Computers & Geosciences Vol. 18, No. 7, pp. 823–837, 1992
Printed in Great Britain.

0098-3004/92 $5.00 + 0.00

A MICROCOMPUTER-BASED ROUTINE FOR OBTAINING MEAN WATERSHED PRECIPITATION FROM POINT VALUES

RICHARD A. COOKE and SAIED MOSTAGHIMI
Department of Agricultural Engineering, Virginia Polytechnic and State University, Blacksburg, VA 24061-0303, U.S.A.

(Received 9 October 1991; accepted 17 February 1992)

Abstract—An interactive microcomputer based BASIC routine for averaging precipitation over an irregularly shaped watershed is presented. Averaging methods used include the station average, Thiessen, and isohyetal methods, with the user having the option of using either inverse distance weighting schemes or punctual kriging for interpolation in the isohyetal method. The routine may be used to compare different averaging methods or to examine the effects of missing data. Data may be entered from a stored ASCII file, interactively from the keyboard from the application of the routine to a specific watershed also are presented.

Key Words: Thiessen polygons, Isohyetal method, Interplation, Punctual kriging, BASIC.

INTRODUCTION

Many of the models used in engineering hydrology require either a single estimate of average watershed precipitation or several subwatershed or cell estimates. Invariably, these estimates have to be derived from point values measured at a few gaging stations located on or near the area of interest. Various methods have been used to obtain areal averages from point measurements. These measurements differ in complexity from the use of the unweighted arithmetic mean to the finite-element methods developed by Hutchinson and Walley (1972), and more recently the cokriging techniques used by Seo, Krajewski, and Bowles (1990) to merge gage and radar measurements. Hall and Barclay (1975) listed the advantages and disadvantages of 13 of these methods. Singh and Chowdhury (1986) compared the results obtained by applying 13 methods to 3 watersheds, with differing climatological and physiographic features. They concluded that although one method may be preferred to another under certain conditions, they all yielded similar estimates of areal precipitation.

In this paper, an interactive microcomputer based routine for evaluating mean watershed precipitation, using a variety of widely used averaging techniques is presented and applied to the Nomini Creek watershed located in Westmoreland County, Virginia. The computerized Thiessen and isohyetal procedures developed by Diskin (1970) and Diskin and Davis (1970), respectively, in FORTRAN IV, are modified to take advantage of the color attributes, interactive features, and the ease of communication with external devices, inherent in BASIC. In addition, being microcomputer based, the routine can be widely distributed.

Although developed specifically for application to precipitation, the routine will be useful in any situation where it is necessary to use a number of point values to obtain an average of any continuous spatial stochastic variable. A spatial stochastic variable as used here refers to any random variable which can be associated with a spatial axis.

AVERAGE AREAL PRECIPITATION THEORY

Linsley, Kohler, and Paulhus (1982) have listed some widely used methods of estimating average areal precipitation, and the situations under which they are suitable. These methods include the station average, Thiessen, and isohyetal methods.

The "Station Average Method" is a simple arithmetic average of the precipitation recorded at all the gages in the watershed. This method is most suitable whenever there is a systematic network of gages on the watershed or when the variation in gage catch over the watershed is small. The "Thiessen method" is most suitable when the gage network is nonsystematic. This is a weighted-average method in which the weight for each gage is the proportion of the total area closer to it than to any other gage. The area closest to each gage is a polygon whose vertices are the points of intersection of the perpendicular bisectors of the lines joining that gage and its neighbors. The gage weights are independent of any precipitation event, and once established, they can be used for any storm, storm interval, or combination of storms occurring on the watershed. Thus the Thiessen method can be used for evaluating temporal variations in spatially averaged precipitation. The "isohyetal method" is somewhat subjective. In addition to the measured gage values, it requires some interpretation of orographic effects and storm morphology. All the information then is used to draw lines of equal precipitation (isohyets). A weighted mean of the isohyetal values then is calculated where the weight for each value is the proportion of the total watershed area that can be represented by that value. In the absence of any information of storm morphology or orographic effects, there is no significant difference between averages obtained by the Thiessen and the isohyetal methods.

COMPUTER IMPLEMENTATION

In situations with nonsystematic gage networks with highly variable precipitation, the estimates obtained from the use of the Thiessen and isohyetal methods are more accurate than those obtained from the station-average method, but they are more difficult to computerize.

The Thiessen method is a specific application of the Voronoi tessellation procedure. Based on research in the field of computational geometry, Preparata and Shamos (1985) have listed some of the most efficient algorithms for determining the polygons associated with the sample points. The time required to evaluate the polygons, at best, is proportional to the product of the number of gages and the logarithm of this number. When applied to irregular shaped areas, such as watersheds, the computational time is increased because the points of intersection between the polygons and the watershed boundary have to be determined.

Computer implementations of the isohyetal method, usually follow the two stage process outlined by Morrison (1971). In the first stage, a point interpolation method is used to provide precipitation estimates at nodes located either at the intersection of regularly spaced, orthogonal grid lines, or centrally between the grid lines. The isohyets then are produced by linear interpolation between these nodal values. Morrison (1971) concluded that the error introduced by the second stage linear interpolation may be assumed to be insignificant.

Lam (1983) listed several methods to perform the primary interpolation as mentioned. Included among these are the inverse distance weighting method and punctual kriging.

The inverse-distance weighting method is one of many point interpolation methods. It is versatile, easy to program, easy to understand, and fairly accurate under a wide range of conditions (Lam, 1983). Using this method, the precipitation at each node is given by

$$P_i = \frac{\sum_{j=1}^{G} P_j / D_{ij}^n}{\sum_{j=1}^{G} 1/D_{ij}^n} \qquad (1)$$

where P_i is the precipitation at node i; P_j is the precipitation at gage j; D_{ij} is the distance from i to j; G is the number of gages; and n is the inverse distance weighting power.

When n is set equal to 0, then the method is identical to the station-averaging method, and as n approaches infinity the method approximates the Thiessen method. Usually however, n is arbitrarily set at a value of 2. Values of 1.65 and 2 were used for n by Kelway (1974) and NOAA (1972), respectively.

In the punctual kriging method, a weighted mean of the gage values is evaluated for each node such that the estimation error is minimized. According to Davis (1986), this is accomplished by simultaneously solving the following set of equations.

$$P_i = \sum_{j=1}^{G} W_j P_j \qquad (2)$$

$$\sum_{j=1}^{G} W_j = 1 \qquad (3)$$

$$\sum_{j=1}^{G} W_j \gamma(D_{kj}) + \lambda = \gamma(D_{ki}) \quad k = 1, 2, \ldots G \qquad (4)$$

where λ is a variable added to ensure that the estimation error is the minimum possible, and $\gamma(D_{nm})$ is the semivariogram value between points n and m.

In situations where the nature of the semivariogram is unknown, the set of equations listed here can be solved by assuming that there is no nugget effect and that the semivariogram is given by

$$\gamma(D_{kj}) = A * f(D_{kj}) \qquad (5)$$

where f is a known function of D_{kj} and A is a constant. There is only a small group of functions which is suitable for use in defining a semivariogram. McBratney and Webster (1986) have listed these functions and explained why other functions are inappropriate. In the simplest situation $F(D_{kj}) = D_{kj}$, and Equation (4) may be replaced by

$$\sum_{j=1}^{G} W_j D_{kj} + \beta = D_{ki} \quad k = 1, 2, \ldots G \qquad (6)$$

because it is not necessary to determine the value of λ or β in order to solve Equation (2). This simplest situation, which is essentially the assumption of a linear semivariogram, is useful in situations where the actual shape of the semivariogram is unknown.

PROGRAM DEVELOPMENT

Thiessen procedure

A modified Thiessen procedure, as described here, was implemented.

(1) The watershed was inscribed in a bounding rectangle so that the watershed just touches the

rectangle on all four sides. A color attribute then was assigned to all the points within the boundary and another to the external points.

(2) The rectangle was divided into rectangular cells of equal area by superimposing a regular grid. The size of each cell was dependent on the fineness of the grid and the area of the inscribing rectangle. If, for example, a 50 × 50 grid was superimposed over a rectangle of area 1000 ha, each cell would be 0.4 ha in size.

(3) Each cell was assigned to the gage closest to its centroid.

(4) The number of cells assigned to each gage was determined. The contribution of each cell to the total count was based on the number of vertices of that cell that had the color attribute assigned to the interior of the watershed. Possible contributions were 1.0 for four internal vertices, 0.75 for three, 0.5 for two, 0.25 for one and 0.0 for no internal vertices. Vertices on the boundary line were treated in the same manner as internal vertices.

(5) The weight for each gage was the number of cells attributed to that gage, expressed as a fraction of the total number of cells within the watershed.

Isohyetal procedure

In this implementation of the isohyetal procedure the evaluation of mean precipitation, based on the area between isohyets, is replaced by the arithmetic mean of the precipitation in the rectangular cells formed by the intersecting grid lines. The steps in the procedure were as follows:

(1) The watershed was inscribed in a bounding rectangle so that the watershed just touches the rectangle on all four sides. A color attribute then was assigned to all the points within the boundary and another to the external points.

(2) The rectangle was divided into rectangular cells of equal area as previously described.

(3) An interpolation method, either punctual kriging or the inverse distance weighting method, was used to estimate the precipitation at the centroid of each cell.

(4) The sum of the product of precipitation value for each cell and the proportion of the cell inside the watershed boundaries was determined. The proportion of a cell inside the watershed boundaries was evaluated in the manner outlined in the fourth step of the modified Thiessen procedure.

(5) The sum obtained was divided by the number of cells inside the watershed.

The routine is written for use in the MS DOS environment and can be run on any IBM PC, or compatible machine, which has a graphics card. Although a color monitor is preferable, it is not essential. Data may be read in from an existing ASCII file in the format shown in Table 1 or may be entered interactively using a digitizer.

The routine is set up for the NUMONICS digitizer Model 2200, or a SUMMAGRAPHICS bit pad with an RS232 interface and with default communication

Table 1. Format of input data file

CARD	FIELD	VARIABLE	VARIABLE TYPE	VARIABLE DESCRIPTION
1	1	NG	INTEGER	NUMBER OF GAGES ON/NEAR WATERSHED
	2	NP	INTEGER	NUMBER OF POINTS USED TO DEFINE WATERSHED BOUNDARY
	3	DMAX	REAL	MAXIMUM DIMENSION OF WATERSHED
2	1	X(I)	REAL	X COORDINATE OF GAGE I
	2	Y(I)	REAL	Y COORDINATE OF GAGE I
	3	P(I)	REAL	PRECIPITATION AT GAGE I
		(Card 2 is written NG times)		
3	1	X(J)	REAL	X COORDINATE OF POINT J
	2	Y(J)	REAL	Y COORDINATE OF POINT J
		(Card 3 is written NP times)		

Table 2. Default digitizer communication protocol

LOCATION	:COM1	BAUD RATE	:2400
PARITY	:EVEN	DATA BITS	:7
STOP BITS	:2		

protocol as shown in Table 2. The program provides actual configuration of the digitizer. Modified versions of the program for use with other digitizers can be obtained from the authors.

The program provides mean watershed precipitation, an isohyetal map for the isohyetal method, and Thiessen polygons and gage weights for the Thiessen method.

ANALYSIS OF RAINFALL AT NOMINI CREEK

The Department of Agricultural Engineering at the Virginia Polytechnic Institute and State University operates several monitoring stations on the Nomini Creek watershed, located in Westmoreland County, Virginia, as part of an ongoing water quality study. Seven rain gages are located on or near the watershed. The location of these gages are shown in Figure 1. Models are being developed to simulate surface and groundwater on the watershed. Precipitation is of major importance in these models. The routine presented herein was used to evaluate the mean annual watershed precipitation; mean watershed precipitation for May 1990; temporal rainfall distribution for a storm occurring on 29 May 1990; and mean watershed precipitation for a storm occurring on 15 June 1990, during which one gage malfunctioned. Data for these rainfall events are given in Table 3.

The analysis was done using data from five of the seven gauges shown in Figure 1 as the data from the other two gages are not yet in an easily accessible form.

Mean annual precipitation

The mean annual watershed precipitation was obtained by averaging the mean annual station precipitation values given in Table 3. Table 4 shows the results obtained from the different averaging methods. For the Thiessen and isohyetal methods, a 30 × 30 grid was superimposed over the watershed. The inverse distance weighting power was set to 2. All the values are within 1% of each other. Figures 2 and 3 show the Thiessen polygons and isohyetal maps for the watershed, respectively, with the shaded regions in Figure 2 representing the areas of influence of the rain gages. The area of influence of a rain gage is defined as the area within the watershed that is closer to the rain gage than to any other.

Theoretically, the smaller the size of each cell the more accurate are the results obtained. This is especially true for watersheds with irregular boundaries because smaller cells define the external

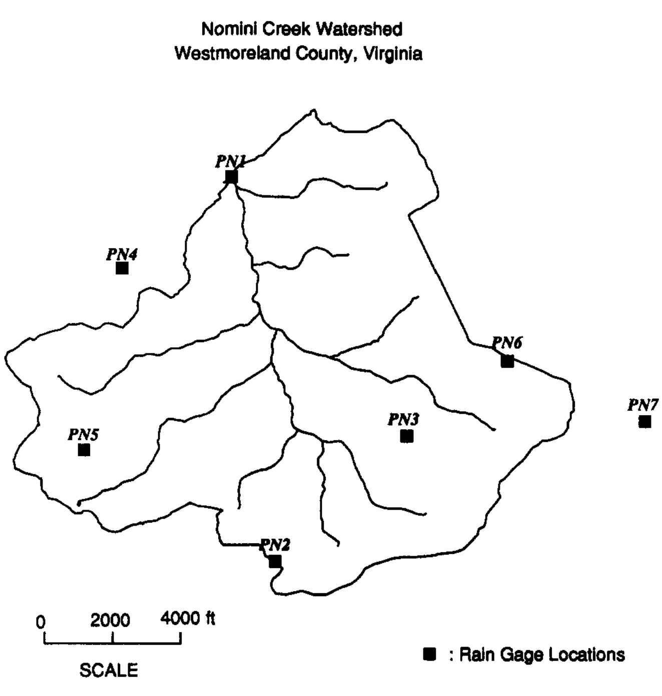

Figure 1. Map of Nomini Creek watershed showing location of rain gages.

Table 3. Selected precipitation record for Nomini Creek

STATION	Precipitation (mm) Mean Annual	May, 1990	June 15, 1990
PN1	1227.96	314.96	35.30
PN3	1268.04	287.27	-----
PN4	1214.64	307.85	21.33
PN5	1215.84	275.84	17.52
PN7	1218.96	276.10	20.57

Cumulative Station Precipitation (mm) on May 29,1990

TIME (hr)	PN1	PN3	PN4	PN5	PN7
0.0*	0.000	0.000	0.000	0.000	0.000
1.0	0.254	0.508	0.000	0.254	0.648
2.0	0.882	0.941	0.000	0.635	0.812
3.0	3.810	2.286	2.286	3.810	3.048
4.0	6.924	3.556	4.572	5.334	4.572
5.0	9.398	4.572	5.588	5.842	11.430
6.0	26.924	17.526	22.564	18.034	25.146
7.0	40.132	31.242	35.306	36.576	35.560
8.0	59.182	42.372	51.054	55.118	48.514
9.0	69.596	56.388	65.532	70.104	55.626
10.0	74.422	60.452	69.596	73.406	61.468
11.0	77.216	64.770	74.676	78.232	64.516
12.0	84.582	69.342	79.502	82.296	69.088
13.0	99.822	81.026	92.202	94.488	79.454
14.0	102.362	91.186	104.140	105.156	85.852

* : The rainstorm began at 5:00 am on May 29, 1990

and internal areas with greater precision. However, beyond a certain limit the gain in accuracy is insignificant and does not justify the increased computational time. Figure 4 shows the effect of cell size on the mean annual precipitation. Over the range considered, the results are independent of cell size for grids smaller than 30 × 30.

Mean rainfall for May 1990

Isohyetal maps and Thiessen polygons for precipitation occurring in May 1990 are shown in Figures 5 and 6, respectively. The inverse distance weighting power was 2 in Figure 5A and 15 in Figure 5B. The value of 15 used in Figure 5B is outside the range of values that is normally used. The purpose of this inflated value is to demonstrate the fact that as the inverse distance weighting power increases, the isohyets tend to bunch close together, and the resulting map approaches the map obtained by the Thiessen method. The Thiessen polygons are identical to those in Figure 2 demonstrating the fact that they are independent of rainfall amount. In all three situations, the values of mean watershed precipitation were not significantly different from each other.

Temporal rainfall distribution

The temporal distribution for a storm occurring on 29 May 1990 was evaluated using the mean of the precipitation at the five stations, and by using the Thiessen weights to obtain a weighted average. The resulting rainfall intensity histograms are shown in Figure 7. Both the station average and the Thiessen methods gave similar results for this storm.

Missing precipitation data

Carelessness or instrument failure may result in the loss of data from a precipitation gage. When

Table 4. Mean annual watershed precipitation at Nomini Creek

Station average method	1229.1
Thiessen method	1241.6
Isohyetal method (Inverse distance)	1236.3
Isohyetal method (Kriging)	1238.4

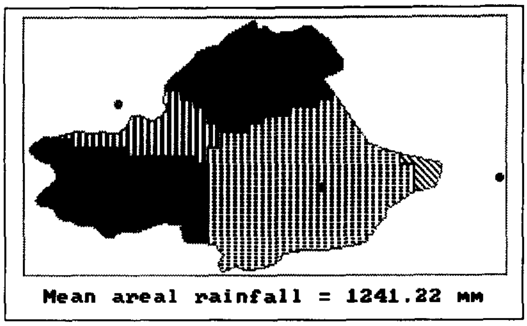

Figure 2. Thiessen polygons for mean annual precipitation at Nomini Creek.

this occurs, it may be necessary to either estimate the precipitation at that gage, or to ignore that gage when evaluating the mean watershed precipitation. The normal ratio method may be used to estimate missing precipitation at a gage for which the mean annual precipitation is known. In this method the missing precipitation is estimated by

$$P_x = \frac{A_x}{m} \sum_{i=1}^{m} \frac{P_i}{A_i} \qquad (7)$$

where P_x is the estimated precipitation at station x; A_x is the mean annual precipitation at station x; P_i is the measured precipitation at station i; A_i is the mean annual precipitation at station i; and m is the number of rain gages surrounding the gage with missing data.

By using this method the estimated precipitation at station PN3 is given by

$$P = \frac{1268.04}{4} \times \left(\frac{35.3}{1227.96} + \frac{21.33}{1214.64} + \frac{17.52}{1215.84} + \frac{20.57}{1218.96} \right) = 24.6\ \text{mm}$$

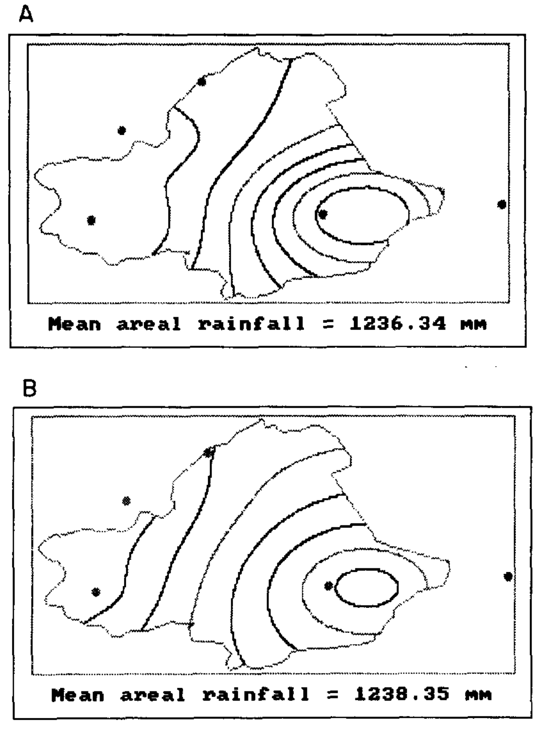

Figure 3. Isohyetal map for mean annual precipitation at Nomini Creek with point values interpolated by (A) inverse distance weighted method and (B) punctual kriging.

and the resulting mean watershed precipitation would be given by

$$P_m = 0.209 * 35.3 + 0.447 * 24.6 + 0.103 * 21.33 + 0.223 * 17.52 + 0.018 * 20.57 = 24.84\ \text{mm}.$$

Alternately, the mean precipitation could be estimated by ignoring gage PN3, and using the other gages to obtain a mean watershed value. When this is done, the mean watershed precipitation becomes 23.68 mm. The new Thiessen polygons are shown in

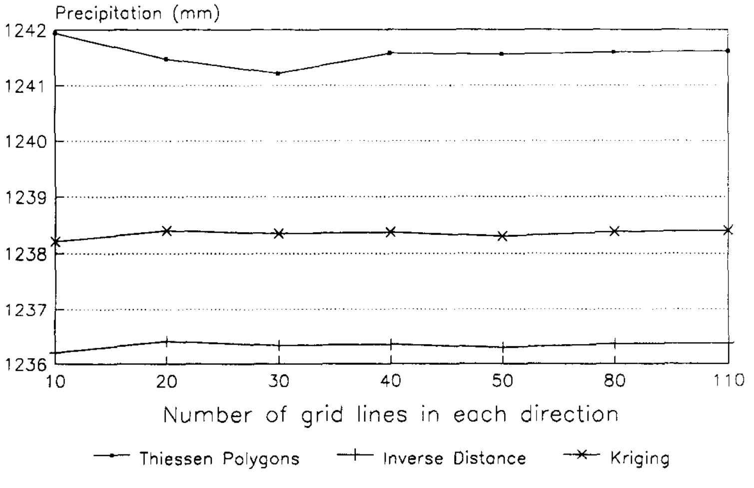

Figure 4. Effect of grid size on mean annual precipitation at Nomini Creek.

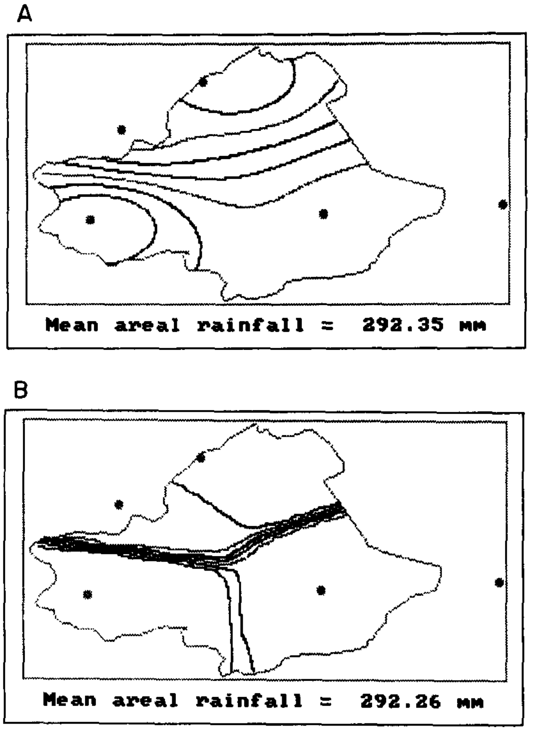

Figure 5. Isohyetal map for rainfall at Nomini Creek for May 1990. Inverse distance weighing power = 2 (A) and = 15 (B).

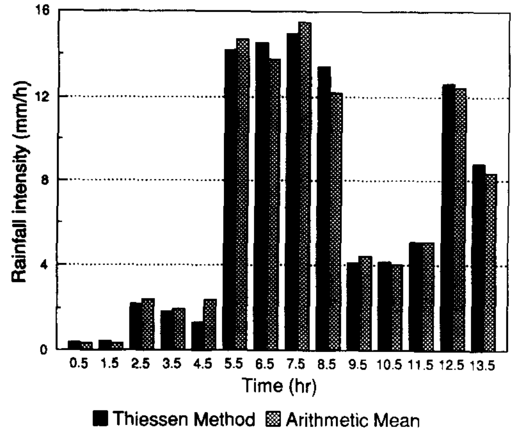

Figure 7. Rainfall histogram for storm occurring 29 May 1990 at Nomini Creek.

Figure 8. There is almost a 5% difference between these two values. The value obtained by using the normal ratio method can be considered to be more accurate because it includes any bias that the gage with missing precipitation might impose on the watershed.

DISCUSSION AND CONCLUSIONS

A routine was developed for averaging precipitation over an irregularly shaped watershed. In this routine the user can select one of three averaging methods, and one of two interpolation algorithms for the isohyetal method. An application of the routine to the Nomini Creek watershed is presented.

The interpolation methods did not yield significantly different values of mean precipitation in this application. No conclusions therefore could be drawn about the accuracy of the method. In situations where the method may yield different values, it is difficult to draw conclusions about their accuracy, because the true areal rainfall is never known. When both point and radar estimates are available, areal distribution patterns derived from the method developed by Seo, Krajewski, and Bowles (1990), may be used to approximate true areal rainfall and used to compare the accuracy of point methods.

The sensitivity of areal rainfall distribution to the power used in the inverse-distance weighted method needs to be determined separately for convective and cyclonic rainfall as well as for combination of both types. The Nomini Creek data set did not contain enough information to perform this analysis.

The optimum grid size to achieve a balance between precision and computational time is dependent on the degree of irregularity of the watershed boundary and on the speed of the computer. In this routine the BASIC compiler used prevented the use of grids finer than 120 × 120.

Acknowledgments—This study was supported, in part, by the Virginia Department of Conservation and Recreation, Division of Soil and Water Conservation, Richmond, Virginia.

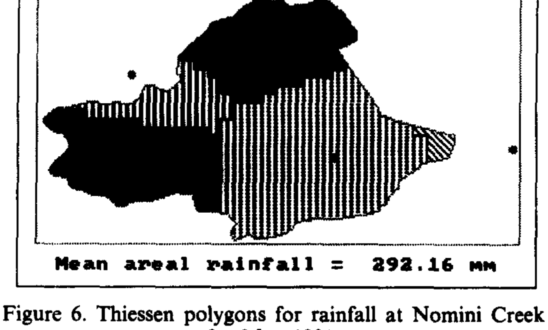

Figure 6. Thiessen polygons for rainfall at Nomini Creek for May 1990.

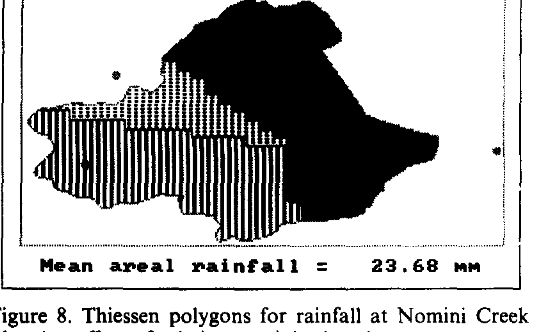

Figure 8. Thiessen polygons for rainfall at Nomini Creek showing effect of missing precipitation data at one gage.

REFERENCES

Davis, J. C., 1986, Statistics and data analysis in geology (2nd ed): John Wiley & Sons, New York, 646 p.

Diskin, M. H., 1970, On the computer evaluation of Thiessen weights: Jour. Hydrology, v. 11, no. 1, p. 69–78.

Diskin, M. H., and Davis, P. R., 1970, A computer program for interpolation and plotting of isohyetal points: U.S. Dept. Agriculture, ARS, Tucson, Arizona, 22p.

Hall, A. J., and Barclay, P. A., 1975, Metods of determining areal rainfall from observed data, *in* Chapman, T. G., and Dunin, F. X., eds., Prediction in catchment hydrology: Australian Acad. Science, Canberra, p. 47–57.

Hutchinson, P., and Walley, W. J., 1972, Calculation of areal rainfall using finite element techniques with altitudinal corrections: Bull. Intern. Assoc. Hydrol. Sci., v. 17, no. 3, p. 259–272.

Kelway, P. S., 1974, A scheme for assessing the reliability of interpolated rainfall estimates: Jour. Hydrology, v. 21, no. 3, p. 247–267.

Lam, N. S., 1983, Spatial interpolation methods: review: The American Cartographer, v. 10, no. 2, p. 129–149.

Linsley, R. K., Kohler, M. A., and Paulhus, J. L. H., 1982, Hydrology for engineers (3rd ed.): McGraw-Hill Book Co., New York, 508 p.

McBratney, A. B., and Webster, R., 1986, Choosing functions for semi-variograms of soil properties and fitting them to estimates: Jour. Soil Science, v. 37, no. 4, p. 617–639.

Morrison, J. L., 1971, Method-produced error in isarithmic mapping: Technical Monograph No. CA-5, Cartography Division, American Congress on Surveying and Mapping, 76 p.

N.O.A.A., 1972, National Weather Service River Forecast System forecast procedures: TM NWS HYDRO-14, U.S. Dept. Commerce, Silver Spring, Maryland.

Preparata, F. P., and Shamos, M. I., 1985, Computational geometry: Springer-Verlag, New York, 390 p.

Seo, D., Krajewski, W. F., and Bowles, D. S., 1990, Stochastic interpolation of rainfall data from rain gages and radar using cokriging. I. Design of experiments: Water Resour. Res., v. 26, no. 3, p. 469–477.

Singh, V. P., and Chowdhury, P. K., 1986, Comparing some methods of estimating mean areal rainfall: Water Res. Bull. v. 22. no. 2, p. 275–281.

APPENDIX

Program Listing

```
'***********************************************************************
'**                                                                   **
'** INTRODUCTORY SCREEN                                               **
'**                                                                   **
'***********************************************************************

    WIDTH 80: xm = 2: ym = 2: NG = 3
    REM $DYNAMIC
    DIM f(xm, ym), RGX(20), RGY(20), RGZ(20), RZ(xm, ym), aa(xm, ym)
    DIM x(200), Y(200), vtx(2), vty(2), c(20), x$(9)
    DIM SM(NG), PTB(NG)
    KEY OFF
    SCREEN 0
    COLOR 7, 0, 7
    LOCATE 1, , 0
    CLS : COLOR , , 2
    PRINT : COLOR 3
    PRINT SPC(15); "Department of Agricultural Engineering"
    PRINT SPC(28); "Virginia Tech"
    COLOR 14
    PRINT
    PRINT SPC(14); "Developed by: R. Cooke and S. Mostaghimi"
    PRINT : COLOR 5
    PRINT
    PRINT
    PRINT SPC(10); "        Spatial Averaging of Precipitation"
    PRINT SPC(10); "        =================================="
    PRINT : COLOR 2
    PRINT SPC(10); "    Estimation of mean watershed rainfall depth"
    PRINT SPC(10); "    using any of the following averaging schemes:"
    PRINT : COLOR 13
    PRINT SPC(15); "   * Station Average Method"
    PRINT SPC(15); "   * Theissen Polygon Method"
    PRINT SPC(15); "   * Isohyetal Method"
    PRINT : PRINT : PRINT
    PRINT : COLOR 4, 0
    PRINT SPC(20); " PRESS ANY KEY TO CONTINUE..."
360 IF INKEY$ = "" THEN 360
    COLOR 7, 0, 0
    WIDTH 80
    CLS

'***********************************************************************
'**                                                                   **
'** SELECTING ITEM FROM MENU                                          **
'**                                                                   **
'***********************************************************************
```

```
530 CLEAR : KEY OFF: SCREEN 0, 0, 0: WIDTH 80:
    DIM tile$(20)
    tile$(1) = CHR$(&HFF): tile$(2) = CHR$(&HF) + CHR$(&H0)
    tile$(4) = CHR$(&HFA) + CHR$(&H7A) + CHR$(&HAF) + CHR$(&HA7)
    tile$(5) = CHR$(&HC0) + CHR$(&H30) + CHR$(&HC) + CHR$(&H3):
    tile$(6) = CHR$(&H80) + CHR$(&H20) + CHR$(&H8) + CHR$(&H2)
    tile$(7) = CHR$(&HF8) + CHR$(&HE2) + CHR$(&H8B) + CHR$(&H2F):
    tile$(8) = CHR$(&H3C): tile$(3) = CHR$(&HA0)
    FOR i = 9 TO 20: tile$(i) = tile$(i - 8): NEXT i
    COLOR 14, 0: CLS : LOCATE 1, 20, 0: PRINT STRING$(43, "-")
    LOCATE 3, 20: PRINT STRING$(43, "-"): LOCATE 1, 19: PRINT "+"
    LOCATE 1, 63: PRINT "+": LOCATE 3, 63: PRINT "+": LOCATE 3, 19
    PRINT "+": LOCATE 2, 19: PRINT "¦"; SPC(43); "¦": COLOR 4, 7
    LOCATE 2, 25: PRINT "ESTIMATION OF AREAL PRECIPITATION": COLOR 3, 0
    LOCATE 5, 24: PRINT "        Available Options        "
    LOCATE 16, 14: COLOR 23: PRINT "     "; : COLOR 24, 0
    PRINT "+------------------------------------------+";
    COLOR 23: PRINT "   ": COLOR 7: LOCATE 17, 14: COLOR 24
    PRINT "     ¦"; : COLOR 4, 0
    PRINT "Strike Key Corresponding To Desired Option"; : COLOR 24
    PRINT "¦    ": COLOR 7: LOCATE 18, 14: COLOR 23: PRINT "     ";
    COLOR 24, 0: PRINT "+------------------------------------------+";
    COLOR 23: PRINT "    ": COLOR 7: COLOR 0, 5: LOCATE 7, 28
    PRINT " A ": LOCATE 9, 28: PRINT " B ": LOCATE 11, 28: PRINT " C "
    LOCATE 13, 28: PRINT " D ": LOCATE 7, 34: COLOR 2, 0
    PRINT "Station Average Method": LOCATE 9, 34
    PRINT "Theissen Polygon Method": LOCATE 11, 34
    PRINT "Isohyetal Method": LOCATE 13, 34: PRINT "Return to DOS"
710 RP$ = UCASE$(INKEY$): IF RP$ = "" THEN 710
620 IF RP$ < "A" OR RP$ > "D" THEN 710
    ON ASC(RP$) - 64 GOTO 650, 660, 670, 680
    END
650 GOSUB 3000
660 GOSUB 6000: GOSUB 4000
670 FLG = 99: GOSUB 6000: GOSUB 5000
680 CLS : SYSTEM
    END

'*********************************************************************
'**                                                                 **
'** ROUTINE FOR STATION AVERAGE METHOD                              **
'**                                                                 **
'*********************************************************************

3000 SCREEN 0: WIDTH 80: CLS
     INPUT "How many point gauges are located on/near the watershed"; NG
     DIM RGZ(NG)
     PRINT : SUM = 0
     FOR M = 1 TO NG:
     PRINT "What is the rainfall catch at gauge "; : PRINT M, : INPUT RGZ(M)
     PRINT : SUM = SUM + RGZ(M)
     NEXT M
     MEAN = SUM / NG
     PRINT "    Mean areal rainfall = "; : PRINT USING "####.##"; MEAN;
     PRINT " mm": PRINT "    Press the SPACE BAR to continue"
3230 IF INKEY$ <> " " THEN 3230
     WIDTH 80: SCREEN 0: CLS : RETURN 530

'*********************************************************************
'**                                                                 **
'** ROUTINE FOR THIESSEN POLYGON METHOD                             **
'**                                                                 **
'*********************************************************************

4000 FOR i = 1 TO xm: FOR j = 1 TO ym
       HD = XMIN + (2 * i - 1) / 2 * dx: VD = YMIN + (2 * j - 1) / 2 * dy
       SUM1 = 0: SUM2 = 0: sume = 0: DBMIN = 999999999#
       FOR M = 1 TO NG: DB = (RGX(M) - HD) ^ 2 + (RGY(M) - VD) ^ 2
       DB = DB ^ .5: IF DB < DBMIN THEN DBMIN = DB: TS = M
       NEXT M
       vtx(1) = XMIN + (i - 1) * dx: vty(1) = YMIN + (j - 1) * dy
       vtx(2) = XMIN + i * dx: vty(2) = YMIN + j * dy
       FOR K = 1 TO 2: FOR L = 1 TO 2
       IF POINT(vtx(L), vty(K)) <> 2 THEN sume = sume + 1
       NEXT L, K
       f(i, j) = RGZ(TS):  RZ = f(i, j) * sume / 4
       IF sume > 0 THEN aa(i, j) = TS
       SUM3 = SUM3 + RZ: SUM4 = SUM4 + sume / 4
       SM(TS) = SM(TS) + sume / 4
```

```
     NEXT j, i
     FOR i = 1 TO xm: FOR j = 1 TO ym
      vtx(1) = XMIN + (i - 1) * dx: vty(1) = YMIN + (j - 1) * dy
      vtx(2) = XMIN + i * dx: vty(2) = YMIN + j * dy
      IF i > 1 AND aa(i, j) <> aa(i - 1, j) THEN LINE (vtx(1), vty(1))-(vtx(1), vty(2)), 2
      IF j > 1 AND aa(i, j) <> aa(i, j - 1) THEN LINE (vtx(1), vty(1))-(vtx(2), vty(1)), 2
     NEXT j, i: FOR N = 1 TO NG: PTB(N) = 0: NEXT N
     FOR j = 1 TO ym: FOR i = 1 TO xm: N = aa(i, j)
     IF N = 0 OR PTB(N) = 5 THEN 4327
      HD = XMIN + (2 * i - 1) / 2 * dx: VD = YMIN + (2 * j - 1) / 2 * dy
      PAINT (HD, VD), tile$(N), 2: IF j > 2 THEN PTB(N) = PTB(N) + 1
4327 NEXT i, j
     FOR i = 2 TO NP: LINE (x(i - 1), Y(i - 1))-(x(i), Y(i)), 1: NEXT i
     PAINT ((XMIN - dx / 4), (YMIN - dy / 4)), 0, 1
     LINE ((XMIg - dx / 2), (YMIg - dy / 2))-((XMAg + dx / 2), (YMAg + dy / 2)), 1, B
     FOR i = 1 TO NG: CIRCLE (RGX(i), RGY(i)), DR, 1, , , 1
     PAINT (RGX(i), RGY(i)), 1, 1: NEXT i
     LOCATE 22, 1: PRINT "    Mean areal rainfall = ";
     PRINT USING "####.##"; SUM3 / SUM4; : PRINT " mm"
     LOCATE 2, 1: PRINT "    Press the SPACE BAR to continue"
4333 IF INKEY$ <> " " THEN 4333
     WIDTH 80: SCREEN 0: CLS
     SST = 0: GMAX = 0
     FOR i = 1 TO NG: IF SM(i) > GMAX THEN GMAX = SM(i):
     NEXT i
     PRINT "Station                Weight"; : LOCATE , 62: PRINT "Cum. Weight"
     PRINT
     FOR N = 1 TO NG: CL1 = (N MOD 6) + 1: COLOR CL1: PRINT N;
       LP = INT(SM(N) / GMAX * 40 + 1)
       COLOR 0, CL1: LOCATE , 10: FOR i = 1 TO LP: PRINT "#"; : NEXT i
       COLOR CL1, 0: PRINT USING "    #.### "; SM(N) / SUM4;
       LOCATE , 62: SST = SST + SM(N): COLOR 7
       PRINT USING "   #.###"; SST / SUM4
     NEXT N: COLOR 14
     LOCATE 23, 21: PRINT "    Press any key to continue"
4452 IF INKEY$ = "" THEN 4452
     RETURN 530

'**********************************************************************
'**                                                                  **
'** ROUTINE FOR ISOHYETAL METHOD                                     **
'**                                                                  **
'**********************************************************************

5000 IF WW = 1 THEN 5400

'----------------------------------------------------------------------
'--                                                                  --
'-- KRIGING ROUTINE                                                  --
'--                                                                  --
'----------------------------------------------------------------------

     REM $DYMAMIC
     REDIM COEFF(21, 42)
     FOR i = 1 TO NG: FOR j = 1 TO NG
     COEFF(i, j) = (RGX(i) - RGX(j)) ^ 2 + (RGY(i) - RGY(j)) ^ 2
     COEFF(i, j) = COEFF(i, j) ^ .5: NEXT j
     COEFF(i, NG + 1) = 1: COEFF(NG + 1, i) = 1: NEXT i
     COEFF(NG + 1, NG + 1) = 0: N = NG + 1

     FOR j = 1 TO N
     FOR K = N + 1 TO 2 * N
     IF j = K - N THEN 5110
     COEFF(j, K) = 0
     GOTO 5112
5110 COEFF(j, K) = 1
5112 NEXT K
     NEXT j
     Z = 0
     FOR R = 1 TO N
     GOSUB 5140
     IF S = 0 THEN 5136
     GOSUB 5154
     GOSUB 5164
     GOSUB 5176
     Z = Z + 1
     NEXT R
     GOTO 5194
```

```
5136 PRINT "COEFFICIENT MATRIX IS NOT INVERTIBLE"
     GOTO 5194
5140 FOR j = Z + 1 TO N
     IF COEFF(j, R) = 0 THEN 5148
     S = j
     RETURN
5148 NEXT j
     S = 0
     RETURN
5154 IF Z + 1 = S THEN 5162
     FOR K = 1 TO 2 * N
     T = COEFF(S, K): COEFF(S, K) = COEFF(Z + 1, K): COEFF(Z + 1, K) = T
     NEXT K
5162 RETURN
5164 T = COEFF(Z + 1, R)
     IF T = 1 THEN RETURN
     FOR K = 1 TO 2 * N
     COEFF(Z + 1, K) = COEFF(Z + 1, K) / T
     NEXT K
     RETURN
5176 FOR j = 1 TO N
     IF j = Z + 1 THEN 5190
     T = COEFF(j, R)
     IF T = 0 THEN 5190
     FOR K = 1 TO 2 * N
     COEFF(j, K) = COEFF(j, K) - COEFF(Z + 1, K) * T
     NEXT K
5190 NEXT j
     RETURN
5194 ' End of inversion routine

'-------------------------------------------------------------------
'--                                                               --
'-- INVERSE DISTANCE WEIGHTING ROUTINE                            --
'--                                                               --
'-------------------------------------------------------------------

5400 FOR i = 1 TO xm: FOR j = 1 TO ym
       HD = XMIN + (2 * i - 1) / 2 * dx: VD = YMIN + (2 * j - 1) / 2 * dy
       SUM1 = 0: SUM2 = 0: sume = 0
     IF WW = 2 THEN
      FOR MW = 1 TO NG: WGT = 0:
       FOR M = 1 TO NG: DB = (RGX(M) - HD) ^ 2 + (RGY(M) - VD) ^ 2
       DB = DB ^ .5: WGT = WGT + DB * COEFF(MW, M + N): NEXT M
       WGT = WGT + COEFF(MW, NG + 1 + N): SUM1 = SUM1 + RGZ(MW) * WGT
      NEXT MW
      SUM2 = 1
      GOTO 5460
     END IF
       FOR M = 1 TO NG: DB = (RGX(M) - HD) ^ 2 + (RGY(M) - VD) ^ 2
       DB = DB ^ .5: SUM1 = SUM1 + RGZ(M) / DB ^ idwp
       SUM2 = SUM2 + 1 / DB ^ idwp
       NEXT M
5460   vtx(1) = XMIN + (i - 1) * dx: vty(1) = YMIN + (j - 1) * dy
       vtx(2) = XMIN + i * dx: vty(2) = YMIN + j * dy
       FOR K = 1 TO 2: FOR L = 1 TO 2
         IF POINT(vtx(L), vty(K)) <> 2 THEN sume = sume + 1
       NEXT L, K
       f(i, j) = SUM1 / SUM2
       IF idp = 1 AND sume > 0 THEN PRINT #2, vtx(1) + dx / 2, vty(1) + dy / 2, f(i, j)
       IF f(i, j) < zmig THEN zmig = f(i, j)
       IF f(i, j) > zMAG THEN zMAG = f(i, j)
       RZ = f(i, j) * sume / 4
       SUM3 = SUM3 + RZ: SUM4 = SUM4 + sume / 4
     NEXT j, i
     DR = (XMAg - XMIg) ^ 2 + (YMAg - YMIg) ^ 2: DR = DR ^ .5 / 200
     FOR i = 1 TO NG:
     CIRCLE (RGX(i), RGY(i)), DR, 1, , , 1: PAINT (RGX(i), RGY(i)), 1, 1
     NEXT i
      FOR i = 2 TO NP: LINE (x(i - 1), Y(i - 1))-(x(i), Y(i)), 1: NEXT i
     GOSUB 5700
     WINDOW ((XMIg - dx / 2), (YMIg - dy / 2))-((XMAg + dx / 2), (YMAg + dy / 2))
     FOR i = 2 TO NP: LINE (x(i - 1), Y(i - 1))-(x(i), Y(i)), 1: NEXT i
     PAINT ((XMIN - dx / 4), (YMIN - dy / 4)), 0, 1
     LINE ((XMIg - dx / 2), (YMIg - dy / 2))-((XMAg + dx / 2), (YMAg + dy / 2)), 1, B
     LOCATE 22, 1: PRINT "    Mean areal rainfall = ";
     PRINT USING "####.##"; SUM3 / SUM4; : PRINT " mm"
     LOCATE 2, 1: PRINT "    Press the SPACE BAR to continue"
```

```
5605 IF INKEY$ <> " " THEN 5605
     WIDTH 80: SCREEN 0: CLS
     PRINT "Maximum contour          :"; zMAG - (c(2) - c(1)): COLOR 4
     PRINT "Median  contour          :"; c(mc): COLOR 11
     PRINT "Minimum contour          :"; zmig + (c(2) - c(1)): COLOR 14
     PRINT "Contour interval         :"; c(2) - c(1)
     COLOR 14
     LOCATE 23, 21: PRINT "    Press any key to continue"
5610 IF INKEY$ = "" THEN 5610
     WIDTH 80: SCREEN 0: CLS : IF idp = 1 THEN idp = 0: CLOSE 2
     RETURN 530

'*********************************************************************
'**                                                                 **
'** ROUTINE FOR PLOTTING CONTOURS                                   **
'**                                                                 **
'*********************************************************************

5700 '----  CONTOUR  plotting ----          File: CONDOT (SSC)
     A1 = xm: A2 = ym: c1 = C1A - 1: mc = INT((C1A + 1) / 2)
     FOR L = 0 TO c1: c(L) = zmig + L * (zMAG - zmig) / c1: NEXT L
     C0 = c(1)
     VIEW: VIEW (20, 20)-(300, 160): WINDOW (0, 0)-(A1 + 1, A2 + 1)
      S2 = .05:    S3 = .05
     FOR j = 1 TO A1 - 1: FOR i = 1 TO A2 - 1: '---- Condot routine
      X1 = f(i, j):      X2 = f(i, j + 1)
      X3 = f(i + 1, j): X4 = f(i + 1, j + 1)
      IF X1 < C0 AND X2 < C0 AND X3 < C0 AND X4 < C0 THEN 5764
      FOR K = 0 TO 1 - S2 STEP S2:  '---- I-dim interpolation
          Z1 = X1 - K * (X1 - X3):   Z2 = X2 - K * (X2 - X4)
          IF Z1 < C0 AND Z2 < C0 THEN 5744
          GOSUB 5768:                '---- get contour subscripts
          IF C4 < C3 THEN 5744
          M = Z2 - Z1:    B = Z1 - M * j:  R2 = i + K
          FOR C5 = C3 TO C4: COL = 1: IF C5 = mc THEN COL = 2
          IF C5 > mc THEN COL = 3
            R1 = (c(C5) - B) / M:  IF POINT(R2, R1) <> 2 THEN PSET (R2, R1), COL
          NEXT C5
5744    NEXT K
      FOR K = 0 TO 1 - S3 STEP S3: '---- J-dim interpolation
          Z1 = X1 - K * (X1 - X2): Z2 = X3 - K * (X3 - X4)
          IF Z1 < c(1) AND Z2 < c(1) THEN 5762
          GOSUB 5768:    IF C3 > C4 THEN 5762
          M = Z2 - Z1:    B = Z1 - M * i:  R1 = j + K
          FOR C5 = C3 TO C4: COL = 1: IF C5 = mc THEN COL = 2
          IF C5 > mc THEN COL = 3
            R2 = (c(C5) - B) / M:  IF POINT(R2, R1) <> 2 THEN PSET (R2, R1), COL
          NEXT C5
5762    NEXT K
5764 NEXT i:    NEXT j
     RETURN:                 '---- Subroutine for Z-crossing
5768 IF Z1 > Z2 THEN Y1 = Z2:  Y2 = Z1:     GOTO 5772
     Y1 = Z1: Y2 = Z2
5772 FOR C3 = 1 TO c1
      IF Y1 <= c(C3) THEN 5778
     NEXT C3: C4 = 0:       RETURN
5778 FOR C4 = C3 TO c1
      IF Y2 <= c(C4) THEN 5782
     NEXT C4
5782 C4 = C4 - 1:          RETURN
     RETURN

'*********************************************************************
'**                                                                 **
'** DATA INPUT ROUTINE                                              **
'**                                                                 **
'*********************************************************************

6000 CLS : LOCATE 12, 18
     PRINT "Is the data to come from an existing file (Y/N) "
6003 V$ = INKEY$
     IF UCASE$(V$) = "Y" OR UCASE$(V$) = "N" THEN 6005 ELSE 6003
6005 IF UCASE$(V$) = "N" THEN CLS : GOTO 6010
     PRINT : LOCATE 15, 24
     PRINT "What is the name of the data file "
     LOCATE 17, 40: INPUT "", op$
     OPEN op$ FOR INPUT AS 1
     GOSUB 8000: RETURN
```

```
'------------------------------------------------------------------
'--                                                              --
'-- DIGITIZER CONFIGURATION                                      --
'--                                                              --
'------------------------------------------------------------------

6010 REDIM x$(10)
     x$(1) = "COM1:": x$(2) = "2400": x$(3) = "E": x$(4) = "7"
     x$(5) = "2": x$(6) = ",CS,DS,CD "
     LOCATE 1, 22: PRINT "   DIGITIZER SPECIFICATIONS "
     LOCATE 2, 22: PRINT "   ========================="
     LOCATE 3, 22: PRINT "    1    LOCATION     :COM1"
     LOCATE 4, 22: PRINT "    2    BAUD RATE    :2400"
     LOCATE 5, 22: PRINT "    3    PARITY       :E"
     LOCATE 6, 22: PRINT "    4    DATA BITS    :7"
     LOCATE 7, 22: PRINT "    5    STOP BITS    :2"
6170 LOCATE 9, 14: COLOR 14:
     PRINT "ARE YOU SATISFIED WITH THESE VALUES (Y/N)"
6172 IF INKEY$ <> "" THEN 6172
6174 RP$ = UCASE$(INKEY$)
     IF RP$ = "" THEN 6174 ELSE IF RP$ = "Y" THEN GOTO 6220
     IF RP$ = "N" THEN 6177 ELSE 6174
6177 LOCATE 9, 14: PRINT SPC(60); : LOCATE 9, 22
     PRINT "INPUT NEW VALUE OR PRESS RETURN"
     FOR j = 1 TO 5: LOCATE j + 2, 43: COLOR 23, 0: PRINT x$(j)
     LOCATE j + 2, 43: COLOR 5, 0: INPUT "", G$
     IF G$ <> "" THEN x$(j) = G$
     LOCATE j + 2, 43: PRINT x$(j); "      "
     NEXT j: GOTO 6170
6220 op$ = x$(1) + x$(2) + "," + x$(3) + "," + x$(4) + "," + x$(5) + x$(6)
     PRINT : PRINT
     PRINT "             What distance (in metres) is represented"
     INPUT "             by a length of 1 centimetre on the map "; SCALE
     SCREEN 0: WIDTH 80: CLS : COLOR 14: INPUT "How many grid points"; xm
     ym = xm
     IF FLG = 99 THEN
6221 PRINT
     PRINT "Input    1  for Inverse Distance Weighting Method"
     PRINT "         2  for Punctual Kriging"
     INPUT "        "; WW: WW = INT(WW)
     IF WW < 1 OR WW > 2 THEN 6221
     PRINT : INPUT "How many contour levels are to be plotted"; C1A: PRINT
     C1A = C1A + 2
     IF WW = 1 THEN INPUT "What is the inverse distance weighting power"; idwp: PRINT
     PRINT : PRINT "Do you wish to print the distributed precipitation (Y/N) "
6223 V$ = INKEY$: idp = 0
     IF UCASE$(V$) = "Y" OR UCASE$(V$) = "N" THEN 6225 ELSE 6223
6225 IF UCASE$(V$) = "N" THEN GOTO 6230
     PRINT : LOCATE 15, 24: idp = 1
     PRINT "Please specify a file to send the output data to"
     LOCATE 17, 40: INPUT "", dp$
     OPEN dp$ FOR OUTPUT AS 2
6230 END IF
     REM $DYNAMIC
     DIM f(xm, ym), RGX(20), RGY(20), RGZ(20), RZ(xm, ym), aa(xm, ym)
     DIM x(2000), Y(2000), vtx(2), vty(2), c(20), COEFF(20, 20)
     OPEN op$ FOR RANDOM AS 1
     PRINT #1, II$
     PRINT

'------------------------------------------------------------------
'--                                                              --
'-- INPUTTING DATA WITH DIGITIZER                                --
'--                                                              --
'------------------------------------------------------------------

     INPUT "How many point gauges are located on or near the watershed"; NG
     SCREEN 1: CLS : LINE (20, 20)-(300, 160), 2, B: VIEW (20, 20)-(300, 160)
     WINDOW (0, 0)-(30, 30)' ****** value digitizer dependent *******
     DIM SM(NG), PTB(NG)
     XMAg = 0: YMAg = 0: YMIg = 99999999#: XMIg = 99999999#: zMAG = 0: zmig = 9999999
     FOR M = 1 TO NG: LOCATE 1, 1: PRINT "   INPUT THE LOCATION OF GAUGE"; : PRINT M
     INPUT #1, A$
     V = VAL(RIGHT$(A$, 5)) / 400: H = VAL(MID$(A$, 3, 5)) / 400
     CIRCLE (H, V), .2, 1: LOCATE 1, 1: PRINT "                                          "
     LOCATE 22, 4: INPUT "What is the gauge catch"; RGZ(M)
     LOCATE 22, 4: PRINT "                                   "
       RGX(M) = H * SCALE: RGY(M) = V * SCALE
```

```
     IF RGX(M) < XMIg THEN XMIg = RGX(M)
     IF RGY(M) < YMIg THEN YMIg = RGY(M)
     IF RGX(M) > XMAg THEN XMAg = RGX(M)
     IF RGY(M) > YMAg THEN YMAg = RGY(M)
     IF RGZ(M) < zmig THEN zmig = RGZ(M)
     IF RGZ(M) > zMAG THEN zMAG = RGZ(M)
     NEXT M
     LOCATE 1, 1:  PRINT "    Digitize the watershed boundaries"
             PRINT "       starting with any point "
     LOCATE 22, 1: PRINT "    Press the SPACE BAR just before"
             PRINT "       entering the final point"
     WINDOW (0, 0)-(30, 30)' ****** value digitizer dependent *******
     N = 1: XMAx = 0: YMAx = 0: YMIN = 99999999#: XMIN = 99999999#
     PRINT #1, ":TM15"
7294 INPUT #1, A$
     IF LEFT$(A$, 1) = "U" THEN 7294 ELSE 7310
7300 INPUT #1, A$
7310 V = VAL(RIGHT$(A$, 5)) / 400: H = VAL(MID$(A$, 3, 5)) / 400
     IF H < XMIN THEN XMIN = H
     IF V < YMIN THEN YMIN = V
     IF H > XMAx THEN XMAx = H
     IF V > YMAx THEN YMAx = V
     IF N = 1 THEN 7370 ELSE 7380
7370 A = H: B = V: X1 = H: Y1 = V: N = 11: i = 1
7380 x(i) = H * SCALE: Y(i) = V * SCALE: i = i + 1
     LINE (A, B)-(H, V), 3: A = H: B = V
     V$ = INKEY$: IF V$ = " " THEN 7410 ELSE 7300
7410 H = X1: V = Y1: x(i) = H * SCALE: Y(i) = V * SCALE
     LINE (A, B)-(H, V), 3: NP = i
     XMIN = XMIN * SCALE
     YMIN = YMIN * SCALE
     XMAx = XMAx * SCALE
     YMAx = YMAx * SCALE
     IF XMIN < XMIg THEN XMIg = XMIN
     IF XMAx > XMAg THEN XMAg = XMAx
     IF YMIN < YMIg THEN YMIg = YMIN
     IF YMAx > YMAg THEN YMAg = YMAx
     PRINT #1, ":PT"
     PRINT #1, SS$: CLOSE 1
     SCREEN 0: SCREEN 1: VIEW: LINE (20, 20)-(300, 160), 1, B
     VIEW (20, 20)-(300, 160)
     dx = (XMAx - XMIN) / xm: dy = (YMAx - YMIN) / ym
     WINDOW ((XMIg - dx / 2), (YMIg - dy / 2))-((XMAg + dx / 2), (YMAg + dy / 2))
     FOR i = 2 TO NP: LINE (x(i - 1), Y(i - 1))-(x(i), Y(i)), 3: NEXT i
     PAINT ((XMIN - dx / 4), (YMIN - dy / 4)), 2, 3: SUM3 = 0: SUM4 = 0
     LINE ((XMIg - dx / 2), (YMIg - dy / 2))-((XMAg + dx / 2), (YMAg + dy / 2)), 2, B
     DR = (XMAg - XMIg) ^ 2 + (YMAg - YMIg) ^ 2: DR = DR ^ .5 / 200
     LOCATE 2, 1: PRINT "        Please wait for a moment"
     RETURN

'-----------------------------------------------------------------
'--                                                             --
'-- INPUTTING DATA FROM ASCII FILE                              --
'--                                                             --
'-----------------------------------------------------------------

8000 SCREEN 0: WIDTH 80: CLS : COLOR 14: INPUT "How many grid points"; xm
     ym = xm
     IF FLG = 99 THEN
8001 PRINT
     PRINT "Input    1  for Inverse Distance Weighting Method"
     PRINT "         2  for Punctual Kriging"
     INPUT "        "; WW: WW = INT(WW)
     IF WW < 1 OR WW > 2 THEN 8001
     PRINT : INPUT "How many contour levels are to be plotted"; C1A: PRINT
     C1A = C1A + 2
     IF WW = 1 THEN INPUT "What is the inverse distance weighting power"; idwp: PRINT
     PRINT
     PRINT "Do you wish to print the distributed precipitation (Y/N) "
8103 V$ = INKEY$: idp = 0
     IF UCASE$(V$) = "Y" OR UCASE$(V$) = "N" THEN 8105 ELSE 8103
8105 IF UCASE$(V$) = "N" THEN GOTO 8110
     PRINT : LOCATE 15, 24: idp = 1
     PRINT "Please specify a file to send the output data to"
     LOCATE 17, 40: INPUT "", dp$
     OPEN dp$ FOR OUTPUT AS 2
8110 END IF: CLS
     REM $DYNAMIC
```

```
DIM f(xm, ym), RGX(20), RGY(20), RGZ(20), RZ(xm, ym), aa(xm, ym)
DIM x(2000), Y(2000), vtx(2), vty(2), c(20)
INPUT #1, NG, NP, DXMAX
SCREEN 1: CLS : LINE (20, 20)-(300, 160), 2, B: VIEW (20, 20)-(300, 160)
WINDOW (0, 0)-(DXMAX, DXMAX)
DIM SM(NG), PTB(NG)
XMAg = 0: YMAg = 0: YMIg = 99999999
XMIg = 99999999#: zMAG = 0: zmig = 9999999
FOR M = 1 TO NG
INPUT #1, RGX(M), RGY(M), RGZ(M)
IF RGX(M) < XMIg THEN XMIg = RGX(M)
IF RGY(M) < YMIg THEN YMIg = RGY(M)
IF RGX(M) > XMAg THEN XMAg = RGX(M)
IF RGY(M) > YMAg THEN YMAg = RGY(M)
IF RGZ(M) < zmig THEN zmig = RGZ(M)
IF RGZ(M) > zMAG THEN zMAG = RGZ(M)
NEXT M
FOR i = 1 TO NP
INPUT #1, H, V
IF H < XMIN THEN XMIN = H
IF V < YMIN THEN YMIN = V
IF H > XMAx THEN XMAx = H
IF V > YMAx THEN YMAx = V
x(i) = H: Y(i) = V
NEXT i
x(NP + 1) = x(1): Y(NP + 1) = Y(1)
NP = NP + 1
IF XMIN < XMIg THEN XMIg = XMIN
IF XMAx > XMAg THEN XMAg = XMAx
IF YMIN < YMIg THEN YMIg = YMIN
IF YMAx > YMAg THEN YMAg = YMAx
SCREEN 0: SCREEN 1: VIEW: LINE (20, 20)-(300, 160), 1, B
VIEW (20, 20)-(300, 160)
dx = (XMAx - XMIN) / xm: dy = (YMAx - YMIN) / ym
WINDOW ((XMIg - dx / 2), (YMIg - dy / 2))-((XMAg + dx / 2), (YMAg + dy / 2))
FOR i = 2 TO NP: LINE (x(i - 1), Y(i - 1))-(x(i), Y(i)), 3: NEXT i
PAINT ((XMIN - dx / 4), (YMIN - dy / 4)), 2, 3: SUM3 = 0: SUM4 = 0
LINE ((XMIg - dx / 2), (YMIg - dy / 2))-((XMAg + dx / 2), (YMAg + dy / 2)), 2, B
DR = (XMAg - XMIg) ^ 2 + (YMAg - YMIg) ^ 2: DR = DR ^ .5 / 200
LOCATE 2, 1: PRINT "       Please wait for a moment"
RETURN
```