UR1985_42

1/17

1985/42. Contouring using the Department of Mines Perkin-Elmer mini-computer. Part A: Small quantities of raw data (Revision 1)

R.G. Richardson

Abstract

The program described here uses a modified version of ACORD (Watson, 1972) for contouring up to 1200 data points without prior gridding. Singularities in the data, which cause a divide by zero error when running the program, are removed by slightly perturbing the x co-ordinate of the data value and then returning to the start of the subroutine. This revision corrects an error in the call of ACORD and modifies the method of perturbation.

THE PROGRAM

NUCNT5 (Appendix 1)

This program was written for contouring gravity data, but any data in a suitable format can be contoured. The program prompts for commands on logical unit 3, reads commands on logical unit 5, and reads the data from logical unit 2. Logical unit 7 is a temporary work-file and should have a record length of 80 characters.

Control data read from logical unit 5 is:

- XMIN - the western boundary of the area to be contoured (km)
- XMAX - the eastern boundary of the area to be contoured (km)
- YMIN - the southern boundary of the area to be contoured (km)
- YMAX - the northern boundary of the area to be contoured (km)
- TITLE - the title for the map followed by &
- CONTLOW - the minimum contour value
- CONTHIGH - the maximum contour value
- CONTOUR INTERVAL - the contour interval
- NPT - should be set to a value between 0 and 6. The value is not, however, used
- NTHK - every NTHK contour starting at CONTLOW will be made thicker. If NTHK = 0 no contours will be thickened.
- IOPT - select the functions desired from the program
  - (a) IOPT = 0
    draw contours only
  - (b) IOPT = 1
    draw contours and mark data points
  - (c) IOPT = 2
    draw contours, mark data points and write data values at the data points.
- SCALE - the plotting scale. *e.g.* 50 000.
  The labelling used depends on the value of SCALE. If the map will not plot on the available paper width, a new scale will be requested.

Data input from logical unit 2 is;

SURVNO, STATNO, X, Y, BA - format (I5, 1X, I4, 2 F 9.1, 36X, F8.2) where SURVNO is the survey number
STATNO is the station number within the survey
X, Y are the east and north co-ordinates of the data point in metres (X, Y in the range 0.0 to 700000).
BA is the data value.

The program should be given no more than 1200 data points to contour. The contours are not labelled and are drawn as short, widely dispersed segments so plotting times may be very long.

The sample plots took 4.7 seconds of processor time to generate the plot file for the 57 point data set (Appendix 2) and 157.2 seconds for the 567 point data set (Appendix 1).

REFERENCE

WATSON, D.F. 1982. ACORD: automatic contouring of raw data. *Computers and Geosciences*. 8:97-101.

[25 July 1985]

APPENDIX 1

Program NUCNT5

```
L NUCNT5
AS 2,@1.DAT,ERO
AS 3,CON:
AS 5,CON:
TEMPFILE 7,IN,80
ST
$EXIT
```

```
$TITL NUCNT5 - A PROGRAM FOR CONTOURING SMALL QUANTITIES OF RAW DATA
C LU  2  ORIGINAL FIELD DATA
C LU  3  PROMPTS FOR DATA ENTRY
C LU  5  UNIT FROM WHICH MAP SCALE,CONTOUR RANGE AND INTERVAL ARE READ
C LU  7  A TEMPORARY WORK FILE
C PRECISION MAY BE A PROBLEM - HENCE THE USE OF REAL*8
      DIMENSION AMAP(1203,3),TTLE(18),CONT(50)
      REAL*8 AMAP,CONT
      COMMON /TRFORM/ SFACT
      LOGICAL PNUP,THICK,LABL
      INTEGER*2 SURVEY(50),STRING
C SFACT IS
C THE SCALE FACTOR TO CONVERT FROM CONTOUR COORDS TO PLOTTER INCHES
      REWIND 7
C TEMP FILE
      WRITE(3,102)
  102 FORMAT(' XMINNN  XMAXXX IN KM')
      READ(5,*) XMIN,XMAX
      WRITE(3,104)
  104 FORMAT(' YMINNN  YMAXXX IN KM')
      READ(5,*) YMIN,YMAX
      WRITE(3,106)
  106 FORMAT(' MAP TITLE - TERMINATE BY &')
      READ(5,107) TTLE
  107 FORMAT(18A4)
      SPACE=1.0
      WRITE(7,108) XMIN,XMAX,YMIN,YMAX,SPACE,TTLE
  108 FORMAT(5F10.3/18A4)
C READ THE LIMITS AND TITLE AND PUT IN TEMP FILE TO ALLOW
C USE OF BORDER
      CALL DATAIN(AMAP,NX,XMIN,XMAX,YMIN,YMAX,FXMIN,FYMIN)
C READ THE DATA FILE
      CALL BORDER(NUMCON,ABOT,AINTT,TTLE,SURVEY,NPT,
     . NTHK)
C GET ALL ARGUMENTS AND DRAW FRAME
      CALL PLOT(0.0,0.0,0)
      XTMP=FXMIN-XMIN
      YTMP=FYMIN-YMIN
      CALL XFORM(XTMP,YTMP)
      CALL PLOT(XTMP,YTMP,-3)
C SET THE ORIGIN AS THE SOUTH-WEST CORNER OF THE DATA
      DO 20 I=1,NUMCON
      CONT(I)=ABOT
      ABOT=ABOT+AINTT
   20 CONTINUE
C STORE CONTOUR VALUES
      WRITE(3,999) NX
  999 FORMAT(' NO OF POINTS',I6)
      CALL ACORD(AMAP,NX+3,NUMCON,CONT,NTHK)
      CALL RSTR(2)
      END
      SUBROUTINE DATAIN(AMAP,NX,XMIN,XMAX,YMIN,YMAX,FXMIN,FYMIN)
      DIMENSION AMAP(1203,3)
      REAL*8 AMAP
      NX=4
C FIRST VALUE IS TO BE IN POSITION 4 FOR ACORD
      FXMIN=9.9E9
      FYMIN=FXMIN
C LOOKING FOR THE ACTUAL X AND Y MIN COORD FOR THE DATA SELECTED
```

```
      FZMIN=FXMIN
      FZMAX=-FZMIN
C USED TO GET THE DATA RANGE
      REWIND 2
   10 READ(2,100,END=110) XTMP,YTMP,ZTMP
  100 FORMAT(10X,2F9.1,36X,F8.2)
      XTMP=XTMP*0.001
      YTMP=YTMP*0.001
C CONVERT TO KM
      IF (XTMP .LT. XMIN .OR. XTMP .GT. XMAX .OR. YTMP .LT. YMIN
     .    .OR. YTMP .GT. YMAX) GOTO 10
C DATA POINT NOT WITHIN THE SELECTED AREA
      IF (NX .GT. 4) THEN
      NUP=NX-1
      DO 20 I=1,NUP
      IF (DABS(AMAP(I,1)-XTMP) .LT. 0.0015 .AND. DABS(AMAP(I,2)-YTMP)
     .   .LT. 0.0015) GOTO 10
C TAKE THE FIRST OF CLOSE VALUES
   20 CONTINUE
      ENDIF
      FXMIN=AMIN1(FXMIN,XTMP)
      FYMIN=AMIN1(FYMIN,YTMP)
      FZMIN=AMIN1(FZMIN,ZTMP)
      FZMAX=AMAX1(FZMAX,ZTMP)
C FIND THE RANGE OF THE DATA VALUES
      AMAP(NX,1)=XTMP
      AMAP(NX,2)=YTMP
      AMAP(NX,3)=ZTMP
      NX=NX+1
      IF (NX .GT. 1203) PAUSE 'TOO MANY DATA POINTS'
      GOTO 10
  110 NX=NX-4
      WRITE(3,200) FZMIN,FZMAX
  200 FORMAT(' RANGE OF INPUT DATA',F10.4,' TO',F10.4)
      RETURN
      END
$INCLUDE BORDCNT.FTN,-
$INCLUDE ACORD.FTN,-
```

```
      SUBROUTINE XFORM(A,B)
      COMMON /TRFORM/ SFACT
      A=A*SFACT
      B=B*SFACT
      RETURN
      END
      SUBROUTINE BORDER(NUMCON,ABOT,AINTT,TITLE,SURVEY,NPT,
     . NTHK)
      COMMON /TRFORM/ SFACT
      DIMENSION TITLE(18)
      INTEGER*2 SURVNO,NSURV,SURVEY(50),STATNO,AOK,YES,NO
      DOUBLE PRECISION XMIN,XMAX,YMIN,YMAX
      INTEGER*4 IXMIN,IXMAX,IYMIN,IYMAX,SPCE,DFACT,TXSP,DX,DY,
     . IFXMIN,IFXMAX,IFYMIN,IFYMAX,XSP,YSP
      LOGICAL SURV,LABEL,METRES,LINE
      DATA YES/1HY/,NO/1HN/
      REWIND 7
C REWIND GRIDDED DATA
      READ(7,101) XMIN,XMAX,YMIN,YMAX,SPACE,TITLE
  101 FORMAT(5F10.3/18A4)
      WRITE(3,108)
  108 FORMAT(' CONTLOWWW CONTHIGHH')
      READ(5,*) ABOT,ATOP
      WRITE(3,105)
  105 FORMAT(' CONTOUR INTERVAL')
      READ(5,*) AINTT
    9 WRITE(3,106)
  106 FORMAT(' NUMBER OF PLACES AFTER DEC. PT. ON CONTOUR LABELS')
      READ(5,*) NPT
      IF (NPT .GT. 6) GOTO 9
C TOO MANY PLACES
      IF (NPT .LE. 0) NPT=-1
C IF DONT WANT DECIMAL PLACES DONT WANT DEC. PT.
   22 WRITE(3,110)
  110 FORMAT(' EVERY N TH. CONTOUR HEAVY')
      READ(5,*) NTHK
      IF (NTHK .LT. 0) GOTO 22
C MAKE EVERY NTHK CONTOUR THICK
      WRITE(3,103)
  103 FORMAT(' OPTIONS'/
     .  '  0  CONTOURS ONLY'/
     .  '  1  STATION LOCATIONS'/
     .  '  2  STATION NUMBERS')
      READ(5,*) IOPT
      CALL INITAL(9,200,36,0,0,0)
      NSURV=1
C COUNTER FOR NUMBER OF SURVEYS
      NUMCON=((ATOP-ABOT)/AINTT)+1.0
C NUMBER OF CONTOURS
      SURV=.FALSE.
C
      XMIN=XMIN*1000.0D00
      IXMIN=XMIN
      XMAX=XMAX*1000.0D00
      IXMAX=XMAX
C CONVERT TO METRES
      YMIN=YMIN*1000.0D00
      IYMIN=YMIN
      YMAX=YMAX*1000.0D00
```

```
      IYMAX=YMAX
C CONVERT TO METRES
  102 WRITE(3,100)
  100 FORMAT(' SCALE E.G.  50000.')
      READ(5,*) SCALE
      FACT=1.E+2/(2.54*SCALE)
C THE FACTOR TO CONVERT FROM GRID METRES TO PLOTTER INCHES
      SFACT=FACT*1000.0
C TO CONVER FROM GRID KM TO PLOTTER INCHES
      LABEL=.TRUE.
      METRES=.TRUE.
      IF (SCALE .LE.  2001.) GOTO 851
      IF (SCALE .LE.  5001.) GOTO 852
      IF (SCALE .LE.  25005.) GOTO 853
      IF (SCALE .LE.  100010.) GOTO 854
C
C COPE WITH SCALE GREATER THAN 100000.  HERE
      METRES=.FALSE.
      DFACT=10000
      SPCE=100000
      TXSP=10000
      GOTO 855
C
C SCALE .LE.  100000,SCALE .GT.  25000
  854 METRES=.FALSE.
      DFACT=1000
      TXSP=1000
      SPCE=10000
      GOTO 855
C
C SCALE .LE.  25000,SCALE .GT.  5000
  853 DFACT=250
      TXSP=250
      SPCE=1000
      GOTO 855
C
C SCALE .LE.  5000,SCALE .GT.  2000
  852 DFACT=100
      TXSP=100
      SPCE=500
      GOTO 855
C
C SCALE .LE.  2000
  851 DFACT=100
      LABEL=.FALSE.
      TXSP=10
      SPCE=100
  855 CONTINUE
      IFXMIN=((IXMIN+1)/DFACT)*DFACT
      FXMIN=IFXMIN
      IFXMAX=((IXMAX+DFACT-1)/DFACT)*DFACT
      FXMAX=IFXMAX
      IFYMIN=((IYMIN+1)/DFACT)*DFACT
      FYMIN=IFYMIN
      IFYMAX=((IYMAX+DFACT-1)/DFACT)*DFACT
      FYMAX=IFYMAX
C GET THE RANGES AND ROUND UP AND DOWN TO THE NEAREST PIP
C GET THE MAXIMUM Y VALUE TO WORK OUT IF THE PLOT CAN POSSIBLY
C FIT ON THE PAGE
      YTMP=FYMAX
```

```
      CALL GDFORM(YTMP,FYMIN,FACT)
      YTMP=YTMP+2.1
C GET TOTAL WIDTH OF NEW PLOT INCLUDING TICKS AND LABELS
      IF (YTMP .LE. 35.5) GOTO 870
C WILL FIT ON PAPER
      WRITE(3,230)
  230 FORMAT(' PLOT WILL NOT FIT')
      GOTO 102
  870 CONTINUE
      CALL PLOT(1.0,2.0,-3)
C MOVE TO 1.0,2.0 AND CALL IT 0.,0.
C
      FACTMK=1.0
C THE FACTOR FOR SCALING THE SYMBOLS
  232 WRITE(3,233)
  233 FORMAT(' SMALL SYMBOLS? (Y OR N)')
      READ(5,229) AOK
  229 FORMAT(A1)
      IF (AOK .EQ. NO) GOTO 234
      IF (AOK .NE. YES) GOTO 232
      FACTMK=0.5
  234 CONTINUE
C
C
C NOW FOR SOME AXES
      DX=IFXMIN-TXSP
   10 DX=DX+TXSP
      XTMP=DX
      CALL GDFORM(XTMP,FXMIN,FACT)
      CALL PLOT(XTMP,0.,2)
C MOVE TO START OF PIP
      LINE=MOD(DX,SPCE) .EQ. 0
C CHECK TO SEE IF NEED TO DRAW LINE
      CALL XLABEL(DX,FYMIN,FYMAX,FACT,XTMP,LABEL,LINE,METRES)
      IF (IFXMAX .GT. DX) GOTO 10
C GO ALONG BOTTOM AND FINISH AT A PIP WITH DX=FXMAX,XTMP=FXMAX IN PLOTTER INCH
C
C UP R.H.S.
      DY=IFYMIN-TXSP
   11 DY=DY+TXSP
      YTMP=DY
      CALL GDFORM(YTMP,FYMIN,FACT)
      CALL PLOT(XTMP,YTMP,2)
      LINE=MOD(DY,SPCE) .EQ. 0
C CHECK TO SEE IF MULTIPLE
      CALL YLABEL(DY,FXMIN,FXMAX,FACT,YTMP,LABEL,LINE,METRES)
      IF (IFYMAX .GT. DY) GOTO 11
C NOW GONE UP R.H.S.   DY=FYMAX
C                        YTMP=FYMAX IN PLOTTER COORDS.
C
      CALL PENDN
      XUP=XTMP+0.8
      YUP=YTMP+0.5
C USED FOR DRAWING FRAME
   12 XTMP=DX
      CALL GDFORM(XTMP,FXMIN,FACT)
      CALL PLOT(XTMP,YTMP,1)
      CALL PLOT(XTMP,YTMP+0.1,1)
      CALL PLOT(XTMP,YTMP,1)
      DX=DX-TXSP
```

```
      IF (DX .GE. IFXMIN) GOTO 12
C DRAWN PIPS ALONG TOP
C
   14 YTMP=DY
      CALL GDFORM(YTMP,FYMIN,FACT)
      CALL PLOT(0.,YTMP,1)
      CALL PLOT(-0.1,YTMP,1)
      CALL PLOT(0.,YTMP,1)
      DY=DY-TXSP
      IF (DY .GE. IFYMIN) GOTO 14
C DRAWN PIPS DOWN L.H.S.
      CALL PLOT(-0.75,-2.0,3)
      CALL PLOT(XUP,-2.0,2)
      CALL PLOT(XUP,YUP,1)
      CALL PLOT(-0.75,YUP,1)
      CALL PLOT(-0.75,-2.0,1)
C DRAW FRAME
      CALL PENUP
      CALL PWRITE(0.5,-1.0,0.28,0.,TITLE)
C THE TITLE IS PACKED BY THE READ
C WRITE THE TITLE
      ENCODE(SURVEY,301) AINTT
  301 FORMAT('CONTOUR INTERVAL',F8.2,'&')
      DECODE(SURVEY,302) TITLE
  302 FORMAT(18A4)
      CALL PWRITE(1.5,-1.75,0.14,0,TITLE)
C SET UP THE CONTOUR INTERVAL AND WRITE
      IF (IOPT .EQ. 0) GOTO 24
C SKIP IF NOT PLOTTING STATIONS
C
C NOW TO GO THROUGH THE DATA
      REWIND 2
C REWIND RAW DATA FILE
   30 READ(2,104,END=50) SURVNO,STATNO,X,Y
  104 FORMAT(I5,1X,I4,2F9.1)
      IF (X .LT. XMIN .OR. X .GT. XMAX .OR. Y .LT. YMIN
     . .OR. Y .GT. YMAX) GOTO 30
C CHECK TO SEE IF IN AREA
      IF (SURV) GOTO 53
      SURV=.TRUE.
      SURVEY(1)=SURVNO
   53 CONTINUE
      DO 15 ISYMB=1,NSURV
      IF (SURVEY(ISYMB) .EQ. SURVNO) GOTO 20
C ALREADY KNOW THIS NUMBER
   15 CONTINUE
      NSURV=NSURV+1
      ISYMB=NSURV
      SURVEY(NSURV)=SURVNO
C ADD NEW SURVEY NUMBER TO INDEX
   20 CONTINUE
      ASURV=STATNO
C GOT STATION NUMBER ALMOST AS AN INTEGER
      CALL GDFORM(X,FXMIN,FACT)
      CALL GDFORM(Y,FYMIN,FACT)
      CALL PLOT(X,Y,1)
      CALL FACTOR(FACTMK)
      IF (ISYMB .GT. 14) ISYMB=14
C ONLY HAVE 14 SYMBOLS AVAILABLE
      CALL MARKER(ISYMB)
```

```
      CALL FACTOR(1.)
      IF (IOPT .EQ. 1) GOTO 30
C ONLY WANTED THE MARKER
      IF (IOPT .EQ. 2) CALL
     . NUMBER(X+0.07,Y,FACTMK*0.07,ASURV,0.,-1)
      GOTO 30
C PLOTTED AND LABELLED STATION
   50 CONTINUE
C END OF RUN WHEN GET TO HERE SO TIDY UP A LITTLE
      IF (.NOT. SURV) GOTO 24
C NO STATIONS IN AREA
      IF (NSURV .EQ. 1 .AND. SURVEY(1) .EQ. 0) GOTO 24
C HAD DATA WITHOUT SURVEY NUMBERS
      X=0.5
      Y=-1.2
      DO 51 I=1,NSURV
      IMIN=12000
      DO 52 J=1,NSURV
      IF (SURVEY(J) .EQ. 0) GOTO 52
C ALREADY AND PRINTED IN ORDER
      IF (SURVEY(J) .GT. IMIN) GOTO 52
C ALREADY LOOKING AT AN EARLIER SURVEY
      IMIN=SURVEY(J)
      ISYMB=J
   52 CONTINUE
      CALL PLOT(X,Y,1)
      J=ISYMB
      IF (ISYMB .GT. 14) ISYMB=14
      CALL MARKER(ISYMB)
      ASURV=IMIN
      CALL NUMBER(X+0.15,Y-0.07,.14,ASURV,0.,-1)
      SURVEY(J)=0
      X=X+1.0
C AND SPACE ALONG A BIT
   51 CONTINUE
   24 CONTINUE
      XTMP=XMIN
      YTMP=YMIN
      CALL GDFORM(XTMP,FXMIN,FACT)
      CALL GDFORM(YTMP,FYMIN,FACT)
      CALL PLOT(XTMP,YTMP,1)
      SFACT=SFACT*SPACE
      CALL PLOT(SFACT,SFACT,0)
C GET CONVERSION FROM CONTOUR COORDS TO PLOTTER INCHES
C AND RESET ORIGIN
      RETURN
      END
      SUBROUTINE GDFORM(X,XBASE,FACT)
C FIND THE POSITION OF X RELATIVE TO XBASE IN PLOTTER INCHES
C AND RETURNS THE ANSWER IN X!!!!!!!!!!
      X=(X-XBASE)*FACT
      RETURN
      END
      SUBROUTINE XLABEL(IX,YMIN,YMAX,FACT,XT,LABEL,LINE,METRES)
      LOGICAL LABEL,LINE,METRES
      REAL TITLE(2)
      DATA TITLE/3HM E,4HKM E/
      CALL PLOT(XT,-0.1,1)
      IF (.NOT. LABEL .AND. .NOT. LINE) GOTO 10
C ONLY WANT PIP
```

```
      IIX=IX
      IF (.NOT. METRES) IIX=IIX/1000
      CALL NUMPRT(XT-0.035,-0.15,IIX,270.)
      IF (.NOT. LINE) GOTO 10
      CALL WHERE(XX,YY,AX)
C FIND THE PEN
      CALL MOVREL
      IF (METRES) CALL SYMBOL(XX,YY,0.07,TITLE(1),270.,3)
      IF (.NOT. METRES) CALL SYMBOL(XX,YY,0.07,TITLE(2),270.,4)
      CALL MOVABS
      YY=YMAX
      CALL GDFORM(YY,YMIN,FACT)
      CALL PLOT(XT,YY+0.1,1)
      CALL PENDN
   10 CALL PLOT(XT,0.,1)
      RETURN
      END
      SUBROUTINE YLABEL(IY,XMIN,XMAX,FACT,YT,LABEL,LINE,METRES)
      LOGICAL LINE,LABEL,METRES
      REAL TITLE(2)
      DATA TITLE/3HM N,4HKM N/
C USE FOR LABELLING IN SYMBOL
      X=XMAX
      CALL GDFORM(X,XMIN,FACT)
      CALL PLOT(X+0.1,YT,1)
      IF (.NOT. LABEL .AND. .NOT. LINE) GOTO 10
C ONLY WANTED PIP
      IIY=IY
      IF (.NOT. METRES) IIY=IIY/1000
      CALL NUMPRT(X+0.15,YT-0.035,IIY,0.0)
      IF (.NOT. LINE) GOTO 10
      CALL WHERE(XX,YY,AY)
C FIND PEN
      CALL MOVREL
      IF (METRES) CALL SYMBOL(XX,YY,0.07,TITLE(1),0.0,3)
      IF (.NOT. METRES) CALL SYMBOL(XX,YY,0.07,TITLE(2),0.,4)
      CALL MOVABS
      CALL PLOT(-0.1,YT,1)
      CALL PENDN
   10 CALL PLOT(X,YT,1)
      RETURN
      END
      SUBROUTINE NUMPRT(X,Y,IVAR,ANG)
C TO WRITE THE INTEGER IVAR WITH NO LEADING BLANKS
C STARTING AT X,Y AT AN ANGLE ANG
C A HEIGHT OF 0.07 IS ASSUMED
      INTEGER*4 BUFF(3)
      ENCODE(BUFF,100) IVAR
  100 FORMAT(I10,'&')
      K=0
      DO 10 I=1,11
      CALL ILBYTE(IB,BUFF,I-1)
      IF (IB .EQ. 32) GOTO 10
C SKIP SPACES
      CALL ISBYTE(IB,BUFF,K)
      K=K+1
   10 CONTINUE
      CALL PWRITE(X,Y,0.07,ANG,BUFF)
      RETURN
      END
```

```
$TITL  SUBROUTINE ACORD - RAW DATA CONTOURING
      SUBROUTINE ACORD(PNT,N,LC,CONT,NTHK)
C.....ACORD - AUTOMATIC CONTOURING OF RAW DATA - D.F.WATSON
C.....READ DATA POINTS AND FORM ALL 3-TUPLES S.T.  NO OTHER POINT
C     LIES WITHIN THAT 3-TUPLES CIRCUMCIRCLE
C.....PNT HOLDS THE DATA POINTS AND TETR CARRIES THE CIRCUMCIRCLE
C     CENTRE AND RADIUS SQUARED FOR EACH 3-TUPLE
C.....ITETR HOLDS THE DATA POINT INDICES IN INPUT ORDER OF EACH 3-TUPLE
C.....ISTACK IS A LAST-IN-FIRST-OUT STACK OF INDICES OF VACANT 3-TUPLES
C.....KTETR IS A TEMPORARY LIST OF EDGES OF DELETED 3-TUPLES
C.....ID IS POINTER TO ISTACK AND JT IS POINTERTO TETR AND ITETR
C   BASED ON:-
C   WATSON,D.F.,1982.   ACORD:AUTOMATIC CONTOURING OF RAW DATA
C                       COMPUTERS&GEOSCIENCES V8 PP97-101
C
C USES REAL*8 TO TRY TO AND INCREASE ACCURACY -SEEMS TO WORK
C SINGULARITIES ARE OVERCOME BY PERTUBING ONE OF THE ERRANT DATA
C POINTS BY A SMALL AMOUNT AND THEN RESCALING EVERYTHING AND
C RETURNING TO THE START OF THE SUBROUTINE.  THIS IS SLOW BUT IT
C GETS THERE AND THE ERROR WOULD NOT NORMALLY BE SEEN ON A PLOT.
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 PNT(1203,3), TETR(2401,3), XPNT(3,3), CONT(50), DET(2,3)
      INTEGER ITETR(2401,3), ISTACK(2401), KTETR(50,2), ITEMP(3,2)
      COMMON /TRFORM/ SFACT
      REAL*4 X1,X2,Y1,Y2,SFACT,DIST
      DATA ITEMP,XPNT/1,1,2,2,3,3,-1.,5.,-1.,-1.,-1.,5.,2.,2.,18./
      DATA ISP,ID,XMIN,YMIN,XMAX,YMAX/1,2,1.D50,1.D50,-1.D50,-1.D50/
    1 CONTINUE
C LOOP BACK TO HERE IF HAVE TO PERTUBE A DATA POINT
      ISP=1
      ID=2
      XMIN=1.D50
      XMAX=-XMIN
      YMIN=XMIN
      YMAX=XMAX
      N=N-3
C INITIALLISE ALL DATA STATEMENT VARIABLES HERE IF CHANGED TO ALLOW LOOPING
      DO 2 I = 1,3
        ITETR(1,I) = I
        TETR(1,I) = XPNT(I,3)
        DO 2 J = 1,2
    2    PNT(I,J) = XPNT(I,J)
      DO 6 I = 2,2401
    6  ISTACK(I) = I
      N = N + 3
C.....SCAN THE DATA
      DO 10 I = 4,N
        IF(PNT(I,1) .GT. XMAX) XMAX = PNT(I,1)
        IF(PNT(I,1) .LT. XMIN) XMIN = PNT(I,1)
        IF(PNT(I,2) .GT. YMAX) YMAX = PNT(I,2)
   10   IF(PNT(I,2) .LT. YMIN) YMIN = PNT(I,2)
      DATAX = XMAX - XMIN
      Y = YMAX - YMIN
      IF(Y .GT. DATAX) DATAX = Y
      SIZE=DATAX*SFACT
C SET THE OUTPUT SIZE FACTOR
C.....NORMALIZE THE DATA
      DO 12 I = 4,N
        PNT(I,1) = (PNT(I,1) - XMIN) /DATAX
```

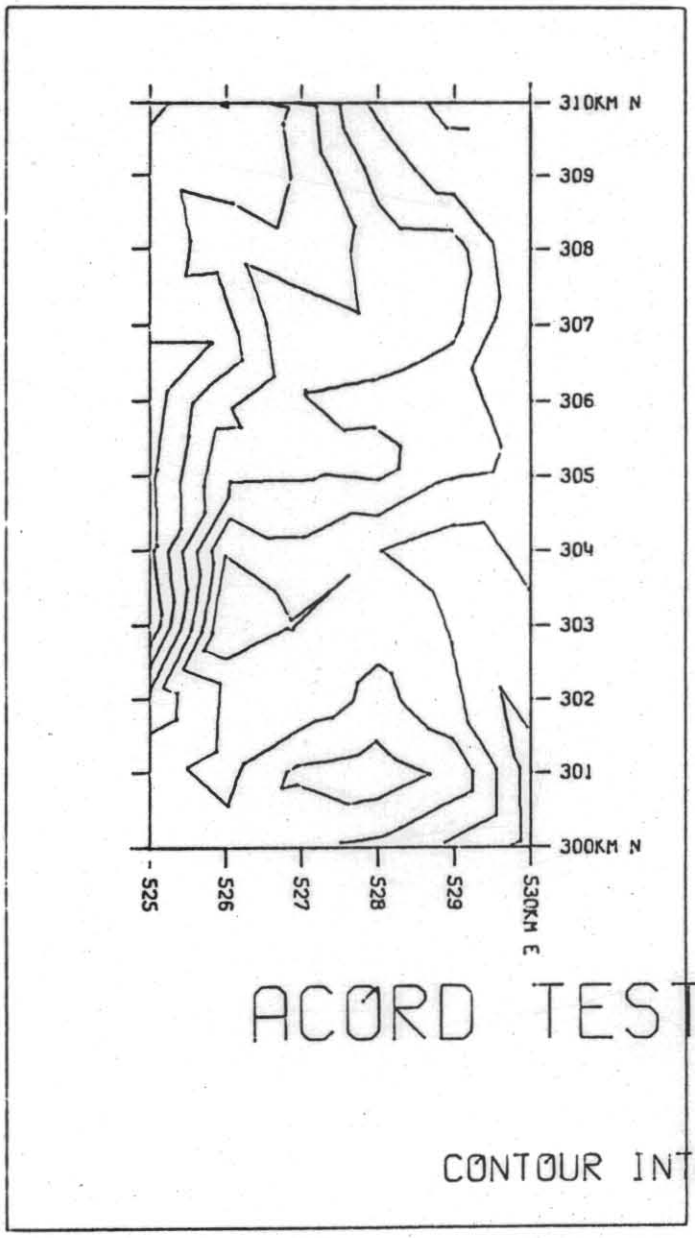
```
   12   PNT(I,2) = (PNT(I,2) - YMIN) /DATAX
      DO 50 NUC = 4,N
        KM = 0
C.....LOOP THRU THE ESTABLISHED 3-TUPLES
        DO 30 JT = 1,ISP
C.........TEST IF NEW DATA POINT IS WITHIN THE JT CIRCUMCIRCLE
          DSQ = TETR(JT,3) - (PNT(NUC,1) - TETR(JT,1))**2
          IF(DSQ .LT. 0.0) GO TO 30
          DSQ = DSQ - (PNT(NUC,2)-TETR(JT,2))**2
          IF(DSQ .LT. 0.0) GO TO 30
C.........DELETE THIS 3-TUPLE BUT SAVE ITS EDGES
          ID = ID - 1
          ISTACK(ID) = JT
C.........ADD EDGES TO KTETR BUT DELETE IF ALREADY LISTED
          DO 28 I = 1,3
            L1 = ITEMP(I,1)
            L2 = ITEMP(I,2)
            IF(KM .LE. 0) GO TO 26
            KMT = KM
            DO 24 J = 1,KMT
              IF(ITETR(JT,L1) .NE. KTETR(J,1)) GO TO 24
              IF(ITETR(JT,L2) .NE. KTETR(J,2)) GO TO 24
              KM = KM - 1
              IF(J .GT. KM) GO TO 28
              DO 20 K = J,KM
                K1 = K + 1
                DO 20 L = 1,2
   20             KTETR(K,L) = KTETR(K1,L)
              GO TO 28
   24         CONTINUE
   26       KM = KM + 1
            KTETR(KM,1) = ITETR(JT,L1)
            KTETR(KM,2) = ITETR(JT,L2)
  28        CONTINUE
  30      CONTINUE
C.....FORM NEW 3-TUPLES
        DO 48 I = 1,KM
          KT = ISTACK(ID)
          ID = ID + 1
C.........CALCULATE THE CIRCUMCIRCLE CENTER AND RADIUS
C         SQUARED OF POINTS KTETR(I,*) AND PLACE IN TETR(KT,*)
          DO 44 JZ = 1,2
            I2 = KTETR(I,JZ)
            DET(JZ,1) = PNT(I2,1) -  PNT(NUC,1)
            DET(JZ,2) = PNT(I2,2) - PNT(NUC,2)
   44       DET(JZ,3) = DET(JZ,1)*(PNT(I2,1) + PNT(NUC,1))/2.0
     +                 + DET(JZ,2)*(PNT(I2,2) + PNT(NUC,2))/2.0
          DD = DET(1,1) * DET(2,2) - DET(1,2) * DET(2,1)
      IF (ABS(DD) .LE. 0.000005) THEN
      IF (NUC .GE. 4) THEN
C NEED TO ENSURE NOT ALTERING ONE OF THE FIRST THREE THAT
C IS RESET LATER
      PNT(NUC,1)=PNT(NUC,1)+0.0001
      ELSE
      IF (I2 .GE. 4) THEN
      PNT(I2,1)=PNT(I2,1)+0.0001
      ELSE
      IF (KTETR(I,1) .LE. 3) THEN
      PAUSE 'ALL THREE POINTS ARE CONSTANTS'
      ELSE
```

```
      PNT(KTETR(I,1),1)=PNT(KTETR(I,1),1)+0.0001
      ENDIF
      ENDIF
      ENDIF
C PERTURB BY SLIGHTLY MORE THAN THE MACHINE ACCURACY  TO FIX SINGULARITY
      WRITE(3,990)
  990 FORMAT(' SINGULARITY AT INDICES AND COORDINATES - ')
      DO 45 KP=4,N
      PNT(KP,1)=PNT(KP,1)*DATAX+XMIN
      PNT(KP,2)=PNT(KP,2)*DATAX+YMIN
   45 CONTINUE
      WRITE(3,992) NUC,KTETR(I,1),I2
  992 FORMAT(3I5)
      WRITE(3,993) PNT(NUC,1),PNT(NUC,2),PNT(KTETR(I,1),1),
     . PNT(KTETR(I,1),2),PNT(I2,1),PNT(I2,2)
  993 FORMAT(6F10.3)
C RESCALE COORDS TO INITIAL VALUES
      GOTO 1
C AND LOOP BACK ROUND AGAIN
      ENDIF
          TETR(KT,1) = (DET(1,3) * DET(2,2) - DET(2,3) * DET(1,2))/DD
          TETR(KT,2) = (DET(1,1) * DET(2,3) - DET(2,1) * DET(1,3))/DD
          TETR(KT,3) = (PNT(NUC,1) - TETR(KT,1))**2
     +                 + (PNT(NUC,2) - TETR(KT,2))**2
          ITETR(KT,1) = KTETR(I,1)
          ITETR(KT,2) = KTETR(I,2)
   40     ITETR(KT,3) = NUC
   50   ISP = ISP + 2
      XLAST=0.
      YLAST=0.
C USE TO OPTIMISE PEN MOVEMENT ON PLOTTER
      DO 90 JT = 1,ISP
        IF(ITETR(JT,1) .LT. 4 .OR. TETR(JT,3) .GT. 1.) GO TO 90
C.......FIND CONTOUR INTERSECTIONS
        ABIT = 0.0
        IF(PNT(ITETR(JT,1),3) .EQ. PNT(ITETR(JT,2),3) .OR.
     +     PNT(ITETR(JT,1),3) .EQ. PNT(ITETR(JT,3),3) .OR.
     +     PNT(ITETR(JT,2),3) .EQ. PNT(ITETR(JT,3),3)) ABIT = 1.E-10
        TOP = DMAX1(PNT(ITETR(JT,1),3),PNT(ITETR(JT,2),3),
     +              PNT(ITETR(JT,3),3))
        BOT = DMIN1(PNT(ITETR(JT,1),3),PNT(ITETR(JT,2),3),
     +              PNT(ITETR(JT,3),3))
        DO 87 JC = 1,LC
          IF(CONT(JC) .GT. TOP .OR. CONT(JC) .LT. BOT) GO TO 87
          CZ = (CONT(JC) - PNT(ITETR(JT,1),3))/(PNT(ITETR(JT,2),3)
     +                   - PNT(ITETR(JT,1),3) + ABIT)
          IF(CZ .LE. 0.0 .OR. CZ .GE. 1.0) GO TO 83
          X1 = (PNT(ITETR(JT,1),1) + (PNT(ITETR(JT,2),1)
     +      - PNT(ITETR(JT,1),1)) * CZ) * SIZE
          Y1 = (PNT(ITETR(JT,1),2) + (PNT(ITETR(JT,2),2)
     +      - PNT(ITETR(JT,1),2)) * CZ) * SIZE
   82     CZ = (CONT(JC) - PNT(ITETR(JT,1),3))/(PNT(ITETR(JT,3),3)
     +                   - PNT(ITETR(JT,1),3) + ABIT)
          IF(CZ .LT. 0.0 .OR. CZ .GT. 1.0) GO TO 84
          X2 = (PNT(ITETR(JT,1),1) + (PNT(ITETR(JT,3),1)
     +      - PNT(ITETR(JT,1),1)) * CZ) * SIZE
          Y2 = (PNT(ITETR(JT,1),2) + (PNT(ITETR(JT,3),2)
     +      - PNT(ITETR(JT,1),2)) * CZ) * SIZE
          GO TO 85
   83     CZ = (CONT(JC) - PNT(ITETR(JT,1),3))/(PNT(ITETR(JT,3),3)
```

```
     +                     - PNT(ITETR(JT,1),3) + ABIT)
           IF(CZ .LT. 0.0 .OR. CZ .GT. 1.0) GO TO 87
           X1 = (PNT(ITETR(JT,1),1) + (PNT(ITETR(JT,3),1)
     +        -  PNT(ITETR(JT,1),1)) * CZ) * SIZE
           Y1 = (PNT(ITETR(JT,1),2) + (PNT(ITETR(JT,3),2)
     +        -  PNT(ITETR(JT,1),2)) * CZ) * SIZE
    84     CZ = (CONT(JC) - PNT(ITETR(JT,2),3))/(PNT(ITETR(JT,3),3)
     +                     - PNT(ITETR(JT,2),3) + ABIT)
           IF(CZ .LT. 0.0 .OR. CZ .GT. 1.0) GO TO 87
           X2 = (PNT(ITETR(JT,2),1) + (PNT(ITETR(JT,3),1)
     +        -  PNT(ITETR(JT,2),1)) * CZ) * SIZE
           Y2 = (PNT(ITETR(JT,2),2) + (PNT(ITETR(JT,3),2)
     +        -  PNT(ITETR(JT,2),2)) * CZ) * SIZE
    85    IF (DIST(X1,Y1,XLAST,YLAST) .GT. DIST(X2,Y2,XLAST,YLAST)) THEN
            XTMP=X1
            YTMP=Y1
            X1=X2
            Y1=Y2
            X2=XTMP
            Y2=YTMP
C ENSURE START LINE AT CLOSEST POINT
            ENDIF
            XLAST=X2
            YLAST=Y2
              CALL PLOT(X1,Y1,3)
              CALL PLOT(X2,Y2,2)
        IF (NTHK .NE. 0) THEN
        IF (MOD(JC-1,NTHK) .EQ. 0) THEN
        CALL PLOT(X2+0.01,Y2,3)
        CALL PLOT(X1+0.01,Y1,2)
        CALL PLOT(X1,Y1+0.01,3)
        CALL PLOT(X2,Y2+0.01,2)
C DRAW THICK LINES IF WANTED
        ENDIF
        ENDIF
    87        CONTINUE
    90        CONTINUE
        RETURN
        END
        REAL FUNCTION DIST(X1,Y1,X2,Y2)
        DIST=(X1-X2)*(X1-X2)+(Y1-Y2)*(Y1-Y2)
        RETURN
        END
```

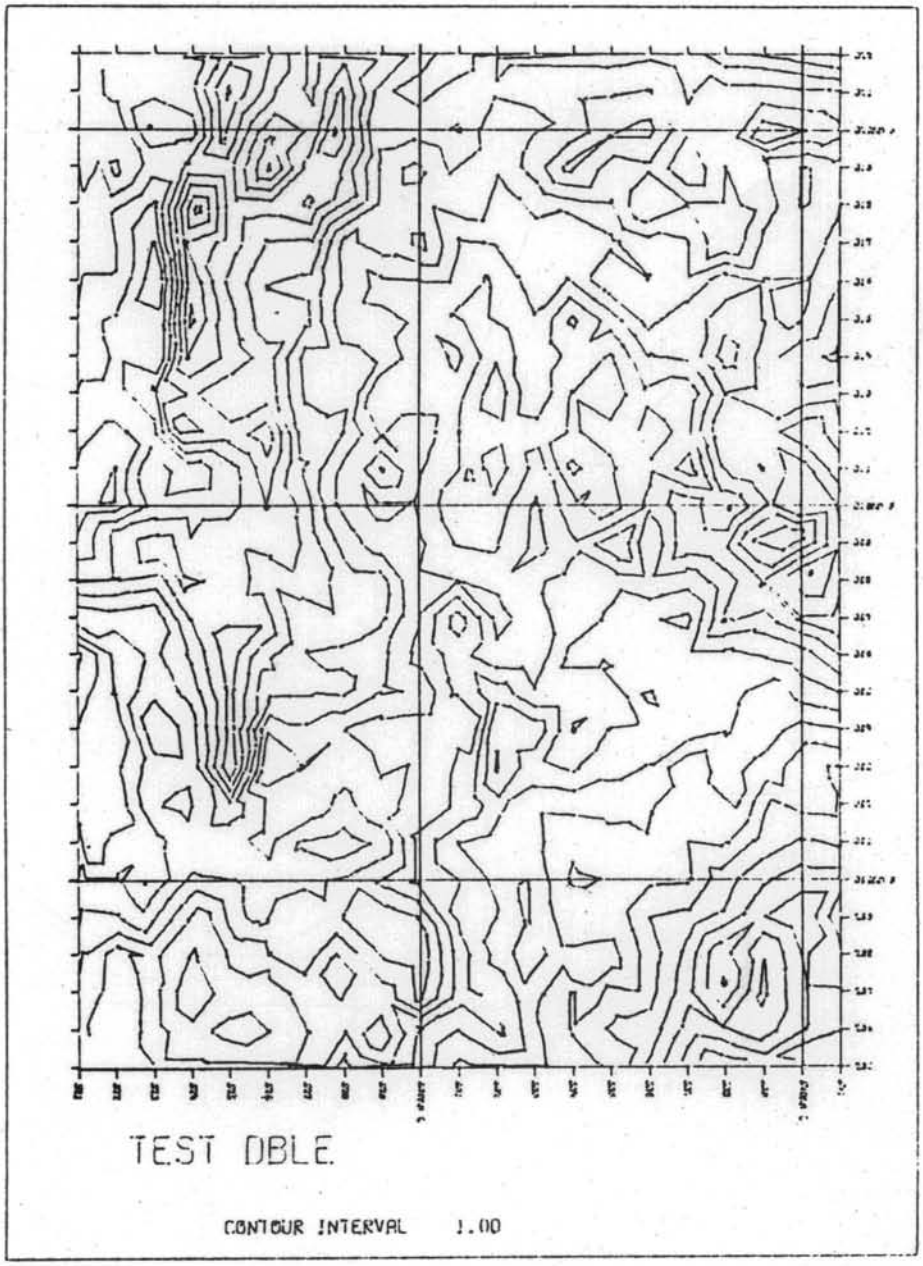
APPENDIX 2

Sample plot; 57 point data set



APPENDIX 3

Sample plot; 567 point data set



5 cm