COMPUTER EVALUATION OF DOUBLE-THEODOLITE DATA

by

W. Gale Biggs

Technical Report No. 4

ORA Project 03632
NATIONAL SCIENCE FOUNDATION GRANT G-11404

College of Engineering
Department of Engineering Mechanics
Meteorological Laboratories

and

Institute of Science and Technology
Great Lakes Research Division
Special Report No. 13

THE UNIVERSITY OF MICHIGAN
ANN ARBOR, MICHIGAN
April 1962

ensm

UMR 8535

ABSTRACT

Conventional methods of evaluating double-theodolite data involve numerical and/or graphical techniques that are laborious and time consuming. The following presentation offers a method whereby the electronic digital computer first computes the position of the balloon in space and then from this determines the wind vector. The program also makes provision for the problem of missing data.

This computing technique is highly valuable when a large number of pilot balloon ascents must be processed.

While conducting field operations at the Enrico Fermi Nuclear Reactor Site during the summer of 1961, it was necessary to take double-theodolite balloon ascents as a means of evaluating the wind speed in the lower few thousand feet of the atmosphere. The basic approaches to evaluating double-theodolite data have been: 1) numerically to calculate the position of the balloon in space and time, and then to plot it on a winds-aloft plotting board to determine wind speed and direction; or 2) to use graphical methods throughout to determine wind speed and direction. Either of these methods requires the use of winds-aloft plotting boards and both methods are laborious and time consuming. Weedfall and Jagodzinski [1961] describe a graphical method in which they can evaluate a 30-interval run in about 25 min. The United States Weather Bureau Circular "O" describes a method which takes 70 min for a 30-interval run. When a large number of observations are to be evaluated, the high speed electronic digital computer offers a much more convenient and rapid technique of analysis.

The following method of analysis requires no plotting of data. The only work is in the conversion of the data from

tabular form to punched cards or any other convenient form of computer input. The data deck consists of one card containing the length of the baseline and the number of observations, followed by four sets of cards. The first of these contains the azimuth angles from station A (the balloon release point), the second contains the elevation angles from station A, the third contains the azimuth angles from station B (the satellite station), and the fourth contains the elevation angles from station B. These angles are read into a three dimensional matrix A_{ijk}, where in this case i is the number of layers, j is the number of rows, and k is the number of columns.

The output of the computer is in the desired form of the horizontal wind vector and the altitude of the balloon for each sounding. Also printed out are the horizontal radial distance of the balloon from the launch site, the u and v components of the wind vector, and the x and y coordinates of the balloon. The mathematical basis for the method presented is simple and is easily handled by the digital computer.

The use of two theodolites necessitates a baseline which is carefully surveyed. The baseline used was 2000 ft long and lay in an east-west direction. Since the terrain

is relatively flat, the two stations were at equal elevations, except for the two kilometer inland site which had seven feet difference in elevation. The other sites had less than a foot difference in elevation. Communications were maintained through the use of two transistorized walkie-talkies. Five men were used, three at the balloon release station and two at the satellite station (figure 1). At the balloon release station one man tracked the balloon and read the two angles, one man recorded, and one man kept time. At the satellite station the radio was turned up so both could hear. One man tracked the balloon and read the angles while the second man recorded the data. This arrangement allowed for reading the theodolites at ten second intervals.

Three sites were surveyed with 2000 ft baselines, one at the plant site, one at two kilometers inland, and one at four kilometers inland. Provisions were also made to survey an inland site at six kilometers which will be done during the spring of 1962.

The two theodolites were adjusted so that both were reading north at 360°. As long as the balloon is not too near the vertical plane containing the baseline, the horizontal triangle ABC (figure 2) is solved first, to obtain the projection of the balloon and its horizontal distance

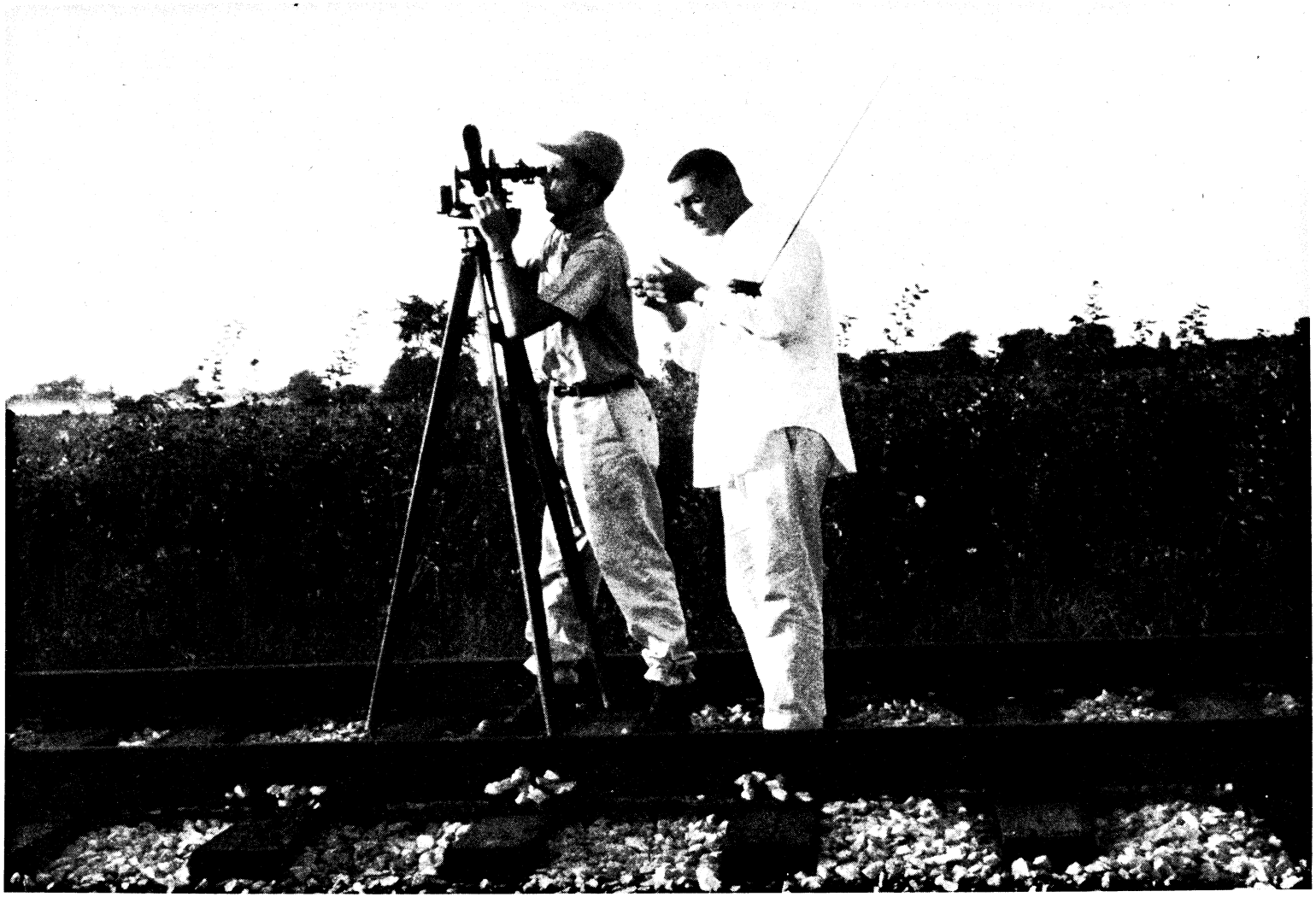

Figure 1. Double-theodolite tracking operations at the satellite station.

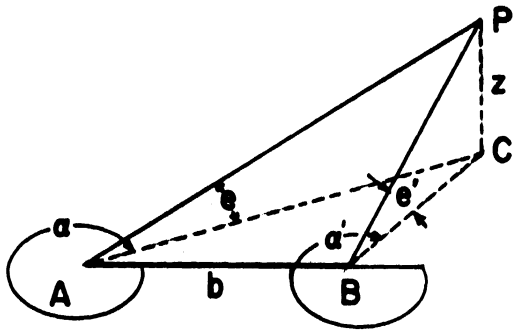

Figure 2. Perspective of two theodolite triangulation.

from the two stations. AB is the baseline of length b ; P is the position of the balloon, and C is its projection on the horizontal plane through A . The angles measured from A are α and ϵ , those measured from B , α' and ϵ' . After the projection of the balloon (C) has been obtained, its height above the level of A is calculated by the relation $Z = AC \tan \epsilon$, where ϵ is the elevation angle measured at A . The elevation above B may be calculated as a check; it is given by $Z' = BC \tan \epsilon'$.

It is seen from figure 2

$$\text{angle } ACB = \alpha - \alpha'$$

and by the sine law

$$\frac{AC}{\sin \alpha'} = \frac{b}{\sin (\alpha - \alpha')}$$

therefore

$$AC = \frac{b \sin \alpha'}{\sin (\alpha - \alpha')}$$

Similarly

$$BC = \frac{b \sin \alpha}{\sin (\alpha - \alpha')}$$

also

$$Z = \frac{b \sin \alpha' \tan \epsilon}{\sin (\alpha - \alpha')}$$

and

$$Z' = \frac{b \sin \alpha \tan \epsilon'}{\sin (\alpha - \alpha')}$$

Since in all but one case, A and B are at the same elevation, Z and Z' should be equal. At the two kilometer site where A and B are not in the same horizontal plane, all the above formulae hold except for Z' where it will differ by h , where h is the difference in elevation of A and B .

The analysis so far has described the position of the balloon in space in a cylindrical coordinate system (r, θ, z) , where $r = AC$, $\theta = \alpha$, and $z = z$. For the

following approach this needs to be converted to a Cartesian coordinate system (x, y, z). This is done as follows:

$$x = r \cos \theta$$

$$y = r \sin \theta$$

$$z = z$$

The Cartesian coordinate system is set up by requiring that the x-axis coincide with the baseline used for the two theodolites and that the origin be at the theodolite where the balloon is released. The baseline was located on an east-west line thus making the y-axis on a north-south line (figure 3).

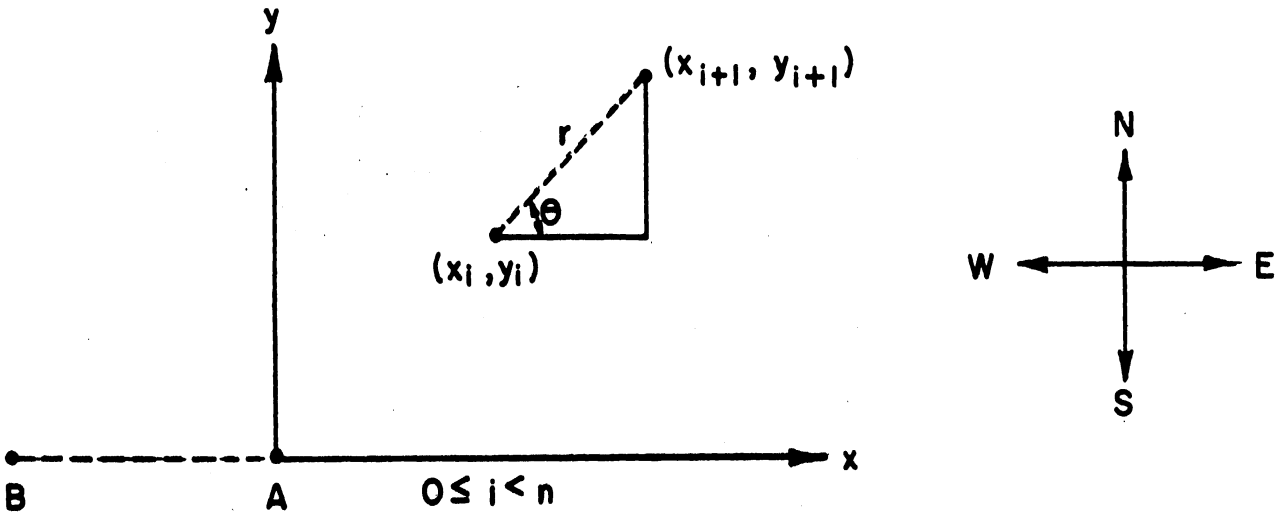

Figure 3. Diagram showing coordinate system used in equations.

The position of the balloon in the x , y plane is designated as x_0 , y_0 for the release position; x_1 , y_1 as the first position, etc., to x_n , y_n as the n^{th} position; then in order to find the distance the balloon traveled in going from x_i , y_i to x_{i+1} , y_{i+1} the following formulae are used

$$r = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}$$

$$\theta = \tan^{-1}\left[\frac{y_{i+1} - y_i}{x_{i+1} - x_i}\right]$$

$$0 \leq i < n$$

Thus, knowing r , the distance traveled, and t , the time between successive readings (for this program it was every 10 sec), the average wind speed $\bar{U}$ is simply r/t . The height of the balloon was previously calculated by the computer in order to describe its position in space. The U. S. Weather Bureau in computing its pibal soundings uses every other point in obtaining a wind speed and direction; this is very simply done by using x_{i+2} , y_{i+2} instead of x_{i+1} , y_{i+1} .

When the computer comes to a missing point, an average is taken by using the next point. This is also true if two consecutive points are missing. However, if three or more consecutive points are missing, then an average is not taken and the computer continues to test for missing data until points are again found and the averaging process starts again. Thus a segment of the run is considered missing only if three or more consecutive readings are missing.

A problem also arises when the balloon crosses the baseline or travels along it. A small error in α or α' will cause a large error in the computed position of the balloon. In this case it is better to solve the vertical triangles ABP and ACP (figure 4). It follows that

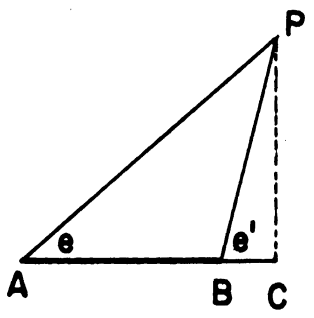

Figure 4. Solution of vertical triangles.

$$AC = b \frac{\sin \epsilon' \cos \epsilon}{\sin (\epsilon \pm \epsilon')}$$

$$BC = b \frac{\cos \epsilon' \sin \epsilon}{\sin (\epsilon \pm \epsilon')}$$

$$Z = b \frac{\sin \epsilon' \sin \epsilon}{\sin (\epsilon \pm \epsilon')}$$

The positive sign will be used when C lies in AB , the negative sign when it lies in AB produced.

Once the position of the balloon in space is computed, the coordinates are converted to Cartesian and the wind speed and direction are computed as before.

Since the angles as recorded by the theodolites are used only in sine and cosine functions, the fact that they go through 360° makes no difference. The sine and cosine functions are symmetric and periodic, thus the sine 362° $\doteq$ sine 2°, and the same is true for the cosine.

Since the arc tangent is multivalued, there is a question of which value to choose. This problem can be easily y solved by looking at the combination of signs on the x and y values used to compute the arc tangent. If both are positive ive then the angle lies in the first quadrant; if y is positive and x is negative then the angle lies in the second quadrant; etc. This was even less of a problem for this program since

the computer has a calling subroutine that will compute the arc tangent for values between $0 \rightarrow 2\pi$.

Just before the results are printed out by the computer, the complement of each angle is taken and then rotated 270°. The complement is taken since the program computes θ in a counterclockwise direction, whereas it is standard practice to use θ in a clockwise direction to express wind direction. The angle is then rotated 270° because in the program 360° is east, and the angle θ defines the direction in which the balloon is traveling. Thus θ is rotated 90° to correspond to the concept of north being 360° and then 180° more to correspond to the concept of wind direction being given as the direction from whence the wind is blowing.

Table 1 shows a comparison between the computer evaluation and the hand calculated values. As a means of showing the computer evaluation with missing data, two values were read in as zeros and the results compared. The first pair of columns presents the radial horizontal distance from the release point to the balloon (in feet). The second pair gives the height of the balloon above the theodolite (in feet) and the third gives the local wind speed (in feet per second). The fourth pair of columns shows the local wind direction (in

TABLE 1

Comparison of values obtained by the computer with those obtained by conventional methods.

R (ft)		Z (ft)		$\overline{U}$ (ft/sec)		θ (deg)		u (ft/sec)	v (ft/sec)
Comp	HC	Comp	HC	Comp	HC	Comp	HC	Comp	Comp
0	0	0	0						
				7	7	69	69	6.6	2.5
70	70	42.3	42.3						
				5	5	94	92	4.8	-0.3
116	116	64.3	64.3						
				6	6	75	75	6.0	1.6
178	178	96.3	96.3						
0	-	0	-						
				18	18	75	76	17.5	4.5
0	-	0	-						
722	722	329.0	329.0						
				15	14	73	67	14.7	4.4
875	875	382.3	382.3						
				11	10	64	62	10.1	4.9
1118	1118	422.3	422.3						
				13	14	68	67	12.3	4.9
1306	1306	453.8	453.8						
				19	20	67	67	17.4	7.5
1438	1438	509.3	509.3						
				13	12	65	65	12.0	5.6
1588	1588	537.6	537.6						

R - Radial horizontal distance of balloon from launch site (ft)
Z - Height of balloon (ft)
$\overline{U}$ - Wind speed (ft/sec)
θ - Azimuth of wind (deg)
u - x-comp of wind (as obtained by computer) (ft/sec)
v - y-comp of wind (as obtained by computer) (ft/sec)
Comp - Computer values
HC - Hand calculated values

degrees), and the last two columns give the local u and v components of the wind speed as evaluated by the computer.

The method of evaluation used to obtain the hand calculated figures was as follows: first the position of the balloon in space was numerically calculated, and then this was plotted on a winds-aloft plotting board and a graphical analysis used to determine the wind speed and direction.

The program, since it is a computer analysis, suffers from none of the human errors involved in a graphical analysis and is therefore exact. Thus the accuracy of the final result is determined only by the accuracy with which the original angles were recorded.

The program has (as an external function) a system whereby a smoothing subroutine is incorporated in it. Any method of smoothing may be used in the subroutine, with the present smoothing function given by

$$A_i = \tfrac{1}{4} B_{i-1} + \tfrac{1}{2} B_i + \tfrac{1}{4} B_{i+1}$$

where

$$0 < i < n$$

The smoothing subroutine also takes care of missing data. In the present program only the wind speed is smoothed.

The merit of the program is the great speed with which a large number of data can be analyzed. One program consisted of 10 pibal soundings, each sounding 10 min long with readings taken every 10 sec for a total of 600 points. The computer time involved was 1.5 min for the IBM 709; it required about 45 min of one person's time for punching the data on cards. This is to be compared with time for the method presented by Weedfall and Jagodzinski [1961] which required about 25 min to complete one run of only 30 points and that for the Circular "O" method which requires 70 min for a 30-point run.

The program was written in the MAD* language and is reproduced here in that form in the Appendix. The program can be broken into four main groups--the first group consists of reading in the data and converting to radians. The second group computes the location of the balloon in space. The third group tests for missing data, computes wind speed and direction, and smoothes the wind speed. The last group consists of input and output statements necessary for the MAD translator.

---

*Michigan Algorithm Decoder

Since the MAD language may not be familiar to everyone, each MAD statement is given with its SAP* translation. Thus the program may be used in most of the standard computers available today.

---

*Share Assembly Program

REFERENCES

1. Middleton, W. E. K., and A. F. Spilhaus, 1953: Meteorological Instruments (3rd ed.). Toronto, University of Toronto Press, 186-187.

2. Weedfall, R. O., and W. M. Jagodzinski, 1961: Comments on Double-Theodolite Evaluations. Bull. Amer. Meteor. Soc., 42, 322-324.

3. Hansen, F. V , and N. H. Taft, 1959: Another Method of Evaluating Double-Theodolite Runs. Bull. Amer. Meteor. Soc., 40, 221-224.

4. U. S. Weather Bureau Circular "O."

A P P E N D I X

```
        PRINT FORMAT JUMP                                                  *001
START   READ FORMAT CON,N,B,LIC                                            *002
        PRINT FORMAT UU                                                    *003
        READ FORMAT DATA,A(1,1,1)...A(1,1,N)                               *004
        READ FORMAT DATA,A(1,2,1)...A(1,2,N)                               *005
        READ FORMAT DATA,A(2,1,1)...A(2,1,N)                               *006
        READ FORMAT DATA,A(2,2,1)...A(2,2,N)                               *007
        EXECUTE ZERO.(C(1)...C(N))                                         *008
        THROUGH S,FOR I=1,1,I.G.N                                          *009
        WHENEVER A(1,1,I).E.0.OR.A(2,1,I).E.0,TRANSFER TO S                *010
        A(1,1,I)=(90.-A(1,1,I))*6.2831853/360.                             *011
        A(1,2,I)=A(1,2,I)*6.2831853/360                                    *012
        A(2,1,I)=(90.-A(2,1,I))*6.2831853/360.                             *013
        A(2,2,I)=A(2,2,I)*6.2831853/360                                    *014
        C(I)=B/SIN.(A(1,1,I)-A(2,1,I))                                     *015
S       CONTINUE                                                           *016
        THROUGH SS,FOR I=1,1,I.G.N                                         *017
        AC(I)=C(I)*SIN.(A(2,1,I))                                          *018
        BC=C(I)*SIN.(A(1,1,I))                                             *019
        ZA(I)=AC(I)*SIN.(A(1,2,I))/COS.(A(1,2,I))                          *020
        ZB(I)=BC*SIN.(A(2,2,I))/COS.(A(2,2,I))                             *021
        XA(I)=AC(I)*COS.(A(1,1,I))                                         *022
SS      YA(I)=AC(I)*SIN.(A(1,1,I))                                         *023
        EXECUTE ZERO.(C(1)...C(N))                                         *024
        THROUGH MM,FORJ=1,1,J.G.N                                          *025
        WHENEVER XA(J).E.0.AND.YA(J).E.0                                   *026
        WHENEVER XA(J+1).E.0.AND.YA(J+1).E.0                               *027
        WHENEVER XA(J+2).E.0.AND.YA(J+2).E.0                               *028
        THROUGH AA,FOR K=J+3,1,K.G.N                                       *029
        WHENEVER XA(K).NE.0.OR.YA(K).NE.0                                  *030
        J=K                                                                *031
        TRANSFER TO MM                                                     *032
AA      END OF CONDITIONAL                                                 *033
        END OF CONDITIONAL                                                 *034
        X=XA(J+2)-XA(J-1)                                                  *035
        Y=YA(J+2)-YA(J-1)                                                  *036
        J=J+2                                                              *037
        TRANSFER TO NN                                                     *038
        END OF CONDITIONAL                                                 *039
        X=XA(J+1)-XA(J-1)                                                  *040
        Y=YA(J+1)-YA(J-1)                                                  *041
        J=J+1                                                              *042
        OTHERWISE                                                          *043
        X=XA(J)-XA(J-1)                                                    *044
        Y=YA(J)-YA(J-1)                                                    *045
        END OF CONDITIONAL                                                 *046
NN      R=SQRT.(X*X+Y*Y)                                                   *047
        THETA=270.-ATN1.(Y,X)*360./6.2831853                               *048
        WHENEVER THETA.L.0,THETA=360.+THETA                                *049
        PRINT FORMAT PP,J,R,THETA,ZA(J),ZB(J),ZA(J)-ZA(J-1),ZB(J)-ZB(      *050
```

```
      1J-1),ZA(J)-ZB(J),X,Y,XA(J),YA(J)                                  *050
       C(J)=R                                                            *051
MM     CONTINUE                                                          *052
       PRINT FORMAT XX,AC(1)...AC(N)                                     *053
       PUNCH FORMAT DATA1,N,C(1)...C(N)                                  *054
       WHENEVER LIC.NE.0.                                                *055
       EXECUTE SMOTH.(C,ZB,N)                                            *056
       PRINT FORMAT  XX,ZB(1)...ZB(N)                                    *057
       END OF CONDITIONAL                                                *058
       TRANSFER TO START                                                 *059
       INTEGER N,I,J,K                                                   *060
       VECTOR VALUES DIM=3,1,2,200                                       *061
       VECTOR VALUES CON=$I5,2F10.0*$                                    *062
       VECTOR VALUES DATA1=$I5/(16F5.1)*$                                *063
       VECTOR VALUES DATA=$16F5.1*$                                      *064
       VECTOR VALUES XX=$1H0/(1H ,14F8.1,F7.1)*$                         *065
       VECTOR VALUES UU=$1H4,S4,1HJ,S7,1HR,S9,1H0,S8,2HZA,S8,2HZB,S7     *066
      1,5HZA-ZA,S5,5HZB-ZB,S5,5HZA-ZB,S7,1HU,S9,1HV,S9,1HX,S9,1HY/1H     *066
      2+,S22,1H-*$                                                       *066
       VECTOR VALUES PP=$1H ,I5,11F10.1*$                                *067
       DIMENSION A(1000,DIM),C(200),XA(200),YA(200),ZA(200),ZB(200),     *068
      1AC(200)                                                           *068
       VECTOR VALUES JUMP=$1H1*$                                         *069
       END OF PROGRAM                                                    *070
```

```
MAD PROGRAM,TYPE 12 MAR 1962   (ALL NUMBERS ARE OCTAL)

 NO. OF LOCATIONS 05601 TRA VECTOR SIZE   00013      TRA VECTOR STARTS 00000   ENTRY PT. 04351   ERASABLE STARTS 77777

VARIABLE STORAGE
     AA      00013          AC      00324          A       02275          BC      02276          B       02277
     CON     02301          C       02612          DATAI   02614          DATA    02616          DIM     02622
     I       02623          J       02624          JUMP    02625          K       02626          LIC     02627
     MM      02630          NN      02631          N       02632          PP      02635          R       02636
     S       02637          SS      00013          START   02640          THETA   02641          UU      02664
     XA      03175          X       03176          XX      03202          YA      03513          Y       03514
     ZA      04025          ZB      04336

FUNCTION DICTIONARY
     ATN1    00000          COS     00001          .MTX    00002          .PRINT  00003          .PUNCH  00004
     .READ   00005          SIN     00006          SMOTH   00007          SQRT    00010          SYSTEM  00011
     ZERO    00012

ABSOLUTE CONSTANTS
04337 +000000000000    04340 +000000000001    04341 +000000000002    04342 +000000000003    04343 +000000000550
04344 +203622077324    04345 +207550000000    04346 +211416000000    04347 +211550000000    04350 +233000000000

STATEMENT DICTIONARY
02630 TXL -305531005530    02631 TXL -305416005404    02637 TXL -304745004744    02640 TXL -304362004354

PROGRAM
                PRINT FORMAT JUMP                                                          *001
04351 TSX +0 07400 4 00003    04352 STR -1 00001 0 02625    04353 STR -1 00000 0 00000
    START       READ FORMAT CON,N,B,LIC                                                    *002
04354 TSX +0 07400 4 00005    04355 STR -1 00001 0 02301    04356 STR -1 00000 0 02632    04357 STR -1 00000 0 02277
04360 STR -1 00000 0 02627    04361 STR -1 00000 0 00000
                PRINT FORMAT UU                                                            *003
04362 TSX +0 07400 4 00003    04363 STR -1 00001 0 02664    04364 STR -1 00000 0 00000
                READ FORMAT DATA,A(1,1,1)...A(1,1,N)                                       *004
04365 TSX +0 07400 4 00005    04366 STR -1 00001 0 02616    04367 TSX +0 07400 4 00002    04370 TXH +3 02622 0 02275
04371 TXH +3 00000 0 04340    04372 TXH +3 00000 0 04340    04373 TXH +3 00000 0 02632    04374 SUB +0 40200 0 04375
04375 TXH +3 00000 0 02275    04376 ALS +0 76700 0 00022    04377 STD +0 62200 0 04410    04400 TSX +0 07400 4 00002
04401 TXH +3 02622 0 02275    04402 TXH +3 00000 0 04340    04403 TXH +3 00000 0 04340    04404 TXH +3 00000 0 04340
04405 SUB +0 40200 0 04406    04406 TXH +3 00000 0 02275    04407 STA +0 62100 0 04410    04410 STR -1 00000 0 00000
04411 STR -1 00000 0 00000
                READ FORMAT DATA,A(1,2,1)...A(1,2,N)                                       *005
04412 TSX +0 07400 4 00005    04413 STR -1 00001 0 02616    04414 TSX +0 07400 4 00002    04415 TXH +3 02622 0 02275
04416 TXH +3 00000 0 04340    04417 TXH +3 00000 0 04341    04420 TXH +3 00000 0 02632    04421 SUB +0 40200 0 04422
04422 TXH +3 00000 0 02275    04423 ALS +0 76700 0 00022    04424 STD +0 62200 0 04435    04425 TSX +0 07400 4 00002
04426 TXH +3 02622 0 02275    04427 TXH +3 00000 0 04340    04430 TXH +3 00000 0 04341    04431 TXH +3 00000 0 04340
04432 SUB +0 40200 0 04433    04433 TXH +3 00000 0 02275    04434 STA +0 62100 0 04435    04435 STR -1 00000 0 00000
04436 STR -1 00000 0 00000
                READ FORMAT DATA,A(2,1,1)...A(2,1,N)                                       *006
04437 TSX +0 07400 4 00005    04440 STR -1 00001 0 02616    04441 TSX +0 07400 4 00002    04442 TXH +3 02622 0 02275
04443 TXH +3 00000 0 04341    04444 TXH +3 00000 0 04340    04445 TXH +3 00000 0 02632    04446 SUB +0 40200 0 04447
04447 TXH +3 00000 0 02275    04450 ALS +0 76700 0 00022    04451 STD +0 62200 0 04462    04452 TSX +0 07400 4 00002
04453 TXH +3 02622 0 02275    04454 TXH +3 00000 0 04341    04455 TXH +3 00000 0 04340    04456 TXH +3 00000 0 04340
04457 SUB +0 40200 0 04460    04460 TXH +3 00000 0 02275    04461 STA +0 62100 0 04462    04462 STR -1 00000 0 00000
04463 STR -1 00000 0 00000
                READ FORMAT DATA,A(2,2,1)...A(2,2,N)                                       *007
04464 TSX +0 07400 4 00005    04465 STR -1 00001 0 02616    04466 TSX +0 07400 4 00002    04467 TXH +3 02622 0 02275
04470 TXH +3 00000 0 04341    04471 TXH +3 00000 0 04341    04472 TXH +3 00000 0 02632    04473 SUB +0 40200 0 04474
04474 TXH +3 00000 0 02275    04475 ALS +0 76700 0 00022    04476 STD +0 62200 0 04507    04477 TSX +0 07400 4 00002
04500 TXH +3 02622 0 02275    04501 TXH +3 00000 0 04341    04502 TXH +3 00000 0 04341    04503 TXH +3 00000 0 04340
04504 SUB +0 40200 0 04505    04505 TXH +3 00000 0 02275    04506 STA +0 62100 0 04507    04507 STR -1 00000 0 00000
```

```
04510 STR -1 00000 0 00000
                EXECUTE ZERO.(C(1)...C(N))                                                    *008
04511 CLA +0 50000 0 02632    04512 SUB +0 40200 0 04513    04513 TXH +3 00000 0 02612    04514 ALS +0 76700 0 00022
04515 STD +0 62200 0 04517    04516 TSX +0 07400 4 00012    04517 TIX +2 00000 0 02611
                THROUGH S,FOR I=1,1,I.G.N                                                     *009
04520 CLA +0 50000 0 04340    04521 STO +0 60100 0 02623    04522 TRA +0 02000 0 04526    04523 CLA +0 50000 0 02623
04524 ADD +0 40000 0 04340    04525 STO +0 60100 0 02623    04526 CLA +0 50000 0 02623    04527 SUB +0 40200 0 02632
04530 TZE +0 10000 0 04532    04531 TPL +0 12000 0 04745
                WHENEVER A(1,1,I).E.0.OR.A(2,1,I).E.0,TRANSFER TO S                           *010
04532 TSX +0 07400 4 00002    04533 TXH +3 02622 0 02275    04534 TXH +3 00000 0 04341    04535 TXH +3 00000 0 04340
04536 TXH +3 00000 0 02623    04537 PAX +0 73400 1 00000    04540 CLA +0 50000 0 04337    04541 ORA -0 50100 0 04350
04542 FAD +0 30000 0 04350    04543 CHS +0 76000 0 00002    04544 ADD +0 40000 1 02275    04545 TZE +0 10000 0 04562
04546 TSX +0 07400 4 00002    04547 TXH +3 02622 0 02275    04550 TXH +3 00000 0 04340    04551 TXH +3 00000 0 04340
04552 TXH +3 00000 0 02623    04553 PAX +0 73400 1 00000    04554 CLA +0 50000 0 04337    04555 ORA -0 50100 0 04350
04556 FAD +0 30000 0 04350    04557 CHS +0 76000 0 00002    04560 ADD +0 40000 1 02275    04561 TNZ -0 10000 0 04563
04562 TRA +0 02000 0 02637
                A(1,1,I)=(90.-A(1,1,I))*6.2831853/360.                                        *011
04563 TSX +0 07400 4 00002    04564 TXH +3 02622 0 02275    04565 TXH +3 00000 0 04340    04566 TXH +3 00000 0 04340
04567 TXH +3 00000 0 02623    04570 PAX +0 73400 2 00000    04571 CLA +0 50000 0 04345    04572 FSB +0 30200 2 02275
04573 STO +0 60100 0 05576    04574 LDQ +0 56000 0 05576    04575 FMP +0 26000 0 04344    04576 FDP +0 24100 0 04347
04577 STQ -0 60000 0 05576    04600 TSX +0 07400 4 00002    04601 TXH +3 02622 0 02275    04602 TXH +3 00000 0 04340
04603 TXH +3 00000 0 04340    04604 TXH +3 00000 0 02623    04605 PAX +0 73400 1 00000    04606 CLA +0 50000 0 05576
04607 STO +0 60100 1 02275
                A(1,2,I)=A(1,2,I)*6.2831853/360                                               *012
04610 TSX +0 07400 4 00002    04611 TXH +3 02622 0 02275    04612 TXH +3 00000 0 04340    04613 TXH +3 00000 0 04341
04614 TXH +3 00000 0 02623    04615 PAX +0 73400 1 00000    04616 LDQ +0 56000 1 02275    04617 FMP +0 26000 0 04344
04620 STO +0 60100 0 05576    04621 CLA +0 50000 0 04343    04622 ORA -0 50100 0 04350    04623 FAD +0 30000 0 04350
04624 STO +0 60100 0 05577    04625 CLA +0 50000 0 05576    04626 FDP +0 24100 0 05577    04627 STQ -0 60000 0 05577
04630 TSX +0 07400 4 00002    04631 TXH +3 02622 0 02275    04632 TXH +3 00000 0 04340    04633 TXH +3 00000 0 04341
04634 TXH +3 00000 0 02623    04635 PAX +0 73400 1 00000    04636 CLA +0 50000 0 05577    04637 STO +0 60100 1 02275
                A(2,1,I)=(90.-A(2,1,I))*6.2831853/360.                                        *013
04640 TSX +0 07400 4 00002    04641 TXH +3 02622 0 02275    04642 TXH +3 00000 0 04341    04643 TXH +3 00000 0 04340
04644 TXH +3 00000 0 02623    04645 PAX +0 73400 2 00000    04646 CLA +0 50000 0 04345    04647 FSB +0 30200 2 02275
04650 STO +0 60100 0 05577    04651 LDQ +0 56000 0 05577    04652 FMP +0 26000 0 04344    04653 FDP +0 24100 0 04347
04654 STQ -0 60000 0 05577    04655 TSX +0 07400 4 00002    04656 TXH +3 02622 0 02275    04657 TXH +3 00000 0 04341
04660 TXH +3 00000 0 04340    04661 TXH +3 00000 0 02623    04662 PAX +0 73400 1 00000    04663 CLA +0 50000 0 05577
04664 STO +0 60100 1 02275
                A(2,2,I)=A(2,2,I)*6.2831853/360                                               *014
04665 TSX +0 07400 4 00002    04666 TXH +3 02622 0 02275    04667 TXH +3 00000 0 04341    04670 TXH +3 00000 0 04341
04671 TXH +3 00000 0 02623    04672 PAX +0 73400 1 00000    04673 LDQ +0 56000 1 02275    04674 FMP +0 26000 0 04344
04675 STO +0 60100 0 05577    04676 CLA +0 50000 0 04343    04677 ORA -0 50100 0 04350    04700 FAD +0 30000 0 04350
04701 STO +0 60100 0 05576    04702 CLA +0 50000 0 05577    04703 FDP +0 24100 0 05576    04704 STQ -0 60000 0 05577
04705 TSX +0 07400 4 00002    04706 TXH +3 02622 0 02275    04707 TXH +3 00000 0 04341    04710 TXH +3 00000 0 04341
04711 TXH +3 00000 0 02623    04712 PAX +0 73400 1 00000    04713 CLA +0 50000 0 05577    04714 STO +0 60100 1 02275
                C(I)=B/SIN.(A(1,1,I)-A(2,1,I))                                                *015
04715 TSX +0 07400 4 00002    04716 TXH +3 02622 0 02275    04717 TXH +3 00000 0 04341    04720 TXH +3 00000 0 04340
04721 TXH +3 00000 0 02623    04722 STO +0 60100 0 05577    04723 TSX +0 07400 4 00002    04724 TXH +3 02622 0 02275
04725 TXH +3 00000 0 04340    04726 TXH +3 00000 0 04340    04727 TXH +3 00000 0 02623    04730 PAX +0 73400 1 00000
04731 LXA +0 53400 2 05577    04732 CLA +0 50000 1 02275    04733 FSB +0 30200 2 02275    04734 STO +0 60100 0 05577
04735 TSX +0 07400 4 00006    04736 TXH +3 00000 0 05577    04737 STO +0 60100 0 05577    04740 CLA +0 50000 0 02277
04741 FDP +0 24100 0 05577    04742 LXA +0 53400 1 02623    04743 STQ -0 60000 1 02612
      S       CONTINUE                                                                        *016
04744 TRA +0 02000 0 04523
                THROUGH SS,FOR I=1,1,I.G.N                                                    *017
04745 CLA +0 50000 0 04340    04746 STO +0 60100 0 02623    04747 TRA +0 02000 0 04753    04750 CLA +0 50000 0 02623
04751 ADD +0 40000 0 04340    04752 STO +0 60100 0 02623    04753 CLA +0 50000 0 02623    04754 SUB +0 40200 0 02632
04755 TZE +0 10000 0 04757    04756 TPL +0 12000 0 05146
                AC(I)=C(I)*SIN.(A(2,1,I))                                                     *018
04757 TSX +0 07400 4 00002    04760 TXH +3 02622 0 02275    04761 TXH +3 00000 0 04341    04762 TXH +3 00000 0 04340
04763 TXH +3 00000 0 02623    04764 SUB +0 40200 0 04765    04765 TXH +3 00000 0 02275    04766 STA +0 62100 0 04770
```

```
04767 TSX +0 07400 4 00006    04770 TXH +3 00000 0 00000    04771 LXA +0 53400 1 02623    04772 STO +0 60100 0 05577
04773 LDQ +0 56000 1 02612    04774 FMP +0 26000 0 05577    04775 LXA +0 53400 1 02623    04776 STO +0 60100 1 00324
                    BC=C(I)*SIN.(A(1,1,I))                                                          *019
04777 TSX +0 07400 4 00002    05000 TXH +3 02622 0 02275    05001 TXH +3 00000 0 04340    05002 TXH +3 00000 0 04340
05003 TXH +3 00000 0 02623    05004 SUB +0 40200 0 05005    05005 TXH +3 00000 0 02275    05006 STA +0 62100 0 05010
05007 TSX +0 07400 4 00006    05010 TXH +3 00000 0 00000    05011 LXA +0 53400 1 02623    05012 STO +0 60100 0 05577
05013 LDQ +0 56000 1 02612    05014 FMP +0 26000 0 05577    05015 STO +0 60100 0 02276
                    ZA(I)=AC(I)*SIN.(A(1,2,I))/COS.(A(1,2,I))                                       *020
05016 TSX +0 07400 4 00002    05017 TXH +3 02622 0 02275    05020 TXH +3 00000 0 04340    05021 TXH +3 00000 0 04341
05022 TXH +3 00000 0 02623    05023 SUB +0 40200 0 05024    05024 TXH +3 00000 0 02275    05025 STA +0 62100 0 05027
05026 TSX +0 07400 4 00001    05027 TXH +3 00000 0 00000    05030 STO +0 60100 0 05577    05031 TSX +0 07400 4 00002
05032 TXH +3 02622 0 02275    05033 TXH +3 00000 0 04340    05034 TXH +3 00000 0 04341    05035 TXH +3 00000 0 02623
05036 SUB +0 40200 0 05037    05037 TXH +3 00000 0 02275    05040 STA +0 62100 0 05042    05041 TSX +0 07400 4 00006
05042 TXH +3 00000 0 00000    05043 LXA +0 53400 1 02623    05044 STO +0 60100 0 05576    05045 LDQ +0 56000 1 00324
05046 FMP +0 26000 0 05576    05047 FDP +0 24100 0 05577    05050 LXA +0 53400 1 02623    05051 STQ -0 60000 1 04025
                    ZB(I)=BC*SIN.(A(2,2,I))/COS.(A(2,2,I))                                          *021
05052 TSX +0 07400 4 00002    05053 TXH +3 02622 0 02275    05054 TXH +3 00000 0 04341    05055 TXH +3 00000 0 04341
05056 TXH +3 00000 0 02623    05057 SUB +0 40200 0 05060    05060 TXH +3 00000 0 02275    05061 STA +0 62100 0 05063
05062 TSX +0 07400 4 00001    05063 TXH +3 00000 0 00000    05064 STO +0 60100 0 05577    05065 TSX +0 07400 4 00002
05066 TXH +3 02622 0 02275    05067 TXH +3 00000 0 04341    05070 TXH +3 00000 0 04341    05071 TXH +3 00000 0 02623
05072 SUB +0 40200 0 05073    05073 TXH +3 00000 0 02275    05074 STA +0 62100 0 05076    05075 TSX +0 07400 4 00006
05076 TXH +3 00000 0 00000    05077 STO +0 60100 0 05576    05100 LDQ +0 56000 0 02276    05101 FMP +0 26000 0 05576
05102 FDP +0 24100 0 05577    05103 LXA +0 53400 1 02623    05104 STQ -0 60000 1 04336
                    XA(I)=AC(I)*COS.(A(1,1,I))                                                      *022
05105 TSX +0 07400 4 00002    05106 TXH +3 02622 0 02275    05107 TXH +3 00000 0 04340    05110 TXH +3 00000 0 04340
05111 TXH +3 00000 0 02623    05112 SUB +0 40200 0 05113    05113 TXH +3 00000 0 02275    05114 STA +0 62100 0 05116
05115 TSX +0 07400 4 00001    05116 TXH +3 00000 0 00000    05117 LXA +0 53400 1 02623    05120 STO +0 60100 0 05577
05121 LDQ +0 56000 1 00324    05122 FMP +0 26000 0 05577    05123 LXA +0 53400 1 02623    05124 STO +0 60100 1 03175
      SS    YA(I)=AC(I)*SIN.(A(1,1,I))                                                          *023
05125 TSX +0 07400 4 00002    05126 TXH +3 02622 0 02275    05127 TXH +3 00000 0 04340    05130 TXH +3 00000 0 04340
05131 TXH +3 00000 0 02623    05132 SUB +0 40200 0 05133    05133 TXH +3 00000 0 02275    05134 STA +0 62100 0 05136
05135 TSX +0 07400 4 00006    05136 TXH +3 00000 0 00000    05137 LXA +0 53400 1 02623    05140 STO +0 60100 0 05577
05141 LDQ +0 56000 1 00324    05142 FMP +0 26000 0 05577    05143 LXA +0 53400 1 02623    05144 STO +0 60100 1 03513
05145 TRA +0 02000 0 04750
                    EXECUTE ZERO.(C(1)...C(N))                                                      *024
05146 CLA +0 50000 0 02632    05147 SUB +0 40200 0 05150    05150 TXH +3 00000 0 02612    05151 ALS +0 76700 0 00022
05152 STD +0 62200 0 05154    05153 TSX +0 07400 4 00012    05154 TIX +2 00000 0 02611
                    THROUGH MM,FORJ=1,1,J.G.N                                                       *025
05155 CLA +0 50000 0 04340    05156 STO +0 60100 0 02624    05157 TRA +0 02000 0 05163    05160 CLA +0 50000 0 02624
05161 ADD +0 40000 0 04340    05162 STO +0 60100 0 02624    05163 CLA +0 50000 0 02624    05164 SUB +0 40200 0 02632
05165 TZE +0 10000 0 05167    05166 TPL +0 12000 0 05531
                    WHENEVER XA(J).E.0.AND.YA(J).E.0                                                *026
05167 LXA +0 53400 1 02624    05170 CLA +0 50000 0 04337    05171 ORA -0 50100 0 04350    05172 FAD +0 30000 0 04350
05173 CHS +0 76000 0 00002    05174 ADD +0 40000 1 03513    05175 TNZ -0 10000 0 05366    05176 LXA +0 53400 1 02624
05177 CLA +0 50000 0 04337    05200 ORA -0 50100 0 04350    05201 FAD +0 30000 0 04350    05202 CHS +0 76000 0 00002
05203 ADD +0 40000 1 03175    05204 TNZ -0 10000 0 05366
                    WHENEVER XA(J+1).E.0.AND.YA(J+1).E.0                                            *027
05205 CLA +0 50000 0 02624    05206 ADD +0 40000 0 04340    05207 PAX +0 73400 1 00000    05210 CLA +0 50000 0 04337
05211 ORA -0 50100 0 04350    05212 FAD +0 30000 0 04350    05213 CHS +0 76000 0 00002    05214 ADD +0 40000 1 03513
05215 TNZ -0 10000 0 05336    05216 CLA +0 50000 0 02624    05217 ADD +0 40000 0 04340    05220 PAX +0 73400 1 00000
05221 CLA +0 50000 0 04337    05222 ORA -0 50100 0 04350    05223 FAD +0 30000 0 04350    05224 CHS +0 76000 0 00002
05225 ADD +0 40000 1 03175    05226 TNZ -0 10000 0 05336
                    WHENEVER XA(J+2).E.0.AND.YA(J+2).E.0                                            *028
05227 CLA +0 50000 0 02624    05230 ADD +0 40000 0 04341    05231 PAX +0 73400 1 00000    05232 CLA +0 50000 0 04337
05233 ORA -0 50100 0 04350    05234 FAD +0 30000 0 04350    05235 CHS +0 76000 0 00002    05236 ADD +0 40000 1 03513
05237 TNZ -0 10000 0 05306    05240 CLA +0 50000 0 02624    05241 ADD +0 40000 0 04341    05242 PAX +0 73400 1 00000
05243 CLA +0 50000 0 04337    05244 ORA -0 50100 0 04350    05245 FAD +0 30000 0 04350    05246 CHS +0 76000 0 00002
05247 ADD +0 40000 1 03175    05250 TNZ -0 10000 0 05306
                    THROUGH AA,FOR K=J+3,1,K.G.N                                                    *029
05251 CLA +0 50000 0 02624    05252 ADD +0 40000 0 04342    05253 STO +0 60100 0 02626    05254 TRA +0 02000 0 05260
```

```
05255 CLA +0 50000 0 02626    05256 ADD +0 40000 0 04340    05257 STO +0 60100 0 02626    05260 CLA +0 50000 0 02626
05261 SUB +0 40200 0 02632    05262 TZE +0 10000 0 05264    05263 TPL +0 12000 0 05306
                  WHENEVER XA(K).NE.0.OR.YA(K).NE.0                                          *030
05264 LXA +0 53400 1 02626    05265 CLA +0 50000 0 04337    05266 ORA -0 50100 0 04350    05267 FAD +0 30000 0 04350
05270 CHS +0 76000 0 00002    05271 ADD +0 40000 1 03513    05272 TNZ -0 10000 0 05302    05273 LXA +0 53400 1 02626
05274 CLA +0 50000 0 04337    05275 ORA -0 50100 0 04350    05276 FAD +0 30000 0 04350    05277 CHS +0 76000 0 00002
05300 ADD +0 40000 1 03175    05301 TZE +0 10000 0 05305
                  J=K                                                                        *031
05302 CLA +0 50000 0 02626    05303 STO +0 60100 0 02624
                  TRANSFER TO MM                                                             *032
05304 TRA +0 02000 0 02630
      AA          END OF CONDITIONAL                                                         *033
05305 TRA +0 02000 0 05255
                  END OF CONDITIONAL                                                         *034
                  X=XA(J+2)-XA(J-1)                                                          *035
05306 CLA +0 50000 0 02624    05307 SUB +0 40200 0 04340    05310 STO +0 60100 0 05577    05311 CLA +0 50000 0 02624
05312 ADD +0 40000 0 04341    05313 PAX +0 73400 1 00000    05314 LXA +0 53400 2 05577    05315 CLA +0 50000 1 03175
05316 FSB +0 30200 2 03175    05317 STO +0 60100 0 03176
                  Y=YA(J+2)-YA(J-1)                                                          *036
05320 CLA +0 50000 0 02624    05321 SUB +0 40200 0 04340    05322 STO +0 60100 0 05577    05323 CLA +0 50000 0 02624
05324 ADD +0 40000 0 04341    05325 PAX +0 73400 1 00000    05326 LXA +0 53400 2 05577    05327 CLA +0 50000 1 03513
05330 FSB +0 30200 2 03513    05331 STO +0 60100 0 03514
                  J=J+2                                                                      *037
05332 CLA +0 50000 0 02624    05333 ADD +0 40000 0 04341    05334 STO +0 60100 0 02624
                  TRANSFER TO NN                                                             *038
05335 TRA +0 02000 0 02631
                  END OF CONDITIONAL                                                         *039
                  X=XA(J+1)-XA(J-1)                                                          *040
05336 CLA +0 50000 0 02624    05337 SUB +0 40200 0 04340    05340 STO +0 60100 0 05577    05341 CLA +0 50000 0 02624
05342 ADD +0 40000 0 04340    05343 PAX +0 73400 1 00000    05344 LXA +0 53400 2 05577    05345 CLA +0 50000 1 03175
05346 FSB +0 30200 2 03175    05347 STO +0 60100 0 03176
                  Y=YA(J+1)-YA(J-1)                                                          *041
05350 CLA +0 50000 0 02624    05351 SUB +0 40200 0 04340    05352 STO +0 60100 0 05577    05353 CLA +0 50000 0 02624
05354 ADD +0 40000 0 04340    05355 PAX +0 73400 1 00000    05356 LXA +0 53400 2 05577    05357 CLA +0 50000 1 03513
05360 FSB +0 30200 2 03513    05361 STO +0 60100 0 03514
                  J=J+1                                                                      *042
05362 CLA +0 50000 0 02624    05363 ADD +0 40000 0 04340    05364 STO +0 60100 0 02624
                  OTHERWISE                                                                  *043
05365 TRA +0 02000 0 05404
                  X=XA(J)-XA(J-1)                                                            *044
05366 CLA +0 50000 0 02624    05367 SUB +0 40200 0 04340    05370 LXA +0 53400 1 02624    05371 PAX +0 73400 2 00000
05372 CLA +0 50000 1 03175    05373 FSB +0 30200 2 03175    05374 STO +0 60100 0 03176
                  Y=YA(J)-YA(J-1)                                                            *045
05375 CLA +0 50000 0 02624    05376 SUB +0 40200 0 04340    05377 LXA +0 53400 1 02624    05400 PAX +0 73400 2 00000
05401 CLA +0 50000 1 03513    05402 FSB +0 30200 2 03513    05403 STO +0 60100 0 03514
                  END OF CONDITIONAL                                                         *046
        NN        R=SQRT.(X*X+Y*Y)                                                           *047
05404 LDQ +0 56000 0 03514    05405 FMP +0 26000 0 03514    05406 STO +0 60100 0 05577    05407 LDQ +0 56000 0 03176
05410 FMP +0 26000 0 03176    05411 FAD +0 30000 0 05577    05412 STO +0 60100 0 05577    05413 TSX +0 07400 4 00010
05414 TXH +3 00000 0 05577    05415 STO +0 60100 0 02636
                  THETA=270.-ATN1.(Y,X)*360./6.2831853                                       *048
05416 TSX +0 07400 4 00000    05417 TXH +3 00000 0 03514    05420 TXH +3 00000 0 03176    05421 STO +0 60100 0 05577
05422 LDQ +0 56000 0 05577    05423 FMP +0 26000 0 04347    05424 FDP +0 24100 0 04344    05425 STQ -0 60000 0 05577
05426 CLA +0 50000 0 04346    05427 FSB +0 30200 0 05577    05430 STO +0 60100 0 02641
                  WHENEVER THETA.L.0,THETA=360.+THETA                                        *049
05431 CLA +0 50000 0 04337    05432 ORA -0 50100 0 04350    05433 FAD +0 30000 0 04350    05434 CHS +0 76000 0 00002
05435 ADD +0 40000 0 02641    05436 TZE +0 10000 0 05443    05437 TPL +0 12000 0 05443    05440 CLA +0 50000 0 04347
05441 FAD +0 30000 0 02641    05442 STO +0 60100 0 02641
                  PRINT FORMAT PP,J,R,THETA,ZA(J),ZB(J),ZA(J)-ZA(J-1),ZB(J)-ZB(              *050
                 1J-1),ZA(J)-ZB(J),X,Y,XA(J),YA(J)                                           *050
```

```
05443 TSX +0 07400 4 00003    05444 STR -1 00001 0 02635    05445 STR -1 00000 0 02624    05446 STR -1 00000 0 02636
05447 STR -1 00000 0 02641    05450 CLA +0 50000 0 02624    05451 SUB +0 40200 0 05452    05452 TXH +3 00000 0 04025
05453 STA +0 62100 0 05454    05454 STR -1 00000 0 00000    05455 CLA +0 50000 0 02624    05456 SUB +0 40200 0 05457
05457 TXH +3 00000 0 04336    05460 STA +0 62100 0 05461    05461 STR -1 00000 0 00000    05462 CLA +0 50000 0 02624
05463 SUB +0 40200 0 04340    05464 LXA +0 53400 1 02624    05465 PAX +0 73400 2 00000    05466 CLA +0 50000 1 04025
05467 FSB +0 30200 2 04025    05470 STO +0 60100 0 05577    05471 STR -1 00000 0 05577    05472 CLA +0 50000 0 02624
05473 SUB +0 40200 0 04340    05474 LXA +0 53400 1 02624    05475 PAX +0 73400 2 00000    05476 CLA +0 50000 1 04336
05477 FSB +0 30200 2 04336    05500 STO +0 60100 0 05577    05501 STR -1 00000 0 05577    05502 LXA +0 53400 1 02624
05503 LXA +0 53400 2 02624    05504 CLA +0 50000 1 04025    05505 FSB +0 30200 2 04336    05506 STO +0 60100 0 05577
05507 STR -1 00000 0 05577    05510 STR -1 00000 0 03176    05511 STR -1 00000 0 03514    05512 CLA +0 50000 0 02624
05513 SUB +0 40200 0 05514    05514 TXH +3 00000 0 03175    05515 STA +0 62100 0 05516    05516 STR -1 00000 0 00000
05517 CLA +0 50000 0 02624    05520 SUB +0 40200 0 05521    05521 TXH +3 00000 0 03513    05522 STA +0 62100 0 05523
05523 STR -1 00000 0 00000    05524 STR -1 00000 0 00000
                C(J)=R                                                                    *051
05525 LXA +0 53400 1 02624    05526 CLA +0 50000 0 02636    05527 STO +0 60100 1 02612
      MM        CONTINUE                                                                  *052
05530 TRA +0 02000 0 05160
                PRINT FORMAT XX,AC(1)...AC(N)                                             *053
05531 TSX +0 07400 4 00003    05532 STR -1 00001 0 03202    05533 CLA +0 50000 0 02632    05534 SUB +0 40200 0 05535
05535 TXH +3 00000 0 00324    05536 ALS +0 76700 0 00022    05537 STD +0 62200 0 05540    05540 STR -1 00000 0 00323
05541 STR -1 00000 0 00000
                PUNCH FORMAT DATA1,N,C(1)...C(N)                                          *054
05542 TSX +0 07400 4 00004    05543 STR -1 00001 0 02614    05544 STR -1 00000 0 02632    05545 CLA +0 50000 0 02632
05546 SUB +0 40200 0 05547    05547 TXH +3 00000 0 02612    05550 ALS +0 76700 0 00022    05551 STD +0 62200 0 05552
05552 STR -1 00000 0 02611    05553 STR -1 00000 0 00000
                WHENEVER LIC.NE.0.                                                        *055
05554 CLA +0 50000 0 02627    05555 SUB +0 40200 0 04337    05556 TZE +0 10000 0 05574
                EXECUTE SMOTH.(C,ZB,N)                                                    *056
05557 TSX +0 07400 4 00007    05560 TXH +3 00000 0 02612    05561 TXH +3 00000 0 04336    05562 TXH +3 00000 0 02632
                PRINT FORMAT  XX,ZB(1)...ZB(N)                                            *057
05563 TSX +0 07400 4 00003    05564 STR -1 00001 0 03202    05565 CLA +0 50000 0 02632    05566 SUB +0 40200 0 05567
05567 TXH +3 00000 0 04336    05570 ALS +0 76700 0 00022    05571 STD +0 62200 0 05572    05572 STR -1 00000 0 04335
05573 STR -1 00000 0 00000
                END OF CONDITIONAL                                                        *058
                TRANSFER TO START                                                         *059
05574 TRA +0 02000 0 02640
                INTEGER N,I,J,K                                                           *060
                VECTOR VALUES DIM=3,1,2,200                                               *061
02617      +0 00000 0 00310    02620      +0 00000 0 00002    02621      +0 00000 0 00001    02622      +0 00000 0 00003
                VECTOR VALUES CON=$I5,2F10.0*$                                            *062
02300      +0 03300 5 46060    02301      +3 10573 0 22601
                VECTOR VALUES DATA1=$I5/(16F5.1)*$                                        *063
02613      +2 60533 0 13454    02614      +3 10561 7 40106
                VECTOR VALUES DATA=$16F5.1*$                                              *064
02615      -1 46060 6 06060    02616      +0 10626 0 53301
                VECTOR VALUES XX=$1HO/(1H ,14F8.1,F7.1)*$                                 *065
03177      +3 30134 5 46060    03200      +1 03301 7 32607    03201      +3 06073 0 10426    03202      +0 13000 6 17401
                VECTOR VALUES UU=$1H4,S4,1HJ,S7,1HR,S9,1HO,S8,2HZA,S8,2HZB,S7             *066
               1,5HZA-ZA,S5,5HZB-ZB,S5,5HZA-ZB,S7,1HU,S9,1HV,S9,1HX,S9,1HY/1H          *066
               2+,S22,1H-*$                                                             *066
02642      +0 27301 3 04054    02643      +0 13020 7 36202    02644      +1 17301 3 07061    02645      -3 30130 6 77362
02646      +0 13065 7 36211    02647      +3 06473 6 21173    02650      +2 27362 0 77301    02651      +0 53071 2 14071
02652      -3 12273 6 20573    02653      -3 30530 7 12240    02654      -0 07121 7 36205    02655      +0 77305 3 07121
02656      +0 23071 2 27362    02657      -3 12173 6 21073    02660      -3 36210 7 30230    02661      -2 21173 0 13000
02662      +0 77301 3 05173    02663      -3 30130 4 17362    02664      +0 13004 7 36204
                VECTOR VALUES PP=$1H ,I5,11F10.1*$                                        *067
02633      +3 30154 6 06060    02634      -3 30101 2 60100    02635      +0 13060 7 33105
                DIMENSION A(1000,DIM),C(200),XA(200),YA(200),ZA(200),ZB(200),             *068
               1AC(200)                                                                   *068
                VECTOR VALUES JUMP=$1H1*$                                                 *069
02625      +0 13001 5 46060
                END OF PROGRAM                                                            *070
05575 TSX +0 07400 4 00011
```

```
$COMPILE MAD                                                    04/17/62    004575


MAD (12 MAR 1962 VERSION) PROGRAM LISTING ... ... ...


                EXTERNAL FUNCTION (A,B,C)                                   *001
                ENTRY TO SMOTH.                                             *002
                EXECUTE ZERO.(B(1)...B(C))                                  *003
                THROUGH Q,FOR I=2,1,I.G.C-1                                 *004
                WHENEVER A(I-1).E.0.OR.A(I).E.0.OR.A(I+1).E.0, TRANSFER TO Q *005
                B(I)=0.25*A(I-1)+0.5*A(I)+0.25*A(I+1)                       *006
        Q       CONTINUE                                                    *007
                B(1)=0.75*A(1)+0.25*A(2)                                    *008
                B(C)=0.25*A(C-1)+0.75*A(C)                                  *009
                FUNCTION RETURN                                             *010
                INTEGER I,C                                                 *011
                END OF FUNCTION                                             *012


MAD PROGRAM,TYPE 12 MAR 1962   (ALL NUMBERS ARE OCTAL)

 NO. OF LOCATIONS 00204 TRA VECTOR SIZE  00001      TRA VECTOR STARTS 00000   ENTRY PT. 00013   ERASABLE STARTS 77777

 PROGRAM IS AN EXTERNAL FUNCTION.   THE FOLLOWING ARE ENTRIES
     SMOTH  00014

VARIABLE STORAGE
     I      00002          Q      00003

FUNCTION DICTIONARY
     ZERO   00000

ABSOLUTE CONSTANTS
00004 +000000000000    00005 +000000000001    00006 +000000000002    00007 +177400000000    00010 +200400000000
00011 +200600000000    00012 +233000000000

STATEMENT DICTIONARY
00003 TXL -300116000115


     $DATA                                                      04/17/62    004575

 MAP
     ERROR  00000*    SYSTEM 00000*    SPRINT 00000*    SKIP6  00000*    SCARDS 00000*    DPUNCH 00000*
     .EXIT  00000*    (MAIN) 10000     SMOTH  15601     .ERR   16005*    .IOH   16061*    .READ  20267*
     .PRINT 20372*    .PUNCH 20433*    ZERO   20506*    SQRT   20542*    COS    20620*    SIN    20620*
     ATNI   21017*    .MTX   21212*    (PROG) 21245     (SUBT) 75414     (ERAS) 77767
54141 LOCS. CAN BE SAFELY USED IN EXPANDING PROG. (OCTAL)
```