 1.46 2.83 1.77 2.9 1.85 3.48 2.12 3.74            BODY 5
2.49 3.41 2.24 2.93 2.87 2.49 2.77 2.07 2.4 1.94 2.32 1.62
13 0 0 0 17.9 62.9 3600 3200
2.12 1.8 1.5 1.86 1.96 2.97 1.92 3.24 2.06 3.62 2.35 3.33 2. 3.01              BODY 6
1.94 2.79 2.18 2.62 2.24 2.7 2.09 2.53 2.7 2.15 2.24 2.11
6 0 0 0 17.9 62.9 3800 3600
1.7 2. 1.9 2.63 2.64 2.5 2.65 2.21 2.15 2.18 2.08 1.84                         BODY 7
4 0 0 0 17.9 62.9 4000 3800
2.04 1.97 1.95 1.97 1.95 2.05 2.06 2.04                                        BODY 8
4 0 0 0 17.9 62.9 4000 3800
1.89 2.19 1.89 2.5 2.55 2.49 2.6 2.31                                          BODY 9
```

Command Segment Examples—Continued

2. This example will perform a forward calculation from the nine model bodies specified below. The fieldpoint locations are read from a user formatted file.

```
&parms unix="mile",dec=17.9,fincl=62.9,earth=-52000,iplotr=1,xscale=.5,
ipodv=1,icalc=1,ires=1,ipos=1,lsqs=5,ifile="magpoly.test1",
name="formatted",height=4500,dc=20,idplot="magpoly test 1",&
36 0 0 0 17.9 62.9 2000 1200
1.14 .46 1.61 .71 .64 1.07 .46 .6 .54 1.04 .05 .62 -.05 1.07 .46 1.56
.05 1.96 .47 1.89
.26 2.16 .52 2.77 .3 3.09 .49 3.92 .83 3.91 .76 3.63 1.3 3.43 1.83 4.39     BODY 1
2.31 4.39 2.42 3.9 2.68 3.57 3.01 3.23 2.84 3.06 3.2 2.7 3.27 2.36
3.1 1.82 3.6 1.52 3.65 1.13 4.03 .77 3.92 .52 3.61 1.06 2.95 1.12
2.31 .68 2.77 .5 2.47 .03 2.17 .46
11 0 0 0 17.9 62.9 2400 2000
1.2 1.21 .54 2.29 .8 2.36 .51 3.34 1.31 3.25 1.95 4.12 2.14 4.06 2.71 3.49  BODY 2
2.61 3.01 3.13 2.54 2.93 1.24
4 0 0 0 17.9 62.9 2400 2000
.4 1.26 .56 1.43 .72 1.34 .6 1.19                                           BODY 3
12 0 0 0 17.9 62.9 2800 2400
2.51 1.48 1.4 1.49 .96 2.04 1. 2.51 1.46 2.97 2. 3.93 2.21 3.86 2.6 3.47    BODY 4
2.43 2.96 3.03 2.54 2.8 1.82 2.58 1.77
13 0 0 0 17.9 62.9 3200 2800
1.52 1.08 1.27 1.94 1.6 2.64 1.46 2.83 1.77 2.9 1.85 3.48 2.12 3.74         BODY 5
2.49 3.41 2.24 2.93 2.87 2.49 2.77 2.07 2.4 1.94 2.32 1.62
13 0 0 0 17.9 62.9 3600 3200
2.12 1.8 1.5 1.86 1.96 2.97 1.92 3.24 2.06 3.62 2.35 3.33 2. 3.01           BODY 6
1.94 2.79 2.18 2.62 2.24 2.7 2.09 2.53 2.7 2.15 2.24 2.11
6 0 0 0 17.9 62.9 3800 3600
1.7 2. 1.9 2.03 2.64 2.5 2.65 2.21 2.15 2.18 2.08 1.84                      BODY 7
4 0 0 0 17.9 62.9 4000 3800
2.04 1.97 1.95 1.97 1.95 2.05 2.06 2.04                                     BODY 8
4 0 0 0 17.9 62.9 4000 3800
1.89 2.19 1.89 2.5 2.55 2.49 2.6 2.31                                       BODY 9
```

Appendix B

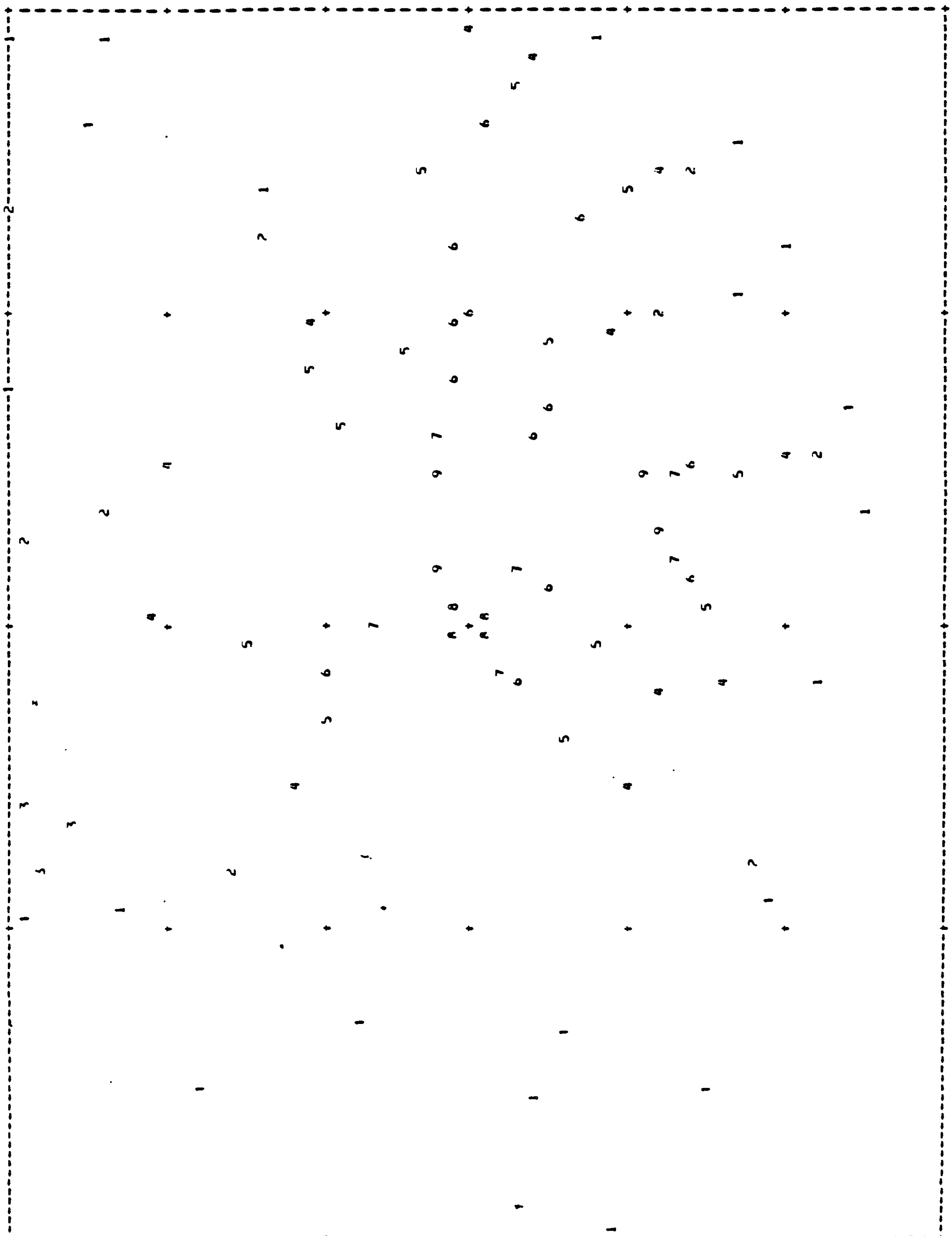
Output Example 1

The following output was produced from command example 1.

```
exact solution for 3d magnetic bodies bounded by horizontal and vertical planes
    magpoly test 1
    expect determination of standard deviation using inputted t-component of magnetic field.
    units of distance measurement are in mile and heights are in feet
    declination=   17.9 and inclination=  62.9 degrees for earths field
        of 52000. gammas.
    magnetic anomaly is assumed to be small compared to earths field.


body  1 between depths   2000.000 and   1200.000 with  36 corners.
        1.140   0.480          1.610    0.710          0.840   1.070        0.460   0.600        0.540   1.040
        0.050   0.620         -0.050    1.070          0.460   1.560        0.050   1.860        0.470   1.890
        0.260   2.160          0.520    2.770          0.300   3.090        0.490   3.920        0.830   3.910
        0.760   3.630          1.300    3.430          1.830   4.390        2.310   4.380        2.420   3.900
        2.880   3.570          3.010    3.230          2.840   3.080        3.200   2.700        3.270   2.360
        3.100   1.820          3.600    1.520          3.650   1.130        4.030   0.770        3.920   0.520
        3.610   1.060          2.950    1.120          2.310   0.680        2.770   0.500        2.470   0.030
        2.170   0.460
body  2 between depths   2400.000 and   2000.000 with  11 corners.
        1.200   1.210          0.540    2.290          0.800   2.360        0.510   3.340        1.310   3.250
        1.950   4.120          2.140,   4.060          2.710   3.490        2.610   3.010        3.130   2.540
        2.930   1.240
body  3 between depths   2400.000 and   2000.000 with   4 corners.
        0.400   1.260          0.560    1.430          0.720   1.340        0.600   1.190
body  4 between depths   2800.000 and   2400.000 with  12 corners.
        2.510   1.480          1.400    1.490          0.980   2.040        1.000   2.510        1.460   2.970
        2.000   3.930          2.210    3.860          2.600   3.470        2.430   2.960        3.030   2.540
        2.800   1.820          2.580    1.770
body  5 between depths   3200.000 and   2800.000 with  13 corners.
        1.520   1.680          1.270    1.940          1.600   2.640        1.460   2.830        1.770   2.900
        1.850   3.480          2.120    3.740          2.490   3.410        2.240   2.930        2.870   2.490
        2.770   2.070          2.400    1.940          2.320   1.620
body  6 between depths   3600.000 and   3200.000 with  13 corners.
        2.120   1.800          1.500    1.860          1.960   2.970        1.920   3.240        2.060   3.620
        2.350   3.330          2.000    3.010          1.940   2.790        2.180   2.620        2.240   2.700
        2.690   2.530          2.700    2.150          2.240   2.110
body  7 between depths   3800.000 and   3600.000 with   6 corners.
        1.700   2.000          1.900    2.630          2.640   2.500        2.650   2.210        2.150   2.180
        2.080   1.840
body  8 between depths   4000.000 and   3800.000 with   4 corners.
        2.040   1.970          1.950    1.970          1.950   2.050        2.060   2.040
body  9 between depths   4000.000 and   3800.000 with   4 corners.
        1.890   2.190          1.890    2.500          2.550   2.490        2.600   2.310


          observed values
      x          y          z          m

    0.5000     0.0000  4500.0000  1830.0000
    1.0000     0.0000  4500.0000  1815.0000
    1.5000     0.0000  4500.0000  1820.0000
    2.0000     0.0000  4500.0000  1825.0000
    2.5000     0.0000  4500.0000  1820.0000
    3.0000     0.0000  4500.0000  1815.0000
    3.5000     0.0000  4500.0000  1815.0000
    0.5000     1.0000  4500.0000  1960.0000
    1.0000     1.0000  4500.0000  2005.0000
    1.5000     1.0000  4500.0000  2025.0000
    2.0000     1.0000  4500.0000  2015.0000
    2.5000     1.0000  4500.0000  1940.0000
```

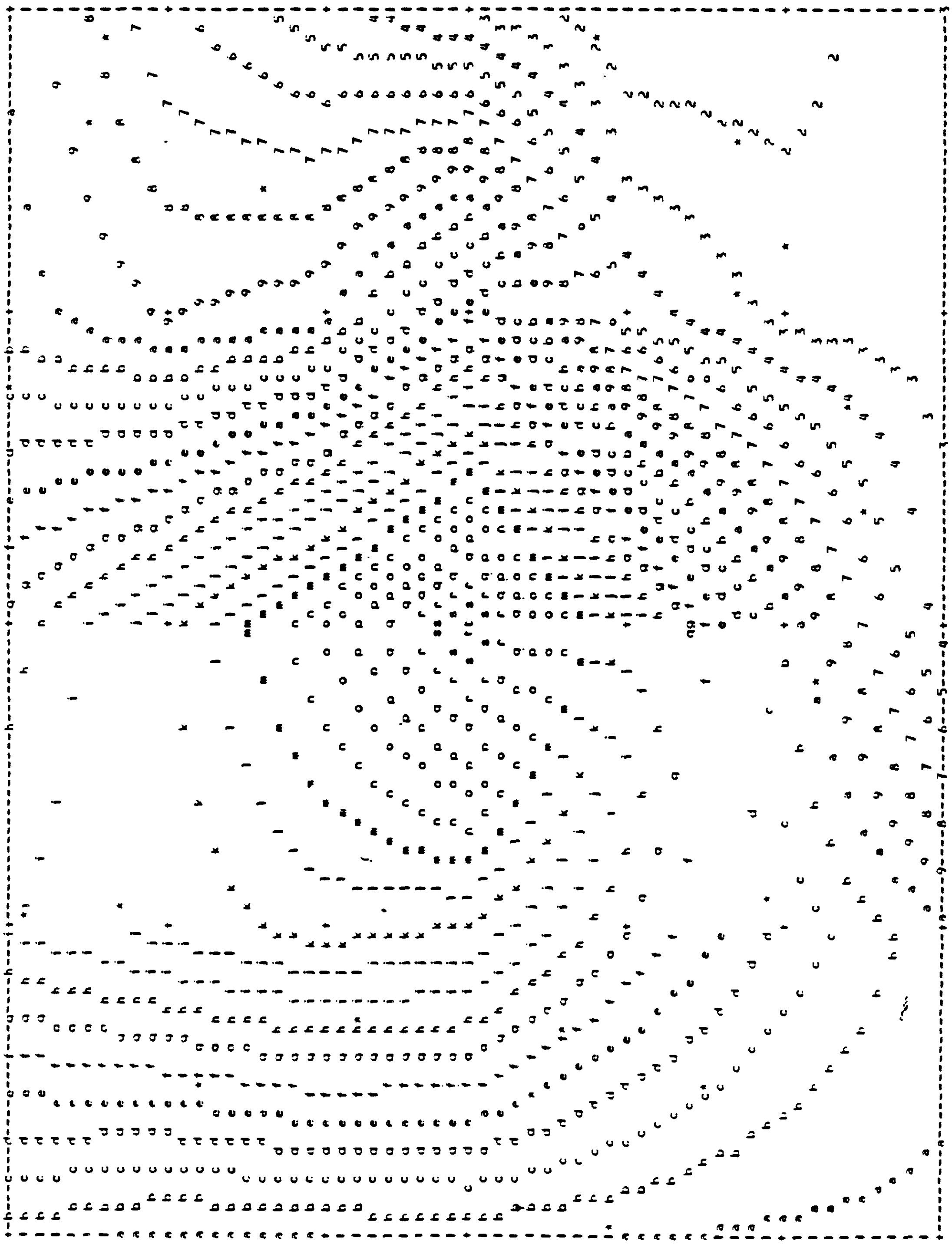
```
    3.0000      1.0000  4500.0000  1875.0000
    3.5000      1.0000  4500.0000  1830.0000
    0.5000      2.0000  4500.0000  1950.0000
    1.0000      2.0000  4500.0000  2015.0000
    1.5000      2.0000  4500.0000  2100.0000
    2.0000      2.0000  4500.0000  2205.0000
    2.5000      2.0000  4500.0000  2010.0000
    3.0000      2.0000  4500.0000  1840.0000
    3.5000      2.0000  4500.0000  1695.0000
    0.5000      3.0000  4500.0000  1830.0000
    1.0000      3.0000  4500.0000  1800.0000
    1.5000      3.0000  4500.0000  1815.0000
    2.0000      3.0000  4500.0000  1920.0000
    2.5000      3.0000  4500.0000  1715.0000
    3.0000      3.0000  4500.0000  1675.0000
    3.5000      3.0000  4500.0000  1670.0000
    0.5000      4.0000  4500.0000  1815.0000
    1.0000      4.0000  4500.0000  1745.0000
    1.5000      4.0000  4500.0000  1710.0000
    2.0000      4.0000  4500.0000  1680.0000
    2.5000      4.0000  4500.0000  1640.0000
    3.0000      4.0000  4500.0000  1647.0000
    3.5000      4.0000  4500.0000  1680.0000

 expect plot of body corners      x=     0.5 to     3.5 y=     0.0 to     4.0
 y increases to the right and x increases downward
```

Output Example 1--Continued



```
expect plot of observed values    of  7 by  5( 35) array. x=    0.5 to     3.5 y=     0.0 to     4.0 contoured at   20.0 gammas

    body point x=     0.460 y=     0.600 is outside plot boundary.
    body point x=     0.050 y=     0.620 is outside plot boundary.
    body point x=    -0.050 y=     1.070 is outside plot boundary.
    body point x=     0.460 y=     1.560 is outside plot boundary.
    body point x=     0.050 y=     1.860 is outside plot boundary.
    body point x=     0.470 y=     1.890 is outside plot boundary.
    body point x=     0.260 y=     2.160 is outside plot boundary.
    body point x=     0.300 y=     3.090 is outside plot boundary.
    body point x=     0.490 y=     3.920 is outside plot boundary.
    body point x=     1.830 y=     4.390 is outside plot boundary.
    body point x=     2.310 y=     4.380 is outside plot boundary.
    body point x=     3.600 y=     1.520 is outside plot boundary.
    body point x=     3.650 y=     1.130 is outside plot boundary.
    body point x=     4.030 y=     0.770 is outside plot boundary.
    body point x=     3.920 y=     0.520 is outside plot boundary.
    body point x=     3.610 y=     1.060 is outside plot boundary.
```

Output Example 1--Continued



symbol value (gammas). y increases to right and x increases downward

symbol	value (gammas)
1	1640.0000
2	1660.0000
3	1680.0000
4	1700.0000
5	1720.0000
6	1740.0000
7	1760.0000
8	1780.0000
9	1800.0000
a	1820.0000
b	1840.0000
c	1860.0000
d	1880.0000
e	1900.0000
f	1920.0000
g	1940.0000
h	1960.0000
i	1980.0000
j	2000.0000
k	2020.0000
l	2040.0000
m	2060.0000
n	2080.0000
o	2100.0000
p	2120.0000
q	2140.0000
r	2160.0000
s	2180.0000
t	2200.0000
+	fieldpoint

* corner of body

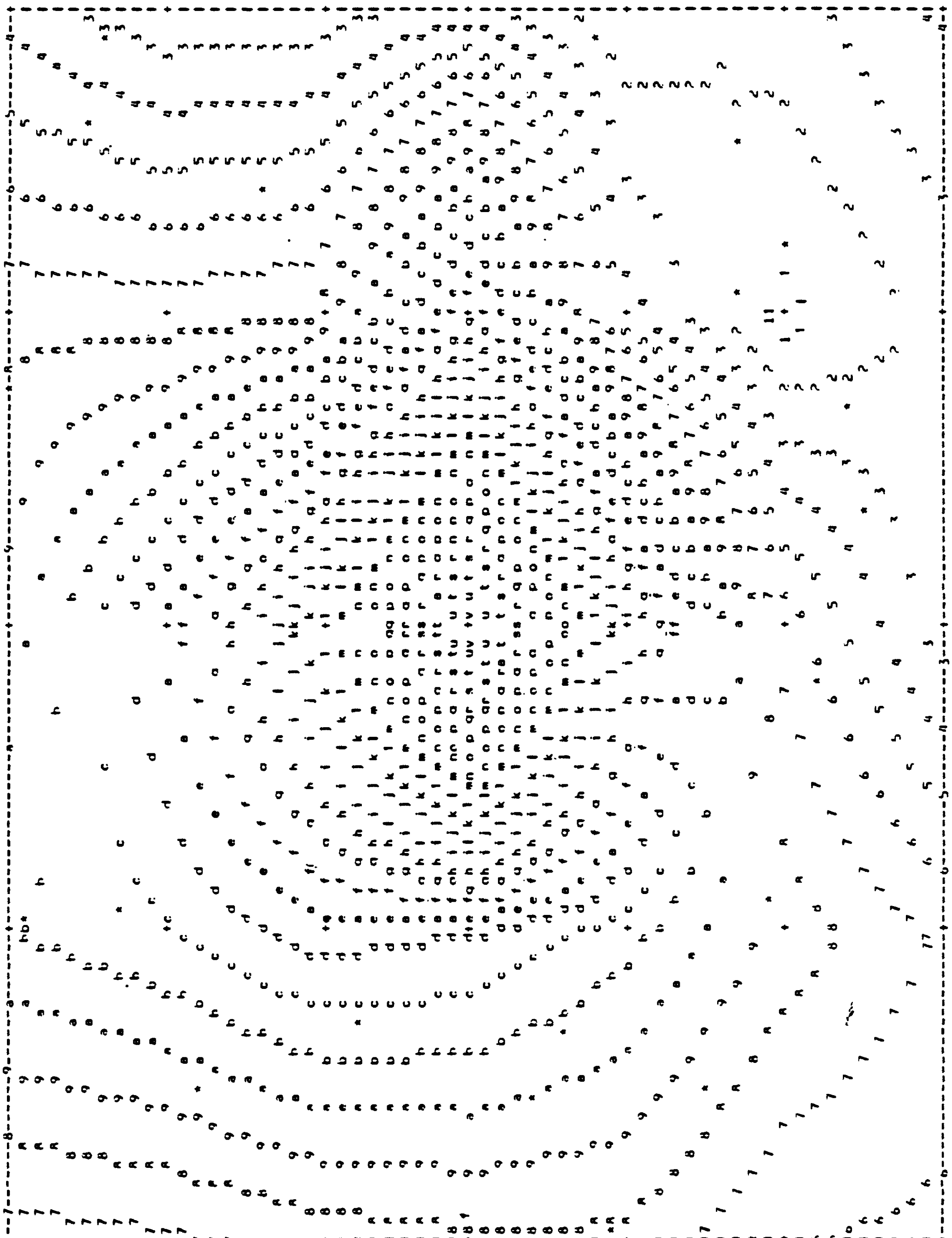
total of 35 fieldpoints.

least-squares susceptibility is 329.4e-5 emu. std dev= 267. gammas. correl coef= 0.92 zero 1796. gammas
12.5
magnetic susceptibility is 329.36e-5 cgs.

0.0 gamma is added to t-output

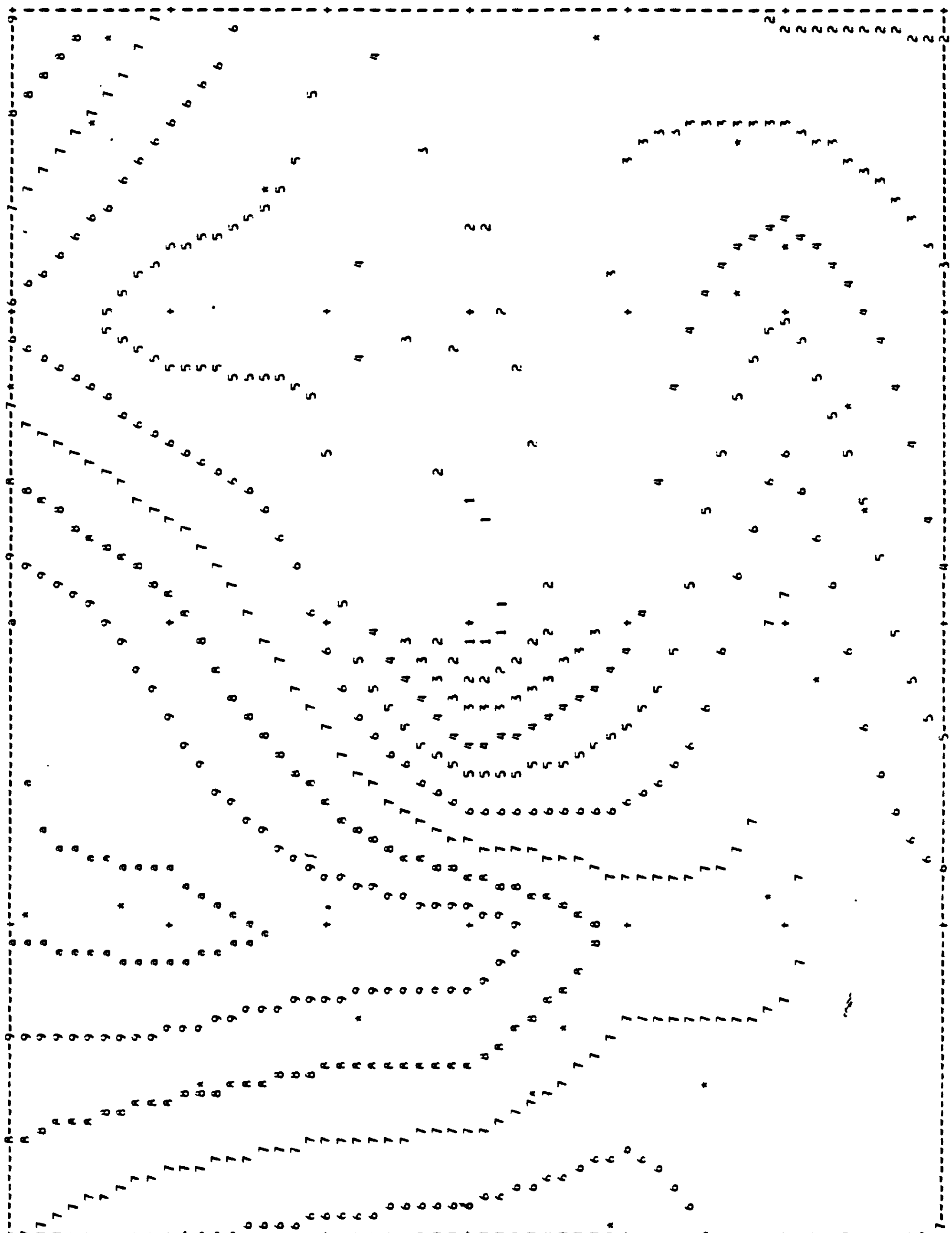
	x	y	z	del-x	del-y	del-z	del-h	del-t	error
1	0.500	0.000	4500.000	17.13	34.70	-9.08	38.26	9.35	1820.65
2	1.000	0.000	4500.000	18.00	46.40	-9.29	49.69	14.37	1800.63
3	1.500	0.000	4500.000	13.63	62.94	-3.51	64.08	26.07	1793.93
4	2.000	0.000	4500.000	1.01	71.34	4.92	68.20	35.44	1789.56
5	2.500	0.000	4500.000	-26.60	62.93	12.65	51.71	34.82	1785.18
6	3.000	0.000	4500.000	-30.94	33.06	-9.92	21.95	1.17	1813.83
7	3.500	0.000	4500.000	-19.52	18.66	-12.02	11.76	-5.34	1820.34
8	0.500	1.000	4500.000	60.52	59.27	72.59	75.00	98.79	1981.21
9	1.000	1.000	4500.000	59.93	100.80	72.54	114.34	116.67	1888.33
10	1.500	1.000	4500.000	35.68	121.44	101.67	126.53	148.15	1876.85
11	2.000	1.000	4500.000	-21.52	118.20	102.06	105.87	139.08	1875.92
12	2.500	1.000	4500.000	-74.77	106.99	78.99	78.83	106.23	1833.77
13	3.000	1.000	4500.000	-82.20	76.25	35.32	47.30	52.98	1822.02
14	3.500	1.000	4500.000	-63.16	27.70	18.29	6.95	19.45	1810.55

```
 15     0.500      2.000   4500.000     135.39     -12.61      62.27      29.61      68.92    1881.08
 16     1.000      2.000   4500.000     150.98      -6.64     164.71      40.09     164.89    1850.11
 17     1.500      2.000   4500.000     180.63     -30.02     324.98      26.95     301.58    1798.42
 18     2.000      2.000   4500.000     -67.30      58.57     565.07      35.05     519.00    1686.00
 19     2.500      2.000   4500.000    -197.37     115.99     263.09      49.72     256.85    1753.15
 20     3.000      2.000   4500.000    -221.37      -7.50      54.49     -75.17      14.26    1825.74
 21     3.500      2.000   4500.000     -95.07     -27.36     -49.10     -55.25     -68.88    1763.88
 22     0.500      3.000   4500.000      98.68     -52.22      49.10     -19.37      34.89    1795.11
 23     1.000      3.000   4500.000      59.81    -107.00      79.08     -83.44      32.38    1767.62
 24     1.500      3.000   4500.000      99.87    -142.04      99.82    -104.47      41.27    1773.73
 25     2.000      3.000   4500.000     -16.60     -63.58     264.46     -65.60     205.54    1714.46
 26     2.500      3.000   4500.000    -194.48     -87.68      40.87    -143.21     -28.85    1743.85
 27     3.000      3.000   4500.000    -116.26     -70.98     -67.14    -103.28    -106.82    1781.82
 28     3.500      3.000   4500.000     -35.93     -29.69     -64.22     -39.30     -75.07    1745.07
 29     0.500      4.000   4500.000       7.81     -46.92     -30.78     -42.25     -46.65    1861.65
 30     1.000      4.000   4500.000       6.68     -55.33     -56.34     -50.60     -73.21    1818.21
 31     1.500      4.000   4500.000      42.46     -85.59     -44.22     -68.40     -70.52    1780.52
 32     2.000      4.000   4500.000     -10.36    -128.64      11.34    -125.59     -47.11    1727.11
 33     2.500      4.000   4500.000     -63.19     -79.85     -59.80     -95.40     -96.69    1736.69
 34     3.000      4.000   4500.000     -18.82     -25.87     -63.76     -30.40     -70.61    1717.61
 35     3.500      4.000   4500.000      -0.50      -7.63     -38.85      -7.42     -37.96    1717.96

standard deviation=  0.18228E+04 and average difference=  0.17958E+04 gammas


expect plot of calculated values  of  7 by  5( 35) array. x=     0.5 to     3.5 y=     0.0 to     4.0 contoured at   20.0 gammas

    body point x=      0.460 y=     0.600 is outside plot boundary.
    body point x=      0.050 y=     0.620 is outside plot boundary.
    body point x=     -0.050 y=     1.070 is outside plot boundary.
    body point x=      0.460 y=     1.560 is outside plot boundary.
    body point x=      0.050 y=     1.860 is outside plot boundary.
    body point x=      0.470 y=     1.890 is outside plot boundary.
    body point x=      0.260 y=     2.160 is outside plot boundary.
    body point x=      0.300 y=     3.090 is outside plot boundary.
    body point x=      0.490 y=     3.920 is outside plot boundary.
    body point x=      1.830 y=     4.390 is outside plot boundary.
    body point x=      2.310 y=     4.380 is outside plot boundary.
    body point x=      3.600 y=     1.520 is outside plot boundary.
    body point x=      3.650 y=     1.130 is outside plot boundary.
    body point x=      4.030 y=     0.770 is outside plot boundary.
    body point x=      3.920 y=     0.520 is outside plot boundary.
    body point x=      3.610 y=     1.060 is outside plot boundary.
```



```
symbol        value (gammas).   y increases to right and x increases downward
   1             -100.0000
   2              -80.0000
   3              -60.0000
   4              -40.0000
   5              -20.0000
   6                0.0000
   7               20.0000
   8               40.0000
   9               60.0000
   a               80.0000
   b              100.0000
   c              120.0000
   d              140.0000
   e              160.0000
   f              180.0000
   g              200.0000
   h              220.0000
   i              240.0000
   j              260.0000
   k              280.0000
   l              300.0000
   m              320.0000
   n              340.0000
   o              360.0000
   p              380.0000
   q              400.0000
   r              420.0000
   s              440.0000
   t              460.0000
   u              480.0000
   v              500.0000
   +      fieldpoint              *      corner of body

expect plot of residual values   of  7 by  5( 35) array. x=    0.5 to    3.5 y=    0.0 to    4.0 contoured at   20.0 gammas

     body point x=     0.460 y=     0.600 is outside plot boundary.
     body point x=     0.050 y=     0.620 is outside plot boundary.
     body point x=    -0.050 y=     1.070 is outside plot boundary.
     body point x=     0.460 y=     1.560 is outside plot boundary.
     body point x=     0.050 y=     1.860 is outside plot boundary.
     body point x=     0.470 y=     1.890 is outside plot boundary.
     body point x=     0.260 y=     2.160 is outside plot boundary.
     body point x=     0.300 y=     3.090 is outside plot boundary.
     body point x=     0.490 y=     3.920 is outside plot boundary.
     body point x=     1.830 y=     4.390 is outside plot boundary.
     body point x=     2.310 y=     4.380 is outside plot boundary.
     body point x=     3.600 y=     1.520 is outside plot boundary.
     body point x=     3.650 y=     1.130 is outside plot boundary.
     body point x=     4.030 y=     0.770 is outside plot boundary.
     body point x=     3.920 y=     0.520 is outside plot boundary.
     body point x=     3.610 y=     1.060 is outside plot boundary.
```

Output Example 1--Continued



Output Example 1--Continued

```
symbol         value (gammas).   y increases to right and x increases downward
   1             1700.0000
   2             1720.0000
   3             1740.0000
   4             1760.0000
   5             1780.0000
   6             1800.0000
   7             1820.0000
   8             1840.0000
   9             1860.0000
   a             1880.0000
   +       fieldpoint              *       corner of body
```

Appendix C

Output Example 2

The following output was produced using command example 2 and a contouring program to produce the plots of the observed, calculated, and residual values.

```
exact solution for 3d magnetic bodies bounded by horizontal and vertical planes
    magpoly test 1
    expect determination of standard deviation using inputted t-component of magnetic field.
    units of distance measurement are in mile and heights are in feet
    declination=   17.9 and inclination=  62.9 degrees for earths field
        of 52000. gammas.
    magnetic anomaly is assumed to be small compared to earths field.


body  1 between depths   2000.000 and    1200.000 with  36 corners.
        1.140     0.480           1.610     0.710         0.840     1.070         0.460     0.600         0.540     1.040
        0.050     0.670          -0.050     1.070         0.460     1.560         0.050     1.860         0.470     1.890
        0.260     2.160           0.520     2.770         0.300     3.090         0.490     3.920         0.830     3.910
        0.760     3.630           1.300     3.430         1.830     4.390         2.310     4.380         2.420     3.900
        2.880     3.570           3.010     3.230         2.840     3.080         3.200     2.700         3.270     2.360
        3.100     1.820           3.600     1.520         3.650     1.130         4.030     0.770         3.920     0.520
        3.610     1.060           2.950     1.120         2.310     0.680         2.770     0.500         2.470     0.030
        2.170     0.460
    magnetic susceptibility is   329.36e-5 cgs.


body  2 between depths   2400.000 and    2000.000 with  11 corners.
        1.200     1.210           0.540     2.290         0.800     2.360         0.510     3.340         1.310     3.250
        1.950     4.120           2.140     4.060         2.710     3.490         2.610     3.010         3.130     2.540
        2.930     1.240
    magnetic susceptibility is   329.36e-5 cgs.


body  3 between depths   2400.000 and    2000.000 with   4 corners.
        0.400     1.260           0.560     1.430         0.720     1.340         0.600     1.190
    magnetic susceptibility is   329.36e-5 cgs.


body  4 between depths   2800.000 and    2400.000 with  12 corners.
        2.510     1.480           1.400     1.490         0.980     2.040         1.000     2.510         1.460     2.970
        2.000     3.930           2.210     3.860         2.600     3.470         2.430     2.960         3.030     2.540
        2.800     1.820           2.580     1.770
    magnetic susceptibility is   329.36e-5 cgs.


body  5 between depths   3200.000 and    2800.000 with  13 corners.
        1.520     1.680           1.270     1.940         1.600     2.640         1.460     2.830         1.770     2.900
        1.850     3.480           2.120     3.740         2.490     3.410         2.240     2.930         2.870     2.490
        2.770     2.070           2.400     1.940         2.320     1.620
    magnetic susceptibility is   329.36e-5 cgs.


body  6 between depths   3600.000 and    3200.000 with  13 corners.
        2.120     1.800           1.500     1.860         1.960     2.970         1.920     3.240         2.060     3.620
        2.350     3.330           2.000     3.010         1.940     2.790         2.180     2.620         2.240     2.700
        2.690     2.530           2.700     2.150         2.240     2.110
    magnetic susceptibility is   329.36e-5 cgs.


body  7 between depths   3800.000 and    3600.000 with   6 corners.
        1.700     2.000           1.900     2.630         2.640     2.500         2.650     2.210         2.150     2.180
        2.080     1.840
    magnetic susceptibility is   329.36e-5 cgs.
```

```
body  8 between depths   4000.000 and   3800.000 with   4 corners.
     2.040    1.970         1.950   1.970         1.950   2.050         2.060   2.040
   magnetic susceptibility is   329.36e-5 cgs.

body  9 between depths   4000.000 and   3800.000 with   4 corners.
     1.890    2.190         1.890   2.500         2.550   2.490         2.600   2.310
   magnetic susceptibility is   329.36e-5 cgs.
```

observed values

x	y	z	m
0.5000	0.0000	4500.0000	1830.0000
1.0000	0.0000	4500.0000	1815.0000
1.5000	0.0000	4500.0000	1820.0000
2.0000	0.0000	4500.0000	1825.0000
2.5000	0.0000	4500.0000	1820.0000
3.0000	0.0000	4500.0000	1815.0000
3.5000	0.0000	4500.0000	1815.0000
0.5000	1.0000	4500.0000	1980.0000
1.0000	1.0000	4500.0000	2005.0000
1.5000	1.0000	4500.0000	2025.0000
2.0000	1.0000	4500.0000	2015.0000
2.5000	1.0000	4500.0000	1940.0000
3.0000	1.0000	4500.0000	1875.0000
3.5000	1.0000	4500.0000	1830.0000
0.5000	2.0000	4500.0000	1950.0000
1.0000	2.0000	4500.0000	2015.0000
1.5000	2.0000	4500.0000	2100.0000
2.0000	2.0000	4500.0000	2205.0000
2.5000	2.0000	4500.0000	2010.0000
3.0000	2.0000	4500.0000	1840.0000
3.5000	2.0000	4500.0000	1695.0000
0.5000	3.0000	4500.0000	1830.0000
1.0000	3.0000	4500.0000	1800.0000
1.5000	3.0000	4500.0000	1815.0000
2.0000	3.0000	4500.0000	1920.0000
2.5000	3.0000	4500.0000	1715.0000
3.0000	3.0000	4500.0000	1675.0000
3.5000	3.0000	4500.0000	1670.0000
0.5000	4.0000	4500.0000	1815.0000
1.0000	4.0000	4500.0000	1745.0000
1.5000	4.0000	4500.0000	1710.0000
2.0000	4.0000	4500.0000	1680.0000
2.5000	4.0000	4500.0000	1640.0000
3.0000	4.0000	4500.0000	1647.0000
3.5000	4.0000	4500.0000	1680.0000

total of   35 fieldpoints.

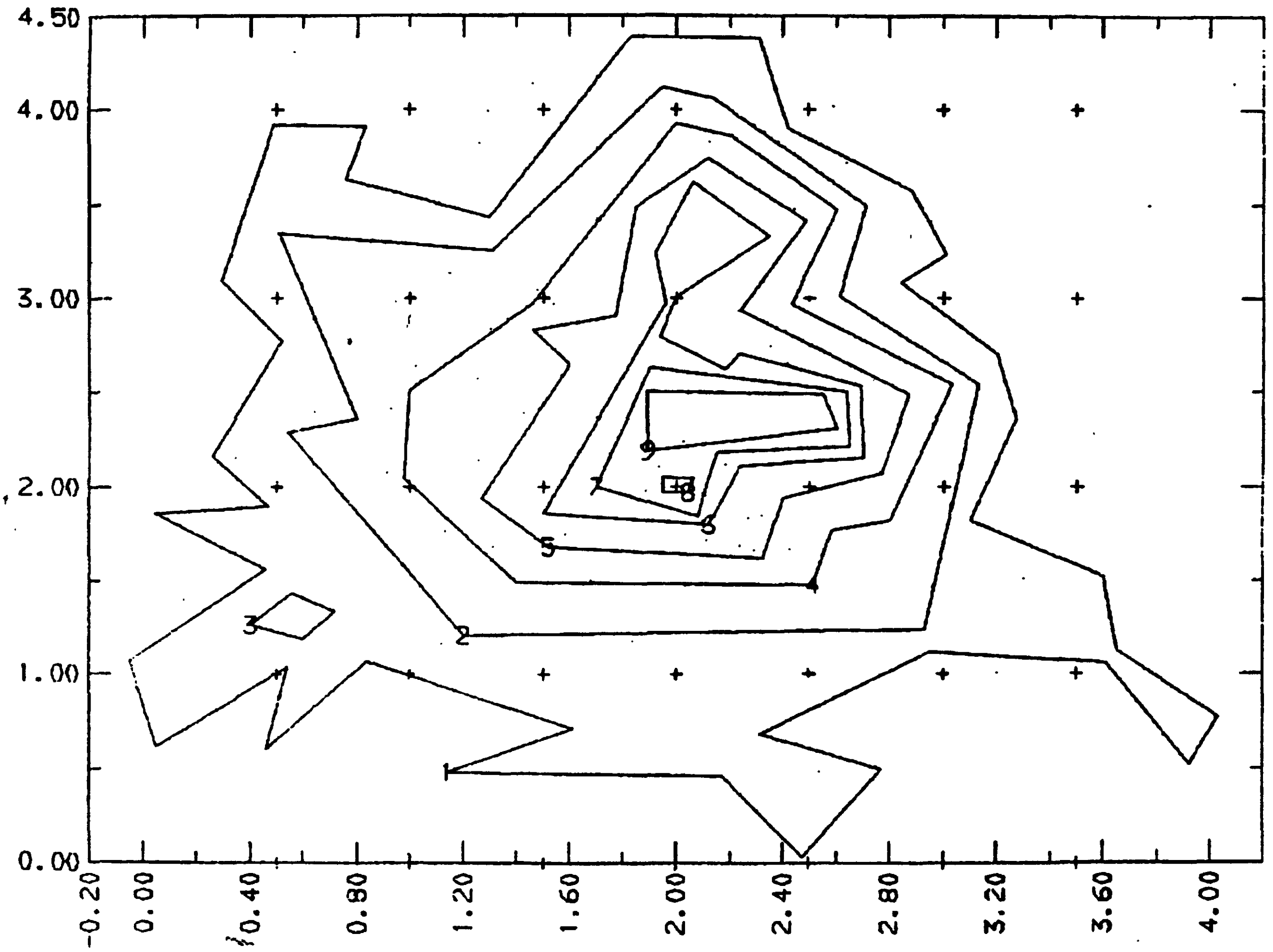
0.0 gamma is added to t-output

	x	y	z	del-x	del-y	del-z	del-h	del-t	error
1	0.500	0.000	4500.000	17.13	34.70	-9.08	38.28	9.35	1820.65
2	1.000	0.000	4500.000	16.00	46.40	-9.29	49.69	14.37	1800.63
3	1.500	0.000	4500.000	13.63	62.94	-3.51	64.08	26.07	1793.93
4	2.000	0.000	4500.000	1.01	71.34	4.92	68.20	35.44	1789.56

Output Example 2--Continued

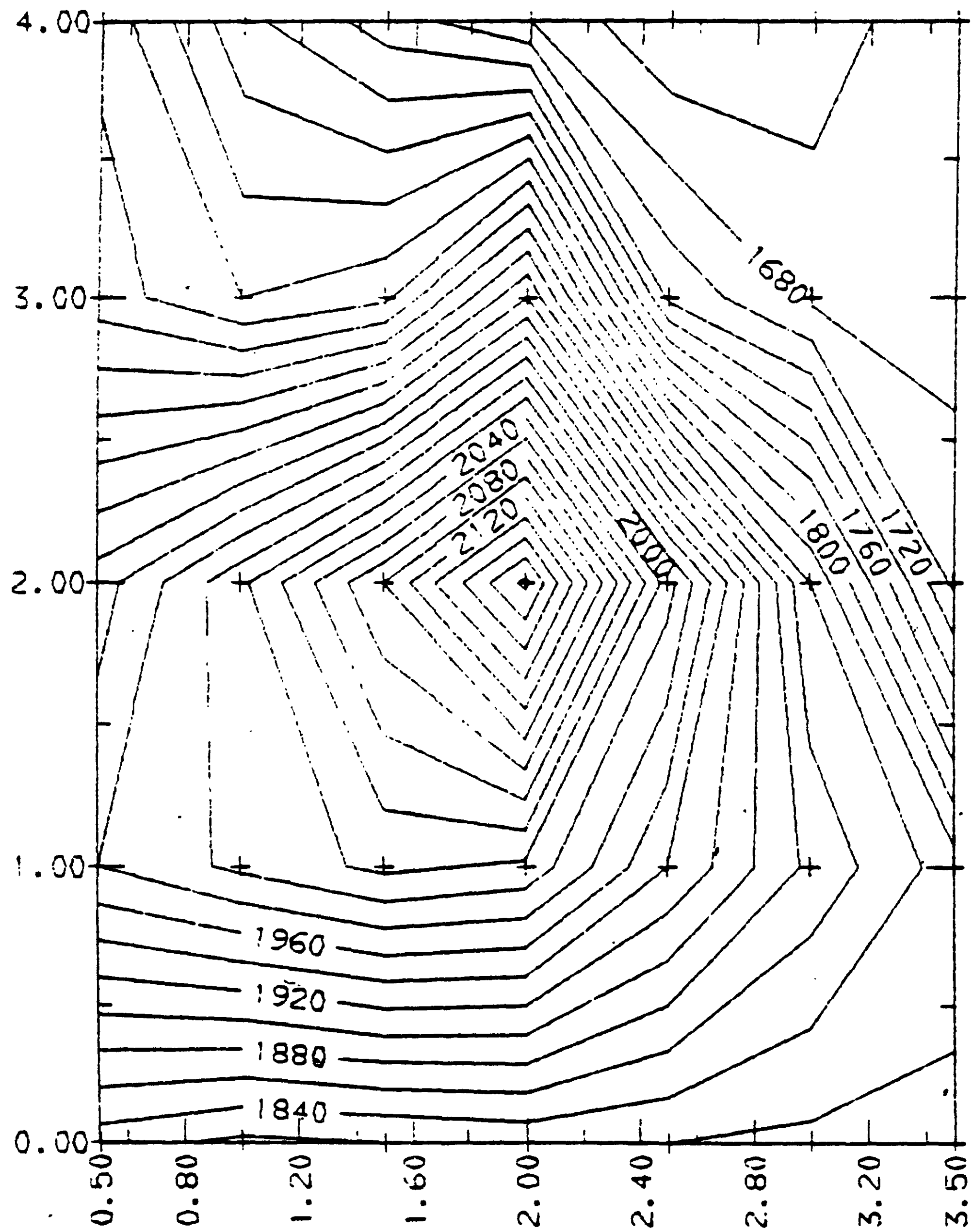
5	2.500	0.000	4500.000	-26.60	62.93	12.65	51.71	34.82	1785.18
6	3.000	0.000	4500.000	-30.94	33.06	-9.92	21.95	1.17	1813.83
7	3.500	0.000	4500.000	-19.52	18.66	-12.02	11.76	-5.34	1820.34
8	0.500	1.000	4500.000	60.52	59.27	72.59	75.00	98.79	1881.21
9	1.000	1.000	4500.000	59.93	100.80	72.54	114.34	116.67	1888.33
10	1.500	1.000	4500.000	35.68	121.45	101.68	126.53	148.15	1876.85
11	2.000	1.000	4500.000	-21.52	118.20	102.06	105.87	139.08	1875.92
12	2.500	1.000	4500.000	-74.77	106.99	78.99	78.83	106.23	1833.77
13	3.000	1.000	4500.000	-82.20	76.25	35.32	47.30	52.99	1822.01
14	3.500	1.000	4500.000	-63.16	27.70	18.29	6.95	19.45	1810.55
15	0.500	2.000	4500.000	135.39	-12.61	62.27	29.61	68.92	1881.08
16	1.000	2.000	4500.000	150.98	-6.64	164.71	40.09	164.89	1850.11
17	1.500	2.000	4500.000	180.63	-30.02	324.98	26.95	301.58	1798.42
18	2.000	2.000	4500.000	-67.30	58.57	565.08	35.05	519.01	1685.99
19	2.500	2.000	4500.000	-197.37	115.99	263.09	49.72	256.85	1753.15
20	3.000	2.000	4500.000	-221.37	-7.50	54.49	-75.17	14.26	1825.74
21	3.500	2.000	4500.000	-95.07	-27.36	-49.10	-55.25	-68.88	1763.88
22	0.500	3.000	4500.000	98.68	-52.22	49.10	-19.37	34.89	1795.11
23	1.000	3.000	4500.000	59.81	-107.00	79.08	-83.44	32.38	1767.62
24	1.500	3.000	4500.000	99.87	-142.04	99.83	-104.47	41.27	1773.73
25	2.000	3.000	4500.000	-16.60	-63.58	264.46	-65.60	205.54	1714.46
26	2.500	3.000	4500.000	-194.48	-87.68	40.87	-143.21	-28.85	1743.85
27	3.000	3.000	4500.000	-116.27	-70.98	-67.14	-103.28	-106.82	1781.82
28	3.500	3.000	4500.000	-35.93	-29.69	-64.22	-39.30	-75.07	1745.07
29	0.500	4.000	4500.000	7.81	-46.92	-30.78	-42.25	-46.65	1861.65
30	1.000	4.000	4500.000	6.68	-55.33	-56.34	-50.60	-73.21	1818.21
31	1.500	4.000	4500.000	42.46	-85.59	-44.22	-68.40	-70.52	1780.52
32	2.000	4.000	4500.000	-10.36	-128.64	11.34	-125.59	-47.11	1727.11
33	2.500	4.000	4500.000	-63.19	-79.85	-59.80	-95.40	-96.69	1736.69
34	3.000	4.000	4500.000	-18.82	-25.87	-63.76	-30.40	-70.61	1717.61
35	3.500	4.000	4500.000	-0.50	-7.63	-38.85	-7.42	-37.96	1717.96

standard deviation= 0.18228E+04 and average difference= 0.17958E+04 gammas

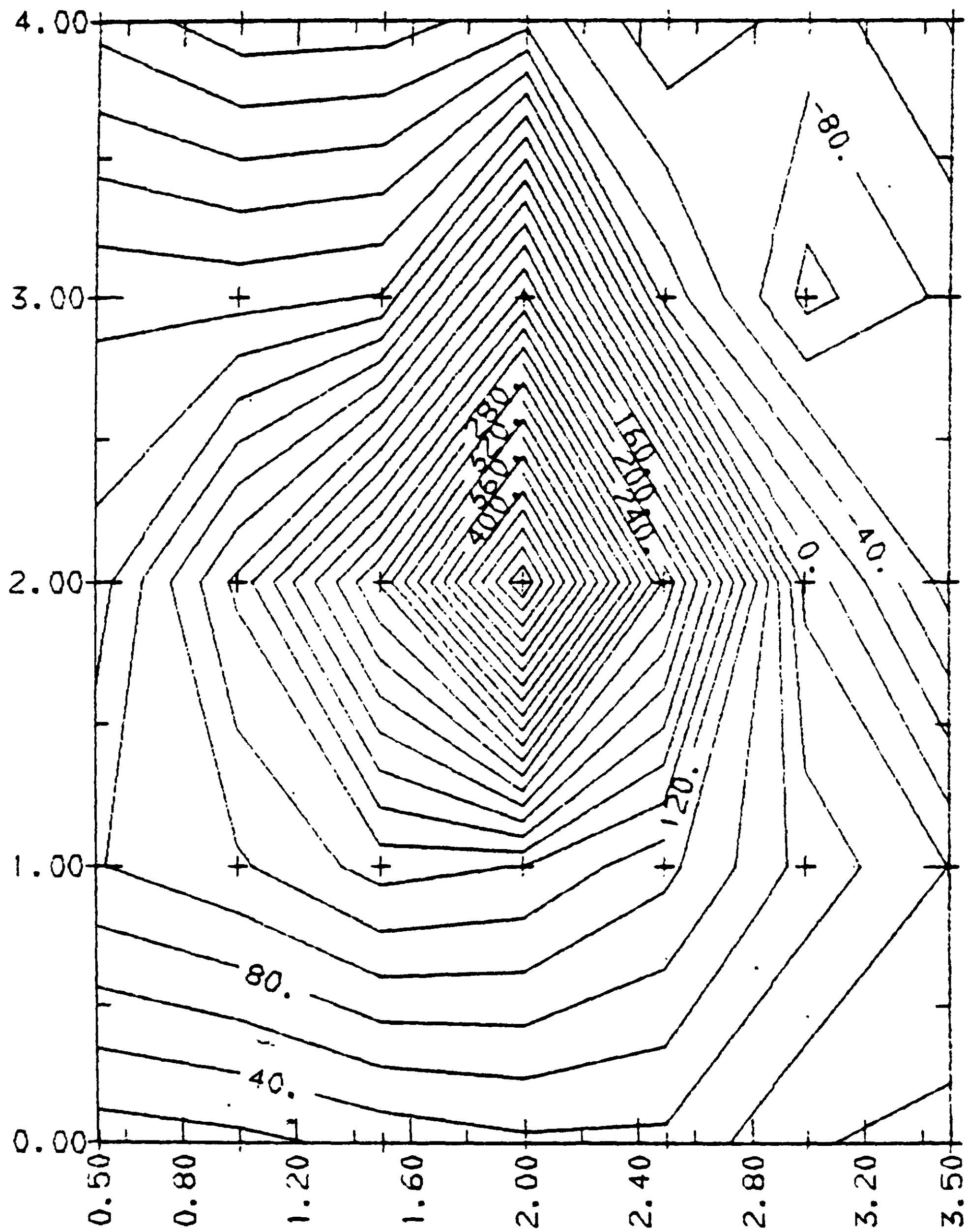


Body corners and fieldpoint locations

Output Example 2--Continued

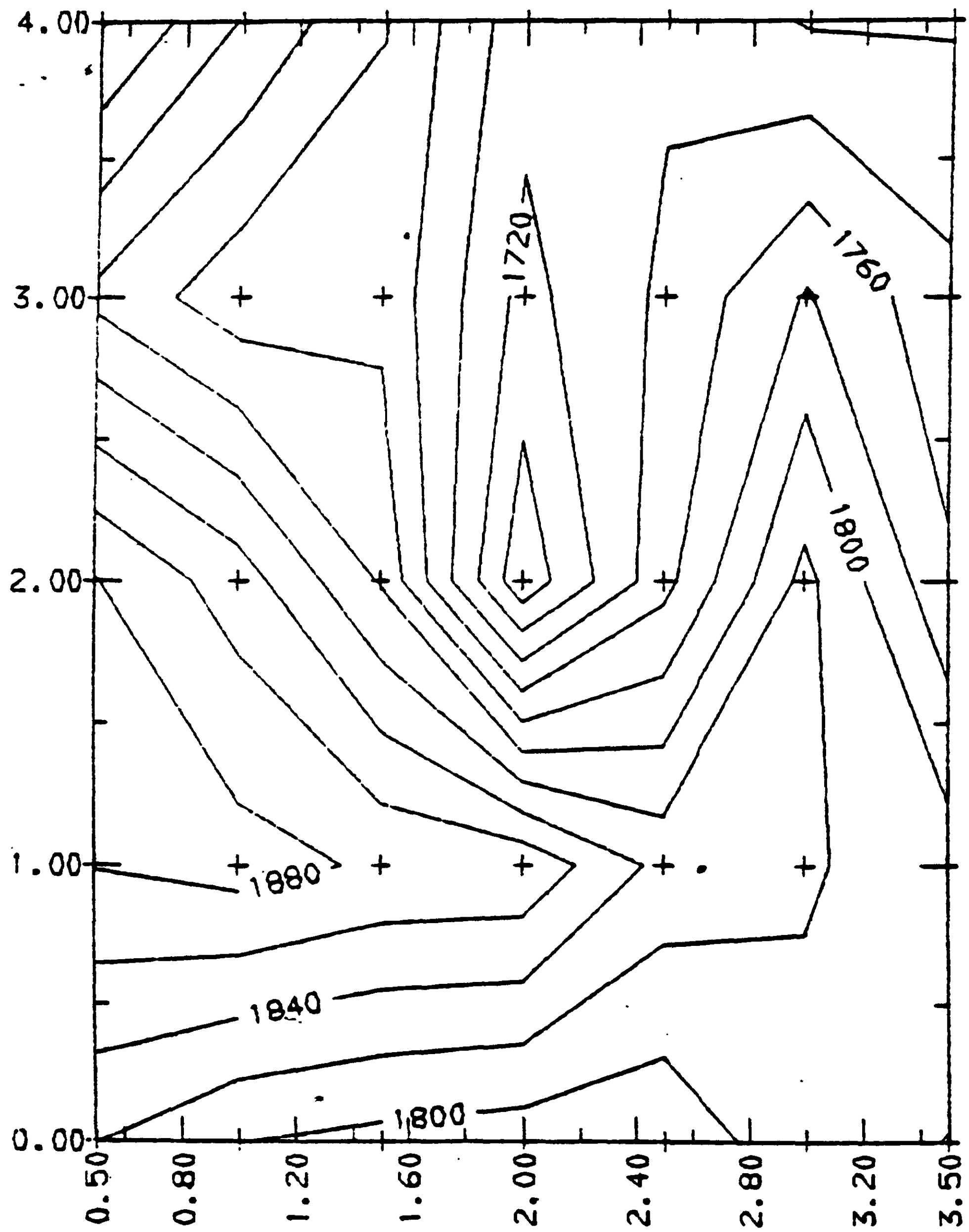


Observed values



Calculated values

Output Example 2--Continued



Residual values

Appendix D

Standard Grid

The grid can be any one of many types: (a) a rectangular grid with equal spacing in the x- and y-directions, (b) a rectangular grid with constant but unequal x- and y-intervals, (c) a rectangular grid with varying distances between grid positions in either the x- or y-direction or both, and d) a quadrilateral grid that consists of connecting quadrilaterals whose interior angles do not exceed 180 degrees.

The file of the gridded data consists of two basic parts: (1) a header record and optionally, a following record that contains the x-coordinates for each column, and (2) a series of data records, each containing the column values for one row.

The following diagram shows the relationship of the grid elements in the usual case where dx and dy are positive.

```
NROW  X      X      X      X      X   last row stored

  3   X      X      X      X      X      X
                    dy
  2   X      X      X  dx  X      X      X

  1   X      X      X      X      X      X   first row stored

      1      2      3      4      5    NCOL
   (xo,yo)
```

A. Header record (23 words long)

id: 56 ASCII characters of identification (14 words).

pgm: 8 ASCII characters of creation program identification (2 words).

ncol: number of columns of data (integer, 1 word).

nrow: number of rows of data (integer, 1 word).

nz: number of words per data element (integer, 1 word). For single precision use 1, double precision or complex use 2, double precision complex use 4. For quadrilateral grids this value is 3.

xo: position of first column of data (real, 1 word).

dx: equal spacing interval of columns (real, 1 word).

If equal to zero, then coordinates for each column are in the following data record; otherwise the following record consists of data for row one.

yo: position of first row (real, 1 word).

dy: equal spacing interval of rows (real, 1 word). If equal to zero, the coordinate for each row is the first word of each data record row.

B. Column coordinate record, present only if dx of header record is equal to zero. Record consists of ncol real words specifying the coordinates of each data column in monotonic order. If nz=3 then this record is present but the values are meaningless.

C. Data record. Each data record contains one row of real data items. The total record length is ncol times (nz plus 1) words. For quadrilateral data the sequence of data values is x, y, and z. The first word contains the row coordinate if dy of the header record is zero, else the value is a dummy. Again, the row coordinates should be in monotonic sequence, if specified.

In general, i/o for this standard file can be stated in fortran as:

```
      dimension g(iz,ix,iy),id(14),pgm(2),x(ix),y(iy)
      read or write (..) id,pgm,ncol,nrow,nz,xo,dx,yo,dy
      if (dx.eq.0.) read or write (..) (x(i),i=1,ncol
      if (dy.ne.0.) go to 15
      do 10 j=1,nrow
10    read or write (..) y(j),((g(k,i,j),k=1,nz),i=1,ncol)
      go to 25
15    do 20 j=1,nrow
```

```
20    read or write (..) dum,((g(k,i,j),k=1,nz),i=1,ncol)
25    continue
```

In the usual case where dx and dy are constant and nz=1. the code simplifies to:

```
      dimension g(ix,iy),id(14),pgm(2)
      read or write(..) id,pgm,ncol,nrow,nz,xo,dx,yo,dy
      do 10 j=1,nrow
10    read or write (..) dum,(g(i,j),i=1,ncol)
```

Appendix E

Example of User-Formatted File

```
.5  0 1830 4500
1.  0 1815 4500
1.5 0 1820 4500
2.  0 1825 4500
2.5 0 1820 4500
3   0 1815 4500
3.5 0 1815 4500
.5  1 1980 4500
1   1 2005 4500
1.5 1 2025 4500
2   1 2015 4500
2.5 1 1940 4500
3   1 1875 4500
3.5 1 1830 4500
.5  2 1950 4500
1   2 2015 4500
1.5 2 2100 4500
2   2 2205 4500
2.5 2 2010 4500
3   2 1840 4500
3.5 2 1695 4500
.5  3 1830 4500
1   3 1800 4500
1.5 3 1815 4500
2   3 1920 4500
2.5 3 1715 4500
3   3 1675 4500
3.5 3 1670 4500
.5  4 1815 4500
1   4 1745 4500
1.5 4 1710 4500
2   4 1680 4500
2.5 4 1640 4500
3   4 1647 4500
3.5 4 1680 4500
```

Appendix F

External Subroutines

The following subroutines are called externally by the program:

Basic Plot System (Evenden and Wahl, 1977)

char (x, y, a, nc, icode, size, theta, xoff, yoff) - writes characters on the plot

endpt (ie) - terminates plot

line (x, y, n, icon, ipn) - draws one or more lines

neatl () - draws a box around data area

pltset (iplotr, xbd, ybd, isl) - initializes plot system

scale (dxp, dyp, xp, yp, nopts, ier) - defines the plot area and units

xaxis (dxp, dyp, xp, del, ip, size, fmt, nfmt) - draws and labels x axis

yaxis (dxp, dyp, xp, del, ip, size, fmt, nfmt) - draws and labels y axis

where:

a = character string to be plotted

del = data unit interval of tick marks

dxp = array defining plot size (see fig. C1)

dyp = array defining plot size (see fig. C1)

fmt = Fortran format for labels

icode = 1 means x, y are arrays, 3 means x, y are single elements

icon = 0 start new line, 1 continue old line

ie = 10 element array containing diagnostics

ier = set to 0 to indicate no problems

ip = interval between tick mark lables

ipn = line type, 0=solid, 1-7 dashed lines

iplotr = plotter number

isl = 5 element array, isl(1)=1 inches, 2 cm

n = dimension of x, y arrays in line call

nc = if icode is 0 or 1, dimension of x and y arrays, otherwise length of a

nfmt = length of format

nopts = length of xp and yp arrays

size = character size

theta = rotation of char in radians

x = x location (array of dimension n)

y = y location (array of dimension n)

xbd = returned maximum x size for selected plotter

ybd = returned maximum y size for selected plotter

xoff = x offset for center of label

yoff = y offset for center of label

xp = array defining plot size (see fig. C1)

yp = array defining plot size (see fig. C1)

scale3(xmin, xmax, nxl, xminp, xmaxp, xdel)

where:

xmin = minimum x value of data (input)

xmax = maximum x value of data (input)

nxl = number of divisions along x axis (input)

xminp = minimum whole number of x value (output)

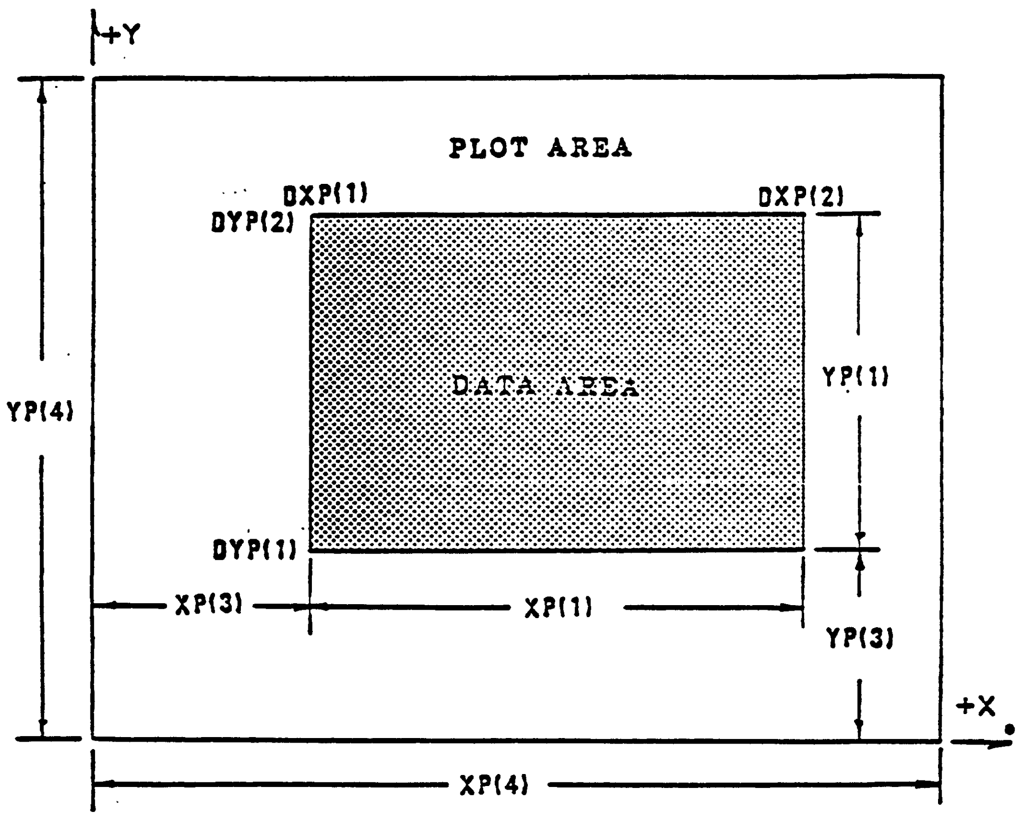
xmaxp = maximum whole number of x value (output)

xdel = increment for labelling x axis (output)

Figure C1. DXP,DYP,XP,YP arrays and their relation to the plot area.

DXP and DYP are in data units.

XP and YP are in plot units (inches).



```
c***************
c exact solution for talwani 3d magnetics. plouff 7-77
c modified by r.godson for denver multics 3/81
c
      character*4 unix,uniz
      character*8 name
      character*18 title
      character*40 idplot
      character*50 ifile,ifile2,cfile
      character*54 rfile
      character*55 cafile
      character*56 pfile
      character*80 ifmt
      common/st/cafile,rfile,pfile
      common/maginp/iplotr,ibody,ifile,ifile2,lsqs,dec,fincl,
     1 datum,xscale,dc,unix,uniz,name,height,
     2 ifmt,idplot,sus,rem,d,zi,iobs,icalc,ires
      common/field/v(1000,5),r(1000),g(1000,3)
      common/fpts/xp(1000),yp(1000),zp(1000)
      common /side/xb(900),yb(900), ll(50)
      common /bodk/ mm,nlim,jbody,short,nend
      common /tjcal/tjx(50),tjy(50),tjz(50),mtest
      common /vec/cde,cie,sde,sie,ex,ey,ez,earth,cdci,sdci
      common /trans/ inpt,iout,jin,iprint
      common /bod/ astnc(900),ifprnt(50),zee(50),zm(50)
      dimension t(5),flsq(7),a(1000),b(1000)
      data flsq/'y','x','z','h','t','t','t'/
c dimension limits for 3 parameters.  corners, bodies, fieldpoints.
      data ncorn,nbod,npt,ier,tx,ty,tz
     1 / 900,50,1001,0,0.0,0.0,0.0/
c
c     get input parameters
c
      call min($390)
      write(iout,10) idplot
   10 format ('1exact solution for 3d magnetic bodies bounded by ',
     1 'horizontal and vertical planes',/,5x,a40)
      if(lsqs.ne.0) go to 20
      go to 40
   20 write (iout,30) flsq(lsqs)
   30 format (5x,'expect determination of standard deviation using ',
     1'inputted ',a1,'-component of magnetic field.')
      if (lsqs .lt. 6) go to 40
      if (earth  .eq. 0.0) earth=-50000.0
      if (fincl  .ne. 0.0) go to 40
      if (dec    .ne. 0.0) go to 40
      fincl=89.999
   40 facx=0.0
      facz=0.0
      call unit(unix,uniz,facx,facz)
      if (facx .eq. 0.0) go to 370
      zfac=1.0/facz
   50 call compon(1,0,dec,fincl,0.0,0.0,0.0,0.0,0.0)
      mtest=0
      short=1.0e20
      zdel =1.0e20
c
c     read body coordinates
c
      call bodm(facz,nbod,ncorn,zdel,tx,ty,tz)
```

```
      zdel=abs(facz*zdel)
c
c mm (common) is total number of layers
      if (mm .eq. 0) go to 370
      if (mtest .eq. 0) go to 70
      if (mtest .eq.mm) go to 70
      write (iprint,60)
   60 format (5x,'magnetization not specified for all bodies')
      go to 370
   70 kk=0
c
c     read fieldpoint locations and maybe values
c
      call ptsm(kk,$390)
c
c     check if printer plot of bodies & fieldpoints
c     desired if fieldpoints are in grid form
c
      title="body corners"
      if(ibody.ne.0.and.iplotr.eq.9.and.name.eq."gridded")
     1 call cplot(dc,a,1,0,title,$365)
c
c     check if location plot of bodies & fieldpoints on
c     plotting device wanted
c
      if(ibody.ne.0.and.iplotr.ne.9) call mag3dplot(kk,$390)
c
c     check if contour plot of observed values desired
c
      if(lsqs.eq.0) go to 80
      title="observed values"
      if(iobs.ne.0.and.iplotr.eq.9.and.name.eq."gridded")
     1 call cplot(dc,r,kk,1,title,$365)
c
   80 nk1=kk+1
      do 90 k=1,kk
   90 zp(k)=zp(k)*facz
      if(lsqs.eq.0) go to 110
      do 100 k=1,kk
      r(k)=r(k)-datum
  100 continue
  110 write (iout,120) kk
  120 format (//,' total of',i5,' fieldpoints.')
      call calc(kk,mm,short,zdel)
      if(lsqs.lt.6) go to 220
      go to 160
  130 read (jin,140,end=360) icalc,earth,sus,rem,d,zi
  140 format(v)
      write(iout,150)
  150 format('1')
      call compon(1,0,dec,fincl,0.0,0.0,0.0,0.0,0.0)
      go to 210
  160 d=0.0
      zi=0.0
      rem=0.0
      if (nk1 .gt. 3) go to 180
      write (iprint,170)
  170 format ('0stop. insufficient fieldpoints for least-squares.')
      go to 370
  180 call tlsqs(zi,d,rem,nk1,lsqs,a,b,datum)
```

```
      if (nk1 .gt. 0) go to 190
      go to 370
  190 if (lsqs .eq. 6) go to 200
      sus=rem
      rem=0.0
      go to 210
  200 sus=0.0
  210 call compon(2,ier,d,zi,sus,rem,tx,ty,tz)
      if (ier .eq. -1) go to 370
      mtest=0
  220 std=0.0
      avg=0.0
      ifdp=5
      if(lsqs.ne.0.and.lsqs.lt.6) ifdp=lsqs
      sym1=flsq(ifdp)
      write (iout,230) datum,sym1
  230 format (/,1x,f10.1,' gamma is added to ',a1,'-output',
     1 /, 12x,'x',10x,'y',9x,'z',9x,'del-x',
     2 6x,'del-y',6x,'del-z',6x,'del-h',6x,'del-t',5x,'error')
      do 300 k=1,kk
      if (mtest .eq. mm ) go to 240
      vk1=v(k,1)
      vk2=v(k,2)
      vk3=v(k,3)
      vk5=v(k,4)
      vk6=v(k,5)
      vk4=-(vk1+vk6)
      t(1)=tx*vk1+ty*vk2+tz*vk3
      t(2)=tx*vk2+ty*vk4+tz*vk5
      t(3)=tx*vk3+ty*vk5+tz*vk6
      go to 250
  240 t(1)=g(k,1)
      t(2)=g(k,2)
      t(3)=g(k,3)
  250 t(4)=0.0
      t(5)=0.0
      call compon(3,0,0.0,0.0,t(4),t(5),t(1),t(2),t(3))
      fx=xp(k)
      fy=yp(k)
      zfk=zp(k)*zfac
      conout=t(ifdp)  +datum
      t(ifdp) =conout
c save for contour plot
      a(k)=conout
  260 if ( lsqs .eq. 0) go to 280
      errk=r(k)-conout+datum
c save for residual contour plot
      h(k)=errk
      std=std+errk*errk
      avg=avg+errk
      write (iout,270) k,fy,fx,zfk,t(2),t(1),(t(j),j=3,5),errk
  270 format (i5,f10.3,2f11.3,6f11.2)
      go to 300
  280 write (iout,290) k,fy,fx,zfk,t(2),t(1),(t(j),j=3,5)
  290 format (i5,f10.3,2f11.3,5f11.2)
  300 continue
      if (lsqs .eq. 0) go to 330
  310 if (nk1 .lt. 3) go to 330
      std=sqrt(std/(nk1-2))
      avg=avg/kk
```

```
      write (iout,320) std,avg
  320 format (//,' standard deviation=',e13.5,' and average difference='
     1 e13.5,' gammas')
  330 mtest=0
c
c     contour plots
c
  340 title="calculated values"
      if (icalc.ne.0.and.iplotr.eq.9.and.name.eq."gridded")
     1 call cplot (dc,a,kk,1,title,$365)
      title="residual values"
      if (ires.ne.0.and.iplotr.eq.9.and.lsqs.ne.0.and.name.eq."gridded")
     1 call cplot (dc,b,kk,1,title,$365)
      if(icalc.ne.0.and.name.eq."gridded")
     1 call stdout(1,a)
      if(ires.ne.0.and.lsqs.ne.0.and.name.eq."gridded")
     1 call stdout(2,b)
  350 if (mtest .eq. mm .and. lsqs.ge.6) go to 160
      if(lsqs.lt.6) go to 130
  360 continue
      go to 390
  365 write(iprint,375)
  375 format(" contour interval(dc) is 0.")
  370 write(iprint,380)
  380 format(" error - program abort")
  390 close(jin)
      close(iout)
      write(iprint,400) pfile
  400 format(/," output is in file ",a56)
      stop
      end
c***************
      subroutine min(*)
c
c     namelist input parameter subroutine
c
c     r.godson - 3/81
c
      character*4 unix,uniz
      character*8 name
      character*54 rfile
      character*55 cafile
      character*56 pfile
      character*40 idplot
      character*50 ifile,ifile2,cfile
      character*80 ifmt
      common/st/cafile,rfile,pfile
      common/maginp/iplotr,ibody,ifile,ifile2,lsqs,dec,fincl,
     1 datum,xscale,dc,unix,uniz,name,height,
     2 ifmt,idplot,sus,rem,d,zi,iobs,icalc,ires
      common/contg/nrow,ncol,dy,dx,xmin,ymin,xmax,ymax
      common /vec/cae,cie,sae,sie,ex,ey,ez,earth,cdci,sdci
      common /trans/ inpt,iout,jin,iprint
      common/widect/naxcol
      namelist/parms/iplotr,ibody,ifile,ifile2,lsqs,earth,dec,fincl,
     1 datum,xscale,dc,unix,uniz,name,height,
     2 ifmt,idplot,iobs,icalc,ires,naxcol
      inpt=5
      iout=8
      iprint=6
```

```
      jin=9
      iplotr=9
      maxcol=130
      ibody=0
      icalc=0
      iobs=0
      ires=0
      idplot=" "
      ifile=" "
      ifile2=" "
      lsqs=0
      earth=0.
      dec=0.
      fincl=0.
      datum=0.
      xscale=0.
      dc=0.
      unix="kilm"
      uniz="feet"
      name="gridded"
      height=0.
      ifmt="(v)"
      write(iprint,10)
   10 format(" enter command file name: ",$)
      read(inpt,20) cfile
   20 format(v)
      open(jin,file=cfile,form="formatted")
      read(jin,parms,end=30)
      if(lsqs.eq.0) ires=0
      if(iplotr.lt.0.or.iplotr.gt.9) iplotr=9
      if(ifile.eq." ") go to 50
c
c     encode output file names
c
      encode(pfile,25) cfile
   25 format("print.",a50)
      encode(cafile,26) cfile
   26 format("calc.",a50)
      encode(rfile,27) cfile
   27 format("res.",a50)
      open(iout,file=pfile,form="formatted",carriage=.true.)
      return
   30 write(iprint,40)
   40 format(" end of file reading namelist command file")
      go to 70
   50 write(iprint,60)
   60 format(" no fieldpoint data file name specified")
   70 return 1
      end
c****************
      subroutine bodm(facz,nbod,ncorn,zgel,tx,ty,tz)
c
c      reads body corner coordinates
c
c      r.godson - 4/81
c
      character*4 unix,uniz
      character*8 name
      character*40 idplot
      character*50 ifile,ifile2,cfile
```

```
      character*80 ifmt
      common/mplot/ymin,xmin,ymax,xmax
      common/conta/nrow,ncol,dy,dx,yo,xo,xma,yma
      common/maginp/iplotr,ibody,ifile,ifile2,lsqs,dec,fincl,
     1 datum,xscale,dc,unix,uniz,name,height,
     2 ifmt,iqplot,sus,rem,d,zi,iobs,icalc,ires
      common /bod/ dstnc(900),ifornt(50),zee(50),zm(50)
c see ncorn dimension limit in main program
      common /side/xb(900),yb(900), ll(50)
      common /bodk/ mm,nlim,jbody,short,nend
      common /tjcal/tjx(50),tjy(50),tjz(50),mtest
      common /trans/ inpt,iout,jin,iprint
      nend=0
      mm=0
      xmin=1.0e38
      xmax=-1.0e38
      ymin=1.0e38
      ymax=-1.0e38
   10 read(jin,20,end=190)num,ifornt(mm+1),sus,rem,d,zi,top,bottom
   20 format(v)
      if(num.eq.0) go to 190
      nst=nend+1
      nend=nend+num
      mm=mm+1
      if(mm.gt.nbod) go to 140
      if(nend.gt.ncorn) go to 70
      ll(mm)=num
      zee(mm)=top*facz
      zm(mm)=bottom*facz
      write(iout,30) mm,top,bottom,num
   30 format (' body',i3,' between depths',f11.3,' and',f11.3,
     1 ' with',i4,' corners.')
   40 if (top .gt. bottom) go to 90
   50 write (iprint,60) top,bottom
   60 format ('0top of body is below bottom',2f7.2)
      go to 180
   70 write(iprint,80) mm,nend,ncorn
   80 format (' body',i3,' includes sufficient corners,',i4,',to exceed'
     1 , ' limitation of',i5,' for all bodies.')
      go to 180
c coordinates of corners of body east,north (x,y) pairs.
c body has at least 3 corners
   90 read(jin,20,end=160) (yb(j),xb(j),j=nst,nend)
      write(iout,100) (yb(j),xb(j),j=nst,nend)
  100 format (5(5x,2f9.3))
      do 105 j=nst,nend
      if(xb(j).lt.xmin) xmin=xb(j)
      if(xb(j).gt.xmax) xmax=xb(j)
      if(yb(j).lt.ymin) ymin=yb(j)
      if(yb(j).gt.ymax) ymax=yb(j)
  105 continue
      if (lsqs.ge.6) go to 120
      ier=0
      call compon(2,ier,d,zi,sus,rem,tx,ty,tz)
      if (ier .gt. 0) go to 110
      if(ier.eq.-1) go to 180
      go to 120
  110 mtest=mtest+1
      tjx(mm)=tx
      tjy(mm)=ty
```

```
      tjz(mm)=tz
  120 np=nend+1
      xb(np)=xb(nst)
      yb(np)=yb(nst)
      do 130 j=nst,nend
      dist=(xb(j+1)-xb(j))**2+(yb(j+1)-yb(j))**2
      if (dist .eq. 0.0) go to 130
      dist=sqrt(dist)
      if (dist .lt. short) short=dist
  130 dstnc(j)=dist
      zd=top-bottom
      if ( zd  .lt.  zdel) zdel=zd
      go to 10
  140 write (iprint,150) mm
  150 format (5x,'number of bodies=',i3,' exceeds limitation.')
      go to 180
  160 write(iprint,170) mm
  170 format(" end of file reading x,y pairs for body no. ",i3)
  180 mm=0
  190 return
      end
c***************
      subroutine compon (num,ier,dec,fin,sus,rem,tx,ty,tz)
c various components of earth and magnetization fields
      common /vec/cde,cie,sde,sie,ex,ey,ez,earth,cdci,sdci
      common /trans/ inpt,iout,jin,iprint
      go to (10,90,190),num
c earths field
   10 arth=abs(earth)
      if (earth .eq. 0.0) go to 50
   20 write (iout,30) dec,fin
   30 format (5x,'declination=',f7.1,' and inclination=',f6.1,
     1 ' degrees for earths field')
      if (arth .ne. 0.0) write (iout,40) arth
   40 format (9x,'of',f7.0,' gammas.')
      d=1.745329e-2*dec
      z=1.745329e-2*fin
      cde= cos(d)
      cie= cos(z)
      sie= sin(z)
      sde= sin(d)
      cdci=cde*cie
      sdci=sde*cie
      if (earth .le. 0.0) go to 70
      eh=earth*cie
      ex=earth*cdci
      ey=earth*sdci
      ez=earth*sie
      return
   50 write (iout,60)
   60 format (5x,'magnetic susceptibilities are assumed to be negligible
     1 .',//)
      go to 20
   70 write (iout,80)
   80 format (5x,'magnetic anomaly is assumed to be small compared to ',
     1 'earths field.',//)
      return
c magnetization parameters
   90 tx=0.0
      ty=0.0
```

```
      tz=0.0
c     if (sus .gt. 0.0) go to 6
      if (sus .ne. 0.0) go to 130
      if (rem .eq. 0.0) go to 140
      write (iout,100) rem,dec,fin
  100 format (5x,'intensity of total magnetization is',f9.2,'e-5 ',
     1 'with a declination=',f7.1,' and inclination=',f6.1,
     2 ' degrees.',//)
  110 d=1.745329e-2*dec
      z=1.745329e-2*fin
      cdr= cos(d)
      cir= cos(z)
      sir= sin(z)
      sdr= sin(d)
      ehr=rem*cir
      tx=tx+ehr*cdr
      ty=ty+ehr*sdr
      tz=tz+rem*sir
  120 ier= 1
      return
  130 if (earth .ne. 0.0) go to 160
  140 ier=-1
      write (ioprint,150)
  150 format (5x,'error. normal field of earth or magnetization not ',
     1 'supplied')
      return
  160 write (iout,170) sus
  170 format (5x,'magnetic susceptibility is',f9.2,'e-5 cgs.',//)
      sus=arth*sus*1.0e-5
      tx=sus*cdci
      ty=sus*sdci
      tz=sus*sie
      if (rem .eq. 0.0) go to 120
      write (iout,180) rem,dec,fin
  180 format (5x,'intensity of remanent magnetization is',f9.2,'e-5 ',
     1 'with a declination=',f7.1,' and inclination=',f6.1,
     2 ' degrees.',//)
      go to 110
  190 if (earth .le. 0.0) go to 200
      delh=(ex+tx  )**2+(ey+ty  )**2
      delt=(ez+tz  )**2+delh
      sus=sqrt(delh)-eh
      rem=sqrt(delt)-earth
      return
  200 sus=tx  *cde+ty  *sde
      rem=tx*cdci+ty*sdci+tz*sie
      return
      end
c***************
      subroutine tlsqs(azi,ad,tm,nk1,lsqs,a,b,datum)
      real*8 fn,sx,sy,sxx,sxy,syy,sz,sxz,syz,d,a2,a1
      real*8 st,stt,stx,sty,stz,szz,a3,b1,b2,b3,c1,c2,c3,d1,d2,d3,det
      real*8 ptx,pty,ptz,pyz,pxz,pxy,sqx,sqy,sqz,siq,p
      integer nm(8)/'x','y','z','t','n','d','r','s'/
      common /vec/cde,cie,sde,sie,ex,ey,ez,earth,cdci,sdci
      common/field/v(1000,5),r(1000),q(1000,3)
      common /trans/ inpt,iout,jin,ioprint
      dimension a(1),b(1)
      p(fn,sxy,sx,sy)=(sxy-sx*sy/fn)/fn
      det(a1,a2,a3,b1,b2,b3,c1,c2,c3)=a1*(b2*c3-b3*c2)
```

```
      1 +a2*(b3*c1-b1*c3)+a3*(b1*c2-b2*c1)
        sig(fn,sx,sxx)=(sxx-sx*sx/fn)/fn
c observed value r
        kk=nk1-1
        fn=kk
        sx =0.0d0
        sy =0.0d0
        syy=0.0d0
        sxy=0.0d0
        sxx=0.0d0
        l=5
        if (lsqs .eq. 6) go to 40
        if (kk .lt. 3) go to 180
        eyy=sdci*sdci
        e1=cdci*cdci-eyy
        e6=sie*sie-eyy
        e2=2.0*cdci*sdci
        e3=2.0*cdci*sie
        e5=2.0*sdci*sie
c determination of best-fit susceptibility
        do 10 k=1,kk
        rk=r(k)
        ak =      e1*v(k,1)+e2*v(k,2)+e3*v(k,3)+e5*v(k,4)+e6*v(k,5)
        sx =sx +ak
        sy =sy +rk
        syy=syy+rk*rk
        sxy=sxy+ak*rk
   10   sxx=sxx+ak*ak
        xx=sxx-sx*sx/fn
        l=1
        if (xx .le. 0.0) go to 180
        l=4
        xy=sxy-sx*sy/fn
        yy=syy-sy*sy/fn
        tm=xy/xx
        tb=(sy-tm*sx)/fn
        coer=9999.0
        if (yy .le. 0.0) go to 20
        coer=xy/sqrt(xx*yy)
   20   eyy=abs(earth)*1.0e-5
        if (eyy .eq. 0.0) eyy=0.5
        stdm=yy*(1.0-coer*coer)/(xx*(kk-2))
        l=8
        if (stdm .lt. 0.0) go to 180
        stdm=sqrt(stdm)
        tm=tm/eyy
        std=sqrt(yy /(kk -1))/eyy
        tb=tb+datum
        write (iout,30) tm,std,coer,tb,stdm
   30   format (//,' least-squares susceptibility is',f9.1,'e-5 emu. std',
      1 ' dev=',f7.0,' gammas. correl coef=',f6.2,' zero',f6.0,' gammas',
      2 /,32x,f9.1)
c       if (tm .le. 0.1) go to 7
        if (tm .gt. 0.7e5) go to 180
        azi=0.0
        ad=0.0
        return
c least squares determination of total magnetization vector
   40   if (kk .lt. 5) go to 180
        sz =0.0d0
```

```
      sxz=0.0d0
      syz=0.0d0
      st =0.0d0
      stt=0.0d0
      stx=0.0d0
      sty=0.0d0
      stz=0.0d0
      szz=0.0d0
      do 50 k=1,kk
      vk1=v(k,1)
      vk2=v(k,2)
      vk3=v(k,3)
      vk5=v(k,4)
      vk6=v(k,5)
      t=r(k)
      x= cdci*vk1+sdci*vk2+sie*vk3
      y= cdci*vk2-sdci*(vk1+vk6)+sie*vk5
      z= cdci*vk3+sdci*vk5+sie*vk6
      st =st +t
      sx =sx +x
      sy =sy +y
      sz =sz +z
      stt=stt+t*t
      stx=stx+t*x
      sty=sty+t*y
      stz=stz+t*z
      szz=szz+z*z
      sxx=sxx+x*x
      sxy=sxy+x*y
      sxz=sxz+x*z
      syz=syz+y*z
      syy=syy+y*y
   50 continue
      ptx=p(fn,stx,st,sx)
      pty=p(fn,sty,st,sy)
      ptz=p(fn,stz,st,sz)
      pyz=p(fn,syz,sy,sz)
      pxz=p(fn,sxz,sx,sz)
      pxy=p(fn,sxy,sx,sy)
      sqx=siq(fn,sx,sxx)
      sqy=siq(fn,sy,syy)
      sqz=siq(fn,sz,szz)
      sqt=siq(fn,st,stt)
      d=det(sqx,pxy,pxz,pxy,sqy,pyz,pxz,pyz,sqz)
      l=6
      if (d  .eq. 0.0d0) go to 180
      x=det(ptx,pty,ptz,pxy,sqy,pyz,pxz,pyz,sqz)/d
      y=det(sqx,pxy,pxz,ptx,pty,ptz,pxz,pyz,sqz)/d
      z=det(sqx,pxy,pxz,pxy,sqy,pyz,ptx,pty,ptz)/d
      a0=-(x*sx+y*sy+z*sz-st)/fn
      tb=a0+datum
      std=x*ptx+y*pty+z*ptz
      coer=std/sqt
      l=7
      if (coer .lt. 0.0) go to 180
      coer=sqrt(coer)
      std=sqt-std
      l=8
      if (std  .lt. 0.0) go to 180
      std =sqrt(std )
```

```
      write(iout,60)y,x,z,std,coer,th
   60 format (//,' least squares determined values. jx=',f8.1,'e-5  jy='
     1,f8.1,'e-5, jz=',f8.1,'e-5',/,5x,' (assuming anomaly is negligibl'
     2, 'e compared to total field of earth)',/,5x,'std',
     3 ' dev=',f7.0,' gammas. correl coef=',f6.2,' zero',f6.0,' gammas')
      th=x*x+y*y
      tm=sqrt(th+z*z)
      th=sqrt(th)
      if (th .gt. 0.0) go to 80
      if (z  .gt. 0.0) go to 70
      azi=-90.
      go to 170
   70 azi=90.
      go to 170
   80 azi=atan (z/th)*57.29578
      if(x) 120,90,130
   90 if(y)100,100,110
  100 ad=-90.
      go to 150
  110 ad=90.
      go to 150
  120 gac=180.
      go to 140
  130 gac=0.0
  140 ad=atan (y/x)*57.29578+gac
  150 write          (iout,160)tm,th,azi,ad
  160 format (/,'    jt=',f8.1,'e-5,   jh=',f8.1,'e-5, zi=',f6.1,', dec=',
     1 f7.1)
      l=4
c     if (tm .le. 0.1) go to 7
      return
  170 ad=0.0
      go to 150
  180 write (iprint,190) nm(l)
  190 format ('0stop. ',a1,'-component of summed field or amplitude ',
     1 'incorrect.')
      nkl=-nkl
      return
      end
c***************
      subroutine calc(kk,mm,short,zdel)
c calculate exact 3d magnetic values
      common /tjcal/tjx(50),tjy(50),tjz(50),mtest
      common/fpts/xp(1000),yp(1000),zp(1000)
      common/field/v(1000,5),r(1000),g(1000,3)
      common /trans/ inpt,iout,jin,iprint
      common /bod/ dstnc(900),ifornt(50),zee(50),zm(50)
      common /side/xb(900),yb(900), ll(50)
      dimension s(6)
c  part  of shortest length of body edges and thinnest layer.
      stest=short*2.0e-4
      ztest= zdel*2.0e-4
      ztem= 10.0*ztest
      do 250 k=1,kk
      xf =xp(k)
      yf =yp(k)
      zf=zp(k)
      delx=0.0
      dely=0.0
      delz=0.0
```

```
      do 10 j=1,5
   10 s(j)=0.0
      nend=0
c number of bodies is mm
      do 230 m =1,mm
      zf1=zee(m)
      zf2= zm(m)
      z1=zf1-zf
      z2=zf2-zf
      if ( abs(z1) .lt. ztest) z1=0.0
      if ( abs(z2) .lt. ztest) z2=0.0
      fact= 1.0
      zz=z1+z2
      z1= abs(z1)
      z2= abs(z2)
      if (zz) 20,230,30
c case where most of mass is above fieldpoint
   20 fact=-1.0
   30 if (z1 .lt. z2) go to 40
      dtest=z2
      z2=z1
      z1=dtest
   40 lk=ll(m)
      ns=nend+1
      nend=ns+lk-1
      z1s=z1*z1
      z2s=z2*z2
      aa1=0.0
      aa2=0.0
      aa3=0.0
      aa5=0.0
      aa6=0.0
      n=ns
      x1=xb(n)-xf
      y1=yb(n)-yf
      if ( abs(x1) .lt. stest) x1=0.0
      if ( abs(y1) .lt. stest) y1=0.0
      r1s=x1*x1+y1*y1
      r1z1= sqrt(r1s+z1s)
      r1z2= sqrt(r1s+z2s)
      r1sf=r1s
      rz1f=r1z1
      rz2f=r1z2
      x1f=x1
      y1f=y1
c number of line segments (vertical sides) or vertices =lk-1
      do 190 l=1,lk
      nm=n
      if (l .ne. lk) go to 50
      x2=x1f
      y2=y1f
      r2s=r1sf
      r2z1=rz1f
      r2z2=rz2f
      go to 60
   50 n=n+1
      x2=xb(n)-xf
      y2=yb(n)-yf
      if ( abs(x2) .lt. stest) x2=0.0
      if ( abs(y2) .lt. stest) y2=0.0
```

```
      r2s=x2*x2+y2*y2
      r2z1= sqrt(r2s+z1s)
      r2z2= sqrt(r2s+z2s)
   60 dx= x2- x1
      dy= y2- y1
      dds= dx* dx+ dy* dy
      if (dds .le. 0.0  ) go to 180
      ds=dstnc(nm)
      cost=dy/ds
      sint=dx/ds
      sncos=sint*cost
      coss=cost*cost
c     sins=sint*sint
      p=(x1*y2-x2*y1)/ds
c test if perpendicular distance to line exceeds 1/5000 segment length
      if ( abs(p) .gt. stest) go to 70
      p=0.0
   70 d1=  x1*sint+y1*cost
      d2=  x2*sint+y2*cost
      dd1= abs(d1)
      dd2= abs(d2)
      if (dd1       .lt. stest) d1=0.0
      if (dd2       .lt. stest) d2=0.0
      d12=d1*d2
c  log(r+d) terms
      if (0.0    .eq. p) go to 140
   80 ph=p*p+z1s
      rdz1=r1z1+d1
      if (d1 .ge. 0.0  ) go to 100
      if (d2 .gt. 0.0  ) go to 90
      al=(r2z1-d2)/(r1z1-d1)
      go to 160
   90 dtest=ph/(d1*d1)
      if (dtest .lt. 0.01  ) rdz1=dd1*0.5  *dtest*(1.0  -0.25  *dtest)
  100 rdz2=r2z1+d2
      if (d2 .ge. 0.0  ) go to 110
      dtest=ph/(d2*d2)
      if (dtest .lt. 0.01  ) rdz2=dd2*0.5  *dtest*(1.0  -0.25  *dtest)
  110 al=rdz1/rdz2
      go to 160
  120 write (iout,130) yf,xf,ztem
  130 format (' fieldpoint at x=',f10.2,', y=',f10.2,' is raised',f9.5,
     1 ' distance units (on edge of body)' )
      z1=ztem
      z1s=z1*z1
      r1z1=sqrt(r1s+z1s)
      r2z1=sqrt(r2s+z1s)
      if (1 .eq. 1) rz1f=r1z1
      go to 80
  140 if (0.0    .ne. z1) go to 80
      if (0.0    .ge. d12) go to 120
      if (0.0    .gt. al) go to 150
c both positive, exterior, in plane of edge, depth zero
      al=d1/d2
      go to 160
c both negative, exterior, in plane of edge, depth zero
  150 al=d2/d1
  160 al=alog(al*(r2z2+d2)/(r1z2+d1))
c  log(r+z) terms
      f1=(z2+r2z2)/(z2+r1z2)
```

```
      if (0.0   .lt. r1z1) f1=f1*(z1+r1z1)
      if (0.0   .lt. r2z1) f1=f1/(z1+r2z1)
      f1=alog(f1)
c  arctangent(zg/pr) terms
      t12=0.0
      if (0.0   .eq. p) go to 170
      call tanget(p,z2*d2,r2z2,z1*d2,r2z1,z1*d1,r1z1,z2*d1,r1z2,t12)
  170 aa1=aa1- coss*t12+sncos*f1
      aa2=aa2+sncos*t12+ coss*f1
      aa3=aa3+       cost*g1
c     aa4=aa4- sins*t12-sncos*f1
      aa5=aa5-       sint*g1
      aa6=aa6+t12
  180 x1=x2
      y1=y2
      r1s=r2s
      r1z1=r2z1
      r1z2=r2z2
c end, corner loop
  190 continue
      s(1)=s(1)+fact*aa1
      s(2)=s(2)+fact*aa2
      s(3)=s(3)+fact*aa3
      s(4)=s(4)+fact*aa5
      s(5)=s(5)+fact*aa6
      aa4=-(aa1+aa6)
      if ( mtest    .eq. 0) go to 210
      tjxm=tjx(m)
      tjym=tjy(m)
      tjzm=tjz(m)
      delx=delx+fact*(tjxm  *aa1+tjym  *aa2+tjzm  *aa3)
      dely=dely+fact*(tjxm  *aa2+tjym  *aa4+tjzm  *aa5)
      delz=delz+fact*(tjxm  *aa3+tjym  *aa5+tjzm  *aa6)
      if (ifprnt(m) .eq. 0) go to 230
      delh=0.0
      delt=0.0
      tx=delx
      ty=dely
      tz=delz
      call compon(3,0,0.0,0.0,delh,delt,tx,ty,tz)
      write (iout,200) k,yf,xf,zf,dely,delx,delz,delh,delt,ifprnt(m)
  200 format (i5,3f11.4,5f11.2,i3)
  210 if (ifprnt(m) .lt. 0)
     1 write (iout,220) k,aa1,aa2,aa3,aa4,aa5,aa6,m,p,dy,dx
  220 format(i6,6e15.8, i4,f8.2,2f7.3)
c end, body loop
  230 continue
      do 240 n=1,5
  240  v(k,n)=s(n)
      if ( mtest    .eq. 0) go to 250
      g(k,1)=delx
      g(k,2)=dely
      g(k,3)=delz
c end, fieldpoint loop
  250 continue
      return
      end
c***************
      subroutine unit(unix,uniz,facx,facz)
c conversion of horizontal (x) and vertical (z) distances-3d magnetics
```

```
      character*4 unix,uniz,feet,kilm,mile,kilf,metr,jblank,
     1 lunx,lunz
      common /trans/ inpt,iout,jin,iprint
      data kilm,  feet,  mile,  kilf,  metr,jblank
     1   /'kilm','feet','mile','kilf','metr','    '/
      write (iout,10) unix,uniz
   10 format (5x,'units of distance measurement are in ',a4,
     1 ' and heights are in ',a4)
   20 if (unix .eq. mile) facx=1.609347
      if (uniz .eq. mile) facz=1.609347
      if (unix .eq. feet) facx=3.048006e-4
      if (uniz .eq. feet) facz=3.048006e-4
      if (unix .eq. kilf) facx=3.048006e-1
      if (uniz .eq. kilf) facz=3.048006e-1
      if (unix .eq. kilm) facx=1.0
      if (uniz .eq. kilm) facz=1.0
      if (unix .eq. metr) facx=1.0e-3
      if (uniz .eq. metr) facz=1.0e-3
      if (facx .ne. 0.0) go to 40
      write(iprint,30)
   30 format(" unix specification is incorrect")
      return
   40 if (facz .ne. 0.0) go to 50
      facx=0.0
      return
c multiplying factor for heights to convert to distance units.
c negative sign to compensate for depths + downward later 'calc'.
   50 facz=-facz/facx
      return
      end
c***************
      subroutine tanget(p,zd1,r1,zd2,r2,zd3,r3,zd4,r4,t12)
c arctangent addition with alternating terms negative
      dimension zd(4),r(4)
      r(1)=r1
      r(2)=r2
      r(3)=r3
      r(4)=r4
      zd(1)=zd1
      zd(2)=zd2
      zd(3)=zd3
      zd(4)=zd4
      i=1
      sum=0.0
       tg=0.0
      do 60 j=1,4
      fnum=zd(j)
      if (fnum .eq. 0.0  ) go to 60
c subroutine by-passed for p=0
      f=fnum/(p*r(j))
      if (i .eq. -1) f=-f
      fnum=tg+f
      den=1.0d0-tg*f
      if (den  .eq. 0.0  ) go to 30
       tg=fnum /den
      if (den  .gt. 0.0  ) go to 60
      if (fnum ) 10, 60,20
   10 sum   =    sum-3.141593
      go to 60
   20 sum   =    sum+3.141593
```

```
      go to 60
   30 if (fnum .lt. 0.0  ) go to 40
      sum=sum+1.570796
      go to 50
   40 sum=sum-1.570796
   50 tg=0.0
   60 i=-i
      t12= atan(tg)+sum
      return
      end
c****************
      subroutine cplot(pc,a,nt,jk,title,*)
c jk must be zero for first call of cplot. later calls use revised xb/yb
c              ci---the initial contour value (calculated)
c           pc,dc---the increment in the contour values (read in)
c          a(y,x)---the input array containing the data to be contoured.
c index for equivalent 1d array is n*(x-1)+y
c               m---the  final  value of x-index in a-array
c               n---the  final  value of y-index in a-array
c              limits of plot are a(yi,xi) to a(n,m)
c nx is number of intervals in x-direction (right) less than 41.
c ny is number of intervals in y-direction (down)  less than 41.
      character*18 title
      integer pnt(131),sym(35),c,iy(41),jx(41)
      common /side/xb(900),yb(900), ll(50)
      common/fpts/xf(1000),yf(1000),zf(1000)
      common /contg/ m,n, dx,dy,  xmn,ymn,xmx,ymx
      common /trans/ inpt,iout,jin,iprint
      common /widect/ naxcol
      common /bodk/ mm,nlim,jbody,short,nend
      dimension a(1),xa(900),ya(900),lll(50)
      data sym/'1','2','3','4','5','6','7','8','9','a','b','c','d','e',
     1'f','g','h','i','j','k','l','m','n','o','p','q','r','s','t',
     2'u','v','w','x','y','z'/,ihor/'-'/,ivert/'|'/,        jplus/'+'/
c needs carriage with 132 columns available or change maxcol
      data maxcol,iblank,jex/131,' ','*'/
      if(pc.eq.0.) return 1
      dc=pc
      if (m .lt. 2) go to 650
      if (m .gt. 41) go to 650
      if (n .lt. 2) go to 650
      if (n .gt. 41) go to 650
      maxcol=naxcol+1
      if (naxcol .eq. 0 .or. naxcol .gt. 130) maxcol=131
      nx=m-1
      ny=n-1
      if(jk.eq.0) go to 60
      write (iout,10) title,n,m,nt,ymn,ymx,xmn,xmx,dc
   10 format (///,' expect plot of 'a18,' of',i3,' by',i3,
     1 '(',i3,') arra',
     2 'y. x=',f8.1,' to',f8.1,' y=',f8.1,' to',f8.1,' contoured at',
     3 f7.1,' gammas',//)
      fmin= 1.0e20
      fmax=-1.0e20
      do 20 i=1,nt
      aij=a(i)
      if (aij .gt. fmax) fmax=aij
      if (aij .lt. fmin) fmin=aij
   20 continue
   30 fc=fmin/dc
```

```
      ic=fc
      if(amod(fc,1.).ne.0.) ic=ic+1
      if (ic .lt. 0) ic=ic-1
      jc=fmax/dc
      if (jc .lt. 0) jc=jc-1
      ncont=jc-ic+1
      if (ncont .lt. 36) go to 50
      write  (iout,40) dc
   40 format (5x,'requested contour interval of',f9.4, ' is doubled',
     1 ' because more than 35 contours are produced.')
      dc=2.0*dc
      go to 30
   50 cd=1.0/dc
      ci=dc*ic
c     cf=dc*ic
   60 width=maxcol-1
      ux=nx/width
c number of columns in one grid interval
      fjinc=1.0/ux
      fjincp=0.5+fjinc
      colefm=1.0-fjinc
c maintains true proportions with x expanded to 130 columns
      uy=  dx*ux/(  dy*0.6)
      if (uy .lt. 1.0) go to 80
      uy=1.0/uy
      write  (iout,70) uy
   70 format(5x,'y-distances are exaggerated by',f9.4,' for contouring')
      uy=1.0
c number of columns per unit distance
   80 dlx=fjinc/dx
      fiinc=1.0/uy
      dly=fiinc/dy
      row=1.0
      do  90 k=1,ny
      row=row+fiinc
   90 iy(k)=row+0.5
      lastrw=iy(ny)
      col=1.0
      do  100 k=1,m
      jx(k)=col+0.5
  100 col=col+fjinc
      if (jk .ne. 0) go to 300
c
c     plot body(s) corners
c
      write (iout,110) title,ymn,ymx,xmn,xmx
  110 format (///,' expect plot of 'a18,
     1 'x=',f8.1,' to',f8.1,' y=',f8.1,' to',f8.1,/,
     2 ' y increases to the right and x increases downward',//)
      do 120 i=1,nend
      xa(i)=xb(i)
      ya(i)=yb(i)
  120 continue
      do 130 i=1,mm
      lll(i)=ll(i)
  130 continue
      np=nend
      jr=1
      do 280 irow=1,lastrw
      do 140 l=1,maxcol
```

```
      pnt(1)=iblank
  140 continue
      ie=0
      do 180 i=1,mm
      is=ie+1
      ie=ie+lll(i)
      if(lll(i).eq.0) go to 180
  150 ky=1.5001+dly*(ya(is)-ymn)
      if(ky.ne.jrow) go to 170
      kx=1.5001+dlx*(xa(is)-xmn)
      if(kx.lt.1.or.kx.gt.maxcol) go to 170
      pnt(kx)=sym(i)
      nb=nb-1
      do 160 k=is,nb
      kk=k+1
      xa(k)=xa(kk)
      ya(k)=ya(kk)
  160 continue
      is=is-1
      ie=ie-1
      lll(i)=lll(i)-1
  170 is=is+1
      if(is.le.ie) go to 150
  180 continue
      itest=0
      if (jrow .eq. lastrw) go to 220
      if (jrow .eq.       1) go to 220
      if (jrow .eq. iy(jr)) go to 210
      if (pnt(maxcol) .eq. iblank) pnt(maxcol)=ivert
      if (pnt(      1) .eq. iblank) pnt(      1)=ivert
  190 write  (iout,200) (pnt(k),k=1,maxcol)
  200 format ( 1x,131a1)
      go to 280
  210 itest=1
      jr=1+jr
  220 do 230 k=1,m
      kx=jx(k)
      if (pnt(kx) .eq. iblank) pnt(kx)=jplus
  230 continue
      if (itest .eq. 1) go to 190
      do 240 k=1,maxcol
      if (pnt(      k) .eq. iblank) pnt(      k)=ihor
  240 continue
      if (jrow .eq. lastrw) go to 290
      if(maxcol.eq.131) go to 260
      write(iout,250)
  250 format("1")
      write  (iout,200) (pnt(k),k=1,maxcol)
      go to 280
  260 write(iout,270) (pnt(k),k=1,maxcol)
  270 format("1",131a1)
  280 continue
  290 write  (iout,200) (pnt(k),k=1,maxcol)
      return
c next statement used if only first body corners to be plotted
c otherwise all could be plotted
c nb is total number of corners of first body of model here
  300 nb= ll(1)
      nf=1
      l=0
```

```
  310 do  320 k=nf,nb
      kk=k
      dum=xb(k)
      if (dum .lt. xmn) go to 330
      if (dum .gt. xmx) go to 330
      kx=1.5001+dlx*(dum-xmn)
      dum=yb(k)
      if (dum .lt. ymn) go to 330
      if (dum .gt. ymx) go to 330
      ky=1.5001+dly*(dum-ymn)
      l=l+1
      xa(l)=kx
  320 ya(l)=ky
      go to 350
  330 nf=kk+1
      write  (iout,340) yb(kk),xb(kk)
  340 format (5x,'body point x=',f10.3,' y=',f10.3,' is outside',
     1 ' plot boundary.')
      if (kk .lt. nb) go to 310
  350 nb=l
  360 cm=ci-dc
      do 370 i=1,nt
c changes values inputted as 'a' (restored later).
  370 a(i)=cd*(a(i)-cm)
      jr=1
c loop of rows from top to bottom
      do 580 irow=1,lastrw
      row=1.0+uv*(irow-1)
c major grid index (i) increases once each fiinc rows
      i=row
      ful=row-i
      ind=0
      kt=i-n
      do 380 l = 1,maxcol
  380 pnt(l) = iblank
      coleft=colefm
c loop of grid intervals from left to right
      do 460 j=1,nx
      coleft=coleft+fjinc
      if (jrow .eq. lastrw) go to 390
      kt=kt+n
      kb=kt+n
      dum = a(kt)
      zl = dum + ful*(a(kt+1) -dum)
      dum = a(kb)
      zr = dum + ful*(a(kb+1) -dum)
      go to 400
  390 ind=ind+n
      zl=a(ind)
      kb=ind+n
      zr=a(kb)
  400 den = zr-zl
      if (abs(den) .gt. 0.000001) go to 420
      izc=zl
      fl=izc
      if (fl .ne. zl) go to 460
      left=coleft+0.5
      jrt=coleft+fjincp
c loop of contours in this grid interval
      do 410 l=left,jrt
```

```
410 pnt(1)=sym(izc)
    go to 460
420 if (zl .gt. zr) go to 430
    fmin=zl
    fmax=zr
    go to 440
430 fmin=zr
    fmax=zl
440 lmin=fmin+0.9999
    lmax=fmax
    if (lmax .eq. 0) go to 460
    if (lmax .lt. lmin) go to 460
    ratio=fjinc/den
    do 450 izc=lmin,lmax
    c=coleft+ratio*(izc-zl)
450 pnt(c)=sym(izc)
460 continue
    if (nb .eq. 0) go to 510
    row=jrow
    nf=1
470 do 480 k=nf,nb
    kk=k
    if (ya(k) .eq. row) go to 490
480 continue
    go to 510
490 nb=nb-1
    kx=xa(kk)+0.5
    pnt(kx)=jex
    if (kk .gt. nb) go to 510
    nf=kk
    do 500 k=nf,nb
    kp=k+1
    xa(k)=xa(kp)
500 ya(k)=ya(kp)
    go to 470
510 itest=0
    if (jrow .eq. lastrw) go to 540
    if (jrow .eq.      1) go to 540
    if (jrow .eq. iy(jr)) go to 530
    if (pnt(maxcol) .eq. iblank) pnt(maxcol)=ivert
    if (pnt(     1) .eq. iblank) pnt(     1)=ivert
520 write  (iout,200) (pnt(k),k=1,maxcol)
    go to 580
530 itest=1
    jr=1+jr
540 do 550 k=1,m
    kx=jx(k)
    if (pnt(kx) .eq. iblank) pnt(kx)=jplus
550 continue
    if (itest .eq. 1) go to 520
    do 560 k=1,maxcol
    if (pnt(     k) .eq. iblank) pnt(     k)=ihor
560 continue
    if (jrow .eq. lastrw) go to 590
    if(maxcol.eq.131) go to 570
    write(iout,250)
    write  (iout,200) (pnt(k),k=1,maxcol)
    go to 580
570 write(iout,270) (pnt(k),k=1,maxcol)
580 continue
```

```
  590 write  (iout,200) (pnt(k),k=1,maxcol)
      write  (iout,600)
  600 format(//,              ' symbol',8x,'value (gammas).   y increases'
     1,' to right and x increases downward')
      zc = ci-dc
      do 610 i=1,ncont
      zc=zc+dc
  610 write  (iout,620) sym(i),zc
  620 format(' ',3x,a1,6x,f14.4)
      write  (iout,630)
  630 format(4x,'+',6x,'fieldpoint', 14x,'*',6x,'corner of body')
      do 640 i=1,nt
  640 a(i)=cm+dc*a(i)
      return
  650 write (iprint,660) m,n
  660 format('0','grid of',i4,' by',i4,' (x,y) is too large (41) or ',
     1 'too small for printer contour.')
      return
      end
c***************
      subroutine ptsm(npts,*)
c
c     input data subroutine for user formatted or gridded data
c     r.godson - 3/81
c
      character*4 unix,uniz
      character*8 name,date1
      character*40 idplot
      character*50 ifile,ifile2,cfile
      character*64 id,id1
      character*80 ifmt
      common/mplot/ymin,xmin,ymax,xmax
      common/fpts/xf(1000),yf(1000),zf(1000)
      common/maginp/iplotr,ibody,ifile,ifile2,lsas,dec,fincl,
     1 datum,xscale,dc,unix,uniz,name,height,
     2 ifmt,idplot,sus,rem,d,zi,iobs,icalc,ires
      common/contg/nrow,ncol,dy,dx,yo,xo,xma,yma
      common/field/v(1000,5),r(1000),g(1000,3)
      common /trans/ inpt,iout,iin,iprint
      dimension z(1001),z2(1001)
      data in/10/,in2/11/,nrcmax/1000/
      write(iout,10)
   10 format(//,14x,"observed values",/,
     1 7x,"x",10x,"y",10x,"z",10x,"m",/)
      if(name.eq."gridded") go to 50
c
c     user formatted data
c
      open(in,file=ifile,form="formatted")
      i=1
   20 read(in,ifmt,end=40) yf(i),xf(i),r(i),zf(i)
      write(iout,30) yf(i),xf(i),zf(i),r(i)
   30 format(1x,4f11.4)
      if(xf(i).lt.xmin) xmin=xf(i)
      if(xf(i).gt.xmax) xmax=xf(i)
      if(yf(i).lt.ymin) ymin=yf(i)
      if(yf(i).gt.ymax) ymax=yf(i)
      i=i+1
      go to 20
   40 k=i-1
```

```
      go to 230
c
c     gridded data
c
   50 open(in,file=ifile)
      read(in,end=190) id,ncol,nrow,nz,xo,dx,yo,dy
      if((ncol*nrow).gt.nrcmax) go to 110
      if(dx.eq.0..or.dy.eq.0.) go to 130
      if(nz.ne.1) go to 170
      if(ifile2.eq." ") go to 60
      open(in2,file=ifile2)
      read(in2,end=190) id1,ncol1,nrow1,nz1,xo1,dx1,yo1,dy1
      if(ncol.ne.ncol1.or.nrow.ne.nrow1) go to 90
      if(dx.ne.dx1.or.dy.ne.dy1.or.xo.ne.xo1.or.yo.ne.yo1) go to 90
      if(nz.ne.nz1) go to 90
   60 mcol=ncol+1
      xo1=xo-(2*dx)
      yo1=yo-dy
      k=0
      do 80 i=1,nrow
      call rowin(z,mcol,in)
      if(ifile2.ne." ") call rowin(z2,mcol,in2)
      do 70 j=2,mcol
      k=k+1
      r(k)=z(j)
      if(ifile2.ne." ") height=z2(j)
      zf(k)=height
      yf(k)=xo1+(j*dx)
      xf(k)=yo1+(i*dy)
      write(iout,30) yf(k),xf(k),zf(k),r(k)
   70 continue
   80 continue
      if(xf(1).lt.xmin) xmin=xf(1)
      if(xf(k).gt.xmax) xmax=xf(k)
      if(yf(1).lt.ymin) ymin=yf(1)
      if(yf(k).gt.ymax) ymax=yf(k)
      xma=xf(k)
      yma=yf(k)
      go to 230
   90 write(iprint,100)
  100 format(" parameters on two input standard files do not match")
      go to 260
  110 write(iprint,120) nrcmax
  120 format(" number of grid points .gt. allowed maximum of ",i4)
      go to 260
  130 write(iprint,140)
  140 format(" present implementation does not allow unequal",
     1 " grid intervals")
      go to 260
  150 write(iprint,160) nrcmax
  160 format(" the number of columns * number of rows requested is",
     1 " greater than the maximum allowed of ",i5)
      go to 260
  170 write(iprint,180)
  180 format(" only single precision values allowed")
      go to 260
  190 write(iprint,200)
  200 format(" end of file while reading input data")
      go to 260
  210 write(iprint,220) k,nrcmax
```

```
  220 format(" number of formatted data values",i5,
     1 " is greater than allowed number of ",i5)
      go to 260
  230 npts=k
      if(npts.lt.1) go to 240
      ierr=0
      go to 270
  240 write(iprint,250)
  250 format(" number of field points less than 1")
  260 close(in)
      if(ifile2.ne." ") close(in2)
      return 1
  270 close(in)
      if(ifile2.ne." ") close(in2)
      return
      end
c***************
      subroutine rowin(z,mcol,in)
      dimension z(mcol)
      read(in) z
      return
      end
c***************
      subroutine mag3gplot(npts,*)
c
c     plotting routine for body corners & fieldpoints
c     r.godson - 4/81
c
      external char(descriptors),pplot(descriptors),graph2(descriptors),
     1 xaxis(descriptors),yaxis(descriptors)
      character*4 unix,uniz
      character*8 name
      character*40 idplot
      character*50 ifile,ifile2,cfile
      character*80 ifmt
      common/mplot/xmin,ymin,xmax,ymax
      common/fpts/xf(1000),yf(1000),zf(1000)
      common/maginp/iplotr,ibody,ifile,ifile2,lsqs,dec,fincl,
     1 datum,xscale,dc,unix,uniz,name,height,
     2 ifmt,idplot,sus,rem,d,zi,iobs,icalc,ires
      common /bodk/ mm,nlim,jbody,short,nend
      common /side/xb(900),yb(900), ll(50)
      common /trans/ inpt,iout,jin,iprint
      common/contg/nrow,ncol,dy,dx,yo,xo,yma,xma
      character*1 revx,sym(50)
      dimension xd(2),yd(2),xs(4),ys(4),jdim(1),
     1 ifmt1(2),nscale(5),xxb(900),yyb(900)
      data jdim(1)/0/,nscale/1,0,2,0,2/,ifmt1/"(f10",".1) "/,
     1 nshl/6/,nsvl/10/,nvl/12/,
     2 nx/2/,ny/2/,vexag/1./,
     3 sym/"1","2","3","4","5","6","7","8","9","a","b","c","d","e",
     4 "f","g","h","i","j","k","l","m","n","o","p","q","r","s",
     5 "t","u","v","w","x","y","z",15*"*"/
c
c
c          denver plotting system
c
   10 if(xscale.eq.0.) go to 170
      if(iplotr.eq.1.or.iplotr.eq.4) go to 20
      nxl=(xmax-xmin)/xscale*2.+.5
```

```
      if(nxl.lt.2) nxl=2
      nyl=nxl
      go to 30
   20 nxl=24
      nyl=17
   30 call scale3(xmin,xmax,nxl,xminp,xmaxp,xdel)
      call scale3(ymin,ymax,nyl,yminp,ymaxp,ydel)
      xd(1)=xminp
      xd(2)=xmaxp
      yd(1)=yminp
      yd(2)=ymaxp
      xs(1)=(xmaxp-xminp)/xscale
      ys(1)=(ymaxp-yminp)/xscale
      if(ys(1).gt.18.) ys(1)=18.
      go to 60
   40 nyl=(ymax-ymin)/xscale*vexag*4.+.5
      if(nyl.lt.2) nyl=2
   50 call scale3(ymin,ymax,nyl,yminp,ymaxp,ydel)
      yd(1)=ymaxp
      yd(2)=yminp
      xs(1)=(xmaxp-xminp)/xscale
      ys(1)=(ymaxp-yminp)/xscale*vexag
   60 call pltset(iplotr,xs(4),ys(4),jdim)
      yyss=ys(4)
      xs(2)=0.
      ys(2)=0.
      xs(3)=2.
      ys(3)=2.
      if((xs(1)+xs(3)+.2).gt.xs(4)) go to 70
      if((ys(1)+ys(3)+.2).gt.ys(4)) go to 70
      go to 100
   70 xs(4)=amin1(xs(4),13.)
   80 ys(4)=amin1(ys(4),10.)
   90 xs(3)=1.0
      ys(3)=1.0
      xs(1)=xs(4)-xs(3)-.2
      ys(1)=ys(4)-ys(3)-.2
      go to 110
  100 xs(4)=xs(1)+xs(3)+.2
      ys(4)=ys(1)+ys(3)+.2
  110 call scale(xd,yd,xs,ys,4,icode)
      if(icode.lt.0) go to 150
      call neatl
      call xaxis(xd,yd,xs,xdel,2,.08,"(f10.2)",10)
      call yaxis(yd,xd,ys,ydel,2,.08,"(f10.2)",10)
      if(iplotr.ne.1.and.iplotr.ne.4)
     1 call char(2.,.1,idplot,40,3,.15,0.,0.,0.)
  120 ie=0
c
c     plot body corners
c
      do 140 i=1,mm
      is=ie+1
      ie=ie+ll(i)
      call char(yb(is),xb(is),sym(i),1,0,.08,0.,0.,0.)
      m=0
      do 130 k=is,ie
      m=m+1
      yyb(m)=yb(k)
      xxb(m)=xb(k)
```

```
  130 continue
      xxb(m+1)=xb(is)
      yyb(m+1)=yb(is)
      call line(yyb,xxb,m+1,0,0)
  140 continue
c
c     plot fieldpoints
c
      do 145 i=1,npts
      call char(yf(i),xf(i),"+",1,0,.08,0.,0.,0.)
  145 continue
      call endpt(jdim)
      return
  150 write(iprint,160)
  160 format(" unable to scale plotter")
      go to 190
  170 write(iprint,180)
  180 format(" xscale is zero")
  190 return 1
      end
c***************
      subroutine stdout(isw,z)
c
c     subroutine to create standard file
c     r.godson - 4/81
c
      character*32 id1(2),id2(2)
      character*54 file2
      character*55 file1
      character*56 file3
      dimension z(1)
      common/st/file1,file2,file3
      common/magino/iplotr,ibody,ifile,ifile2,lsqs,dec,fincl,
     1 datum,xscale,dc,unix,uniz,name,height,
     2 ifmt,idplot,sus,rem,d,zi,iobs,icalc,ires
      common/contg/nrow,ncol,dy,dx,xmin,ymin,xmax,ymax
      data iout/11/,nz/1/,dum/0./,
     1 id1/"magpoly calculated values from m",
     2 "odel                            "/,
     3 id2/"magpoly residual values(observed",
     4 "-calculated)                    "/
      if(isw.eq.1) open(iout,file=file1)
      if(isw.eq.2) open(iout,file=file2)
      if(isw.eq.1) write(iout) id1,ncol,nrow,nz,ymin,dx,xmin,dy
      if(isw.eq.2) write(iout) id2,ncol,nrow,nz,ymin,dx,xmin,dy
      is=1
      ie=ncol
      do 10 i=1,nrow
      write(iout) dum,(z(j),j=is,ie)
      is=ie+1
      ie=ie+ncol
   10 continue
      close(iout)
      return
      end
```