COMMONWEALTH OF AUSTRALIA

DEPARTMENT OF
NATIONAL DEVELOPMENT

BUREAU OF MINERAL
RESOURCES, GEOLOGY
AND GEOPHYSICS

Record 1972/105

KWIK5, A COMPUTER PROGRAM FOR RAPID PLOTTING OF DATA MAPS

by

S. Henley

1972/105

KWIK5, A COMPUTER PROGRAM FOR RAPID PLOTTING OF DATA MAPS

by

S. Henley

CONTENTS

SUMMARY

KWIK5 is a program (with versions in CDC3600 and CDC6600 FORTRAN) which produces rapid plots on the lineprinter of data values, contoured if desired by a moving average algorithm, and presented either as a grey-scale picture or as a numerically coded map. The map occupies up to two full pages of printout, giving maximum dimensions of 13 inches (across) by 20 inches (down). Automatic scaling procedures are incorporated, though one may optionally define the corner co-ordinates.

INTRODUCTION

For plotting large volumes of data as are obtained in regional geochemical surveys, manual methods are very inefficient compared with computer techniques, particularly when a number of variables are involved, and when the same batch of data is required to undergo some statistical treatment as well. Contouring programs normally used for mapping geological and geophysical data within BMR rely on incremental plotters, and produce maps accurate to .01 or .001 inch. However, their use is expensive, and for a large multi-element geochemical survey this expense could become prohibitive.

Additionally, the great accuracy obtained is not always essential to geological interpretation of the data, and a faster, less expensive method of map preparation would be more appropriate. Using the normal high-speed medium of computer output, the lineprinter, a map can be plotted in about 6 seconds, excluding computation time, and the total cost of computation and plotting is typically $2 to $10, compared with several tens of hundreds of dollars for map production on an incremental plotter.

In most plotting programs, one also has to worry about setting the origin and scales, defining boundaries, and avoiding any instructions which might lead the pen off the edge of the paper. In KWIK5, such problems are avoided by the provision of automatic scaling routines, which draw the largest possible map in the space available, and can optionally search the data matrix for highest and lowest co-ordinate values and data values, and set the map boundaries and contour class limits accordingly.

Though originally written for its application to geochemical data, the program is of universal application.

LINEPRINTER MAPPING

Head (1967) and Howarth (1971) have both described programs which produce grey-level pictures on the lineprinter. These programs are at opposite extremes in complexity, Head's HALFTONE being a simple subrountine for plotting a single line of grey-scale output, while Howarth's PLTLP1 is a long, complex program. HALFTONE requires a maximum of only two prints per line, but PLTLP1 uses up to 6, though an alternative routine is given to plot a non-overprinting grey scale.

KWIK5 is a much simpler program than PLTLP1 and uses a modified version of HALFTONE to plot either a grey-scale or a numerical contour map, or optionally prints only a map of data values, without interpolation.

The accuracy of a lineprinter map is, of course, much lower than that of a hand-drafted map or of a map produced by a computer-driven incremental plotter. For example, the 29 inch Calcomp plotter at CSIRO Division of

Computing Research, Canberra, has an accuracy of 0.01 inch, but the line-printer characters are 1/10 inch across by 1.6 inch vertically, giving more than a tenfold reduction in accuracy. In situations where the 'topology' of the map is the important information, and exact location of specific points is unnecessary, such loss of accuracy is quite acceptable, particularly when increased speed and lower cost are taken into account.

Because of the space occupied by each lineprinter character, it quite frequently occurs that two or more data points would plot at the same position, and in such a case it is their mean value that is plotted; Howarth advocates the geometric mean, and for trace element data with positvely skewed distributions this is probably valid, but for general application to major and trace elements and to other geological data, the arithmetic mean is of more general usefulness.

If a contoured map is required, the program at present uses a moving average algorithm to interpolate values between data points.

MOVING AVERAGE ALGORITHM

If the moving average contour option is selected, a value is computed at each character position within the map area, by taking the arithmetic mean of all values lying within a circle centred on the character position. The radius of the circle is set automatically by the program, and is given by the expression:

$$r = \sqrt{P X Y / N}$$

where X and Y are the horizontal and vertical dimensions of the map area, and N is the number of data points. P is an arbitrary factor controlling the size of the circle, set to 3 in the present version of the program, but can be altered by re-writing the statement after statement 412 in KWIK5:

```
PFAC = 3.0 $ AREA = ABS(XX* YY) $ PAREA = AREA*PFAC/(FLOAT(NP)*
3.14159)
```

PFAC may be set to any desired value.

An example of the type of output produced by the moving average option is shown in Figure 1; the data points (also plotted by KWIK5) are shown in Figure 2.

AUTOMATIC SCALING

(a) Geographic co-ordinates. If the automatic scaling option is selected, the data matrix is searched for the maximum and minimum values of x and y. These values are then rounded up (for the maxima) or down (minima) to the next higher or lower number of the form i * 10^n, where i and n are both integers. For example, if the minimum and maximum values of x are 34.2 and 168.5, they are rounded to 30.0 and 200.0 respectively. This rounding is achieved by a subroutine ROUNDUP which is called by the autoscaling subroutine AUTOSC. ROUNDUP uses the CDC 3600 FORTRAN statements 'ENCODE' and 'DECODE' for truncation of the unwanted digits.

(b) z values. Subrountines AUTOSC and ROUNDUP are again used, but in this case zmin is rounded up and zmax is rounded down to give the best class differentiation within the range of values of interest.

PROGRAM OPERATION

Deck Structure

```
*JOB, chargecode, ident , timelimit
*FTN, X, L
        Fortran deck inserted here
*LOAD
*RUN, timelimit, printlimit
        Initial data card (1)
        Variable format card (2)
        Data cards (3)
7
8 EOF
*EOD
```

Input card formats

(1) Initial data card.

Columns 1-10	X co-ordinate of lower left corner of map
11-20	X co-ordinate of upper right corner of map
21-30	Y co-ordinate of lower left corner of map
31-40	Y co-ordinate of upper right corner of map
41-50	Minimum contour value (points with values below this are plotted as zero on the number coded map or blank on the grey-scale map)

51-60	Maximum contour value (points with values above this are plotted as '*' on the number coded map or black on the grey-scale map)
61-70	If any positive value is inserted here, values are interpolated by the moving average algorithm; if left blank, data points only are plotted
74-75	If blank, a grey-scale with logarithmic contour spacing is plotted If -1, a numeric scale with logarithmic contour spacing is plotted If -2, a numeric scale with linear contour spacing is plotted
79-80	Right justified, logical unit number for input of data. If data are submitted on cards in the same deck, this space may be left blank.

The whole of card (1) may, if desired, be left blank (except for punching columns 74-5 and 79-80 when necessary) and scaling is performed automatically.

(2) Variable format card. Fortran-type format definition enclosed in brackets, defining the format of x, y, z data on the data cards, e.g. (F5.2, 10X, 2F8.3)

(3) Data cards if included in the same deck, as specified on cols 79-80 of card (1)

REFERENCES

HEAD, A.K., 1967 - Halftone. CSIRO Computing Res. Sec., tech. Note. 21

HOWARTH, R.J., 1971 - FORTRAN IV program for gey-level mapping of spatial data. J. math. Geol., 3, 95-122.

APPENDIX

Listing of program KWIK5 and subrountines AUTOSC, ROUNDUP, MINMAD, HALFTONE, and SHELSORT.

```
      PROGRAM KWIK5
C-----------------------------------------------------------------------
C-----------------------------------------------------------------------
C
C     WRITTEN BY S.HENLEY, BMR GEOLOGICAL BRANCH, JUNE 1972,
C     USES THE CSIRO SRLIST SUBROUTINES 'HALFTONE' AND 'SHELSORT'
C     KWIKMAP PLOTS A GREY SCALE PICTURE OF DATA VALUES ON THE LINEPRINTER
C     INPUT -
C     CARD 1, FORMAT (7F10,2I5)
C        X COORD BOTTOM LEFT CORNER
C        X COORD TOP RIGHT CORNER
C        Y COORD BOTTOM LEFT CORNER
C        Y COORD TOP RIGHT CORNER
C        MIN, CONTOUR VALUE ('WHITE' ON PLOTTED MAP)
C        MAX, CONTOUR VALUE ('BLACK' ON PLOTTED MAP)
C        DUM  IF BLANK, DATA PTS ONLY PLOTTED, IF POSITIVE, VALUES INTERPOLATED
C        MARK  IF BLANK, GREY SCALE WITH LOGARITHMIC SPACING,
C              IF -1, NUMERIC SCALE WITH LOGARITHMIC SPACING,
C              IF -2, NUMERIC SCALE WITH LINEAR SPACING
C        IL  LOGICAL UNIT NUMBER FOR INPUT OF X Y Z DATA
C     CARD 2 , FORMAT (10A8)  FORMAT OF X,Y,Z DATA (ENCLOSED IN BRACKETS)
C
C
      DIMENSION X(3000),Y(3000),Z(3000),A(134),FMT(10),KEY(4000),
     *  BX(3000),IX(3000),IY(3000)
      COMMON/1/X,Y,Z,IX,IY
      PRINT 1
    1 FORMAT(*1PROGRAM KWIKMAP - FAST PLOTTING OF*
     ** DATA VALUES*)
      READ 10, XB,XT,YB,YT,COMIN,COMAX,DUM,MARK,IL
   10 FORMAT (7F10,2I5)
      IF (COMIN.LE.0.0) COMIN=1.0
      IF (IL.LE.0) IL=60
      READ 20,FMT
   20 FORMAT (10A8)
      DO 30 I=1,3000
      READ (IL,FMT) X(I),Y(I),Z(I)
      IF(EOF,IL) 40,30
   30 CONTINUE
   40 NP=I-1
      IF (COMAX.GT.0.0) GO TO 401
      CALL AUTOSC(X,NP,XB,XT,1)
      CALL AUTOSC(Y,NP,YB,YT,1)
      CALL AUTOSC(Z,NP,COMIN,COMAX,0)
  401 DO 25 I=1,NP
      BX(I)=X(I)=X(I)-XB
   25 Y(I)=YT-Y(I)
      NROWS=120  $  ROWS=120.0
      XX=XT-XB
      YY=YT-YB
      COLS=200.0*XX/YY
      NCOLS=COLS
      IF (COLS.LE.130) GO TO 41
      ROWS=ROWS*130.0/COLS
      COLS=130.0
      NROWS=ROWS
      NCOLS=130
   41 DO 412 I=1,NP
      IY(I)=Y(I)*ROWS/YY
```

```
      IX(I)=X(I)*COLS/XX
      PRINT 26,I,IX(I),IY(I),X(I),Y(I),Z(I)
   26 FORMAT(1X,3I10,3F12,2)
  412 CONTINUE
      PFAC=3,0 $ AREA=ABS(XX*YY) $ PAREA=AREA*PFAC/(FLOAT(NP)*3,14159)
      RAD=SQRT(PAREA) $ KRAD=(COLS*RAD/XX+0,5)
      PRINT 4011,NP
 4011 FORMAT(1H0,*DATA READ FOR*,I8,* SAMPLE PTS*)
      SCALE=YY*6,0/ROWS
      CALL SHELSORT(BX,NP,KEY)
      DO 402 I=1,NP
       KK=KEY(I)
  402 PRINT 26,I,KK,IX(KK),X(KK),Y(KK),Z(KK)
      PRINT 411, NCOLS,NROWS,SCALE
  411 FORMAT (1H0,*MAP OCCUPIES*,I10,*  COLUMNS AND*,I10,*  ROWS*//
     ** SCALE OF MAP IS*,F10,2 , *  INPUT UNITS PER PLOTTED INCH*/1H1)
      PRINT 42
   42 FORMAT (1HZ)
      AMAX=0,0
      CALL SETSCALE(DUM,DUM,MARK,COMAX,COMIN)
      CALL BORDER(NCOLS,DUM,DUM,DUM,DUM)
      IF (MARK,EQ,0) MARK=48
      DO 60 I=1,NROWS
      DO 50 J=1,NCOLS
      PTS=0,0  $  SUM=0,0
      DO 45 K=1,NP
      KK=KEY(K)
      IF(IX(KK),GT,J) GO TO 46
      IF (IX(KK),LT,J) GO TO 45
      IF(IY(KK),NE,I) GO TO 45
      SUM=SUM+Z(KK)
      PTS=PTS+1,0
   45 CONTINUE
   46 IF (PTS,LE,0,0) 47,49
   47 IF (DUM,GT,0,00001) GO TO 48
  471 A(J)=1,0E11
      GO TO 50
   48 SUM=PTS=0,0 $ ISTART=J-KRAD $ ISTOP=J+KRAD
      DO 481 K=1,NP
      KK=KEY(K)
      IF(IX(KK),LT,ISTART) GO TO 481
      IF (IX(KK),GT,ISTOP) GO TO 482
      XDIST=ABS(J*XX/COLS-X(KK))
      YDIST=ABS(I*YY/ROWS-Y(KK))
      DIST=SQRT(XDIST**2 + YDIST**2)
      IF (DIST,GE,RAD) 481,480
  480 SUM=SUM+Z(KK) $ PTS=PTS+1,0
  481 CONTINUE
  482 IF(PTS,LE,0,0) 471,49
   49 A(J)=SUM/PTS
      IF(A(J),GT,AMAX) AMAX=A(J)
   50 CONTINUE
      CALL HALFTONE(NCOLS,A,MARK)
   60 CONTINUE
      CALL BORDER(NCOLS,DUM,DUM,DUM,DUM)
      PRINT 601
  601 FORMAT(1HY)
      PRINT 61
   61 FORMAT (*1TONE SCALE*//)
      CALL PRNSCALE
```

```
      PRINT 611,AMAX
  611 FORMAT(*0AMAX=*,F8,2)
      PRINT 62
   62 FORMAT(*0END KWIKMAP*)
      CALL EXIT
      END
```

```
      SUBROUTINE AUTOSC(X,N,XB,XT,L)
C
C  AUTOMATIC SCALING ROUTINE. IF L=1, XB AND XT ARE XMIN AND
C  XMAX, ROUNDED DOWN AND UP RESPECTIVELY. IF L=0 XB AND XT
C  ARE XMIN AND XMAX ROUNDED UP AND DOWN.
C
      DIMENSION X(N)
      CALL MINMAX(X,N,XMIN,XMAX)
      CALL ROUNDUP(XMIN,XB,1-L)
      CALL ROUNDUP(XMAX,XT,L)
      PRINT 10,XMIN,XB,XMAX,XT
   10 FORMAT(*0AUTOSCALE ROUTINE CALLED - MIN=*,F8.2,* ROUNDED TO*,
     .F8.2/28X,*MAX=*,F8.2,* ROUNDED TO*,F8.2)
      RETURN
      END
```

```
      SUBROUTINE ROUNDUP(A,R,L)
C
C  ROUNDS A GIVEN VALUE UP (L=1) OR DOWN (L=0) FOR SCALING
C  PLOTS ETC; AUTOMATICALLY,
C
C  ROUNDED VALUES ARE OF THE FORM (I+L) * 10 ** N
      ENCODE (8,10,M) A
      PRINT 11,A
   11 FORMAT(1H ,E8,1)
   10 FORMAT(E8,1)
      DECODE (8,20,M) J,K,N
   20 FORMAT(I2,1X,I1,I4)
      B=FLOAT(J)*10**N
      IF (B,GT,A) J=J-1
      J=J+L
      R=FLOAT(J)*10**N
      RETURN
      END
```

```
      SUBROUTINE MINMAX(X,N,XMIN,XMAX)
C
C  FINDS MINIMUM AND MAXIMUM VALUES OF ARRAY X(N)
C
      DIMENSION X(N)
      XMIN=XMAX=X(1)
      DO 30 I=2,N
      XMIN=AMIN1(XMIN,X(I))
   30 XMAX=AMAX1(XMAX,X(I))
      RETURN
      END
```

```
      SUBROUTINE HALFTONE(N,ARRAY,MARK,BLACK,WHITE)                     HALFT001
      DIMENSION ARRAY(134),JX(134),JXX(134),JA(11),JAA(11),X(10),Y(10)  HALFT002
      COMMON/HALFTN/X,Y,JA,JAA
      IF(N*(N-135))10,1,1                                               HALFT003
    1 PRINT 2,N $ RETURN                                                HALFT004
    2 FORMAT(8H N WRONG,I15)                                            HALFT005
   10 IF(JA(2).LT.27)11,3                                               HALFT
   11 PRINT 17 $ RETURN                                                 HALFT007
   17 FORMAT(18H NO GRAY SCALE SET)                                     HALFT008
    3 JS=(137-N)/2 $ JF=(135+N)/2                                       HALFT009
      DO 4 J=2,JS                                                       HALFT010
    4 JX(J-1)=JXX(J-1)=48                                               HALFT011
      DO 6 J=JS,JF $ K=J-JS
      IF (ARRAY(K+1).GT.1.0E10) GO TO 91
      Z=ARRAY(K+1)
      DO 5 L=1,10                                                       HALFT013
      IF(Z-X(L))9,9,5                                                   HALFT014
    5 CONTINUE                                                          HALFT015
      L=11                                                              HALFT016
    9 JX(J)=JA(L)                                                       HALFT017
      JXX(J)=JAA(L)                                                     HALFT018
      GO TO 6
   91 JX(J)=JXX(J)=48
    6 CONTINUE
      JSM=JS-1   $   JFP=JF+1
      JXX(JSM)=JXX(JFP)=1RI
      IF (MARK.GE.-0) PRINT 7,MARK,(JX(M),M=1,JF)                       HALFT019
      PRINT 8,(JXX(M),M=1,JFP)                                          HALFT02
    7 FORMAT (1H+,135R1)
    8 FORMAT(2X,134R1)                                                  HALFT022
      RETURN                                                            HALFT023
      ENTRY PRNSCALE                                                    HALFT024
      DO 12 L=1,10 $ K=L+1 $ Z=X(L)
   12 PRINT13,(JA(L),M=1,5),Z,(JA(K),M=1,5),(JAA(L),M=1,5),             HALFT026
     1 (JAA(K),M=1,5)                                                   HALFT027
   13 FORMAT(1H+5X,5R1,E13.4,3X,5R1/6X,5R1,16X,5R1)                     HALFT028
      RETURN                                                            HALFT029
      ENTRY SETSCALE                                                    HALFT030
      IF (MARK.EQ.-2) GO TO 19
      JA(1)=48 $ JAA(1)=48 $ Y(1)=1.2633                                HALFT034
      JA(2)=27 $ JAA(2)=48 $ Y(2)=1.2151                                HALFT035
      JA(3)=10 $ JAA(3)=48 $ Y(3)=1.1780                                HALFT036
      JA(4)=12 $ JAA(4)=59 $ Y(4)=1.1527                                HALFT037
      JA(5)=11 $ JAA(5)=48 $ Y(5)=1.1310                                HALFT038
      JA(6)=46 $ JAA(6)=27 $ Y(6)=1.1051                                HALFT039
      JA(7)=46 $ JAA(7)=62 $ Y(7)=1.0773                                HALFT040
      JA(8)=24 $ JAA(8)=30 $ Y(8)=1.0519                                HALFT041
      JA(9)=46 $ JAA(9)=61 $ Y(9)=1.0266                                HALFT042
      JA(10)=24 $ JAA(10)=57 $ Y(10)=1.0000                             HALFT043
      JA(11)=18 $ JAA(11)=14                                            HALFT044
      DO 16 L=1,10                                                      HALFT045
      F=FLOAT(L-1)/9.0
   16 X(L)=WHITE*(BLACK/WHITE)**F
      IF (MARK.GE.-0) RETURN
  160 DO 161 I=2,10
      JA(I)=48
  161 JAA(I)=I-1
      JA(1)=JA(11)=48
      JAA(1)=0
        JAA(11)=44
```

```
      RETURN
   19 X(1)=WHITE
      DO 20 I=2,10
   20 X(I)=WHITE+(I-1)*(BLACK-WHITE)/9.0
      GO TO 160
      ENTRY BORDER
      JS=(137-N)/2 -1
      JF=(135+N)/2 +1
      DO 18 I=1,JF
      JX(I)=1R
      IF (I.GE.JS) JX(I)=1R-
   18 CONTINUE
      PRINT 8,(JX(M),M=1,JF)
      RETURN
      END                                          HALFT047
```

```
      SUBROUTINE SHELSORT(X,N,KEY)
C
C     SHELL, MODIFIED FRANK AND LAZARUS,CACM 3,20 (1960)
C     TO SORT INTEGERS, USE   INTEGER X,TEMP
C
C     MODIFIED BY S,HENLEY, JUNE 1972, FOR USE WITH KWIKMAP
C     PRODUCES A KEY WITH POSITIVE VALUES OF N, NEGATIVE VALUES
C     NOT ALLOWED
C
      DIMENSION X(1),KEY(1)
      MO=NO=N
      DO 11 I=1,NO
   11 KEY(I)=I
   10 IF(MO,LE,15) 1,3
    1 IF (MO,GT,1) 2,9
    2 MO=2*(MO/4)+1
      GO TO 4
    3 MO=2*(MO/8)+1
    4 KO=NO-MO
      JO=1
    5 I=JO
    6 IMO=I+MO
      IF (X(I),GT,X(IMO)) 7,8
    7 TEMP=X(I)
      X(I)=X(IMO)
      X(IMO)=TEMP
   14 KEMP=KEY(I)
      KEY(I)=KEY(IMO)
      KEY(IMO)=KEMP
   13 I=I-MO
      IF(I,LT,1) 8,6
    8 JO=JO+1
      IF (JO,GT,KO) 10,5
    9 END
```

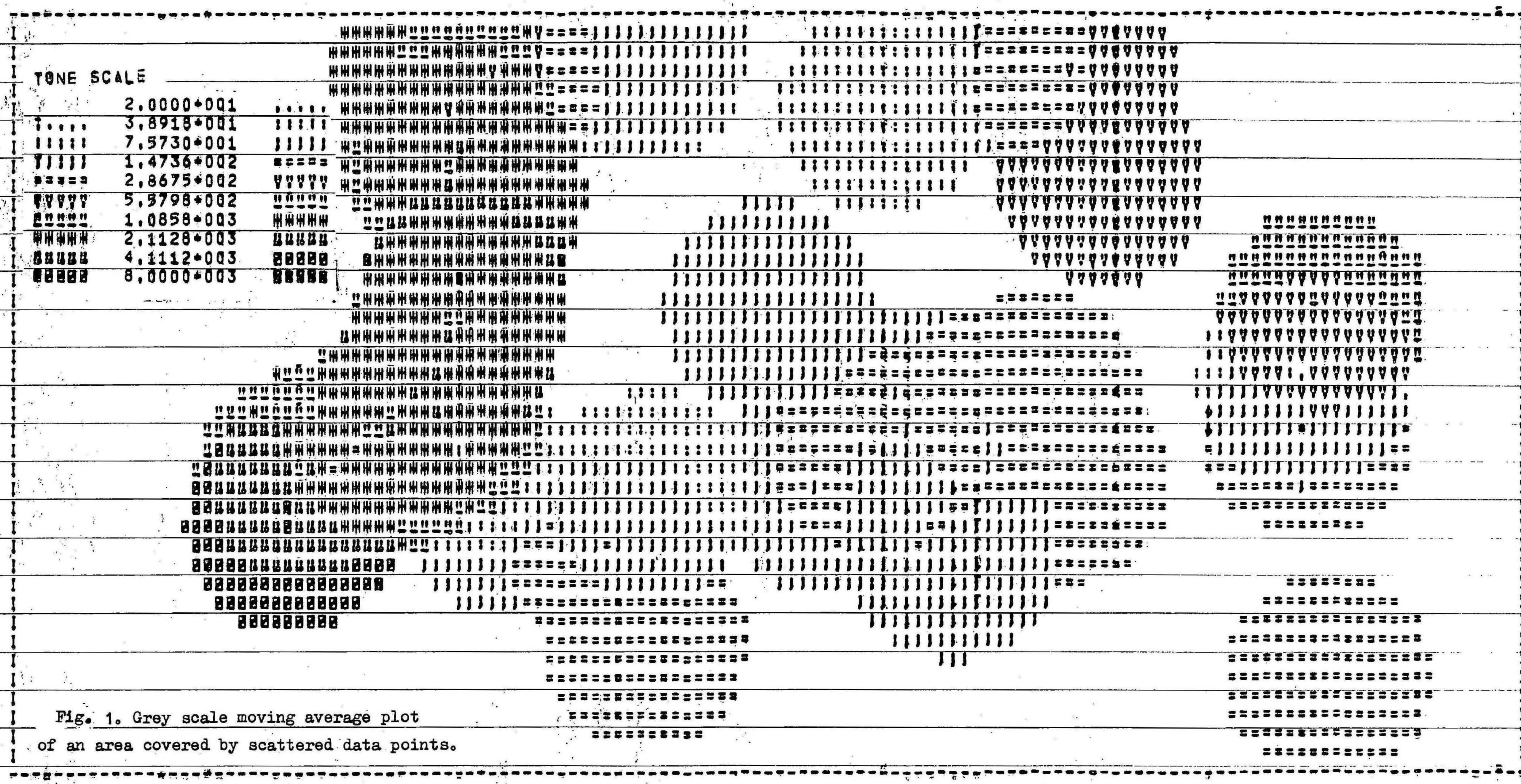

Fig. 1. Grey scale moving average plot of an area covered by scattered data points.

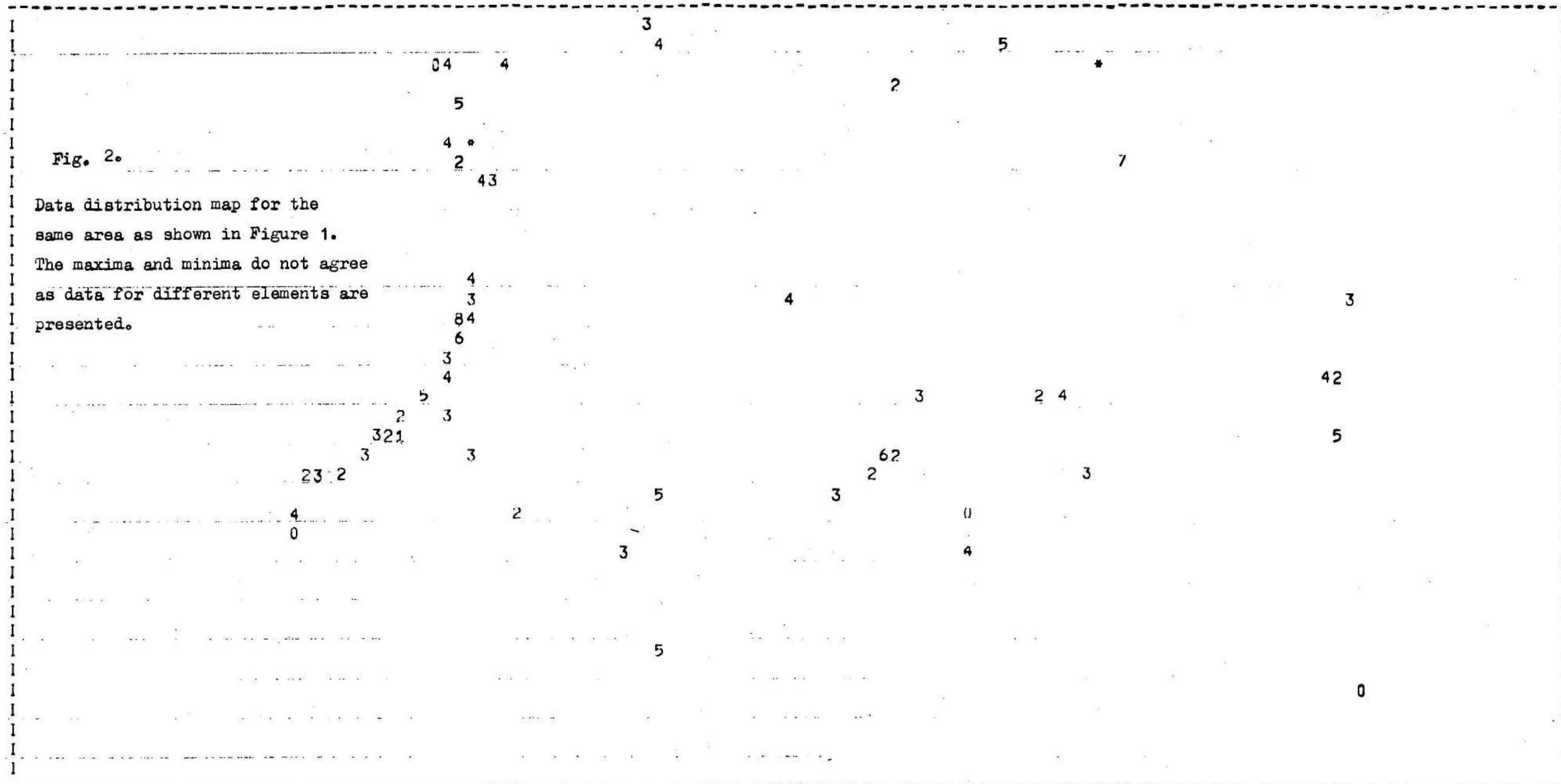

Fig. 2.

Data distribution map for the same area as shown in Figure 1. The maxima and minima do not agree as data for different elements are presented.