

A USERS' GUIDE TO THE CONTOURING ROUTINE
THREAD

By

Carol Withrow

Earth Science Laboratory
University of Utah Research Institute
Salt Lake City, Utah

August, 1980

Table of Contents

ABSTRACT.	1
INTRODUCTION.	2
USAGE.	3
EXAMPLE.	8
ACKNOWLEDGEMENTS.	9
APPENDIX A.	10
APPENDIX B.	13
REFERENCES.	15

ABSTRACT

THREAD is a subroutine that produces a contour map based on gridded three-dimensional data. The user supplies the gridded data and specifies the contours, the area to be contoured, and the scale factor. Major and minor contours with three different pen thicknesses may be specified. Major contours are labeled. Optional latitude and longitude tic marks may be specified.

Map projection options have not yet been implemented.

The output is produced on a plotting device.

INTRODUCTION

The THREAD subroutine package provides the user with the capability of producing a contour plot on a plotting device. Prior to using THREAD, the user has converted his data to grid form. These gridded data, together with parameters describing the contour area, contour lines, and optional tic marks, provide the input to THREAD.

The quality of the plot depends on the quality of the input data. The contouring algorithm is a simple but rapid one, using no more sophisticated mathematics than linear interpolation. We have found at ESL, using the gridding program GRIDIT and a grid size of 0.1 inch, that the results are similar to hand contours. An exception to this is seen occasionally at edges, where anomalies may occur.

The user has a good amount of freedom in selecting the characteristics of the contour plot. The data may be specified in miles, kilometers, meters, feet, inches or centimeters. The vertical and horizontal spacing of the grid may be different. The contour lines selected may be specified as major or minor and are accordingly drawn with different pen widths. Major contour lines are labeled. The user selects the number of decimals to appear in this label and the character size. There can be up to eight minor contours between major contours.

The area to be contoured can be less than the total data area.

One may optionally specify latitude and longitude tic marks. These may be specified in units different from the data units. They may be rotated and labeled. The character size of the label is user-specified.

USAGE

The user's program interfaces with the subroutine package by calling the single subroutine THREAD. All of the data are passed with block common statements.

An example of a main program that calls this package may be seen in Appendix A. Sample output is in Appendix B.

THREAD uses Fortran unit 19 as a scratch file, so the user program should avoid using unit 19.

The user's program specifies the plotting device. Therefore the program must both open and close the plotting file. It may also add such things as labels or page numbers to the plot. Commands initiating plotting, such as SPLOT, are of course typed by the user.

The common blocks should appear in the user's program as follows:

```
PARAMETER MAXGRD = 500
PARAMETER NTIC = 30
```

```
COMMON/THRED1/ AM(MAXGRD, MAXGRD)
```

```
COMMON/THRED2/ M, NN, DELTAY, DELTAX, XO, YO, IUNIT, SCALE
```

```
COMMON/THRED3/ CXO, CYO, CXENT, CYENT, ICUNIT, BORDER(3)
```

```
COMMON/THRED4/ CBGN, CINMAJ, GEND, NMINOR, NDEC, NPOINT, HT,
* SLPTOL
```

```
COMMON/THRED5/ NTREC, XT(NTIC), YT(NTIC), ITUNIT(NTIC), TLEN(NTIC),
* TANGLE(NTIC), LABCOD(NTIC), THT(NTIC), TLABEL(12, NTIC)
```

All the above variable types are the Fortran default types.

All the values in the first four blocks must be defined in the calling program. The values in the fifth block, THRED5, all refer to the latitude and longitude tic marks. If no tic marks are to be drawn, omit this block, or set the first value, NTREC, to zero and ignore the remaining variables in the block.

The variables that must be defined are the following:

AM	=	GRIDDED DATA
M	=	NO. ROWS OF DATA
NN	=	NO. COLUMNS
DELTAX	=	INCREMENT BETWEEN COLUMN VALUES
DELTAY	=	INCREMENT BETWEEN ROWS

XO, YO = DATA ORIGIN
 IUNIT = UNIT CODE FOR DATA
 SCALE = NUMBER INCHES REPRESENTED BY 1" ON MAP
 CXO, CYO = CONTOUR AREA ORIGIN
 CXENT = EXTENT OF CONTOUR AREA IN X DIRECTION
 CYENT = EXTENT OF CONTOUR AREA IN Y
 ICUNIT = UNIT CODE
 BORDER = 'BORDER' IF WANT BORDER DRAWN
 CBGN = VALUE OF BEGINNING CONTOUR
 CEND = VALUE OF LAST CONTOUR
 CINMAJ = INCREMENT BETWEEN MAJOR CONTOURS
 NMINOR = NUMBER OF MINORS BETWEEN MAJORS
 NDEC = NO. DIGITS TO RT OF DECIMAL IN LABEL
 NPOINT = NO. POINTS TO DEFINE STRAIGHT SEGMENT (MIN=4)
 HT = HEIGHT OF CHARACTERS LABELING CONTOUR LINE
 SLPTOL = ARBITRARY NUMBER USED IN LABELING

These variables will now be discussed in more detail in the order in which they have been listed.

AM A doubly dimensioned array containing the gridded data values to be contoured.

M, NN The number of rows and columns in the gridded data. M represents the first subscript, and NN the second in array AM. The upper left corner of the grid is AM(1,1). M and NN each may have a maximum value of 500.

DELTAX Increment between column (second subscript) values of AM in map units.

DELTAY Increment between row (first subscript) values of AM in map units.

XO, YO The coordinates in map units of the data origin, AM(M,1). This is the lower left corner of the grid.

IUNIT The code for the data units:

1 = miles
 2 = kilometers
 3 = feet
 4 = meters
 5 = inches
 6 = centimeters

The following are expressed in these units:
DELTAX, DELTAY, XO, YO.

SCALE Number of inches of mapped area represented by one inch on the plot.

- CXQ,CYQ** Contoured area origin. In the case of contouring a subset of the gridded data, this may be different than XQ,YQ. It represents the lower left corner of the area to be contoured. It may be expressed in different units.
- CXENT** Extent of the contoured area in the X or horizontal direction, expressed in contoured area units. If CXQ is greater than XQ, then this should reflect the remaining width of the contour area. Also, the right side of the data area may be truncated by setting this value appropriately.
- CYENT** Extent of the contoured area in the Y or vertical direction, expressed in contoured area units. If CYQ is greater than YQ, then this should reflect the remaining height of the contour area. Also, the top of the data area may be truncated by setting this value appropriately. If the entire data area is to be contoured, set the two origins the same, and make CXENT and CYENT equal to or greater than the data area.
- ICUNIT** Code for contoured area units (interpreted the same as for IUNIT). The following are expressed in these units: CXQ, CYQ, CXENT, CYENT.
- BORDER** Must be set to the characters 'BORDER' if an outline is to be drawn about the contoured area.
- CBGN** The value of the lowest contour line. This will be interpreted as a major contour and will be drawn in the wide pen width. It will be labeled with its value if there is room.
- CEND** The value of the highest contour line. Be sure CBGN and CEND are within the data range.
- CINMAJ** This is the incremental value to use between major contour lines. The second major contour line has the value CBGN + CINMAJ.
- NMINOR** The number of minor contour lines between each pair of major contour lines.

These are drawn in two narrower pen widths, alternating. Minor contour lines are not labeled. NMINOR may range in value from 0 - 8.

- NDEC The number of digits to the right of the decimal point to be used in the label of major contour lines.
- NPOINT This is the number of points to be removed from a major contour line in order to insert a label. The program searches for a segment this long that is relatively straight. If this search fails, then no label is drawn. The minimum value of NPOINT is four. It will default to this if it is assigned a lesser value. This value assignment is a user option, because it will vary depending on the scale of the map, the size of the grid, the magnitude of the contour values, and the number of decimals. Some experimentation may be required.
- HT This is the height in inches of the label on major contour lines. If not specified, it assumes the default value of 0.07 inches.
- SLPTOL This is a constant used by the algorithm that searches for a straight segment of the contour line where a label can be inserted. The rigor by which "straight" is defined is inversely related to the magnitude of SLPTOL. Ordinarily the user will not be concerned with this variable, except that it must appear in common block THRED4. Its default value is 0.65 .
- NTREC Number of latitude-longitude tic marks to be drawn. If there are none, this is zero. Maximum value is 30.

The variables associated with tic marks are the following:

XT, YT = COORDINATES OF TIC CENTER IN MAP UNITS
ITUNIT = UNIT CODE OF TIC COORDINATES
TLEN = TIC LENGTH IN INCHES
TANGLE = TIC ANGLE OF ROTATION CLOCKWISE IN DEGREES
LABCOD = LABEL CODE, 0 OR 1

THT = CHARACTER HEIGHT OF LABELS IN INCHES
TLABEL = LABELS, 3A4 EACH

All of these are dimensioned variables, and values are assigned for each tic. For example, the unit code for the third tic is in ITUNIT(3). TLABEL is doubly dimensioned. The first dimension is 12. This allows for the first 3 words to be alphanumeric data representing the first label, or the label on the arm pointing east. The succeeding labels are placed in a clockwise order. They are stored in words 4 - 6, 7 - 9, and 10 - 12. The second dimension of TLABEL references the tic.

The origin for tic placement is the contoured area origin (CXD,CYD).

There may be up to 30 tic marks.

LABCOD has the value 1 if labels are to be plotted on the tic, or 0 if there are no labels.

Once the necessary values have been defined, the main program invokes THREAD with the following statement:

```
CALL THREAD
```

There will be a line of text written at the terminal for each contour line found, giving the number of points in the line.

EXAMPLE

The use of THREAD will be better understood if one refers to the sample calling program mentioned above. It may be found in Appendix A. Also there is a sample load module for the program. Notice that the calling program is compiled with the "-BIG" option.

One will notice that the sample program does not have listed all the parameter and block common statements that were listed on page 3. Instead we see:

```
$INSERT CONTOUR>PARAM  
$INSERT CONTOUR>THREAD.COM
```

This is equivalent to entering the nine lines of parameter and block common statements seen on page 3. The choice is left to the user.

One may further notice that the sample program truncates the left and bottom edges of the original data grid by moving the contouring origin and setting the contouring area to be less than the data area. This was done to hide anomolous edge values that came from the gridding program.

The illustration in Appendix B is the output from the program in Appendix A.

ACKNOWLEDGEMENTS

The overall design of this contour mapping package was by Terry Killpack. It was implemented by Carol Withrow. The contour threading algorithm was designed by Dayhoff (1) and implemented by Tobler (2).

APPENDIX A

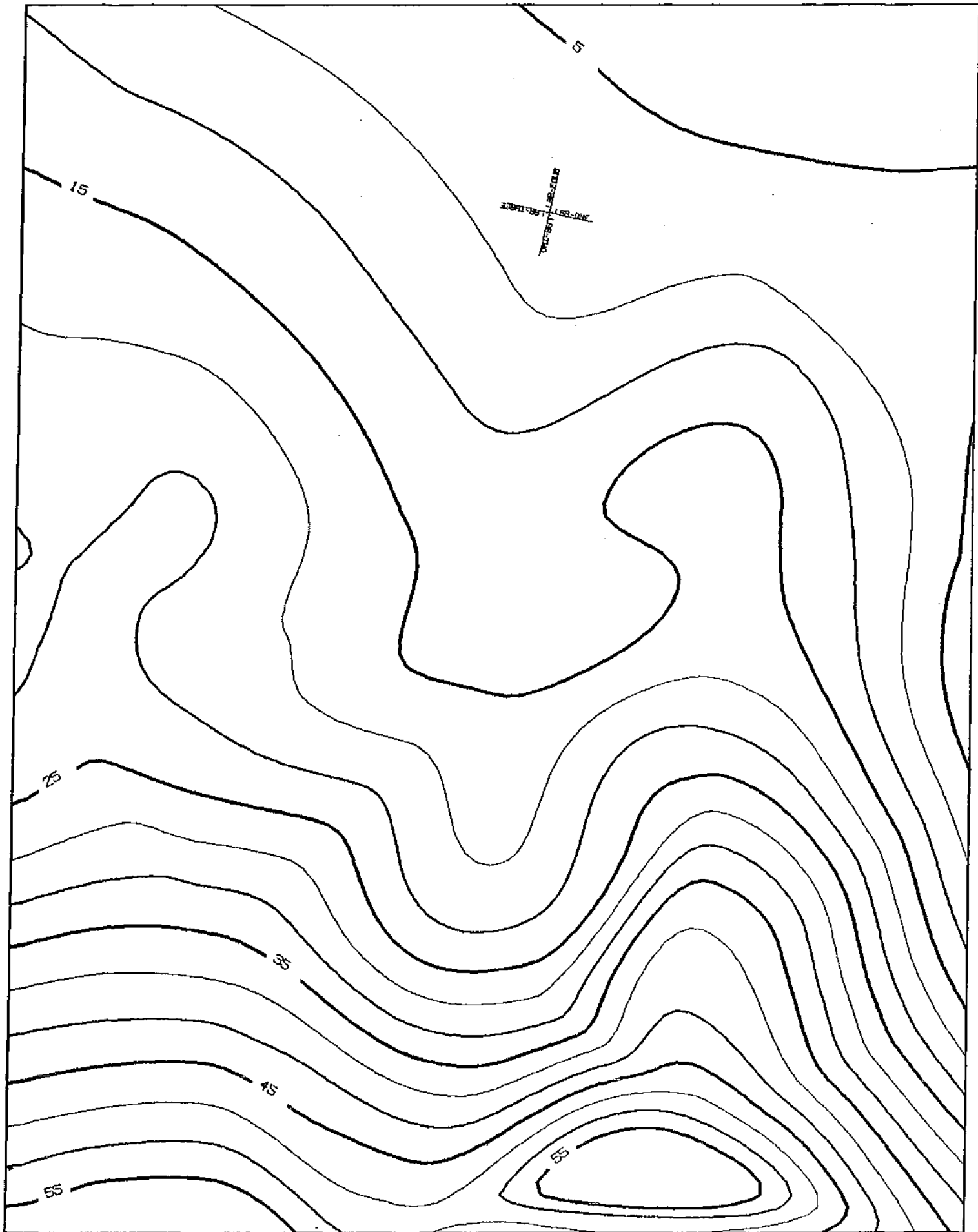
```

C***** EXAMPLE *****
C
C   PROGRAM EXAMPLE
C   THIS IS A TEST OF SUBROUTINE THREAD AND ITS ASSOCIATED SUBROUTINES.
C   INPUT IS FILE GRIDOUT PRODUCED BY PROGRAM TESDAV.
C
C*****
C
$INSERT CONTOUR>PARAM
$INSERT CONTOUR>THREAD.COM
C
C--THE FOLLOWING PARAMETERS DESCRIBE THE GRIDDED DATA
C
    DATA M/100/, NN/80/, DELTAY/.1/, DELTAX/.1/, XO/0./, YO/0./
    DATA IUNIT/3/, SCALE/12./
C
C--THE FOLLOWING DESCRIBE THE CONTOUR AREA AND LINES
C
    DATA CX0/0.3/, CY0/.125/, CXENT/7.7/, CYENT/9.8/, ICUNIT/3/
    DATA BORDER/'BORDER      '/
    DATA CBGN/5./, CINMAJ/10./, CEND/55./, NMINOR/2/, NDEC/0/,
    * NPOINT/10/
C
C--THE FOLLOWING IS THE NUMBER OF TICS TO BE DRAWN
C
    DATA NTREC/1/
C
C--THE FOLLOWING DATA NEED BE SUPPLIED ONLY WHEN PLOTTING TICS
C
    DATA XT(1)/4.2/, YT(1)/8.13/, ITUNIT(1)/3/, TLEN(1)/.7/,
    * TANGLE(1)/12./, LABCOD(1)/1/, THT(1)/.04/,
    * TLABEL(1,1)/'LAB-'/, TLABEL(2,1)/'ONE'/, TLABEL(4,1)/'LAB-'/,
    * TLABEL(5,1)/'TWO'/, TLABEL(7,1)/'LAB-'/, TLABEL(8,1)/'THRE'/,
    * TLABEL(9,1)/'E'/, TLABEL(10,1)/'LAB-'/, TLABEL(11,1)/'FOUR'/,
    * TLABEL(12,1)/' '/, TLABEL(3,1)/' '/, TLABEL(6,1)/' '/
C
C--READ IN THE DATA
C
    CALL OPNRED(8, 'GRIDOUT', 7)
    READ(8, 3000) ((AM(J, K), K=1, NN), J=1, M)
    CALL CLOSEF(8)
C
C--NOW OPEN A PLOT FILE
C
    CALL OPENS(10., 14.)
C
C--SEND EVERYTHING TO THE CONTOURING ROUTINE
C
    CALL THREAD
C
C--CLOSE THE PLOT FILE
C
    CALL CLOSEP
    CALL EXIT
3000  FORMAT(BF7.2)
    END

```

```
* CREATE #EXAMPLE
*
* NOTICE MAIN MUST BE COMPILED
* WITH THE "-BIG" OPTION
*
FOR EXAMPLE -DEBUG -BIG
FILMEM ALL
SEG
LOAD #EXAMPLE
LD B_EXAMPLE
LIB CW
LIB UUPLT
LIB ESL
LIB VAPPLB
LIB
SAVE
MA MAPIT
QUIT
DELETE B_EXAMPLE
CO -END
```

APPENDIX B



REFERENCES

1. M. D. Dayhoff, "A Contour-Map Program for X-ray Crystallography," Communications ACM, 6, 10 (October, 1963), pp. 620 - 622.
2. Waldo R. Tobler, Ed., Selected Computer Programs, Department of Geography, Univ. of Mich., Ann Arbor, 1970, pp. 89 - 109.

