MAGDEP - A Computer Program for Finding Depth
To Basement From Total Field Marine Magnetic Profiles

By

Alan K. Cooper

USGS Open File 1974
74-1017

This report is preliminary and has not been edited or reviewed for conformity with Geological Survey standards and nomenclature.
INTRODUCTION

The purpose of the Fortran program MAGDEP is to provide a method for finding the depth to two-dimensional magnetic source bodies when given only the total field magnetic intensity profile. The assumptions made in this depth analysis are that the causitive source body is:

1. A two-dimensional structure
2. Uniformly magnetized
3. A finite or infinite polygon in cross section

Unlike other methods for magnetic depth determinations which assume a given body shape, this technique allows the shape of the source body to emerge during the analysis. The final result is a sequence of spot depths which locate the position and depth to the corners of the magnetic source body. The correct outline of the body is controlled by the shape of the magnetic intensity profile which has been reduced to the pole. This reduced polar profile is also determined by program MAGDEP. Two depth analyses are made by the program; one for thick polygonal bodies and the other for thin (thickness less than depth) or narrow bodies. The determination of the correct magnetic source body shape depends upon a careful interpretation of the results from both depth analyses.
Theoretical Aspects

The method used by program MAGDEP is based on the theoretical work of Nabighian (1972). The reader interested in the details of the theory should refer to his paper; this section will only outline the theoretical aspects which have been applied to program MAGDEP. Nabighian (1972) shows that all bodies of polygonal cross section can be obtained by the superposition of a finite number of magnetized steps. He gives the magnetic attraction for such a magnetized step as $\Delta M(x)$ (in eq. 1) and then uses it as the starting point for the analysis. Differentiating the attraction of the magnetized step with respect to $x$ yields the horizontal derivative and this represents the magnetic anomaly due to a thin infinite sheet: $T(x) = \frac{3}{\delta x} \Delta M(x)$. The vertical derivative of the magnetized step is obtained by differentiating with respect to $y$: $T_1(x) = \frac{3}{\delta y} \Delta M(x)$. A comparison of the two derivatives shows that $T_1(x)$ is the negative Hilbert transform of $T_1(x)$. When one forms the complex analytic signal function from the two derivatives: $A(x) = T(x) - i T_1(x)$ many interesting properties arise. Those properties which are used by MAGDEP include:

1. The amplitude $|A(x)|$ is a symmetric function with respect to $x=0$.
2. The complex function $A(x)$ has simple poles at each corner of the polygon.
3. The bell-shaped function $a(x) = |A(x)|^2 = T(x)^2 + T_1(x)^2$ has the property that its half-maximum half-width is equal to the depth to the polygon corner.
From this brief introduction to the theory, the procedure for computing the depths to the magnetic source bodies becomes more defined. In the depth analysis for thick source bodies:

1. The total field intensity profile (residual values) is differentiated with respect to \( x \) to yield \( T(x) \).

2. A fourier transformation, spectrum modification, and inverse fourier transformation is performed to obtain the complex analytic signal function: \( A(x) = T(x) - iT_1(x) \).

3. The amplitude square of the signal function is computed to give: \( a(x) = T(x)^2 + T_1(x)^2 \). The deconvolution of the \( a(x) \) curve into a finite number of bell-shaped symmetric curves gives the locations and depths to the corners of the magnetic source body.

The depth analysis for thin (thickness less than depth) magnetic source bodies is accomplished by the same procedure with the exception that in step 1, the total field intensity profile is not differentiated.

In the general case, where the relief of the magnetic source body is irregular, each corner of the body will contribute a symmetric bell-shaped component to the \( a(x) = T(x)^2 + T_1(x)^2 \) curve. To identify each of these symmetric components, it is necessary to deconvolve the \( a(x) \) curve using a nonlinear least square optimization routine. The procedure used by MAGDEP was derived by Marquardt (1963) and is similar to the technique described by Johnson (1969). See Appendix I for more complete details. The amplitude square curve can also be written as: \( a(x) = \frac{x^2}{h^2 + (x-c)^2} \) where \( x \) is a proportional to the magnetic susceptibility and the dip of the polygon side, \( h \) is the depth to the polygon corner, \( x \) is the dependent position variable, and \( c \) is
the location of the polygon corner along the profile. Each corner of
the magnetic source body will contribute a bell-shaped symmetric curve
controlled by the three parameters $\alpha$, $h$, and $c$. At the start of the
least square optimization process, initial values must be assigned to
each of the parameters; the 'goodness of fit' of the final solution
depends to a large extent upon the quality of the initial guesses.
Program MAGDEP uses the following procedure for locating the symmetric
bell-shaped curves and determining the initial values for each of the
parameters (no. of parameters = 3 x no. of bell-shaped curves):

1. All local maxima of the $a(x)$ curve are checked to see if:
   a. The $a(x)$ value is larger than an arbitrary cutoff value.
   b. The second derivative of the $a(x)$ curve is large enough.
      (controlled by maximum expected source depth $Z_{MAXKM}$)

2. If step 1 is true, the initial parameter values are found
depending upon:
   a. $h$ (depth): find the local half-maximum half-width from
      the $a(x)$ curve
   b. $\alpha$: use the $a(x)$ value at the peak of the bell-shaped
curve ($=\frac{\alpha^2}{h^2}$)
   c. $c$ (position): use the x-position of the peak of the bell-
      shaped curve

After the initial symmetric curves are located, the least square routine
is used to obtain a trial solution. The residual values ($a(x)$ -
computed values) are then examined by a similar procedure to determine
whether more bell-shaped curves are present. The final solution is
the 'best fit' of the $a(x)$ curve by the least squares computed curve
(using all bell-shaped curves). Depths and locations of the magnetic
source body corners are derived from this final least squares approxi-
imation.
Using the Program

Program MAGDEP has been designed so that it can be used in one or two sections depending upon the desires of the user. The first section computes the \( a(x) \) curve and the magnetic profile reduced to the pole; the second section does the least squares analysis of the \( a(x) \) curve (see theory section). Experience working with MAGDEP has shown that section 2 generally is from 5 to 15 times more time consuming (and expensive) than section 1 and under some circumstances, where high precision is not required, satisfactory results can be obtained by omitting section 2 and using only a visual inspection of the \( a(x) \) curve. A second point is that since computation time in the least squares routine increases rapidly with the addition of each bell-shaped curve, the user should consider examining the \( a(x) \) curve for each new profile prior to using the least squares routine. Knowing the approximate number of bell-shaped curves present will not only save computation time but will also reduce the chances of obtaining a poor solution because too few bell-shaped curves were specified by the user.* With these notes of warning, the following procedure is suggested for the new user of MAGDEP:

1. Read the theory section in this program description of MAGDEP to become familiar with the terminology used here
2. Look over the program output given in the example (Appendix II)
3. Use the program thru section 1 to obtain the plot of the \( a(x) \) curve
4. Examine the \( a(x) \) curve to get a feel for the number and type of bell-shaped curves present. The shapes and sizes of the curves present will give a rough estimate of the depths to the body

*Example of computation time for 100 data points on an IBM 360/70 system:
Section 1 only: 4 sec. execution
Section 2 only: thick body analysis only, 7 bell-shaped curves, 4 LSQR iterations: 20 sec. execution
5. Refer to the do's and don'ts section for help in deciding whether the profile should be broken into smaller segments for analysis.

6. Resubmit those parts of the profile which require precision depth estimates using the least squares routine.

Input constants required: (must be right-justified)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>(required)</td>
<td>1-80</td>
<td>(required)</td>
<td>1-10</td>
<td>SPDKT</td>
<td>11-20</td>
<td>ID</td>
</tr>
<tr>
<td>(20A4)</td>
<td></td>
<td>(required)</td>
<td></td>
<td>: ship speed in knots</td>
<td></td>
<td>: data interpolation flag</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(F10.1,2I10)</td>
<td></td>
<td>=1 want data to be interpolated (must use card 4)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>=2 use input data as is. (data must be evenly spaced)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>=1 compute a(x) curve and profile reduced to the pole (optional)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>=2 do this and use the least squares routine</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>=1 Thick bodies only</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>=2 Thin bodies only</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>=3 Both thick and thin bodies</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>=0 do not want the reduced profile</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>=1 do want the reduced profile (must use Card 5)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Cols: 31-40 INPLT : Request for the initial plot containing the a(x) curve, the initial parameter curve, and the residual total field profile. (Recommended as being useful)

=0 do not want the plot
=1 do want the plot

41-50 CUTAX : A factor which is to be multiplied by the internal value of the a(x) curve cutoff point. This is used in determining whether local maxima of the a(x) curve are to be considered as bell-shaped curves. (Recommend using CUTAX = 1 on first run)

Card 4: (use only if ID = 1)
(2F10.2, F10.1, I10) Cols: 1-10 XSHR : Starting point for the data interpolation in hours of elapsed time from the start of the profile (must be \( \geq 0 \))

11-20 XEHR : Ending point for the data interpolation in hours of elapsed time from the start of the profile (can be larger than the time of the last data point)

21-30 DXMIN : New time increment at which data is to be sampled (in minutes) (recommend DXMIN = 2 or 5 for ships magnetic profiles)

31-40 NP : Maximum number of points desired to be in the data set used for the depth analysis (must be less than or equal to 200 - the absolute maximum number of data points allowed)
Card 5: (use only if IPOLE = 1)
(2F10.1) Cols: 1-10 FINC : Inclination of the earth's magnetic field where the magnetic profile was measured (in degrees)
Cols: 11-20 ANG : Angle between magnetic north and the ship's heading (in degrees where 180 > ANG > 0)

Card 6: (use only if ISECT = 2)
(4I10,F10.3) Cols: 1-10 NRES : The number of residual checks which are to be made during the execution of section 2. The normal cycle is: 1st sqr.-residual check - 1st sq - residual check - 1st sq - etc., depending upon NRES. (Recommend initial use of NRES=1)

11-20 NPEAKD : Thick body analysis:
The number of bell-shaped curves which are to be used by the least square routine during its first series of iterations. If NPEAKD=0, all bell-shaped curves (max. of 10) with peak values greater than(AXCUT*internal cutoff) will be used. NPEAKD must be less than or equal to 15. (Recommend either: 1. use the two step procedure described above to determine NPEAKD or 2. use NPEAKD=0)

21-30 NPEAKT : Thin body analysis: (same description, requirements, and recommendations as for NPEAKD)

31-40 IOUT : Flag to control the extra output for the values for: 1. the least square approximation and 2. the residual values (a(x) - computed) =0 no output is desired =1 output is desired (Recommend initial use of IOUT=1)
Cols: 41-50   CUTRSD : A factor which is to be multiplied by the internal value of the residual curve cutoff point. This is an arbitrary criteria used in determining whether a local maxima in the residual curve is large enough to allow another bell-shaped curve to be added. \( \text{(Recommend using CUTRSD}=1, \text{on the first run)} \)

Card 7: (use only if ISECT=2)

(8I10)   Col: 1-10   ITIR(1) : The number of iterations of the least squares routine which are to be made during the first call of MARQL. \( \text{(MARQL is the name used for the least squares routine)} \)\( \text{(recommend using ITIR}(I) \leq 3 \text{ on first run)} \)

11-20   ITIR(2) : The number of iterations of the least squares routine which are to be made after the first residual check and during the second call of MARQL. \( \text{(as required up to ITIR}(\text{NRES}+1)) \)

Card 8 to end of data set: Input of time and total field values

\( (T60,I2,T6,I3,I2,2F2.0,F5.0) \)

6-8   ID : Julian day (sequential starting from 1 on January 1)

9-10   IH : Hour (based on 24 hour day)

11-12   AM : Minute

13-14   AS : Second
Cols: 15-19  **TOTMAG** : Measured total field intensity value. (The program will subtract a constant field (= average field value) from all measured total field values. If residual field intensity values are used, a minor change will have to be made to the main program (those statements directly following the data interpolation section))

60-61  **IY** : Year

Note: Data cards of a different format can readily be used by changing statement 100 in Subroutine INPUT to fit the new data format.
<table>
<thead>
<tr>
<th>Name</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name 4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Some Do's and Don'ts

At the time of writing of this 'suggestion' section, the program MAGDEP has had limited use; consequently, many of the following suggestions may eventually become obsolete or possibly even misleading. It is hoped that helpful additions and deletions will be made to this description as they are discovered. The current state of the art in computers and algorithms to do non-linear least squares requires a great deal of computing time and memory storage space. It is thus advantageous for the user of MAGDEP and other similar programs, to take a few extra precautions to insure a less expensive and higher quality solution on the first attempt. The two step procedure, as described earlier, is a worthy precaution and consequently a highly recommended DO. Some observations concerning the operation of the least squares routine, which might be useful are:

Least Squares

1. The better the initial guesses to the parameter values, the faster the routine will reach a good solution. It should be noted that the two techniques described by Nabighian (1972) for determining trial depth values (Eqs. 19 and 21) were attempted and found to give adequate results. Unfortunately, occasional bad depth estimates (probably the result of noise in the spectral transformations) were encountered which adversely affected the least square solution. The current method for finding trial depths, although basically simple, has provided more reliable trial values for the least square solution.

2. The least squares routine converges most rapidly for the amplitude of the symmetric curve, less rapidly towards the width of the curve, and least rapidly for the position of the curve.
3. The rapid amplitude convergence also means that smaller amplitude curves adjacent to or on the flanks of large amplitude curves, are often wiped out during the first series of iterations of the least square routine. These small peaks seem to have a better chance for survival after the large peaks are 'locked in'. The residual analysis has been designed to restore these small peaks and other less obvious ones so that they will be included in the depth analysis.

4. To improve a solution when there is a region of small amplitude symmetric curves near an area of large amplitude peaks, it is often best to break the profile into two parts and analyze them separately.

5. Since the number of multiplications for I iterations of the least square routine is approximately equal to $13 \times I \times K \times N$, where $K$ is the number of parameters (i.e., $3 \times$ number of bell-shaped curves) and $N$ is the number of data points, it is advantageous to either:
   a. Use the largest possible sample interval which does not jeopardize the depth resolution or
   b. use a smaller sample interval for only a short time period. Both possibilities will minimize the total number of data points.

A list of some other do's and don'ts includes:

<table>
<thead>
<tr>
<th>DO</th>
<th>DON'T</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Input and Interpolation</td>
<td></td>
</tr>
<tr>
<td>1. Do modify the subroutine INPUT to accommodate a different data input format if necessary.</td>
<td>1. Don't input residual magnetic values without first changing three statements in the main program.</td>
</tr>
<tr>
<td>2. Do interpolate raw data to the largest sample interval which will not affect the depth resolution. (max resolution is probably about 1/2 the sample interval)</td>
<td>2. Don't use unevenly sampled data without using the data interpolation option.</td>
</tr>
<tr>
<td>3. Do use raw data sampled at the closest interval possible. It is far better to interpolate to a larger sample interval from raw data than to use fewer data cards.</td>
<td>3. Don't use more than 200 input raw data cards or request more than 200 interpolated data points.</td>
</tr>
<tr>
<td>4. Do check the sample listing of the raw data and interpolated data to insure against any input reading errors.</td>
<td></td>
</tr>
</tbody>
</table>
Computing $a(x)$ and initial deconvolution

**Do**

1. Do request initial plot on the first run for any new data set (INPLT=1).
2. Do use a two step procedure and for the first step let the program decide the number of bell-shaped curves to use (let NPEAKT and NPEAKD = 0).
3. Do consider carefully what the maximum source depth might be (+2 km, i.e., don't guess wildly and set to 100).
4. Do examine the plot of the $a(x)$ curve carefully for all symmetric curves which might be detected visually.
5. Do break up a data profile into smaller segments or change the sample interval if the $a(x)$ curve has groups or areas of high and low amplitude peaks.

**Don't**

1. Don't request more than 15 bell-shaped curves (i.e., don't let NPEAKD or NPEAKT = 16 or more).
2. Don't request a large number (>7) of bell-shaped curves on a first run, unless the least squares routine will not be used.
3. Don't set the cutoff for the $a(x)$ curve too low (i.e., AXCUT); otherwise noise becomes a problem.
4. Don't be afraid to request a plot of the profile reduced to the pole - it's cheap.
5. Don't worry if the initial parameter curve looks like a poor fit to the $a(x)$ curve - least squares does a good job.

Least squares and residual analysis

**Do**

1. Do remember that the time required for each least squares analysis will be proportional to the number of multiplications performed (about 13 \times \# iter \times \# data pts \times \# parameters).

**Don't**

1. Don't use a cutoff value for the residuals which is too low (i.e., RSDCUT), it may cause a false symmetric curve to be added which may affect the least squares analysis. The internal cutoff has worked all right so far (RSDCUT=1).
2. Do use at least one residual analysis.

3. Do use enough iterations (min. of 3) on the first least square analysis to allow the residual values to develop adequately.

4. Do expect that the least square solution may 'seek' or oscillate through the correct value from iteration to iteration.

2. Don't be discouraged if a seemingly poor fit is obtained after only a few iterations. The program will inform you if major problems arise and will give words of encouragement and advice.
Flow Diagrams

The detailed flow diagrams for the entire program and some of the more important subroutines are included for two reasons:

1. To give the interested (or confused) user an opportunity to look into the 'black box' of MAGDEP

2. To assist in trouble shooting should difficulties arise

For the purpose of illustration, the entire program has been broken into the two sections which were described previously. The physical layout of the program is such that this separation into two sections is not readily apparent. Those subroutines which do not have a detailed flow diagram included here are indicated by dashed boxes; those with flow diagrams are indicated by double solid boxes. The name of the subroutine lies in one corner of the outer most box.

Explanation for flow diagrams:

- Start and stop
- Return from subroutines
- Go to
- Computational procedure
- Input and output
- Decision

- Description of a subroutine which has a detailed flow diagram
- Description of a subroutine which does not have a detailed flow diagram
Flow diagram of entire program MAGDEP:

1. **Start Program MAGDEP**
   - Read control parameters
   - Input data
   - Data interpolation and derivatives
   - Subtract regional field
   - Write control parameters

2. **A**
   - Use % x TOTMags for analysis
   - Compute analytic signal function
   - Compute a(x) curve
   - Compute a(x) and residual cutoff values
   - Write constants found by the computer
   - Write profile reduced to the pole
   - Compute profile to pole
   - Plot TOTMags and pole mags

3. **B**
   - Write heading for initial plots
   - Write a(x) values
   - Find symmetric components of the a(x) curve
   - Compute profile to pole
   - Compute initial parameter curve and residual curve
   - Initial plots
   - Stop
   - Go to another body analysis
   - Plot a(x) curve, initial parameter curve, TOTMags
   - Compute correlation coefficients for each peak

4. **C**
   - Least squares parameters rejected
   - All parameters rejected
   - All residual analysis finished
   - Check residuals
   - Go to another body analysis
   - Stop
Flow diagrams of important subroutines:

**SUBROUTINE INPUT**

**SUBROUTINE INTERP**

**SUBROUTINE HILBRT**
SUBROUTINE SYMPR3

START SYMPR3

FIND LOCAL MAXIMA FOR \( a(x) \) CURVE

CHECK \( a(x) \) VALUE CUT-OFF

YES

NO

COMPUTE 2ND DERIV.

THEORETICAL DEPTH

YES

NO

TWO DEPTHS MAX.

DEPTH

YES

NO

SAVE PEAKS BY:

+ INCREASING DISTANCE

- DECREASING \( a(x) \) VALUE

Determine NO. OF PEAKS TO USE

YES

NO

COMPUTE INITIAL DEPTHS

COMPUTE OTHER PARAMETERS

WRITE PARAMETER VALUES FOR INITIAL PEAKS FOUND

RETURN

SUBROUTINE MARQ1

START MARQ1

COMPUTE SUM OF SQUARES OF RESIDUALS

COMPUTE 1 ITERATION OF MARQUARDT NON-LINEAR LST SQR ALGORITHM

NO

YES

PLOT \( a(x) \) CURVE, PARAMETER CURVE, TANGENT CURVE

WRITE RESULTS FROM THIS ITERATION

NO

YES

BAD PARAMETER VALUES

REJECT BAD CURVES

REARRANGE PARAMETER ARRAY

WRITE ALL REJECTED PARAMETERS

RETURN

ALL PARAMETERS REJECTED

WRITE COR. COEFF. FOR EACH PEAK

FIND LOCAL MAXIMA FOR \( a(x) \) CURVE

WRITE FINAL VALUES OF PARAMETERS

NO

YES

ALL ITERATIONS COMPLETED

NO

YES

ANY PEAKS REJECTED DURING ITERATIONS

WRITE EXPLANATION

NO

RETURN
SUBROUTINE RSDCHK

START RSDCHK

1. FIND LOCAL MAXIMA IN RESIDUAL CURVE
2. CHECK RESIDUAL VALUE > CUTOFF
   - YES: COMPUTE 2ND DERIV. THEORETICAL DEPTH
   - NO: DEPTH > MAX. DEPTH
3. NO: PEAKS > 13
   - YES: GO TO C

   1. COMPUTE INITIAL DEPTH OF NEW TEMPORARY SYMMETRIC PEAKS
   2. COMPUTE OTHER PARAMETERS
      - IS THIS THE FIRST NEW PEAK
         - YES: GO TO B
         - NO: DO PREVIOUS NEW PEAK CORRESPOND WITH OLD PEAK
            - NO: SAVE PREVIOUS NEW PEAK AND ADD TO LIST OF OLD PEAKS
            - YES: DISCARD PREVIOUS NEW PEAK

   3. IS THIS THE LAST RESIDUAL VALUE
      - YES: DOES LAST NEW PEAK CORRESPOND WITH AN OLD PEAK
          - NO: SAVE LAST NEW PEAK AND ADD TO LIST OF OLD PEAKS
          - YES: DISCARD LAST NEW PEAK
   - NO: REARRANGE THE PARAMETER ARRAY

   4. COMPUTE NEW PARAMETER CURVE
   5. COMPUTE COR. COEFS. FOR EACH PEAK
   6. INITIAL PLOTS DESIRED
      - NO: RETURN
      - YES: WRITE HEADING FOR PLOT

COMPUTE OTHER PARAMETERS

WRITE:
- NO. PEAKS ADDED
- NO. PEAKS DELETED
- WRITE PARAMETER VALUES FOR ALL PEAKS SAVED

DISCARD PREVIOUS NEW PEAK

DISCARD LAST NEW PEAK

WRITE:
- NO. PEAKS ADDED
- NO. PEAKS DELETED

Re-ARRANGE THE PARAMETER ARRAY

WRITE PARAMETER VALUES FOR ALL PEAKS SAVED

RETURN
Programming Considerations

Listed below are suggestions for making minor changes to program MAGDEP which might be desired at some later time:

A. To change input data format:
   1. Modify statement 100 in subroutine INPUT.

B. To use residual magnetic values as input:
   1. Modify the statement immediately preceeding statement 120 in the main program. Do not subtract AVG from an input (or interpolated) values.

C. To increase the number of data points which may be used, up to N, modify the following statements: Note: wherever N appears, use the desired integer number for N.
   1. Main Program:
      COMMON TIME (N), B(N), C(N), D(N), E(N), F(N)
      COMMON /PRMCRV/XF(N)
      DIMENSION AX(N), TMAG(N), DTMAG(N)
      COMPLEX CAX (next even power of 2 above N)
   2. Subroutine INTERP
      COMMON XAXIS(N), B(N), C(N), D(N), E(N), XY(N)
   3. Subroutine SPLINE
      REAL*4 H(N), SPP(N)
   4. Subroutine HILBRT
      COMPLEX AXM (next even power of 2 above N)
   5. Subroutine POLRED
      COMMON X(N), XX(4N)
6. Subroutine SYMPR3
   COMMON X(N), Q(N), T(N), XX(3N)

7. Function DEP
   COMMON X(N), XX(5N)

8. Subroutine PEAKFT
   COMMON X(N), Y(N), C(N), D(N), DIFF(N), XX(N)
   COMMON /PRMCRV/ F(N)

9. Subroutine PLOT
   COMMON X(N), XX(5N)
   DIMENSION U(1), V(1), W(1), Y(N,3), SYM(3)

10. Subroutine PRLPLT
    DIMENSION SYM(NY), ZY(11), GRAPH(101), Y(N,3), X(1)

11. Subroutine MARQ1
    COMMON X(N), TM(N), SS(N), TEM(N), DIFF(N), XX(N)
    COMMON /PRMCRV/ F(N)
    DIMENSION P(N,45), Y(1)

12. Subroutine DERIV
    COMMON X(N), XX(5N)
    DIMENSION P(N,45)

13. Subroutine FUNCT
    COMMON X(N), XX(5N)

14. Subroutine PRMCHK
    COMMON X(N), TT(N), TM(N), XX(3N)

15. Subroutine RSDCHK
    COMMON X(N), Q(N), REJ(N), W(N), R(N), F(N)
    COMMON /PRMCRV/ FF(N)

16. SUBROUTINE INPUT
    IF(N.GT N) Go to 10
D. To increase the number of parameters allowed to KMAX (i.e., 3 times the no. of symmetric curves) change the following: Note: wherever KMAX appears, use the desired integer number for KMAX.

1. Main Program
   COMMON /SYM/ PARM(KMAX), NPAR

2. Subroutine SYMPR3
   COMMON /SYM/ B(KMAX), K
   DIMENSION AX(1), IDIST(KMAX), IAX(KMAX)
   KMAX='KMAX' (directly below statement 104)

3. Subroutine PEAKFT
   COMMON /SYM/ B(KMAX), K

4. Subroutine MARQI
   COMMON /SYM/ B(KMAX, K)
   DIMENSION P(200,KMAX)
   DIMENSION GC(KMAX), GS(KMAX), GU(KMAX), D(KMAX), DS(KMAX),
           DG(KMAX), Bl(KMAX), A(KMAX, KMAX), AS(KMAX, KMAX)

5. Subroutine DERIV
   COMMON /SYM/ B(KMAX), K
   DIMENSION P(200, KMAX)

6. Subroutine PRMCHK
   COMMON /SYM/ B(KMAX), K

7. Subroutine RSDCHK
   COMMON /SYM/ B(KMAX), K
   KMAX='KMAX' (directly after statement 108)

E. The following subroutines have been written so that they may be used outside of MAGDEP; the only changes required are to dimension or common statements:
INTERP (SPLINE)

SPLINE

HILBRT (FORK, FORSET, FORMOD)

FORSET

FORK

FORMOD

CORLAT

MINMAX

PLOT (MINMAX, PRLPLT)

PRLPLT

Note: other subroutines which are called by the main subroutine are enclosed in parenthesis.
APPENDIX II - Program Listing for MAGDEP
FORTRAN IV LEVEL G USGS OPEN FILE 74-1017

THE PROGRAM IS DESIGNED TO PROVIDE A METHOD FOR FINDING THE DEPTH TO CORNERS OF A TWO-DIMENSIONAL MAGNETIC SOURCE BODY WHEN GIVEN ONLY THE TOTAL FIELD MAGNETIC INTENSITY PROFILE. ASSUMPTIONS MADE ABOUT CAUSITIVE SOURCE BODY:

1. A TWO-DIMENSIONAL STRUCTURE
2. UNIFORMLY MAGNETIZED
3. A FINITE OR INFINITE POLYGON IN CROSS SECTION

TWO DEPTH ANALYSES ARE MADE BY THE PROGRAM:
1. THICK BODIES AND
2. THIN BODIES (THICKNESS LESS THAN DEPTH).

THE DETERMINATION OF THE CORRECT SOURCE BODY SHAPE DEPENDS UPON A CAREFUL INTERPRETATION OF RESULTS FROM BOTH DEPTH ANALYSES. THE TECHNIQUES USED BY THE PROGRAM ARE BASED ON THE WORK OF:

1. THEORY: NABIGHIAN, MISAC N., GEOPHYSICS, V. 37, P. 507-517, 1972
2. LEAST SQUARES: MARGERITZ, DON W., J. CUR. SOC. APPL. MATH, V. 11, P. 431-441, 1963
3. JOHNSON, WILLIAM W., GEOPHYSICS, V. 34, N. 1, P. 65-74, 1969

FOR FURTHER DETAILS SEE DOCUMENTATION AVAILABLE FROM OFF. MARINE GEOLOGY.

USE OF PROGRAM: THE PROGRAM CAN BE USED IN ONE OR TWO SECTIONS WHICH:

SECTION 1: COMPUTES A(X) CURVE, MAG PROFILE REDUCED TO POLE, INITIAL TRIAL VALUES FOR PARAMETERS OF THE SYMMETRIC CURVES (1 CURVE PER BODY CORNER AND 3 PARAMETERS PER CURVE(A, H, C))

SECTION 2: LEAST SQUARE ANALYSIS, RESIDUAL ANALYSIS

GENERAL PROCEDURE TO FOLLOW WHEN EXAMINING A NEW DATA PROFILE:

1. USE PROGRAM THRU SECT 1 TO SEE PLOT OF A(X) CURVE - USE 5 MIN VALUES
2. EXAMINE A(X) CURVE FOR NO. AND TYPES OF SYMMETRIC CURVES - THE HALF-MAX HALF-WIDTH IS DEPTH TO BODY CORNER. CHOOSE GOOD VALUE FOR CUTAX.
3. BREAK PROFILE INTO SMALLER SEGMENTS: A. TO SEPARATE AREAS OF LARGE AND SMALL PEAKS B. TO DELETE OR LOOK CLOSER AT 'FLAT' AREAS C. TO STAY WITHIN THE MAX. OF POINTS (INPUT OR INTERPOLATION: MAX=200)
4. COMPUTE THRU SECTION 2 USING ITIR(162)=SMALL NO. DO ONLY 1 RESIDUAL ANALYSIS. AFTER RUN, CHECK VALUE OF CUTRSD
5. MAKE FINAL RUN BASED UPON RESULTS FROM STEPS 1 THRU 4

NOTE: THE EXECUTION TIME FOR THE LEAST SQUARES ROUTINE(SECT 2) IS USUALLY FROM 5 TO 15 TIMES GREATER THAN FOR SECT 1. UNDER SOME CIRCUMSTANCES, WHERE HIGH PRECISION IS NOT REQUIRED, SATISFACTORY RESULTS CAN BE OBTAINED BY OMITTING SECT 2 AND USING ONLY A VISUAL INSPECTION OF THE A(X) CURVE.

INPUT CONSTANTS REQUIRED:

CARD 1: HEADING FORMAT(20A4) REQUIRED
CARD 2: REQUIRED FORMAT(F10.1,2I10)
SPOKT = SHIP SPEED IN KNOTS
C ID = USER DESIRES DATA INTERPOLATION = 1 YES (MUST USE CARD 4)
C = 2 NO (DATA MUST BE EVENLY SPACED)
C
C ISECT = PROGRAM SECTIONS TO USE = 1 THRU SECT 1 = 2 THRU SECT 2
C
C CARD 3: REQUIRED FORMAT (110, F10.1, 210, F10.3)
C IBOD = BODY TYPES TO ANALYSE FOR: = 1 THICK = 2 THIN = 3 BOTH
C
C ZMAXKM = MAX, ESTIMATE) DEPTH TO SOURCE BODY IN KM
C
C IPOL = WANT PROFILE REDUCED TO POLE: = 0 NO = 1 YES (USE CARD 5)
C
C INPLT = WANT INITIAL PLOT (CONTAINS A(X), INITIAL PARAMETER, AND
C RESIDUAL TOTAL FIELD PROFILE): = 0 NO = 1 YES
C
C CUTAX = FACTOR WHICH WILL BE MULTIPLIED BY THE INTERNAL VALUE FOR
C THE A(X) CURVE CUTOFF POINT AND USED TO DETERMINE WHETHER
C LOCAL MAXIMA IN A(X) CURVE WILL BE KEPT AS SYMMETRIC CURVE
C (ON FIRST RUN USE CUTAX=1.)
C CARD 4: USE ONLY IF ID=1 FORMAT (3F10.2, 110)
C XSHR = START INTERPOLATION--HRS OF ELAPSED TIME FROM START OF LINE
C XEHR = END INTERPOLATION--HRS OF ELAPSED TIME FROM START OF LINE
C DXMIN = TIME INCREMENT AT WHICH DATA IS TO BE INTERPOLATED IN MIN
C
C CARD 5: USE ONLY IF IPOL=1 FORMAT (2F10.1)
C
C FINC = INCLINATION OF EARTHS MAG FIELD WHERE PROFILE WAS MEASURED
C
C ANG = ANGLE BETWEEN MAG NORTH AND SHIP HEADING (0) ANG 180 DEG
C CARD 6: USE ONLY IF ISECT = 2 FORMAT (4110, F10.3)
C
C NRES = NO. RESIDUAL CHECKS TO BE MADE (REC NRES=1 ON FIRST RUN)
C NPEAKD = THICK BODY: NO. OF A(X) PEAKS TO BE USED FOR 1ST LST SQR
C NPEAKT = THIN BODY: NO. OF A(X) PEAKS TO BE USED FOR 1ST LST SQR
C
C IOUT = WANT EXTRA OUTPUT (LST SQR VAL & RESID VAL): = 0 NO = 1 YES
C CUTRSD = FACTOR USED IN RESIDUAL ANALYSIS (SAME EXPLAIN AS CUTAX)
C CARD 7: USE ONLY IF ISECT=2 FORMAT (8110)
C
C ITIR(1) = NO. ITERATIONS TO BE MADE DURING 1ST CALL OF LST SQR
C ITIR(2) = NO. ITERATIONS OF LST SQRs TO BE MADE AFTER RESIDUAL
C NUMBER 1 AND DURING LST SQRs ANALYSIS NUMBER 2
C
C ITIR(N) = AS REQUIRED UP TO ITIR(NRES+1) = NO. OF ITERATIONS TO BE
C MADE AFTER THE NRES RESIDUAL CHECK AND DURING THE
C NRES+1 CALL OF LST SQRs.
C DATA CARDS: FORMAT (160, I2, I6, I3, I2, 2F2.0, F5.0)
C THE REQUIRED INPUT VALUES ON EACH DATA CARD (ACCORDING TO THE FORMAT) ARE:
C IY = YEAR ID = JULIAN DAY IH = HOUR (24 HR DAY) AM = MINUTE
C AS = SECOND TOTMAG = MEASURED TOTAL FIELD INTENSITY VALUE
C NOTE: A MAX. OF 200 DATA CARDS OR 200 INTERPOLATED VALUES ARE ALLOWED
C THE PROGRAM SUBTRACTS A CONSTANT REGIONAL FIELD FROM THE DATA
C
C DEVELOPED BY: ALAN K COOPER
C OFFICE OF MARINE GEOLOGY
COMMON_TIME(200), B(200), C(200), D(200), E(200), F(200)
COMMON/LSTSQ/ XA, XU, EPS, OUTP, SPD, IER, INPLT
COMMON/RESID/ ZMAX, ICALL
COMMON/SYM/ PARM(45), NP4R
COMMON/PRMCVR/XF(200)
COMMON/CUT/ CUTA, CUTRD, AMIN, AMAX
COMMON/LSQERR/ IERR
DIMENSION AX(200), TMAG(200), DTMAG(200), ITIR(10)
DIMENSION TITLE(20)
COMPLEX CAI(256)
LOGICAL OUTPT

--- FORMAT STATEMENTS ---
800 FORMAT(F10.2, 10I0)
801 FORMAT(I10, F10.1, 2110, F10.3)
802 FORMAT(2F10.2, F10.1, 110)
803 FORMAT(2F10.1)
804 FORMAT(4F10.4, F10.3)
805 FORMAT(8F10.2)
806 FORMAT(20A4)
900 FORMAT(///, ! VALUES FOR THE A(X) = T(X)**2 + T1(X)**2 CURVE:
  $/ ,10(/,1P5E20.9))
901 FORMAT(1H1, 'ESTIMATION OF DEPTH TO THICK MAGNETIC SOURCE BODIES -
  $USING THE HORIZONTAL DERIVATIVE OF THE TOTAL FIELD ', ' MAGNETICS
  $ FOR THE ANALYSIS;')
902 FORMAT(1H1, 'ESTIMATION OF DEPTH TO THIN MAGNETIC SOURCE BODIES - U
  $Sing THE TOTAL FIELD MAGNETICS FOR THE ANALYSIS;')
903 FORMAT(///, IX, 'THE MATRIX IN SUBROUTINE MARQ1 IS SINGULAR', '/
  $! CHECK THE INITIAL VALUES OF THE PARAMETERS OR REFER TO THE MARQ1
  $ PROGRAM DESCRIPTION')
904 FORMAT(1H1, 'PLOT OF A(X) CURVE(+), INITIAL PARAMETER CURVE(*), AND
  $ THE RESIDUAL TOTAL FIELD MAGNETIC CURVE( ):')
905 FORMAT(///, ' CORRELATION COEFFICIENT BETWEEN A(X) AND INITIAL PARA
  $METER CURVE=1, E17.6)
906 FORMAT(1H1, 'PLOT OF A(X) CURVE(+) AND THE RESIDUAL TOTAL FIELD MAG
  $NETICS CURVE(*):')
907 FORMAT(///, ' SAMPLE OF INPUT DATA AND INTERPOLATED DATA: (FIRST
  $20 DATA POINTS)', ' NO. OF INPUT DATA POINTS=14, 15X,
  $NO. OF INTERPOLATED DATA POINTS=14, 10X, TIME! 7X, TOTMAGS!')
The depth analysis for thick bodies - based on the horizontal derivative of the total field magnetics - has terminated successfully.

The depth analysis for thin bodies - based on the total field magnetics - has terminated successfully.

Input information: parameters supplied by the user:
- Ship speed (kts): SPDKT = F6.1
- Data interpolation: ID = I14
- Program use: ISECT = I14
- Body types: IBOD = I4
- Profile to pole: IPOLE = I14
- Initial plots: INPLT = I
- Body depth (km): ZMAXKM = F6.1
- Profile to pole: IPOLE = I14
- Initial plots: INPLT = I
- A(x) cutoff: CUTAX = F7.3

Start interpolation (hrs): XSHP = F7.2
End interpolation (hrs): XEHR = F7.2
Sample increment (min): DXMIN = F6.1
Max. no. points: NP = I6

Field inclination (deg): FINC = F7.1
Mag heading (deg): FANG = F7.1

No. residual checks: NRES = I14
No. peaks - thick bodies: NPEAKD = I14
No. peaks - thin bodies: NPEAKT = I14
Iterations: ITIR(NRES+1) = 1016

Plot of the input total field magnetics and the total field magnetics reduced to the pole (*) (constant regional field removed).

Constants determined by the computer:
- Ship speed (km/hr): SPD(KM/HR) = F6.1
- Sample increment (min): DXMIN = F6.1
- Constant regional field removed from input: DX = F8.0
Max. body depth (hrs): F8.4
A(x) cutoff value: CUTAX = E15.7
Residual cutoff value: CUTRSD = F7.3

Data after interpolation from F6.2 to F6.2 hrs:

Read in parameters:

```fortran
READ(5,806) TITLE(I), I=1,20
WRITE(6,915) TITLE(I), I=1,20
READ(5,800) SPDKT, ID, ISECT
READ(5,801) IBOD, ZMAXKM, IPOLE, INPLT, CUTAX
WRITE(6,910) SPDKT, ID, ISECT, IBOD, ZMAXKM, IPOLE, INPLT, CUTAX
IF(ID.EQ.2) GO TO 10
READ(5,802) XSHR, XEHR, DXMIN, NP
WRITE(6,911) XSHR, XEHR, DXMIN, NP
DX=DXMIN/60.
XS=XSHR
```

Constants:
- ED(KM/HR) = F6.1
- Sample increment (min): DXMIN = F6.1
- Constant regional field removed from input: DX = F8.0
Max. body depth (hrs): F8.4
A(x) cutoff value: CUTAX = E15.7
Residual cutoff value: CUTRSD = F7.3

Data after interpolation from F6.2 to F6.2 hrs:
{X}E=XEH\n10 IF(IPOLE.EQ.0) GO TO 20
READ(5,803) FINC,ANG
WRITE(6,912) FINC,ANG
20 IF(ISECT.EQ.0) GO TO 90
READ(5,804) NRES,NPEAKD,NPEAKT,OUT,CUTRSD
OUTPT=.TRUE.
IF(IOUT.EQ.0) OUTPT=.FALSE.,
KT=NRES+1
READ(5,805) (ITIR(I),I=1,KT)
WRITE(6,913) NRES,NPEAKD,NPEAKT,OUT,CUTRSD,(ITIR(I),I=1,KT)
90 CONTINUE
C ---------INPUT CONSTANTS----------
IF(ISECT.EQ.1) NPEAKT=0
IF(ISECT.EQ.1) NPEAKD=0
IF(ISECT.EQ.1) CUTRSD=0,
SPD=1.852*SPDKT
ZMAX=ZMAXKM/SPD
IERR=0
IPM=0
EPS=0.15
XU=10.
XLAM=.01
C ---------INPUT DATA----------
CALL INPUT(N,TIME,TMAG)
C ---------INTERPOLATE DATA----------
CALL INTERP(N,TIME,TMAG,XS,XE,DX,N,NI,E,F,DTMAG,ID,NP)
WRITE(6,917) XSHR,XTHR,IF(I1.EQ.2) GO TO 70
WRITE(6,907) N,NI,(TIME(I),TMAG(I),E(I),F(I),I=1,20)
N=NI
CALL MINMAX(N,F,TMIN,TMAX)
AVG=(TMAX+TMIN)/2.
DO 120 I=1,N
TIME(I)=E(I)
TMAG(I)=F(I)-AVG
120 CONTINUE
LE=1
IF(LEBOD.EQ.3) LE=2
DO 1001 JK=1,LE
ICALL=1
ITK=1
KBOD=IBOD+JK
C

---------COMPUTE A(X) CURVE---------

GO TO (200,200,201,200,201), KBOD

200 CALL HILBRT(N,NC,DTMAG,CAX)

GO TO 210

201 CALL HILBRT(N,NC,TMAG,CAX)

210 CONTINUE

DO 211 I=1,N

211 AX(I)=CABS(CAX(I))**2

C

---------COMPUTE CUTOFF POINT FOR A(X) AND RESIDUAL CURVES---------

IF(JK.EQ.2) GO TO 212

CALL MINMAX(N,AX,AMIN,AMAX)

CUTOFF=SQRT(AMAX*AMIN)

CUTAX=CUTOFF*CUTAX

CUTRSD=CUTOFF*CUTRSD

---------OUTPUT COMPUTED CONSTANTS---------

WRITE(6,916) SPD,DXMIN,DX,AVG,ZMAX,CUTAX,CUTRSD

212 CONTINUE

C

---------MAGNETIC PROFILE REDUCED TO THE POLE---------

IF(IPOLE.EQ.0.or.IP=NE.0) GO TO 228

IF(ANG.EQ.90.) GO TO 221

CR=0.01745329

BINC=ATAN(TAN(FINC*CR)/ABSCOS(ANG*CR)))/CR

GO TO 222

221 BINC=90.

222 PHI2=180.-2.*BINC

225 CALL POLRED(N,CAX,PHI2)

WRITE(6,914)

CALL PLOT(N,TMAG,B,B)

228 IPM=1

C

---------FIND SYMMETRIC COMPONENTS OF A(X)---------

GO TO (231,231,232,231,232), KBOD

231 WRITE(6,901)

WRITE(6,900) (AX(I),I=1,N)

CALL SYMPR3(N,NPEAKD,AX)

GO TO 230

232 WRITE(6,902)

WRITE(6,900) (AX(I),I=1,N)

AX(I)=AX(2)

CALL SYMPR3(N,NPEAKD,AX)

230 CONTINUE

IF(NPAR.EQ.0) GO TO 233

CALL FUNCT(PARM,NPAR,AX,N)

DO 234 JT=1,N
234 E(JT)=AX(JT)-XF(JT)
    CALL PEAKFT(N,AX)
C    ----------------------PLOT A(X), TOTMAGS, & INITIAL PARAMETER CURVE----------------------
233 IF(INPLT.EQ.0) GO TO 250
    CALL MINMAX(N,AX,AMIN,AMAX)
    CALL SETUP(N,TMAG,B,AMAX,AMIN)
    IF(NPAR.EQ.0) GO TO 235
    WRITE(6,904)
    IF(ISECT.GE.2) GO TO 252
    CALL FUNCT(PARM,NPAR,C,N)
    CALL CORLAT(N,AX,C,VAL)
    WRITE(6,905) VAL
    GO TO 237
235 WRITE(6,906)
    CALL PLOT(N,2,AX,B,AX)
    GO TO 1000
237 CALL PLOT(N,3,AX,C,B)
    GO TO 1000
250 IF(ISECT.LT.2) GO TO 1000
252 CONTINUE
C    ------------------------LEAST SQUARES ROUTINE------------------------
    ITER=ITIR(ITK)
300 CALL MARQUM(N,AX,$310)
    IF(NPAR.EQ.0) GO TO 1000
    GO TO 320
310 WRITE(6,903)
    GO TO 1000
320 IF(INRES-ITK.LT.0) GO TO 1000
    ITK=ITK+1
C    -------------------------CHECK RESIDUALS------------------------
    CALL RSDCHK(N,AX)
    GO TO 252
1000 CONTINUE
    GO TO (400,400,401,400,401,KBOD
400 WRITE(6,908)
    GO TO 1001
401 WRITE(6,909)
1001 CONTINUE
    STOP
END
SUBROUTINE INTERP(N, X, Y, XSTART, XSTOP, XINC, N1, X1, Y1, DY1, ID, NP)
C THIS SUBROUTINE WILL INTERPOLATE THE VALUES FROM ARRAY Y(N) FROM XSTART
C TO XSTOP AT INCREMENTS OF XINC. THREE ARRAYS ARE RETURNED:
C X1(N1) = REAL ARRAY CONTAINING THE INTERPOLATED X VALUES
C Y1(N1) = REAL ARRAY CONTAINING THE EVENLY SPACED INTERPOLATED VALUE
C DY1(N1) = REAL ARRAY CONTAINING THE DERIVATIVES OF Y1(N) AT EVENLY
C SPACED INCREMENTS XINC(IE AT SAME LOCATIONS AS Y1(N1))
C THE OTHER VARIABLES INCLUDE:
C X(N) = REAL ARRAY CONTAINING THE INITIAL VALUES ALONG THE X-AXIS
C Y(N) = REAL ARRAY CONTAINING THE INITIAL VALUES TO BE INTERPOLATED
C XSTART = FIRST X VALUE WHERE THE INTERPOLATION IS TO START
C XSTOP = LAST X VALUE FOR THE INTERPOLATION
C XINC = INCREMENT OF X AT WHICH EVENLY SPACED DATA WILL BE RETURNED
C N = NUMBER OF DATA POINTS IN THE INPUT ARRAY Y(N)
C N1 = NUMBER OF DATA POINTS IN THE OUTPUT ARRAY Y1(N1) AND DY1(N1)
C NP = MAXIMUM NUMBER OF POINTS ALLOWED IN OUTPUT ARRAYS
C ID = IDENTIFICATION MARKER TO INSTRUCT THE SUBROUTINE, THE VALUES:
C = 0 : DO AN INTERPOLATION USING ALL PARAMETERS, THREE ARRAYS
C = 1 : SAME AS FOR 0 EXCEPT SUBROUTINE SPLINE IS CALLED
C = 2 : THE SUBROUTINE WILL ONLY RETURN THE DERIVATIVE OF THE
C INPUT CURVE AT EACH OF THE EVENLY SPACED X VALUES
C THIS IS A GENERAL SUBROUTINE FOR USE OUTSIDE OF THIS PROGRAM
C MAXIMUM COMMON USAGE: XAXIS(200), B(N), C(N), D(N), E(N), XY(200)
COMMON XAXIS(200), B(200), C(200), D(200), E(200), XY(200)
DIMENSION X(1), Y(1), X1(1), Y1(1), DY1(1)
100 FORMAT(/I1X,'XSTART IS OUT OF THE RANGE OF THE INPUT DATA')
101 FORMAT(/I1X,'XSTOP IS OUT OF THE RANGE OF THE INPUT DATA')
IEND=0
IF(ID.EQ.2) GO TO 15
C ------------CHECK INTERPOLATION RANGE-------------
IF(XSTART.LE.X(1),AND,XSTART.GE.X(1)) GO TO 8
WRITE(6,100)
IEND=1
8 IF(XSTOP.GE.X(1),AND,XSTOP.LE.XSTART) GO TO 9
WRITE(6,101)
RETURN
9 IF(IEND.EQ.1) RETURN
IF(ID.EQ.0) GO TO 1
C ------------COMPUTE THE INTERPOLATING CONSTANTS------------
CALL SPLINE(N, X, Y, B, C, D)
C ------------BINARY SEARCH-------------
G LEVEL 20  INTERP  DATE = 73332  17/28/02

1 XX=XSTART
   I=1
   J=N
2 IF(J-1.EQ.1) GO TO 6
   K=(I+J)/2
3 IF(XX-X(K)) 3,5,4
   J=K
   IF(X(K)-XX,GT,X(K)-X(K-1)) GO TO 2
   J=K-1
   GO TO 7
4 I=K
   IF(XX-X(K),GT,X(K+1)-X(K)) GO TO 2
   J=K
   GO TO 7
5 J=K
   GO TO 7
6 J=I
7 IF(XSTOP,GT,X(N)) XSTOP=X(N)
   M=(XSTOP-XSTART)/XINC+1
   IF(M.GT.NP) M=NP
C -------DO INTERPOLATION - COMPUTE DERIVATIVES---------
10 DO 10 I=1,M
11 IF(XX.LT.X(J+1)) GO TO 13
   IF(J+1.NE.N) GO TO 14
   NI=I-1
   RETURN
14 J=J+1
   GO TO 12
13 IF(XX,GT,XSTOP) RETURN
   NI=I
   DX=XX-X(J)
   Y1(I)=Y(J)+DX*(B(J)+DX*C(J)+DX*D(J))
   DY1(I)=B(J)+DX*(2*C(J)+3*D(J))
   X1(I)=XX
   XX=XX+XINC
10 CONTINUE
   RETURN
C -------ONLY COMPUTE DERIVATIVES---------
15 CALL SPLINE(N,X,Y,B,C,D)
   KK=N-1
   N1=N
   DO 16 I=1,N
   XXX=XX+XINC
   Y1(I)=Y(XXX)
   N2=N1+1
   IF(N2.GE.M) N2=M
   DO 13 J=1,N1
      Y3=Y1(J)+Y2(J)*DX+C(J)*DX+D(J)*DX
      Y2(J)=Y3
      XXX=XX+XINC
      Y1(J)=Y(XXX)
13 CONTINUE
   RETURN
G LEVEL 20
INTERP
DATE = 73332
17/28/02

X1(I) = X(I)
16 Y1(I) = Y(I)
   DO 17 I = 1, KK
17 DY1(I) = B(I)
   DY1(N) = B(KK)
20 RETURN
END
SUBROUTINE SPLINE(N, X, Y, B, C, D)

C THIS SUBROUTINE COMPUTES THE SPLINE COEFFICIENTS NECESSARY TO DO DATA INTERPOLATION. THE INPUT PARAMETERS ARE:

C N = NO. OF DATA POINTS
C X(N) = REAL ARRAY CONTAINING THE ABSCISSA OF THE DATA POINTS
C Y(N) = REAL ARRAY CONTAINING THE ORDINATES OF THE DATA POINTS
C B(N-1), C(N-1), & D(N-1) = REAL ARRAYS CONTAINING THE SPLINE COEFFICIENTS

REAL*4 X(1), Y(1), B(1), C(1), D(1)
REAL*8 DELTA, DELTAL
REAL*4 H(200), SPP(200)

NM1 = N-1

DO 10 I=1,NM1
   H(I) = X(I+1) - X(I)
   DELTAL = (Y(I+1) - Y(I))/H(I)
   IF (I .NE. 1) B(I) = DELTAL - DELTA
   DELTA = DELTAL
10 CONTINUE

DO 20 I = 3,NM1
   C(I) = 2.*O.DO*(H(I) + H(2))
20 CONTINUE

C --------------FORWARD ELIMINATION-------------
C SPP(1) = 0.*O.DO
C SPP(N) = 0.*O.DO
SPP(N-1) = B(N-1)/C(N-1)
NM2 = N-2
DO 30 I = 2,NM2
   J = N-I
   SPP(J) = (B(J) - H(J)*SPP(J+1))/C(J)
30 CONTINUE

DO 40 I = 1,NM1
   B(I) = (Y(I+1) - Y(I))/H(I) - (SPP(I+1) + 2.*O.DO*SPP(I))*H(I)
C(I) = 3.*O.DO*SPP(I)
D(I) = (SPP(I+1) - SPP(I))/H(I)
40 CONTINUE

RETURN
END
SUBROUTINE FORSET(N,F,NC,CF)

C THIS SUBROUTINE PREPARES THE DATA SET F(N) FOR ENTRY INTO THE FAST
C FOURIER TRANSFORM SUBROUTINE FORK. NC IS THE NUMBER OF POINTS IN THE
C MODIFIED DATA SET (IE ZEROS ARE ADDED TO THE NEAREST POWER OF 2). CF(NC)
C IS A COMPLEX ARRAY WHICH CONTAINS THE DATA READY FOR INPUT INTO FORK

DIMENSION F(1)
COMPLEX CF(1)

C --------------FIND NEXT HIGHEST POWER OF 2-------------
IR=ALOG(1.0*N)/.69315
NC=2**(IR+1)
NX=N+1

C --------------FILL THE COMPLEX ARRAY-------------
DO 1 I=1,N
   CF(I)=F(I)
1 CONTINUE
IF(N.EQ.NC) RETURN

C --------------ADD ZEROS-------------
DO 2 I=NX,NC
   CF(I)=0.
2 CONTINUE
RETURN
END
SUBROUTINE HIIBRT(N, NC, FX, FXM)
C THIS SUBROUTINE WILL COMPUTE THE COMPLEX ARRAY AX(NC) = T(X) - ITI(X)
C FROM A REAL ARRAY FX(N). THE REAL ARRAY IS FOUR, TRANS., MODIFIED, AND
C INVERSE FOUR, TRANS. SUCH THAT; THE REAL PART OF THE COMPLEX OUTPUT ARRAY
C EQUALS THE INPUT ARRAY, AND THE IMAGINARY PART OF THE OUTPUT ARRAY IS THE
C HILBERT Transform Of THE INPUT ARRAY
C FX(N) = A 1-DIMENSIONAL REAL ARRAY CONTAINING THE DATA TO BE TRANSFORMED
C FXM(NC) = 1-DIMENSIONAL COMPLEX ARRAY CONTAINING THE TRANSFORM
DIMENSION FX(1)
COMPLEX FXM(1), AXM(256)

CALL FORSET(N, FX, NC, AXM)
CALL FORK(NC, AXM, 1, 1)

CALL FORMOD(NC, AXM, FXM)

CALL FORK(NC, FXM, 1, 1)
RETURN
END
SUBROUTINE FORMLX(CX,_SIGNI)
C FAST FOURIER TRANSFORM ROUTINE WRITTEN BY JON CLARBOUT, GEOPHYSICS
C DEPARTMENT, STANFORD UNIVERSITY, 1970.
C
C CX(K) = SQRT(1/LX) SUM (CX(J)*EXP(2*PI*SIGNI*I*(J-1)*(K-1)/LX))
C
FOR K=1,2,...,LX=2**INTEGER

COMPLEX CX(LX),CARG,CEXP,CW,CTEMP
SC=SQRT(1./LX)
DO 5 J=1,LX
IF(J.GT.J)GO TO 2
CTEMP=CX(J)*SC
CX(J)=CX(I)*SC
CX(I)=CTEMP
2 M=LX/2
3 IF(J.LE.M)GO TO 5
J=J-M
M=M/2
IF(M.GE.1)GO TO 3
5 J=J+M
L=1
6 ISTEP=2*L
DO 8 M=1,L
CARG=(0.,1.)*(3.14159265*SIGNI*(M-1))/L
CW=CEXP(CARG)
DO 8 I=M,LX,ISTEP
CTEMP=CW*CX(I+L)
CX(I+L)=CX(I)-CTEMP
8 CX(I)=CX(I)+CTEMP
L=ISTEP
IF(L.LT.LX)GO TO 6
9 RETURN
END
SUBROUTINE POLRED(N, A, PHI2)
C THIS SUBROUTINE TAKES THE COMPLEX ANALYTIC FUNCTION: A(X)=T(X)-IT1(X)
C AND USES IT TO OBTAIN A MAGNETIC PROFILE WHICH HAS BEEN REDUCED TO THE
C POLE. THE PROFILE OUTPUT IS STORED IN ARRAY B
C MAXIMUM COMMON USAGE: X(N), B(N), XX(0)
C
COMMON X(200),B(200),XX(800)
COMPLEX A(1)

DX=X(2)-X(1)
CON=3.141597/180.

PHI2=PHI2*CON
R1=COS(PHI2)
R2=SIN(PHI2)

POLMAG=0.

100 FORMAT (6,100)

DO 1 I=1,N
   POLMAG=POLMAG+REAL(A(I))*R1-AIMAG(A(I))*R2)*DX
   B(I)=POLMAG
   WRITE (6,200) I,B(I)
1 CONTINUE

200 FORMAT (5X,I3,5X,12.5)

RETURN
END
SUBROUTINE FORMOD(NC, CF, CFM)
C THIS SUBROUTINE WILL MODIFY THE FOURIER TRANSFORM INTO A HILBERT
C TRANSFORM BY MAKING THE FOLLOWING MODIFICATIONS:
C H(W) = F(W) FOR W = 0
C H(W) = 2F(W) FOR F(W) > 0
C H(W) = 0 FOR F(W) < 0
C NC IS THE NUMBER OF POINTS IN CF(NC)
C THE ORIGINAL ARRAY CF(NC) IS SAVED AND THE MODIFIED TRANSFORM ARRAY CFM(NC)
C IS RETURNED
COMPLEX CF(I), CFM(I)
NB = NC/2+1
NA = NB + 1
CFM(1) = CF(1)
DO 1 I = 2, NB
     CFM(I) = 2 * CF(I)
 1 CONTINUE
DO 2 I = NA, NC
     CFM(I) = 0,
 2 CONTINUE
RETURN
END
**Subroutine SYMPP3**

<table>
<thead>
<tr>
<th><strong>NAME</strong></th>
<th><strong>LEVEL</strong></th>
<th><strong>DATE</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>SYMPP3</td>
<td>20</td>
<td>17/29/02</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>17</strong></th>
<th><strong>LEVEL 20 SYMP3</strong></th>
<th><strong>DATE = 73332</strong></th>
</tr>
</thead>
</table>

**Subroutine SYMPP3(N, IPEAK, AX)**

This subroutine finds the location of each symmetric component of the A(X) test curve by the following criteria:

1. Find the local maxima of A(X) - a theoretical, adjacent points. Check the 2nd derivative at each maxima. This must be less than the parameter **ZMAX**. The subroutine computes the parameters for a number of these symmetric components which is determined by:

   - **IPEAK** = 0, which uses all components which have an A(X) value larger than the A(X) cutoff value.
   - **IPEAK** = any number > 0 indicates user wishes the computer to use the largest **N = number of data points**.
   - **IPEAK** = 0 indicates user wishes the computer to use the largest **N = number of data points**.
   - **IPEAK** = 0 indicates user wishes the computer to use the largest **N = number of data points**.

**Parameters Used for the Decomposition of the A(X) Curve**

- **AX(N)** = real array containing the A(X) values.
- **N** = number of data points.
- **IPEAK** = number of symmetric peaks with A(X) values larger than the cutoff value.
- **KMAX** = maximum number of parameters allowed by dimension statements.
- **ZMAX** = estimate of maximum depth to magnetic source.
- **K = number of parameters**.
- **B(?)** = real array containing the parameter values.
- **KMAX = maximum number of parameters**.

**COMMON States**

- **COMMON /RESID/ ZMAX**
- **COMMON /SYM/ B(?)**
- **COMMON /CUT/ CUTAX, CUTMIN**

**Format Statements**

- **$100 FORMAT(1H11), parameters used for the decomposition of the A(X) curve**
- **$101 FORMAT(1H11), parameters used for the decomposition of the A(X) curve**
- **$102 FORMAT(1H11), parameters used for the decomposition of the A(X) curve**
- **$103 FORMAT(1H11), parameters used for the decomposition of the A(X) curve**
- **$104 FORMAT(1H11), parameters used for the decomposition of the A(X) curve**
103 FORMAT(1H1, '(3F15.3,E19.7)')
104 FORMAT(/, 'NO PEAKS WERE FOUND IN THE A(X) CURVE')
105 FORMAT(/, '***WARNING*** NO SYMMETRIC PEAKS HAVE A(X) VALUES
$> CUTOFF -- ANALYSIS ALLOWED TO CONTINUE')
KMAX=45
L=2
M=N-1
LL=1
KL=0
DX=X(2)-X(1)

C ---------FIND LOCAL MAXIMA---------
DO I I=L,M
IF(A(I).LE.A(I+1),OR.A(I).LE.A(I-1)) GO TO 1

C ---------CHECK 2ND DERIVATIVE CRITERIA---------
DER2=ABS((A(I+1)-2*A(I)+A(I-1))
DEP1=SQR((2*A(I)/DER2)*DX
IF(DEP1.GT.ZMAX) GO TO 1
IF(LL.GT.KMAX) GO TO 1

C ---------STORE PEAK LOCATIONS (INCREASING X-DIST)---------
IF(A(I).GE.CUTAX) KL=KL+1
IDIST(ll)=I
IF(LL.NE.1) GO TO 3
IAX(I)=1
GO TO 10

3 KK=I

C ---------STORE PEAK LOCATIONS (DECREASING VALUES OF A(X))---------
4 IT=IAX(KK)
IF(A(I).GT.AX(IT)) GO TO 5
KK=KK+1
IF(KK.GE.LL) GO TO 6
GO TO 4

6 IAX(KK)=I
GO TO 10

C ---------REARRANGE ARRAY IAX---------
5 J=KK
JJ=J
IM=IAX(J)

7 JJ=JJ+1
IF(JJ.EQ.LL) GO TO 8
IL=IAX(JJ)
IAX(JJ)=IM
IM=IL
GO TO 7
8 IAIX(JJ)=IM
    IAIX(J)=1
10 LL=LL+1
C -----------DETERMINE NO. OF PEAKS TO USE-----------
1 CONTINUE
    LL=LL-1
    IF(IPEAK.NE.0) GO TO 9
    KUSE=KL
    IF(KL.GT.10) KUSE=10
    GO TO 11
9 KUSE=IPEAK
    IF(KUSE.GT.LL) KUSE=LL
    IF(IPEAK.GT.KMAX/3) KUSE=KMAX/3
11 CONTINUE
    WRITE(6,100) AMAX, AMIN, CUTFX, KL, IPEAK, LL, KUSE
    NO=-1
    IF(LL.EQ.0) GO TO 50
C -----------FIND PARAMETERS (INCREASING X-DIST)-----------
DO 20 J=1,LL
    NO=NO+2
    I=IDIST(J)
    Q(NO)=X(I)
    Q(NO+1)=AX(I)
20 CONTINUE
    WRITE(6,101) (Q(I), I=1,NO)
    NO=-2
    IF(KUSE.NE.0) GO TO 21
    WRITE(6,105)
    KUSE=1
21 CONTINUE
C -----------FIND PARAMETERS (DECREASING AX(I) VALUES-----------
DO 30 J=1,KUSE
    NO=NO+3
    I=IAIX(J)
C -----------COMPUTE INITIAL DEPTHS-----------
    TEM=DEP(N,1,AX)
    B(NO)=SQRT(TEM**2*AX(I))
    B(NO+1)=TEM
    B(NO+2)=X(I)
    T(J)=AX(I)
30 CONTINUE
    WRITE(6,102)
L = -2
DO 40 J = 1, KUSE
  L = L + 3
  M = L + 2
  WRITE(6, 103) (B(I), I = L, M, T(J))
40 CONTINUE
C -----------FINAL NO. OF PARAMETERS USED--------------
K = KUSE * 3
RETURN
50 WRITE(6, 104)
K = 0
RETURN
END
SUBROUTINE CORLAT(N,A,B,COR)

C THIS SUBROUTINE COMPUTES THE PRODUCT-MOMENT CORRELATION COEFFICIENT BETWEEN
C TWO DATA ARRAYS A(N) AND B(N). THE MATHEMATICAL EXPRESSION IS:
C
C COR = (SUM A*B)/SQRT(SUM A**2)(SUM B**2))

DIMENSION A(1),B(1)
REAL*8 X,Y,Z
X=0.0D0
Y=0.0D0
Z=0.0D0

DO 1 I=1,N
X=X+A(I)*B(I)
Y=Y+A(I)**2
1 Z=Z+B(I)**2

C ---------------CORRELATION COEFFICIENT--------------
COR=DABS(X/DSQRT(Y*Z))
RETURN
END
FUNCTION DEP(N, I, AX)
C THIS FUNCTION ESTIMATES THE INITIAL DEPTHS TO THE MAGNETIC SOURCE BY
C FINDING THE 1/2 WIDTH OF THE SYMMETRIC PEAKS IN THE AX(N) CURVE
C N = NO. OF DATA POINTS IN A(N)
C I = THE SUBSCRIPT OF THE ARRAY A(I) AT WHICH THE INITIAL DEPTH IS
C TO BE ESTIMATED
C AX(N)= A REAL ARRAY CONTAINING THE DATA CURVE
C MAXIMUM COMMON USAGE: X(N), XX(100)
COMMON X(200), XX(1000)
DIMENSION AX(1)
D=AX(1)
JK=I-1
IF(N-I.LT.1) JK=N-1
LO=1
DO 21 JL=1, JK
    IF(LO.EQ.JK) GO TO 25
    IF(AX(I+1+LO).GE.AX(I+LO), OR, AX(I-1-LO).GE.AX(I-LO)) GO TO 25
    LO=LO+1
21 CONTINUE
C ---------------COMPUTE 1/2 WIDTH------------
25 DEP=(DX*FLOAT(LO))/2.
RETURN
END
SUBROUTINE PEAKFT(N, AX)
C THIS SUBROUTINE GIVES A QUANTITATIVE ESTIMATE AS TO THE QUALITY OF THE
C FIT BETWEEN THE A(X) CURVE AND THE COMPUTED CURVE - FOR EACH SYMMETRIC
C PEAK SEPARATELY. THE ANALYSIS FOR EACH PEAK IS DONE BETWEEN:
C PEAK POSITION +/- 3*COMPUTED DEPTH
C AT THESE DISTANCES, THE CURVE FOR THE SYMMETRIC PEAK SHOULD THEORETICALLY
C BE AT 10% OF ITS MAXIMUM VALUE. THE PARAMETERS USED ARE:
C N=NO. OF DATA POINTS IN THE AX(N) CURVE
C AX(N)=REAL ARRAY CONTAINING THE A(X) CURVE
C MAXIMUM COMMON USAGE: X(N), Y(0), C(0), D(0), E(0), DIFF(N), B(KMAX)
C F(N), AA(100), BB(100), XX(0)
REAL AA, BB
REAL MARQIP
COMMON X(200), Y(200), C(200), D(200), DIFF(200), XX(200)
COMMON SYM/B(45), K
COMMON PRMCRV/F(200)
COMMON WORK/ AA(100), BB(100)
DIMENSION AX(L)
100 FORMAT(///, ' ESTIMATION OF THE "GOODNESS OF FIT" FOR EACH SYMMETRIC
$ PEAK - BASED UPON THE COMPARISON OF THE A(X) CURVE AND THE', '
$ COMPUTED PARAMETER CURVE IN THE VICINITY OF THE SYMMETRIC PEAK', '
$ PEAK LOCATION(HRS) COMPARISON PTS RANGE(HRS); FROM', '
$ SUM OF SQUARES OF RESIDUALS CORRELATION COEFFICIE
$NT')
WRITE(6,100)
DX=X(2)-X(1)
DO 1 I=1, K, 3
TDEP=3.*B(I+1)
DISP=B(I+2)-X(I)
BDIST=(DISP-TDEP)/DX
FDIST=(DISP+TDEP)/DX
IA=IFIX(FDIST)+2
IB=IFIX(BDIST)+1
IF(IA.GT.N) IA=N
IF(IB.LT.1) IB=1
NPT=IA-IB+1
IF(NPT.LT.100) GO TO 4
NR=(NPT-99)/2
IA=IA-NR
IB=IB-NR
NPT=IA-IB+1
4 DO 2 J=1, NPT
2
AA(J) = AX(IB+J-1)

2  BB(J) = F(IB+J-1)
   CALL CORLAT(NPT, AA, BB, COR)
   DO 3  J = 1, NPT
3  AA(J) = DIFF(IB+J-1)
   SQR = MROIP(AA, AA, NPT)
   WRITE(6,102) B(1+2), NPT, X(IB), X(IA), SQR, COR
1  CONTINUE
   RETURN
   END
SUBROUTINE SETUP(N,TMAG,V,YMAX,YMIN)
C THIS SUBROUTINE SETS UP THE SCALE FACTORS AND MODIFIES TMAG(N) SO THAT THE
C TOTAL FIELD MAGNETICS CAN BE PLOTTED WITH THE \( f(x)^{1/2} \) CURVE. THE SCALE
C IS SET SUCH THAT 25 SMALL DIVISIONS ON THE Y-AXIS = 1000 GAMMA (2000 GAMMA
C IN SPECIAL CASES). V(N) CONTAINS THE MODIFIED TMAG VALUES, YMIN IS
C THE MINIMUM VALUE OF THE \( f(x)^{1/2} \) CURVE, YMAX IS THE MAXIMUM VALUE.
C DIMENSION TMAG(1),V(1)
CALL MINMAX(N,TMAG,TMIN,TMAX)
CS=(YMAX-YMIN)/4000,
CP=(YMAX+YMIN)/2.
IF(TMAX-TMIN).GT.2000. GO TO 2
DO 1 I=1,N
1 V(I)=((TMAG(I)-TMIN)*CS)+CP
RETURN
2 CS=0.5*CS
DO 3 I=1,N
3 V(I)=((TMAG(I)-TMIN)*CS)+CP
RETURN
END
SUBROUTINE MINMAX(N, X, XMIN, XMAX)

C THIS SUBROUTINE FINDS THE MINIMUM AND MAXIMUM VALUES WITHIN A 1-DIMENSIONAL
C ARRAY X(N).
DIMENSION X(N)
XMIN=X(1)
XMAX=X(1)
DO 10 I=2,N
  IF(X(I).GT.XMAX) XMAX=X(I)
  IF(X(I).LT.XMIN) XMIN=X(I)
10 CONTINUE
RETURN
END
SUBROUTINE PLOT(N, NCUR, U, V, W)

C THIS WILL PLOT UP TO A MAXIMUM OF THREE (3) CURVES: U(N), V(N), W(N) WITH A
C COMMON X-AXIS GIVEN BY X(N). THE ARRAY X(N) MUST BE STORED IN COMMON
C CURVES ARE PLOTTED USING SYMBOLS + * . RESPECTIVELY
C NCUR = NO. OF CURVES TO BE PLOTTED
C THIS SUBROUTINE REQUIRES THE USE OF SUBROUTINE PRLPLT
C MAXIMUM COMMON USAGE: X(N), XX(0)

COMMON X(200), XX(1000)
DIMENSION U(1), V(1), W(1), Y(200, 3), SYM(3)
DATA SYM(1)'/**/', SYM(2)'/**/', SYM(3)'./', '/'/

-------SETUP PLOT ARRAYS------

2 DO 1 I=1,N
    Y(i,1)=U(i)
1   Y(i,2)=V(i)
    Y(i,3)=W(i)

-------FIND SMALLEST AND LARGEST VALUE-------
10 CALL MINMAX(N,U,YMIN,YMAX)
    CALL MINMAX(N,V,XHMIN,XHMAX)
    CALL MINMAX(N,W,ZHMIN,ZHMAX)
    CALL MINMAX(N,X,XMIN,XMAX)
    YMIN=AMIN1(YMIN,XHMIN,ZHMIN)
    YMAX=AMAX1(YMAX,XHMAX,ZHMAX)

-------PLOT THE CURVES-------
    CALL PRLPLT(XMAX, XMIN, YMAX, YMIN, NCUR, N, N, SYM, Y, X)
RETURN

END
SUBROUTINE PRLPLT(XMAX, XMIN, YMAX, YMIN, NY, LINES, LAST, SYM, Y, X)
  C  THIS SUBROUTINE IS A PLOT ROUTINE WHICH HAS THE PARAMETERS:
  C   NY = NUMBER OF PLOTS TO BE MADE (FOR EACH PLOT A SYMBOL MUST
  C    BE ASSIGNED IN THE CALLING PROGRAM - EX. DATA SYM(1)/***/
  C   LINES = THE TOTAL NUMBER OF LINES ALONG THE X AXIS TO BE PLOTTED
  C   LAST = THE NUMBER OF THE LAST POINT TO BE PLOTTED
  C   SYM = AN ARRAY CONTAINING THE SYMBOLS TO BE USED IN PLOTTING
  C   Y = A 2-DIMENSIONAL ARRAY CONTAINING THE Y VALUES TO BE PLOTTED
  C   X = A 1 DIMENSIONAL ARRAY CONTAINING THE X AXIS VALUES
  DATA BLANK/', CR/''', PL/'''
  DIMENSION SYM(NY), ZY(11), GRAPH(101), Y(200,3), X(1)
  15 FORMAT(//,8X,1P11E10.2)
  20 FORMAT(1H,3X,'X VALUES',2X,20('....'),15X,'Y VALUES')
  75 FORMAT(1H,2X,1PE9.2,2X,101A1,2(1PE9.2))
  76 FORMAT(1H,2X,1PE9.2,2X,101A1,2X,1PE11.4)
  85 FORMAT(1H,8X,1P11E16,2*X SCALE=',1PE11.4')
  XSCALE=(XMAX-XMIN)/(LINES-1.)
  YSCALE=(YMAX-YMIN)/100.
  DO 10 K = 1,11
  10 ZY(K)=10.*(K-1)*YSCALE+YMIN
  WRITE(6,15) (ZY(K), K=1,11)
  WRITE(6,20)
  INDEX=1
  40 X1=XMIN
  DO 80 I=1,LINES
  IF(MOD(I,5),EQ,1) GO TO 45
  GRAPH(I)=CR
  GRAPH(101)=CR
  GO TO 50
  45 GRAPH(I)=PL
  GRAPH(101)=PL
  50 DO 55 J=2,100
  55 GRAPH(J)=BLANK
  57 DO 60 J=1,NY
  59 IY=(Y(INDEX,J)-YMIN)/YSCALE+1.5
  GRAPH(IY)=SYM(J)
  CONTINUE.
  INDEX=INDEX+1
  GO TO 70
  70 IF(NY,EQ,1) GO TO 71
  WRITE(6,75)(X(I), (GRAPH(J), J=1,101), Y(I,1), Y(I,2))
  GO TO 80
71 WRITE(6,76)X(I),(GRAPH(J),J=1,101),Y(I,1)
80 CONTINUE
WRITE(6,20)
WRITE(6,85) (ZY(K),K=1,11)
WRITE(6,90) YSCALE,XSCALE
RETURN
END
SUBROUTINE MARQ1(N,Y,*)

C MODIFICATION OF STANFORD UNIVERSITY
C LIBRARY PROGRAM NUMBER CO08
C WILLIAM E RIDDLE (SCC)
C MARCH 9,1967 REVISED APRIL 4,1968
C
C THIS SUBROUTINE DOES A NONLINEAR LEAST SQUARE ANALYSIS BASED ON THE METHOD
C GIVEN BY MARQUARDT. THE ROUTINE FINDS THE BEST FIT FOR THE K PARAMETERS
C (OR K/3 SYMMETRIC CURVES) OF THE AX(N) CURVE. THE INITIAL VALUES OF THE
C PARAMETERS (FOUND BY SYMPR3) ARE IMPROVED BY MINIMIZING THE SUM OF THE
C SQUARES OF THE DIFFERENCE BETWEEN THE AX(N) CURVE AND THE LEAST SQUARE
C CURVE. THE EFFECTIVENESS OF THE METHOD IS HIGHLY DEPENDENT UPON THE
C QUALITY OF THE INITIAL VALUES FOR THE PARAMETERS.
C FOR EACH ITERATION, THE 1ST PARTIAL DERIVATIVES OF ALL PARAMETERS ARE
C COMPUTED ONCE
C FOR EACH FUNCTION EVALUATION, THE PARAMETER VALUES ARE USED TO COMPUTE THE
C LEAST SQUARE APPROXIMATION CURVE.
C MAXIMUM COMMON USAGE: X(N),TM(N),SS(5*KMAX/3),TEM(O),DIFF(N),XX(O)
C
C
C SUBROUTINE PARAMETERS
C INTEGER N,K,IMAX
C REAL LAMRD,P,NU,EPSP,B,Y,F
C
C LOCAL VARIABLES
C REAL TAU,FACT,SUM,S,NORMD,NORMG,DIFG,GS,Gu,D,DS,DG,Bl,A,AS
C INTEGER C1,C2,I,J,L,C3,I1
C DOUBLE PRECISION T1,T2,TSUM
C COMMON X(200),TM(200),SS(200),TEM(200),DIFF(200),XX(200)
C COMMON /LSTSO/ LAMBDA,NU,EPSP,OUTPT,SPD,IMAX,INPLT
C COMMON /RESID/ ZMAX,ICALL
C COMMON /SYM/ Bi(45),K
C COMMON /PRMCRV/F(200)
C COMMON /LSQERR/ IERR
C DIMENSION P(200,45),Y(I)
C LOGICAL OUTPT
C DIMENSION G(45),GS(45),GU(45),D(45),DS(45),DG(45),Bl(45),
C A(45,45),AS(45,45)
C
C SUBPROGRAMS CALLED: MARQGS(SUBROUTINE),MARQ1P(FUNCTION)
C
C 89 FORMAT(1H1, ' MARQUARDT NONLINEAR LEAST SQUARES ESTIMATION OF THE
$\text{SYMMETRIC COMPONENTS OF THE A(X) CURVE:}$

$90 \text{ FORMAT}(/, 'GOODNESS OF FIT PRIOR TO LEAST SQUARE PROCESS:/',$
$\text{SUM OF SQUARES OF RESIDUALS} = E16.7, '/,$$
\text{CORRELATION COEFFICIENT} = E18.7, '/,$$
\text{NUMBER OF ITERATIONS REQUIRED: TED} = 1, 15)$

$551 \text{ FORMAT}(/, 'THE LEAST SQUARE ROUTINE HAS CONVERGED TO A PROPER SOLUTION DURING ITERATION: 13')$

$601 \text{ FORMAT}(/, 'RESULTS FROM ITERATION: 14', /, \text{NO. OF FUNCTION EVALUATIONS} = E16.8, '/,$$
\text{CORRELATION COEFFICIENT} = E17.7, '/,$$
\text{NO. OF CURVES REJECTED DURING THIS ITERATION: 4}$


$902 \text{ FORMAT}(/, 'THE VALUES FOR THE APPROXIMATION TO THE A(X) CURVE ARE: /, 10(/, 1P5E20.9))$

$903 \text{ FORMAT}(/, 'ALL SYMMETRIC CURVES HAVE BEEN REJECTED DURING ITERATION 13 OF THE LEAST SQUARE ALGORITHM-----', /,$$
\text{NO. OF FUNCTION EVALUATIONS} = 14, /, \text{PARAMETERS REJECTED DURING THIS ITERATION: (3F15.5))}$

$904 \text{ FORMAT}(/, 'THE RESIDUAL VALUES (A(X) - COMPUTED) ARE: /, 10(/, 1P5E20.9))$

$905 \text{ FORMAT}(/, 'PLOT OF THE LEAST SQUARE APPROXIMATION TO THE A(X) CURVE: (AFTER 13 CHECKS OF THE RESIDUAL VALUES): /,' (+) A(X) CURVE (*) LST SQ CURVE (-) RESIDUAL CURVE)$

$906 \text{ FORMAT}(/, 'PARAMETERS OF REJECTED CURVES: /, (3F15.5))$

$907 \text{ FORMAT}(/, 'RECHECK THE A(X) CURVE AND THE INITIAL VALUES OF THE PARAMETERS------$)

$908 \text{ FORMAT}(/, 'NOTE: THIS PLOT DOES NOT INCLUDE THE 13 SYMMETRIC CURVES WHICH HAVE BEEN REJECTED DURING THE LEAST SQUARE PROCESS: ', /,$$ TO IMPROVE THE RESULTS REFER TO THE FOLLOWING EXPLANATION: $,$$, 9X, 'REJECTION CRITERIA', 27X, 'POSSIBLE CAUSE', 34X,$$
\text{'REMEDY'}, /, \text{COMPUTED POSITION OUT OF DATA RANGE: ', T45, 'SMALL PEAK NEAR OR ON LARGE PEAK OR ', T90, 'USE MORE PEAKS FOR ANALYSIS OR'},$
\text{COMPUTED DEPTH NEGATIVE: ', T45, 'MANY LARGE PEAKS CLOSE TOGETHER: ', T90, 'USE ADDITIONAL RESIDUAL ANALYSIS: /, COMPUTED DEPTH LARGER THAN ZMAXKM: ', T45, 'ZMAXKM TOO SMALL OR LARGE: ', T90, 'INCREASE ZMAXKM $M')$

$909 \text{ FORMAT}(/, '****ERROR**** THE LEAST SQUARE SOLUTION HAS BLOWN $UP. PROGRAM EXECUTION HAS TERMINATED ON AN EXPONENT OVERFLOW: ', /,$$ OR UNDERFLOW. THE ERROR HAS BEEN CAUSED BY THE REJECTION OF AN $IMPORTANT SYMMETRIC CURVE, THUS CAUSING THE LEAST SQUARE: /,$$$
C ---------FILL ARRAY G(K)---------

TSUM=0.DO
DO 121 JJ=1,N
   T1=DIFF(JJ)
   T2=P(JJ,1)
121 TSUM=TSUM+T1*T2
120 G(J)=TSUM

C ---------CONTINUE WITH ITERATIVE PROCEDURE---------

DO 150 I=1,K
   DO 160 J=I,K
      GS(I,J) = G(I)*DG(I)
   DO 160 J=1,K
      AS(I,J) = A(I,J)*DG(I)*DG(J)
   FACT = 1.0
   IF (LAMBDA.GT.1.0E-12) FACT = 1.0/NU
200 DO 220 I=1,K
   DO 210 J=I,K
      A(I,J) = AS(I,J)
210 A(I,I) = AS(I,I) + LAMBDA*FACT
220 GUU(I) = GSI(I)

MM=K
CALL MAPQGS(MM,A,GU,DS*,1010)
DO 230 I=1,K
230 DI(I) = DS(I)*DG(I)

C

300 DO 310 I=1,K
310 BI(I) = BI(I) + DI(I)

C -------EVALUATE THE FUNCTION-------

CALL FUNCT(B1,K,F,N)
C1 = C1+1
DO 320 I=1,N
320 DIFF(I) = Y(I) - F(I)
S = MARQIP(DIFF,DIFF,N)

C

C -------CHECK SUM OF SQUARES--------

IF (S-SUM) .LT. 400,400,500
SOLUTION TO BECOME UNSTABLE. REMEDY: CHANGE THE NECESSARY INPUT PARAMETERS (IZMAXKM, SHPSPD) OR INPEAKT, NPEAKD, OR ICUTRSD) SO THAT THE CURVE IS NOT REJECTED. CHECK AND CHANGE:

1. ZMAXKM, SHPSPD - IF THE SYMMETRIC CURVE WAS REJECTED BECAUSE THE COMPUTED DEPTH(HRS) * SHPSPD EXCEEDED ZMAXKM.
2. NPEAKT, NPEAKD - IF ERROR OCCURRED BEFORE A RESIDUAL ANALYSIS.
3. SQUARE SOLUTION, ICUTRSD - IF ERROR OCCURRED AFTER A RESIDUAL ANALYSIS. TOO MANY SYMMETRIC CURVES (OR CURVES TOO CLOSE TOGETHER) HAVE BEEN ADDED. TRY LOWERING THE RESIDUAL PEAK CUTOFF LEVEL BY MAKING CUTRSD SMALLER.

If the symmetric curve was rejected because the computed depth exceeded ZMAXKM, check and change:

- If the symmetric curve was rejected because the residual peak cutoff has been exceeded, try lowering the cutoff level by making CUTRSD smaller.

If the error occurred before a residual analysis, more symmetric curves are probably needed for the least squares solution. If an error occurred after a residual analysis, try lowering the residual peak cutoff level by making CUTRSD smaller.

At the time the error occurred, the final values of the parameters for all symmetric curves, X-POSITION(HRS), ZMAXKM, SHPSPD, and NPEAKT, NPEAKD, were as follows:

- X-POSITION(HRS) = [3F15.5]
- ZMAXKM = 0.0
- SHPSPD = 0.0
- NPEAKT = 0
- NPEAKD = 0

For help in finding the problem, the initial parameter curve is computed:

C = COMPUTE INITIAL PARAMETER CURVE

IF(INPLT.EQ.0) GO TO 70
CALL PLOT(N,3,Y,F,TM)
70 CALL CORLAT(N,Y,F,VAL)
WRITE(6,90) SUM, VAL, IMAX
WRITE(6,89)

START ITERATIVE PROCEDURE

DO 120 I=1,K
DO 110 J=1,K
C = FILL ARRAY A(K,K)

TSUM = 0.0
DO 111 JJ=1,N
T1 = P(JJ,I)
T2 = P(JJ,J)
111 TSUM = TSUM + T1 * T2
110 A(I,J) = TSUM
LEVEL 20 MARQ1 DATE = 73332 17/28/02

C --------EXIT FROM ROUTINE---------
400 DO 410 I=1,K
410 B(I) = B(I)
LAMBDA = LAMBDA * FACT
SUM = S
GO TO 600
C
C
500 IF (C3,NE,0) GO TO 550
C
C --------CHANGE CONVERGENCE FACTOR---------
NORMD = MARQIP(D,D,K)
IF(TERR.EQ.0) GO TO 125
WRITE(6,909) (B(I),I=1,K)
STOP
125 CONTINUE
S = NORMD * NORMG
S = MARQIP(D,G,K) / SQRT(S)
IF (S.GT.(.70107) ) GO TO 570
C
C --------CONTINUE WITH ROUTINE---------
FACT = FACT * NU
GO TO 200
C
C
550 IF (C3,NE,40) GO TO 570
WRITE(6,551) C2
GO TO 600
C
C
570 C3 = C3 + 1
DO 580 I=1,K
580 D(I) = D(I)/NU
GO TO 300
C
C --------CHECK PARAMETERS---------
600 CALL PRMCHK(N,NREJ)
NREJ=NREJ*3
IF(NREJ,NE,0) IEX=IEX+NREJ
IF(K.EQ.0) GO TO 700
CALL CORLAT(N,Y,F,VAL)
**** LEVEL 20 ****

C

--------- PRINT RESULTS ----------

WRITE(6,601) C2,C1,SUM,VAL,NREJ
IF(NREJ.NE.0) WRITE(6,901) (SS(I),I=1,NREJ)

602 IF(C2.EQ.IMAX) GO TO 900
IF(VAL.LT.0.98) GO TO 100

900 J=1
DO 910 I=1,K,3
DO 911 JJ=1,3
SS(J)=B(I+JJ-1)
911 J=J+1
DO 910 JJ=1,2
SS(J)=B(I+JJ)*SPD
910 J=J+1

KK=5*K/3
WRITE(6,901) C2,(SS(I),I=1,KK)
IF(NREJ.EQ.0) GO TO 915
CALL FUNCT(B,K,F,N)

915 CALL PEAKFT(N,Y)
IF(.NOT.OUTPT) GO TO 710
WRITE(6,902) (F(I),I=1,N)
WRITE(6,904) (DIFF(I),I=1,N)

C

MARQ1136

C

710 WRITE(6,905) IP

C --------- PLOT RESULTS ---------

CALL PLOT(N,3,Y,F,DFF)
IF(IEX.EQ.0) GO TO 1000
WRITE(6,908) IEX
GO TO 1000

700 WRITE(6,903) C2,C1,(SS(I),I=1,NREJ)
WRITE(6,907)

1000 RETURN

1010 RETURN_1

C LAST CARD OF MARQ1

END
SUBROUTINE MARQGS(M,A,B,X,*)
C   CALLED BY MARQ1 - LIBRARY PROGRAM C 008
C   SOLVES, BY GAUSSIAN ELIMINATION, THE SET OF SIMUL. EQNS. A*X=B
C   PARAMETERS
INTEGER M
REAL A,B,X
DIMENSION A(45,45),B(1),X(1)
C LOCAL VARIABLES
INTEGER I,J,IMAX,K,L,II
REAL MX,T,QUOT

DO 500 K=1,M
   MX = 0.0
   IMAX = K
   DO 100 I = K,M
      IF (MX,GT,ABS(A(I,K))) GO TO 100
      MX = ABS(A(I,K))
      IMAX = I
   100 CONTINUE
   IF (MX.EQ.0.0) GO TO 1010
   I = K
   IF (K.EQ.IMAX) GO TO 400
C   J=IMAX
   T=B(I)
   B(I)=B(J)
   B(J)=T
   DO 200 L=1,M
      T = A(I,L)
      A(I,L) = A(J,L)
   200 A(J,L) = T
C   400 L=K+1
   IF(L.GT.M) GO TO 500
   DO 450 J=L,M
      QUOT = A(J,K) / A(I,K)
      B(J) = B(J) - B(K)*QUOT
   450 A(J,II) = A(J,II) - QUOT*A(K,II)
C

500 CONTINUE

C

DO 590 K=1,M
I = M + 1 - K
T = 0
J = I+1
IF(J.GT.M) GO TO 590
DO 580 L=J,M
580 T = T + A(I,L) * X(L)
590 X(I) = (B(I)-T)/A(I,I)

C

RETURN
1010 RETURN

END
FUNCTION MARQIP (V1, V2, N)  
C CALLED BY MARQ1 - LIBRARY PROGRAM NUMBER C 008  
C CALCULATES DOUBLE PRECISION INNER PRODUCT  
C  
PARAMETERS  
COMMON /LSQERR/ IERR  
INTEGER N  
REAL V1, V2, MARQIP  
DIMENSION V1(N), V2(N)  
C LOCAL VARIABLES  
INTEGER I  
DOUBLE PRECISION T1, T2, SUM  
  
SUM=0.0  
DO 10 I=1, N  
T1 = V1(I)  
T2 = V2(I)  
IF(T1*T2.LT.1.E-60.AND.T1*T2.GT.1.E-60) GO TO 8  
IERR=1  
RETURN  
8 CONTINUE  
10 SUM = SUM + T1 * T2  
MARQIP = SUM  
RETURN  
END
SUBROUTINE DERIV(N,P)
C THIS SUBROUTINE IS USED BY SUBROUTINE MARQ1 TO COMPUTE THE DERIVATIVES
C OF THE ANALYTIC EXPRESSION FOR THE SYMMETRIC BELL SHAPED CURVES WHICH
C COMPOSE THE AX(I) CURVE, THIS EXPRESSION IS F(x) = ALPHA**2/H**2 + (X-POS)**2
C N=NO. OF DATA POINTS
C P(N,K)= ARRAY CONTAINING THE FIRST PARTIAL DERIVATIVES OF THE FUNCTION
C OTHER PARAMETERS USED BY THE SUBROUTINE ARE:
C K= NO. OF PARAMETERS TO BE MINIMIZED
C B(K)= ARRAY CONTAINING THE PARAMETERS
C F(x) WITH RESPECT TO EACH OF THE PARAMETERS AND AT EACH DATA POINT
C
COMMON X(1200),XX(1000)
COMMON /SYM/ B(45), K
REAL*8 D, C, CON, CON2
DIMENSION P(200,45)
L=K/3
DO 10 I=1,N
   DO 10 J=1,K,3
      DIST=X(I)
      ALPHA=B(J)
      DEPTH=B(J+1)
      PKPOS=B(J+2)
      D=2.DO*ALPHA**2
      C=DIST-PKPOS
      CON=DEPTH**2+C**2
      CON2=CON**2
      P(I,J)=D/(CON*ALPHA)
      P(I,J+1)=-D*DEPTH/CON2
10   P(I,J+2)=D*C/CON2
RETURN
END
SUBROUTINE FUNCT (B, K, F, N)

C THIS SUBROUTINE IS USED BY SUBROUTINE MARQ1 TO DETERMINE THE VALUES OF THE
C FUNCTION (WHICH IS BEING USED FOR THE LEAST SQUARE MINIMIZATION) AT EACH
C OF THE DATA POINTS
C K=NO. OF PARAMETERS BEING MINIMIZED
C N= NO. OF DATA POINTS
C B(K)= ARRAY CONTAINING THE PARAMETER VALUES
C F(N)= ARRAY CONTAINING THE OUTPUT FUNCTION VALUES
C MAXIMUM COMMON USAGE: X(N), XX(0)

COMMON X(200), XX(1000)
REAL*8 SUM
DIMENSION B(1), F(1)
SUM=0.0 DO
DO 3 I=1, N
DO 2 J=1, K; 3
DIST=X(I)
ALPHA=B(J)
DEPTH=B(J+1)
PKPOS=B(J+2)
3 SUM=SUM+(ALPHA**2)/((DEPTH**2)+(DIST-PKPOS)**2)
F(I)=SUM
3 SUM=0.0 DO
RETURN
END
SUBROUTINE PRMCHK(N, NREJ)

C THIS SUBROUTINE CHECKS THE PARAMETERS OF THE SYMMETRIC CURVES TO SEE IF:
C 1. SOURCE DEPTHS ARE NEGATIVE
C 2. SOURCE DEPTHS ARE GREATER THAN ZMAX
C 3. X-POSITIONS LIE OUTSIDE THE RANGE OF THE DATA
C IF ANY CONDITION IS FOUND, THE ROUTINE THROWS AWAY THAT SYMMETRIC CURVE
C AND REARRANGES THE ARRAY CONTAINING THE PARAMETERS

C N=NO. OF DATA POINTS
C NREJ=NO. OF SYMMETRIC CURVES REJECTED
C OTHER PARAMETERS USED IN THE SUBROUTINE ARE:
C K=NO. OF PARAMETERS
C B(K)=REAL ARRAY CONTAINING THE PARAMETERS
C MAXIMUM COMMON USAGE: X(N), TT(0), TM(KMAX), XX(0)

C COMMON X(200), TT(200), TM(200), XX(600)
C COMMON /RESID/ ZMAX, ICALL
C COMMON /SYM/ B(45), K

NT=1
L=K

--------CHECK PARAMETERS--------

DO 11 I=1, K, 3
  IF(B(I+2) .GT. X(N) .OR. B(I+2) .LT. X(I)) GO TO 12
  IF(B(I+1) .LT. 0. .OR. B(I+1) .GT. ZMAX) GO TO 12
  GO TO 11
12 L=K-3*NT
   NP=3*NREJ
   NREJ=NREJ+1
   NT=NT+1

--------REJECT BAD CURVES--------

DO 13 J=1, 3
  TM(NP+J)=B(I+J-1)
13 B(I+J-1)=0.

11 CONTINUE
   NT=1

--------REARRANGE PARAMETER ARRAY--------

DO 14 I=1, K
  IF(B(I) .EQ. 0.) GO TO 14
  B(NT)=B(I)
  NT=NT+1
14 CONTINUE
   K=L
RETURN
END
SUBROUTINE PSDCHK(N,AX)
C THIS SUBROUTINE CHECKS THE RESIDUAL CURVE (IE AX(I)-COMPUTED) FOR
C ADDITIONAL SYMMETRIC PEAKS WHICH DO NOT OCCUR AS LOCAL MAXIMA ON THE
C A(X) CURVE. NEW PEAKS FOUND IN THE RESIDUAL CURVE (BY THE SAME CRITERIA
C AS THE A(X)) ARE ADDED TO THE LIST OF OLD SYMMETRIC PEAKS IF THEY ARE
C LOCATED A DISTANCE > 2 * DX AWAY FROM AN OLD PEAK. (STATEMENTS 51 & 52)
C NEW PEAKS ARE NOT KEPT IF THEY ARE LESS THAN THIS DISTANCE
C N = NO. OF DATA POINTS IN A(X)
C AX(N) = REAL ARRAY CONTAINING THE A(X) CURVE
C OTHER PARAMETERS USED IN THE SUBROUTINE ARE:
C R(N) = REAL ARRAY CONTAINING THE RESIDUAL VALUES (A(X) - COMPUTED)
C K = NO. OF PARAMETERS
C B(K) = REAL ARRAY CONTAINING THE PARAMETER VALUES
C ZMAX = MAXIMUM ESTIMATED DEPTH TO THE MAGNETIC SOURCE BODY
C NADD = NO. OF CURVES ADDED FROM RESIDUAL ANALYSIS
C ND = NO. OF CURVES DELETED BY THE RESIDUAL ANALYSIS
C KMAX = MAXIMUM NO. OF PARAMETERS ALLOWED
C MAXIMUM COMMON USAGE: X(N),Q(10),REJ(KMAX),W(2*KMAX),R(N),F(KMAX)
C COMMON X(200),Q(200),REJ(200),W(200),R(200),F(200)
C COMMON /LSTSQ/ LAMBDA,NU,EPS,OUTPT,SPD,IMAX,INPLT
C COMMON /RESID/ ZMAX,ICALL
C COMMON /SYM/ B(45),K
C COMMON /PRMCRV/FF(200)
C COMMON /CUT/ CUTAX,CUTRSP,AMIN,AMAX
C DIMENSION AX(H)

100 FORMAT(//,' THE TOTAL NUMBER OF SYMMETRIC PEAKS HAS REAC
' $HED,13,' - THE MAXIMUM ALLOWABLE--------',)
101 FORMAT(1H1,'RESULTS FROM ANALYSIS NUMBER',13,' OF THE RESIDUAL CUR
' $VE: (AFTER THE LEAST SQUARE PROCESS)',13,' THE CUTOFF LEVEL FOR T
' $HE RESIDUAL PEAKS=',1PE14.6)
102 FORMAT(1H1,' THE',13,' NEW SYMMETRIC CURVES ADDED HAVE THE PARAMETER
' $S: (ONLY RESIDUAL PEAKS WITH VALUES > CUTOFF ARE CONSIDERED)',13,
' $(3F15.5)
103 FORMAT(1H1,' THE',13,' OLD SYMMETRIC CURVES REMOVED HAVE THE PARAMET
' $ERS=',13,'(3F15.5))
104 FORMAT(1H1,' THE',13,' NEW PARAMETERS OF THE',13,' SYMMETRIC CURVES
' $(AFTER RESIDUAL ANALYSIS) ARE=',13,'(3F15.5)
105 FORMAT(1H1,' NO NEW SYMMETRIC CURVES HAVE BEEN ADDED',)
106 FORMAT(1H1,' NO OLD SYMMETRIC CURVES HAVE BEEN REMOVED'
107 FORMAT(1H1,' THE VALUES OF THE PARAMETERS REMAIN UNCHANGED AFTER THE
' $RESIDUAL ANALYSIS')
LEVEL 20  RSDCHK  DATE = 73332  17/28/02

108 FORMAT(1H1, 'PLOT OF THE AX) CURVE(+), NEW PARAMETER CURVE(*), AND
$ TOTMAG CURVE(.): (AFTER RESIDUAL ANALYSIS: 12, & BEFORE LST SQR
$ ANALYSIS: 12, 1)

KMAX=45
NADD=0
NN=1
M=N-1
KN=K+1
DX=X(2)-X(1)
NO=0
WRITE(6,101) ICALL, CUTFSD
DO 10 I=1,K
10 WI(I)=X(I)
DO 1 I=2,M
C -------------FIND LOCAL MAXIMA------------
IF(R(I),LT,CUTFSD) GO TO 1
IF(R(I),LE(R(I+1)),OR.R(I),LE(R(I-1))) GO TO 1
C -------------CHECK 2ND DERIVATIVE CRITERIA-----------
DER2=ABS(R(I+1)-2.*R(I)+R(I-1))
DEP1=SQRT(2.*R(I)/DER2)*DX
IF(DEP1,GT,KMAX) GO TO 1
IF(KN-3*NO,GT,KMAX) GO TO 7
NADD=NADD+1
C -------------FIND INITIAL DEPTH---------------
T=DEP(NN,I,R)
C -------------ADD TEMPORARY NEW PEAKS------------
F(NN)=SQRT(T**2+AX(I))
F(NN+1)=T
F(NN+2)=X(I)
C -------------CHECK FOR REPETITIVE PEAKS---------
IF(NN,NEQ,1) GO TO 4
C -------------DELETE NEW PEAK IF WITHIN 2*DX OF OLD PEAK----------
2 DO 3 JJ=1,K,3
51 IF(ABS(F(NN-1)-W(JJ+2)),GE,2,*DX) GO TO 3
NADD=NADD-1
GO TO 4
3 CONTINUE
C -------------ADD NEW RESIDUAL PEAKS--------------
DO 15 J=1,3
15 W(KN+J-1)=F(NN+J-4)
KN=KN+3
4 NN=NN+3
LL=1
1 CONTINUE
IF(NADD.EQ.0) GO TO 8
C ---------CHECK LAST TEMPORARY PEAK---------
DO 16 JJ=1,K,3
52 IF(ABS(F(NN+1)-W(JJ+1)).GE.2.*UX) GO TO 16
NADD=NADD-1
GO TO 8
16 CONTINUE
DO 17 J=1,3
17 W(KN+J-1)=F(NN+J-4)
KN=KN+3
GO TO 8
7 KPMAX=KMAX/3
WRITE(6,100) KPMAX
8 NT=1
KR=ND*3
KL=K+1
C ---------OUTPUT RESULTS---------
IF(NADD.NE.0) GO TO 20
WRITE(6,105)
GO TO 21
20 K=KN-1
WRITE(6,102) NADD,(W(I),I=KL,K)
21 IF(ND.NE.0) GO TO 22
WRITE(6,106)
IF(NADD+ND.NE.0) GO TO 23
WRITE(6,107)
GO TO 30
22 WRITE(6,103) ND,(REJ(I),I=1,KR)
23 CONTINUE
C ---------REARRANGE THE PARAMETER ARRAY---------
DO 6 I=1,K
IF(W(I).EQ.0) GO TO 6
B(NT)=W(I)
NT=NT+1
6 CONTINUE
K=K-3*ND
KL=K/3
C ---------OUTPUT PARAMETER VALUES FOR NEW PEAKS---------
WRITE(6,104) K,KL,(B(I),I=1,K)
30 CONTINUE
IF(NADD+ND.EQ.0) GO TO 32
CALL FUNCT(B,K,FF,N)
DO 35 JT=1,N
   35 R(JT)=AX(JT)-FF(JT)
   CALL PEAKFT(N,AX)
   IF(INPLT.EQ.0) GO TO 31
   ITEM=2*ICALL
   WRITE(6,108) ICALL,ITEM
31 ICALL=ICALL+1
   RETURN
END
SUBROUTINE INPUT(N,TIME,TMAG)

C THIS SUBROUTINE READS TIMES AND TOTAL FIELD MAGNETIC VALUES FROM CARDS
C (BARTLETT FORMAT) AND CONVERTS TIME INTO HOURS

DIMENSION TIME(1),TMAG(1)

100 FORMAT (3I1,F8.1,F2.0,F10.0)
200 FORMAT(///,' DATA SUPPLIED BY THE USER:',)
201 FORMAT(///,' YEAR DAY TIME',/,' START PROFILE:',I4,I6,I5,)
$F4.1,' END PROFILE:',2I6,I5,F4.1)

TIME(1)=0.
N=1
READ(5,100) IY1,ID1,IH1,AM1,AS1,TMAG(1)
AM=AM+AS/60.
1 READ(5,100,END=10) IY,ID,IH,AM,AS,TOTMAG
AM=AM+AS/60.
TEMP =TMD(TY1,ID1,IH1,AM1,IY,ID,[H,AM])/60.
IF(TEMP.EQ.TIME(N)) GO TO 1
N=N+1
IF(N.GT.200) GO TO 10
TMAG(N)=TOTMAG
TIME(N)=TEMP
GO TO 1
10 WRITE(6,200)
11 CONTINUE
WRITE(6,201) IY1,ID1,IH1,AM1,AM
RETURN
ENTRY PRINT SHR,EHR
TE=SHR+IHI
DO 11 I=1,2
IHR=TEM
AMI=(TE-IHR)*60.+AM1
IDI=ID1
IF(AMI.LT.60.) GO TO 12
AM=AMI-60.
IHRI=IHRI+1
12 IF(IHRI.LT.24) GO TO 13
IHRI=IHRI-24
IDI=IDI+1
13 IF(I,.EQ.2) GO TO 11
IDS=IDI
IHS=IHRI
AMS=AMI
TEM=EHR+IHI
RETURN

END
FUNCTION TMDF(Y1, D1, H1, AM1, Y2, D2, H2, AM2)

C THIS FUNCTION FINDS THE TIME DIFFERENCE IN MINUTES

TMDF = 1440*(D2-D1) + 60*(H2-H1) + AM2-AM1

J = Y2 - Y1

IF (J .EQ. 0) RETURN

TMDF = TMDF + 525600*J + 1440*((Y2-1)/4 - (Y1-1)/4)

RETURN

END