DANIEL F. MERRIAM, Editor

KWIKR8, A FORTRAN IV PROGRAM FOR MULTIPLE REGRESSION AND GEOLOGIC TREND ANALYSIS

By

J. E. ESLER
Control Data Corporation

P. F. SMITH
Atlantic-Richfield Company

and

J. C. DAVIS
Kansas Geological Survey

in cooperation with the
American Association of Petroleum Geologists
Tulsa, Oklahoma

COMPUTER CONTRIBUTION 28
State Geological Survey
The University of Kansas, Lawrence
1968

Editor's Remarks

Nothing in the computer world is static; rapid change is expected, and indeed inevitable. The standard trend-surface program we have been distributing since 1966 (COMPUTER CONTRIBUTION 3) is now being replaced by this program. The new program (KWIKR8) is quite sophisticated when compared with the first trend-surface program made available in 1963 by J.W. Harbaugh (Special Distribution Publication 3). The program represents improvements made possible by increases in machine size, speed, and language elegance during the past few years. Since 1963, we have used several computer languages, including BALGOL, ALGOL, and FORTRAN II. Today the language is FORTRAN IV with PL/1 on the horizon; undoubtedly, both will eventually be replaced by better, more efficient and usable languages. Such is the way of the computer.

Is change necessarily progress? Not always, but we believe that attitudes and abilities are improving in geology and that before too long, earth scientists will be using quantitative techniques routinely and to their great advantage. Of all techniques now available, trend analysis probably is the most widely accepted and universally used. For good reason too, as it has been proved successful.

Part of the success of trend analysis has been that "...geologists are intuitively attracted to these methods which produce and analyze map features...." The authors of this publication have tried to make the program as versatile as possible. They present several convincing problems and outline their solution by trend analysis. Geologists should have little difficulty in finding other problems where the method is equally applicable.

Because of the many computer programs now available through the Geological Survey, the following table should be of help to those seeking programs available in different computer languages. Before ordering decks, however, users should make sure the program they desire is available as many versions are now discontinued.

Language	Geological Survey Publication
BALGOL	SDP3, SDP9, SDP24, B171, CC1
ALGOL	SDP23, CC2, CC8
FORTRAN II	SDP4, SDP12, SDP13, SDP14, SDP26, SDP28, CC4, CC10, CC15, CC19, CC20
FORTRAN IV	SDP12, SDP23, B171, CC2, CC3, CC5, CC6, CC9, CC11, CC13, CC14, CC15, CC16, CC17, CC20, CC21, CC23, CC24, CC25, CC26, CC27, CC28

SDP = Special Distribution Publication; CC = Computer Contribution; B = Bulletin.

Because of the general use of FORTRAN IV in the United States, most programs are available in that language although a few are in machine dependent dialects or contain assembler language subroutines. Where possible, however, the programs are designed to be machine independent.

*Active Member, °Associate Member, +Junior Member, American Association of Petroleum Geologists.

KWIKR8 A FORTRAN IV PROGRAM FOR MULTIPLE REGRESSION AND GEOLOGIC TREND ANALYSIS

by

J.E. Esler, P.F. Smith, and J.C. Davis

INTRODUCTION

Geologic trend-surface analysis, as defined by Grant (1957) and developed by Krumbein (1956, 1959) and his associates (Peikert, 1963; Whitten, 1963) is a special application of multiple regression. Trend analysis has been widely and successfully applied in geologic research and exploration, and is perhaps the most extensively used computer technique in the earth sciences. Geologists are intuitively attracted to these methods which produce and analyze map features, and are finding increasing uses for "four-dimensional" variations of the technique (Harbaugh, 1964; Smith and Harbaugh, 1966; Davis, 1967a). Although most trend-surface applications are not amenable to statistical testing, an increasing number of other geologic problems are being analyzed as multivariable regressions.

The program described in this publication is an outgrowth of Geological Survey research in trend analysis, and incorporates features of several previously published programs. The matrix of regression coefficients contains 35 variables that may be combined in a number of different polynomial expansions. For example, a curvilinear regression on one independent variable may be computed up to X^{35}, a trend surface may be computed up to the seventh order, a "four-dimensional" trend may be computed up to the fourth order, or multivariate linear and curvilinear regressions may be calculated using up to 35 independent variables.

A number of options have been written into this program to make it as versatile as possible. Individual coefficients, for example, may be deleted or incorporated as desired on successive runs. Residuals may be listed or printed on maps. Contour maps, of specified scale and contour interval, may be prepared for trend surfaces or "4-D" trends, including "slice-maps" at any specified coordinate through the solid defined by the independent variables.

Information useful for statistical testing also is produced. These include various sums of squares, the correlation coefficient, an F-value with associated degrees of freedom, and partial regression coefficients and standardized partial regression coefficients. If desired, regression of specified order may be performed on the residuals from a previous regression, as an aid in the search for correlation among residuals.

Numerical methods for computing trend surfaces have been described in detail in earlier publications in this series (Harbaugh, 1963; Good, 1964; Sampson and Davis, 1966, 1967; O'Leary, Lippert, and Spitz, 1966) and in recent textbooks on mathematical geology (Krumbein and Graybill, 1965, Ch. 13; Harbaugh and Merriam, 1968, Ch. 5). In general, they consist of expanding the desired linear regression into a matrix of normal equations, which then is solved by inversion, giving coefficients of the regression. A variety of schemes exist for inverting matrices; this program uses simple Gaussian elimination. However, the matrix is pretreated by an averaging process so that exponents of entries are centered around zero. This minimizes rounding effects created by extremely large entries but allows rapid solution of the matrix.

Coefficients of the matrix are printed out as a numbered list where each number corresponds to an entry in the linear equation $Y = C_1 + C_2X_1 + C_3X_2 + \ldots C_{36}X_{35}$. The first coefficient is the value of the regression at the origin. Successive coefficients depend upon the number of independent variables used and the power of the polynomial equation. Table 1 lists variables, powers, and cross-products for coefficients in the various polynomial expansions. Orientation of geographic variables for trend and "4-D" maps is shown on Figure 1.

OPERATING INSTRUCTIONS

An extensive set of control cards are necessary to take advantage of the versatility of KWIKR8. These are described in detail below, and examples of typical problems are given in the following section, accompanied by listings of the appropriate cards. Numerical parameters are placed in columns that are multiples of ten (cols. 10, 20, 30, etc.). Logic variables (T or F) are placed in columns ending in the number five (15, 25, 35, etc.). This will facilitate checking control cards for mispunched values. The only exception to this arrangement is in col. 71-75 of Card 8 ($X_1X_2X_3$ option card) for three independent variables. All variables are floating-point unless specified integer or logic.

(1) Title Card

Columns	
1-5	TITLE
8-80	Any desired alphanumeric information.

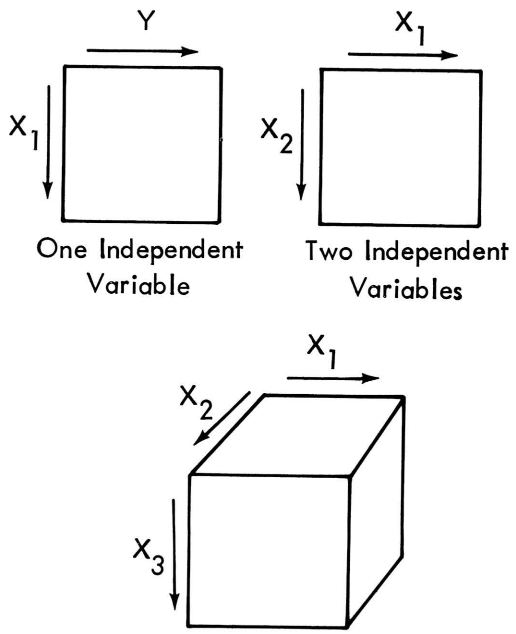

Figure 1.–Orientation of graphic output from KWIKR8, showing orientation of variable axes. Values increase positively in directions shown by arrows.

(2) Control card

Columns	
1–10	Right-justified integer indicating number of independent variables.
11–20	Right-justified integer indicating degree of desired polynomial.
30	SWITCH 1, to delete coefficients. 0 = consider all coefficients 1 = delete some coefficients
35	SWITCH 2, to select data format. T = standard format (see data card) F = read variable format
45	SWITCH 3, last data set indicator. T = last data set F = more data sets to follow
60	SWITCH 4, to list residuals. 0 = omit list of residuals 1 = print list of residuals
65	SWITCH 5, last regression indicator. T = more regressions will be computed on this data F = this is final regression on this data
70	SWITCH 6, to compute regression on residuals. 0 = compute regression on specified data 1 = compute regression on residuals from preceeding regression

(3) Variable Format card (used only if SWITCH 2 is F).
Column 1–80 Variable format, arranged to read in the sequence (independent variable fields, dependent variable field, L1 logic field).

(4) Axis Inversion card

Column 1	SWITCH 7, to invert X_1 axis. T = invert X_1 axis blank = do not invert X_1 axis
Column 2	SWITCH 8, to invert X_2 axis. T = invert X_2 axis blank = do not invert X_2 axis
Column 3	SWITCH 9, to invert X_3 axis. T = invert X_3 axis blank = do not invert X_3 axis

(Note: This card allows the origin of trend maps to be moved, if independent variables are not measured according to the conventions shown in Figure 1.)

(5) Data cards
Up to 425 observations may be entered into the program, under the format specified on Card 3 or under a standard format [(number of variables) F10.4, 9X, L1]. The dependent variable occupies the last data field. The last data card must contain a T in the logic field to indicate end of the data set.

(6) Coefficient elimination card (Used only if SWITCH 1 is 1).
Punch a 1 in each column corresponding to the subscript of the coefficients to be eliminated from the regression.

A sequence of cards are now entered to control the mapping routine. Maps may be produced only for data having one, two, or three independent variables. If no maps are desired, card (8) is blank.

FOR ONE INDEPENDENT VARIABLE (GRAPH)

(7) MAPDTA card
Columns 1–6 MAPDTA

(8) X_1 Option card

Columns 1–10	Size of desired increment step of X_1.
Column 20	SWITCH 10, to control plotting limits. 0 = plot from minimum to maximum value of X_1, minimum to maximum value of Y. 1 = plot between specified limits of X_1 and Y.

(9) X_1 Limits card (Used only if SWITCH 10 is 1).

Columns	
1–10	Maximum limit of plot along X_1
11–20	Minimum limit of plot along X_1
21–30	Maximum limit of plot along Y
31–40	Minimum limit of plot along Y

If SWITCH 5 is T, a new regression will be computed on the same data. New cards (1), (2), (6), (7), (8), and (9) should be entered. New regressions will be computed until SWITCH 5 is F.

FOR TWO INDEPENDENT VARIABLES: (TREND SURFACE):

(7) MAPDTA card

Columns	
1-6	MAPDTA

(8) X_1 X_2 Option card

Columns	
1-10	Value of reference contour (approximately equal to expected mean).
11-20	Size of contour interval.
21-30	Right-justified integer specifying number of maps to be printed.
40	SWITCH 10, to control plotting limits. 0 = plot from minimum to maximum value of X_1, minimum to maximum value of X_2. 1 = plot between specified limits of X_1 and X_2.
41-50	X_1 dimension of trend map, in tenths of inches, up to 131. This prints across lister sheet.
51-60	X_2 dimension of trend map, in tenths of inches. This prints down lister sheet.
61-70	X_1 dimension of residual plot, in tenths of inches.
71-80	X_2 dimension of residual plot, in tenths of inches.

(9) X_1 X_2 Limits card (Use columns 1-40 only if SWITCH 10 is 1).

Columns	
1-10	maximum limit of plot along X_1
11-20	minimum limit of plot along X_1
21-30	maximum limit of plot along X_2
31-40	minimum limit of plot along X_2
45	SWITCH 11, contour map option T = print contour trend map blank = omit contour trend map
55	SWITCH 12, original data option T = plot original data values blank = omit data plot
65	SWITCH 13, residuals option T = plot residual values blank = omit residual plot

A Card (9) should be present for each map specified in Columns 21-30 of Card (8). Switches 11, 12, and 13 can all be on for a given card.

If SWITCH 5 is T, a new regression will be computed on the same data. New cards (1), (2), (6), (7), (8), and (9) should be entered. New regressions will be computed until SWITCH 5 is F.

FOR THREE INDEPENDENT VARIABLES ("4-D" OR RESPONSE SURFACE):

(7) MAPDTA card

Columns	
1-6	MAPDTA

(8) X_1 X_2 X_3 Option card

Columns	
1-10	Value of reference contour (approximately equal to expected mean).
11-20	size of contour interval.
21-30	Right-justified integer specifying total number of maps to be printed (one box = one map).
40	SWITCH 10, to control plotting limits. 0 = plot from minimum to maximum values of X_1, X_2, and X_3. 1 = plot between specified limits of X_1, X_2, and X_3.
41-50	X_1 dimension of "4-D" map, in tenths of inches up to 131. This prints across lister sheet.
51-60	X_2 dimension of "4-D" map, in tenths of inches up to 131. This prints across lister sheet.
61-70	X_3 dimension of "4-D" map, in tenths of inches. This prints down lister sheet.
71-75	Right-justified integer specifying number of desired residual plots.
76-80	Right-justified integer specifying number of desired slice-maps.

(9) X_1 X_2 X_3 Limits card (Used only if SWITCH 10 is 1).

Columns	
1-10	maximum limit of plot along X_1
11-20	minimum limit of plot along X_1
21-30	maximum limit of plot along X_2
31-40	minimum limit of plot along X_2
41-50	maximum limit of plot along X_3
51-60	minimum limit of plot along X_3

(10) Slice-map card (Used only if slice-maps are desired).

Column	
10	SWITCH 14, to select slice-map 1 = slice along X_2-X_3 plane, holding X_1 constant 2 = slice along X_1-X_3 plane, holding X_2 constant 3 = slice along X_1-X_2 plane, holding X_3 constant

(11) Interval card (Used only if slice-maps are desired)

Column	
1-10	value of constant at which slice-map is to be made.

Repeat cards (10) and (11) for each slice map desired, up to the number specified in col. 76-80 of card (8).

(12) Residuals plot card (Used only if residual plots are desired)

Columns	
1-10	Right-justified integer specifying number of slices into which residual plot is to be divided.
11-20	X_1 dimension of residual plots, in tenths of inches up to 131. This prints across lister sheet.
21-30	X_2 dimension of residual plots, in tenths of inches. This prints down lister sheet.
35	SWITCH 12, residuals option T = plot residual values blank = omit residual values
45	SWITCH 13, original data option T = plot original data values blank = omit data plot

(13) X_1 X_2 X_3 Limits card (Used only if SWITCH 10 is 1).

Columns	
1-10	maximum limit of plot along X_1
11-20	minimum limit of plot along X_1
21-30	maximum limit of plot along X_2
31-40	minimum limit of plot along X_2
41-50	maximum limit of plot along X_3
51-60	minimum limit of plot along X_3

Repeat cards (12) and (13) for each residual or original plot desired, up to the number specified in col. 71-75 of card (8). A plot of residuals and original data may both be requested on the same card (12).

If SWITCH 5 is T, a new regression will be computed on the same data. New cards (1), (2), and (6) through (13) should be entered. New regressions will be computed until SWITCH 5 is F.

ERROR MESSAGES

A small number of error messages have been incorporated in this program. Some of these messages were inserted during program development and should rarely appear during normal processing.

Message: BAD DATA, CHECK PARAMETERS

Interpretation: The number of variables, degree of polynomial, or combination of the two is invalid. The program will skip to the next TITLE card (1)and resume operation. If the message is received for the last data set or the set is to be used in more than one regression, the job will terminate.

Message: TOO MUCH DATA

Interpretation: The maximum number of data points (425) has been exceeded. The program will skip to the next MAPDTA card (7) and proceed. If the message is received for the last data set or the set is to be used for more than one regression, the job will terminate.

Message: NO SOLUTION IN SIMEQ

Interpretation: The matrix of normal equations is singular. Data may be incorrect or poorly distributed, resulting in a very unstable solution to the matrix. Results of a regression following this message are incorrect.

Message: OVERPRINT VALUES HAVE EXCEEDED ARRAY LENGTH PLOT HALTED

Interpretation: Specified dimensions of a residual plot are too small, resulting in an excessive number of overprints. The plot is terminated at this point, and the next output or computation request is processed. Increase dimensions of plot or split plot into several larger sections.

Message: SNOOPY FINALLY GOT THE RED BARON

Interpretation: Three attempts have been made to translate a residuals plotting command. Failure indicates a machine malfunction or error in program deck, probably in the residuals plotting routine. This message should never appear from a properly compiled deck.

SUGGESTIONS FOR EFFICIENT USE

If more than one degree is to be computed on a single set of data, the regressions should be computed in decreasing order (i.e., 4th, 3rd, 2nd order, etc.). The most complex matrix will be established initially and then truncated for lower order runs. Computing in increasing order requires that the complete matrix be established for each order.

The time required to produce line printer contour maps increases rapidly with increasing size. Therefore, the physical dimensions of maps should be kept as small as practical.

SUGGESTIONS FOR MODIFICATION

The program may easily be chained by dividing into two links. This will free about 2K usable core with the present deck. The main program should consist of the mainline and subroutine POLYD. The first link should contain STATPK, FIDDLE, EMSLVR, MLTDEG, and LINFIT. The second link will consist of the mapping routines SONG, SHIFTOF, PLTRSD, PLOTER,and COMCON. Statements 500 and 9600 in the mainline can load the first link, and statement 80 can load the second link.

The maximum number of data points to be used may be changed by adjusting the dimensions of all variables now dimensioned 425. Also, the parameter of the DO loop starting on card KWIKR8 50 and the integer on card KWIKR8 64 should be similarly changed.

Table 1.—Numbering of coefficients in polynomial expansions of multivariate regressions, using KWIKR8. Circled numbers denote end of terms of that order.

POLYNOMIALS

Thirty-five variables:

$$Y = C(1) + C(2)\,X_1 + C(3)\,X_2 + C(4)\,X_3 + C(5)\,X_4 \ldots + C(36)\,X_{35} \textcircled{1}$$

Seven variables:

$$Y = C(1) + C(2)\,X_1 + C(3)\,X_2 + C(4)\,X_3 + C(5)\,X_4 + C(6)\,X_5 + C(7)\,X_6 + C(8)\,X_7 \textcircled{1} + C(9)\,X_1^2$$
$$+ C(10)\,X_1X_2 + C(11)\,X_2^2 + C(12)\,X_1X_3 + C(13)\,X_2X_3 + C(14)\,X_3^2 + C(15)\,X_1X_4 + C(16)\,X_2X_4$$
$$+ C(17)\,X_3X_4 + C(18)\,X_4^2 + C(19)\,X_1X_5 + C(20)\,X_2X_5 + C(21)\,X_3X_5 + C(22)\,X_4X_5 + C(23)\,X_5^2$$
$$+ C(24)\,X_1X_6 + C(25)\,X_2X_6 + C(26)\,X_3X_6 + C(27)\,X_4X_6 + C(28)\,X_5X_6 + C(29)\,X_6^2 + C(30)\,X_1X_7$$
$$+ C(31)\,X_2X_7 + C(32)\,X_3X_7 + C(33)\,X_4X_7 + C(34)\,X_5X_7 + C(35)\,X_6X_7 + C(36)\,X_7^2 \textcircled{2}$$

Six variables:

$$Y = C(l) + C(2)\,X_1 + C(3)\,X_2 + C(4)\,X_3 + C(5)\,X_4 + C(6)\,X_5 + C(7)\,X_6 \textcircled{1} + C(8)\,X_1^2 + C(9)\,X_1X_2$$
$$+ C(10)\,X_2^2 + C(11)\,X_1X_3 + C(12)\,X_2X_3 + C(13)\,X_3^2 + C(14)\,X_1X_4 + C(15)\,X_2X_4 + C(16)\,X_3X_4$$
$$+ C(17)\,X_4^2 + C(18)\,X_1X_5 + C(19)\,X_2X_5 + C(20)\,X_3X_5 + C(21)\,X_4X_5 + C(22)\,X_5^2 + C(23)\,X_1X_6$$
$$+ C(24)\,X_2X_6 + C(25)\,X_3X_6 + C(26)\,X_4X_6 + C(27)\,X_5X_6 + C(28)\,X_6^2 \textcircled{2}$$

Five variables:

$$Y = C(1) + C(2)\,X_1 + C(3)\,X_2 + C(4)\,X_3 + C(5)\,X_4 + C(6)\,X_5 \textcircled{1} + C(7)\,X_1^2 + C(8)\,X_1X_2 + C(9)\,X_2^2$$
$$+ C(10)\,X_1X_3 + C(11)\,X_2X_3 + C(12)\,X_3^2 + C(13)\,X_1X_4 + C(14)\,X_2X_4 + C(15)\,X_3X_4 + C(16)\,X_4^2$$
$$+ C(17)\,X_1X_5 + C(18)\,X_2X_5 + C(19)\,X_3X_5 + C(20)\,X_4X_5 + C(21)\,X_5^2 \textcircled{2}$$

Four variables:

$$Y = C(1) + C(2)\,X_1 + C(3)\,X_2 + C(4)\,X_3 + C(5)\,X_4 \textcircled{1} + C(6)\,X_1^2 + C(7)\,X_1X_2 + C(8)\,X_2^2 + C(9)\,X_1X_3$$
$$+ C(10)\,X_2X_3 + C(11)\,X_3^2 + C(12)\,X_1X_4 + C(13)\,X_2X_4 + C(14)\,X_3X_4 + C(15)\,X_4^2 \textcircled{2} + C(16)\,X_1^3$$
$$+ C(17)\,X_1^2 + C(18)\,X_1^2X_3 + C(19)\,X_1^2X_4 + C(20)\,X_1X_2^2 + C(21)\,X_2^3 + C(22)\,X_2^2X_3 + C(23)\,X_2^2X_4$$
$$+ C(24)\,X_1X_3^2 + C(25)\,X_2X_3^2 + C(26)\,X_3^3 + C(27)\,X_3^2X_4 + C(28)\,X_1X_4^2 + C(29)\,X_2X_4^2 + C(30)\,X_3X_4^2$$
$$+ C(31)\,X_4^3 + C(32)\,X_1X_2X_4 + C(33)\,X_1X_3X_4 + C(34)\,X_2X_3X_4 + C(35)\,X_1X_2X_3 \textcircled{3}$$

Three variables:

$$Y = C(1) + C(2)\,X_1 + C(3)\,X_2 + C(4)\,X_3 \textcircled{1} + C(5)\,X_1^2 + C(6)\,X_1X_2 + C(7)\,X_2^2 + C(8)\,X_1X_3 + C(9)\,X_2X_3$$
$$+ C(10)\,X_3^2 \textcircled{2} + C(11)\,X_1^3 + C(12)\,X_1^2X_2 + C(13)\,X_1^2X_3 + C(14)\,X_1X_2^2 + C(15)\,X_2^3 + C(16)\,X_2^2X_3$$

$$+ C(17)\,X_1X_3^2 + C(18)\,X_2X_3^2 + C(19)\,X_3^3 + C(20)\,X_1X_2X_3 \;\textcircled{3}\; + C(21)\,X_1^4 + C(22)\,X_1^3X_2 + C(23)\,X_1^3X_3$$

$$+ C(24)\,X_1X_2^3 + C(25)\,X_2^4 + C(26)\,X_2^3X_3 + C(27)\,X_1X_3^3 + C(28)\,X_2X_3^3 + C(29)\,X_3^4 + C(30)\,X_1^2X_2X_3$$

$$+ C(31)\,X_1^2X_3^2 + C(32)\,X_1X_2^2X_3 + C(33)\,X_1X_2X_3^2 + C(34)\,X_2^2X_3^2 + C(35)\,X_1^2X_2^2 \;\textcircled{4}$$

Two variables:

$$Y = C(1) + C(2)\,X_1 + C(3)\,X_2 \;\textcircled{1}\; + C(4)\,X_1^2 + C(5)\,X_1X_2 + C(6)\,X_2^2 \;\textcircled{2}\; + C(7)\,X_1^3 + C(8)\,X_1^2X_2$$

$$+ C(9)\,X_1X_2^2 + C(10)\,X_2^3 \;\textcircled{3}\; + C(11)\,X_1^4 + C(12)\,X_1^3X_2 + C(13)\,X_1X_2^3 + C(14)\,X_2^4 + C(15)\,X_1^2X_2^2 \;\textcircled{4}$$

$$+ C(16)\,X_1^5 + C(17)\,X_1^4X_2 + C(18)\,X_1^3X_2^2 + C(19)\,X_1^2X_2^3 + C(20)\,X_1X_2^4 + C(21)\,X_2^5 \;\textcircled{5}\; + C(22)\,X_1^6$$

$$+ C(23)\,X_1^5X_2 + C(24)\,X_1^4X_2^2 + C(25)\,X_1^3X_2^3 + C(26)\,X_1^2X_2^4 + C(27)\,X_1X_2^5 + C(28)\,X_2^6 \;\textcircled{6}\; + C(29)\,X_1^7$$

$$+ C(30)\,X_1^6X_2 + C(31)\,X_1^5X_2^2 + C(32)\,X_1^4X_2^3 + C(33)\,X_1^3X_2^4 + C(34)\,X_1^2X_2^5 + C(35)\,X_1X_2^6 + C(36)\,X_2^7 \;\textcircled{7}$$

One variable:

$$Y = C(1) + C(2)\,X_1 + C(3)\,X_1^2 + C(4)\,X_1^3 + \ \dots\ + C(36)\,X_1^{35}$$

ONE INDEPENDENT VARIABLE

This program may be used to produce a polynomial approximation to a functional equation having only one independent variable. Experimental data in this example consist of measurements of the optic axis angles of plagioclases of known composition (Smith, 1956). The angle varies as a complex function of the ratio between two plagioclase end-members, albite and anorthite. Optic axis angles conventionally are measured in positive degrees up to 90°, then in progressively decreasing negative degrees beyond. In order to avoid the hiatus such a convention would produce in the data sequence, they are used here in positive degrees from 70° to 120°.

The data were first fit with a polynomial expansion to the seventh order, giving a correlation of $R = 0.99$. For comparative purposes, a second regression was made using only powers to the fourth order. The correlation of this fit is $R = 0.95$. Although there is little difference in total fit between the two, the seventh-order curve appears to be a better approximator, especially at inflection points. The polynomial approximation seems at least as aesthetic as hand-drawn curves and has the advantage of satisfying the least-squares criterion. The graph produced by the computer of the seventh-order fit is shown in Figure 2.

Cards used in the control sequence are shown in Figure 3. Output from this operation consists of a title, table of statistics, graph of the seventh-order curve, a second title and statistical table, and a graph of the fourth-order curve. Original observations are plotted on the graph as X's, asterisks are used to plot the curve. Dollar signs appear where an X and an asterisk coincide. Note that the control card sequence was set up to produce a graph having grids at specified intervals (2% An, 5° angles).

TWO INDEPENDENT VARIABLES

Geologic trend analysis is an application of multiple regression, using a polynomial expansion of the geographic coordinates of sample points as independent variables. Depth, thickness, or other attributes of the sample may constitute the dependent variable.

Brown (1966) collected data on the thickness of the Pennsylvanian Kanwaka Shale from 566 electric logs of wells drilled in south-central Kansas. To analyze these data, KWIKR8 was modified as described in the text and a series of trend-surface and residual maps were computed. The Kanwaka is part of a sedimentary deltaic complex; results of the investigation will be reported by Merriam and Doria-Medina (in preparation).

Data, consisting of legal location (quarter-quarter section, township, and range) and thickness of Kanwaka, were converted to a cartesian coordinate system using Good's (1964) program. The coordinate system generated by Good's program has an origin southwest of the map area under investigation. Because KWIKR8 assumes an origin in the northwest, it was necessary to invert the second (X_2) axis; this is done by SWITCH 8. As an aid in the search for autocorrelation among residual values, a fifth-order regression was fit to residuals from the third-order surface. In this example, the goodness-of-fit was near zero, indicating that little autocorrelation remains in the

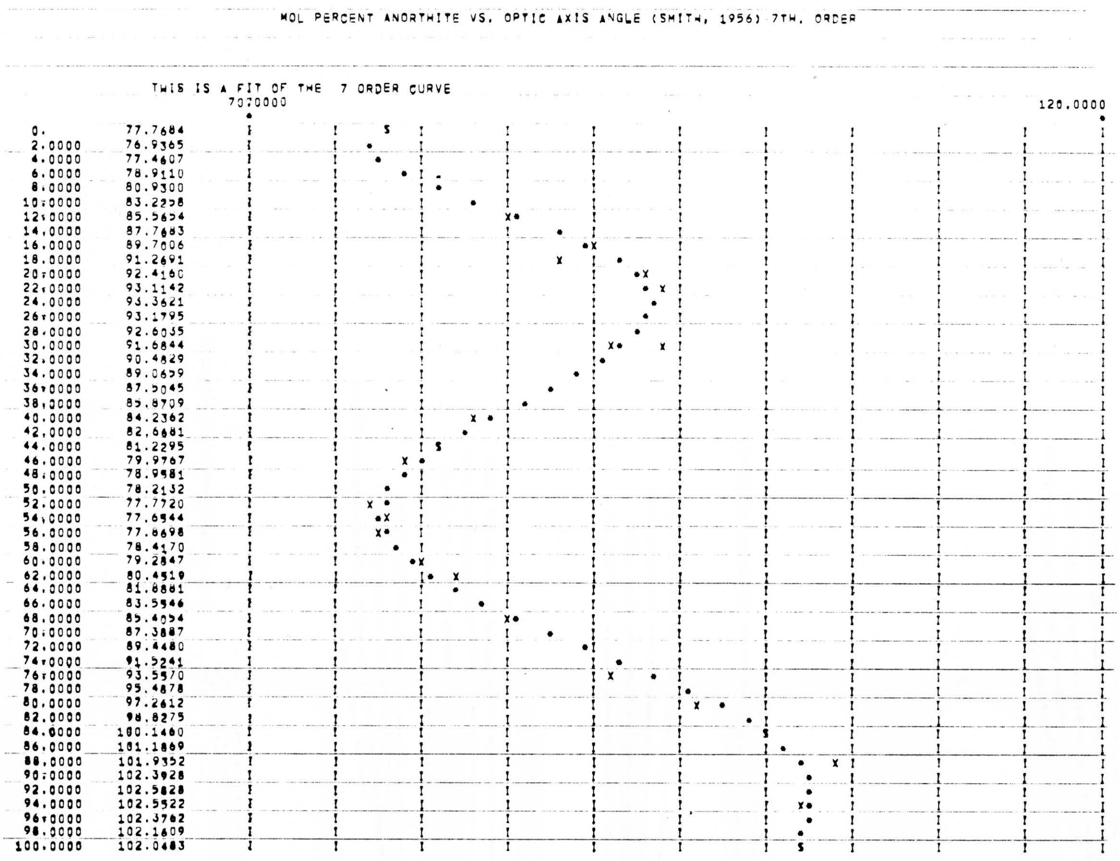

Figure 2.–Graph of mol percent anorthite versus optic axis angle in plagioclases. Output of seventh–order fit.

residuals. The third–order regression is shown in Figure 4. Figure 5 is the fifth–order fit to residuals.

The investigators desired maps produced to a certain scale. This was achieved by specifying an appropriate width and length of plot, and maximum and minimum values of the coordinate system. The plot was then automatically scaled to fit within the size specified. Other options requested included a list of input data, calculated surface, and residuals; a plot of the original data; and a plot of the residuals from the trend surface.

The control cards required to operate the program in this problem are shown in Figure 6, accompanied by an explanation of each parameter. The output sequence consists of:

(1) Title.
(2) Listing of original data, trend surface, and residuals.
(3) Table of statistics and coefficients of trend surface.
(4) Trend surface map.
(5) Plot of original data points.
(6) Plot of residual values.
(7) Table of statistics and coefficients of trend on residuals.
(8) Trend surface map of residuals.

THREE INDEPENDENT VARIABLES

By incorporating depth or stratigraphic location as a third geographic variable, least–squares approximations can be made of a variable embedded in three–dimensional space. The dependent variable commonly is the percentage of a mineral or element at sample points, pH or salinity of brines from various horizons in wells, API gravities or other characteristics of crude oils, indeed, any variable that characterizes samples whose spatial relations are known. Polynomial regression analysis has been applied to data having three geographic variables by Peikert (1963), Harbaugh (1964), Smith and Harbaugh (1966), and Davis (1967a, 1967b). Data in this example are taken from the last reference. Several hundred organic carbon analyses of the Cretaceous Mowry Shale in Wyoming were processed by this technique in order to summarize carbon distribution through the shale lithosome. Using three geographic coordinates as independent variables, carbon concentration was portrayed in three dimensions as a series of "slice–maps". These can be assembled into an egg crate–like set of fence diagrams, or summarized as an isopleth block diagram. Results of the Mowry Shale study are contained in a forthcoming article (Davis, in preparation).

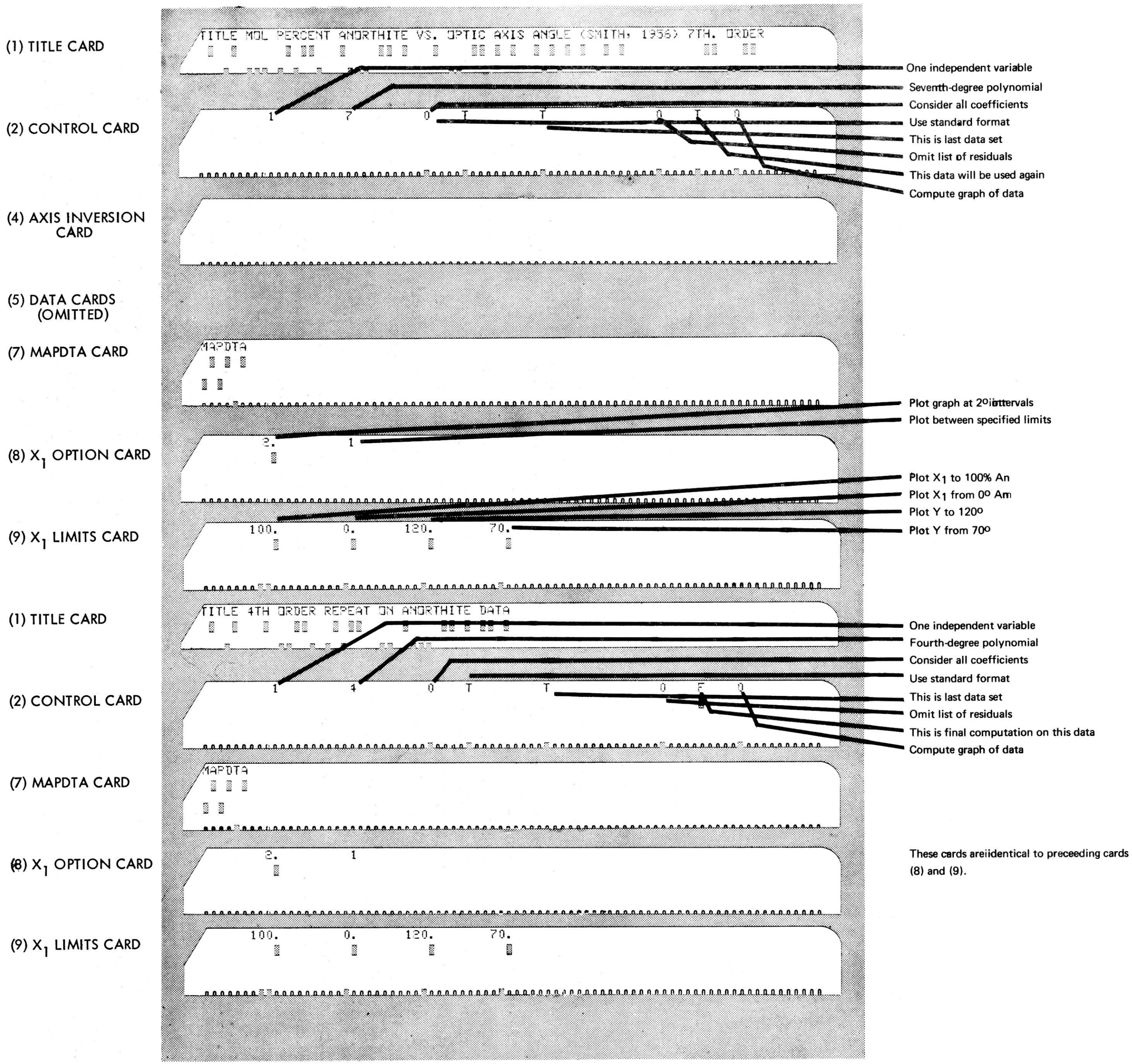

Figure 3.—Control card sequence used to generate output described in text for least-squares polynomial fit of plagioclase optic axis angles.

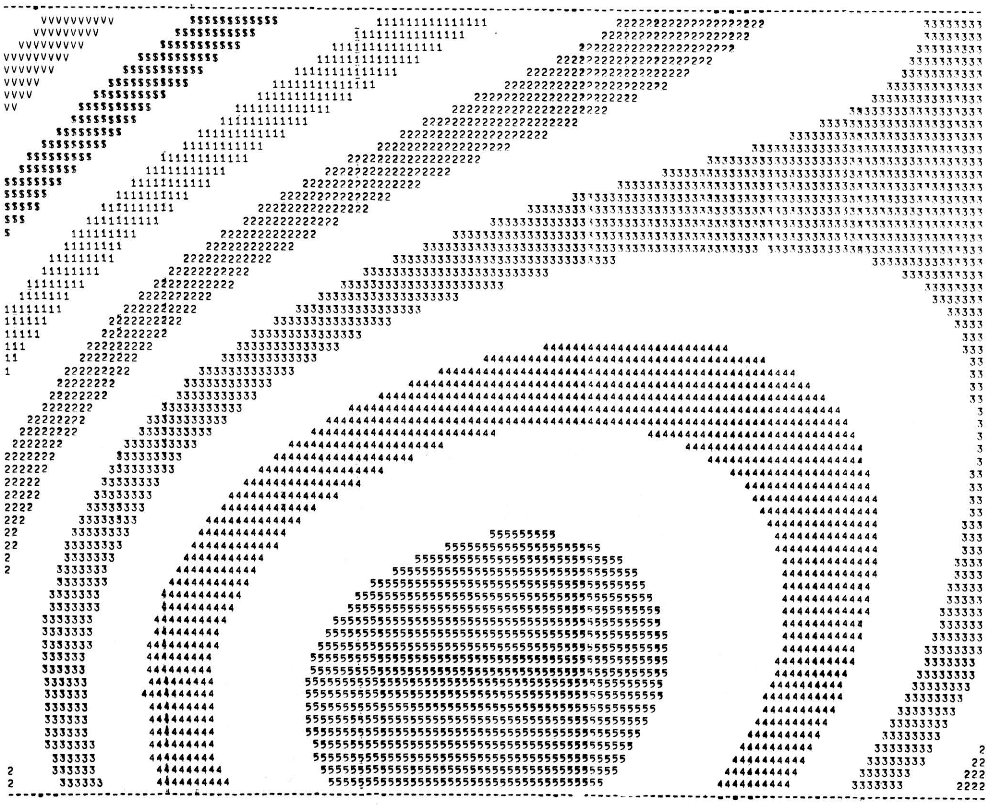

Figure 4.-Third-order trend surface of thickness of Kanwaka Shale in southern Kansas.

Figure 5.-Fifth-order fit of residuals from Figure 4.

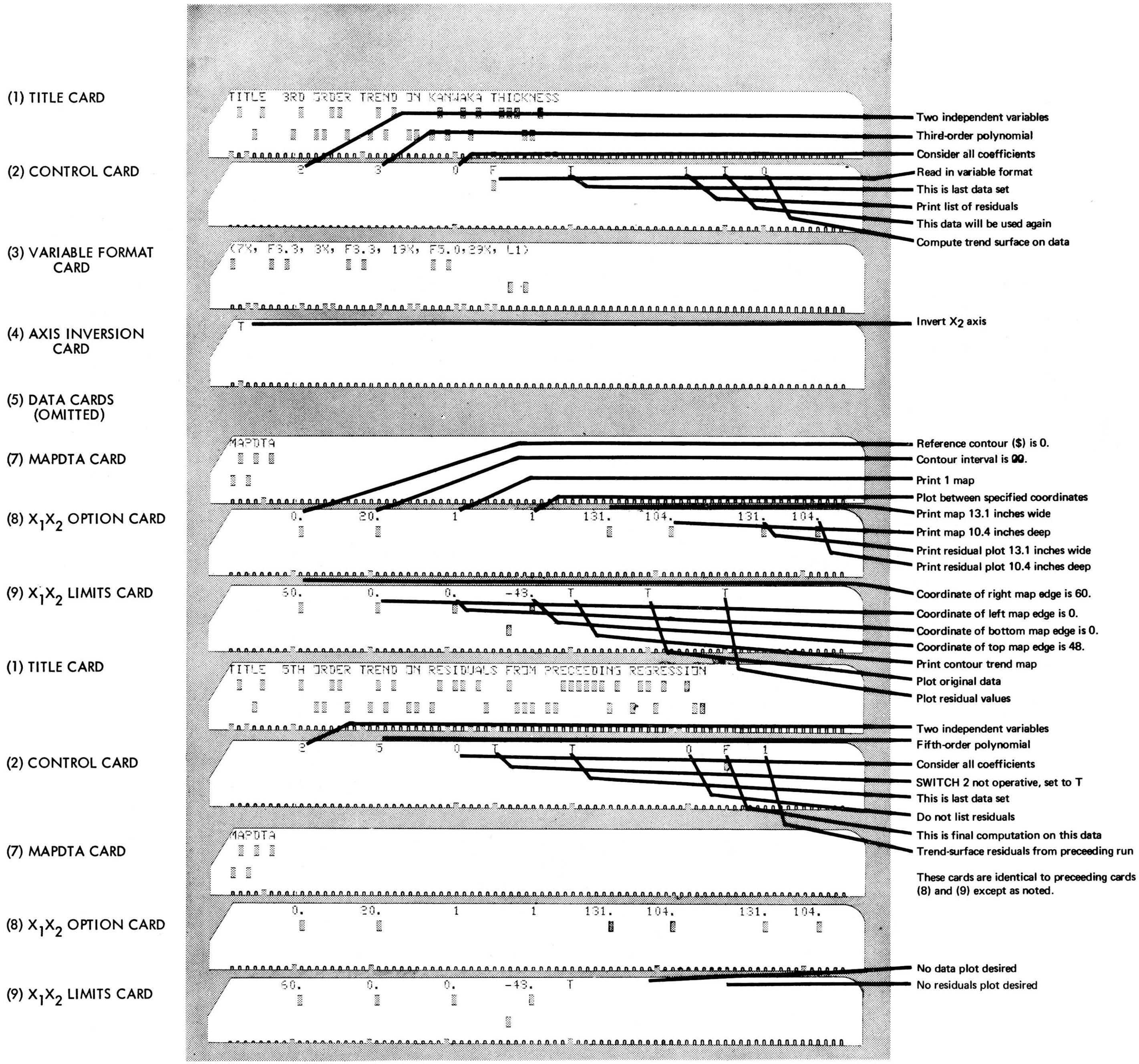

Figure 6.–Control card sequence used to generate output described in Example 2, to fit trend surfaces to Kanwaka Shale thickness.

The purpose of response-surface analysis of these data was to obtain the simplest possible set of isopleth envelopes that would explain most of the variation in carbon content. Therefore, a series of repeated runs on the data were necessary. All output except statistical tables was suppressed on initial runs to save computer time. Standard partial regression coefficients were examined and a backward regression scheme set up to successively eliminate ineffective coefficients from the regression equation (Draper and Smith, 1967). After an optimum equation was obtained, the regression was recalculated and a series of slice-maps computed. From these slice-maps, the isopleth block diagram shown in Figure 7 was then constructed. In this example, only the control sequence from the final run is shown (Fig. 8). Special features of this run are (1) certain coefficients were eliminated from the equation; (2) maps were scaled to match a Wyoming base map; and (3) slice-maps at specified levels were requested. The following sequence of output is created by the control card stack:

(1) Title
(2) List of coefficients eliminated
(3) Table of statistics and coefficients of 4th degree regression
(4) Slice-maps of the outside edges of a box enclosing the sample space (2 maps and 4 cross-sections)
(5) Slice-maps requested across sample space.

It should be noted that the six slice-maps initially produced are correctly oriented so they may be assembled into a box with all printed sides facing out. Also, the box (i.e., all six sides) constitutes a single "map" for purposes of specifying total number of maps on the $X_1X_2X_3$ Option Card (8). Extra slice-maps are produced in pairs, one the mirror image of the other, so they may be assembled into an "egg crate" with printing on both sides. Each pair constitutes a single map.

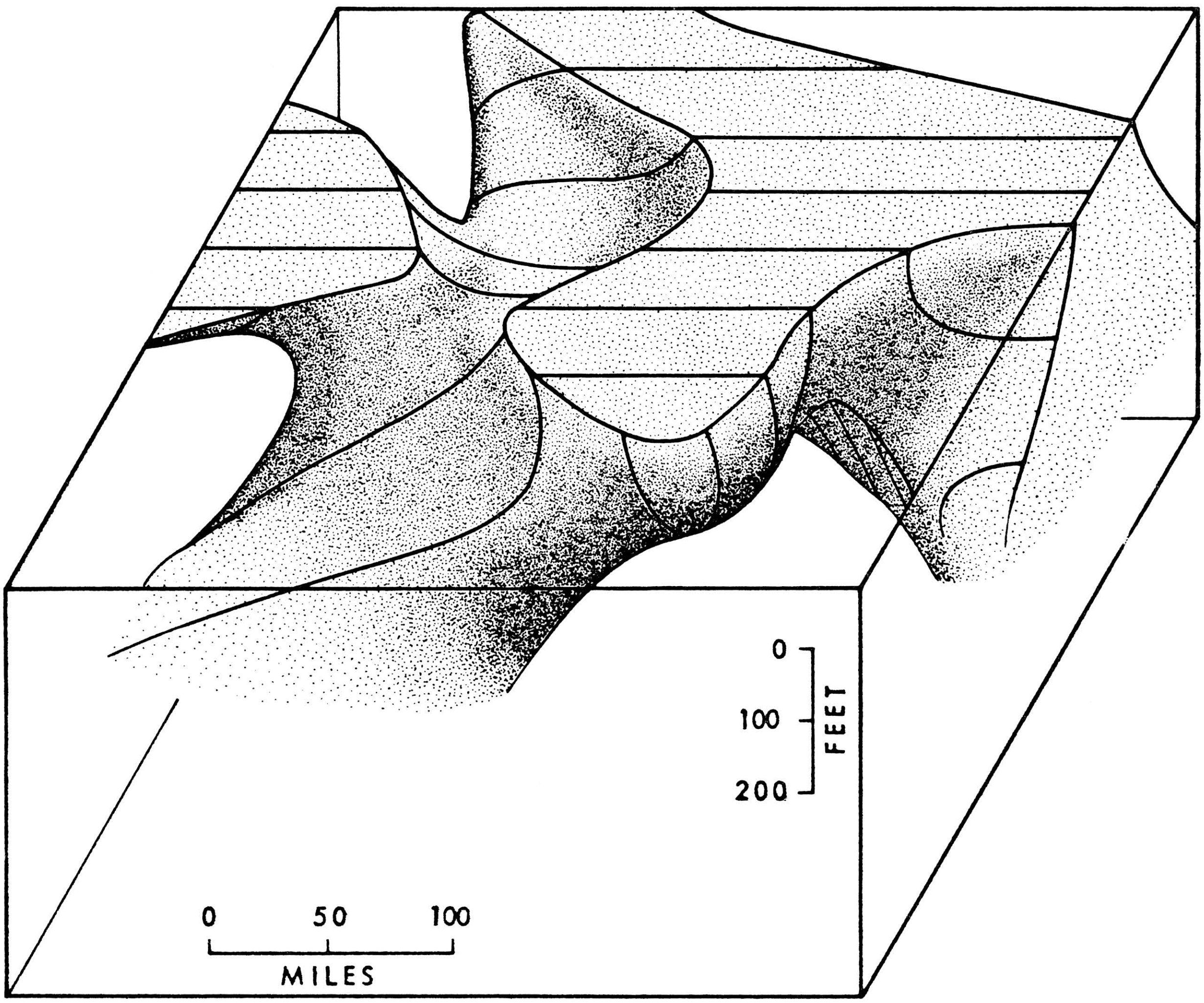

Figure 7.-Isopleth block diagram of organic carbon content in the Mowry Shale of Wyoming. Diagram was constructed from slice-maps generated along lines shown on isopleth. Solid encloses areas having greater than average (72%) organic carbon.

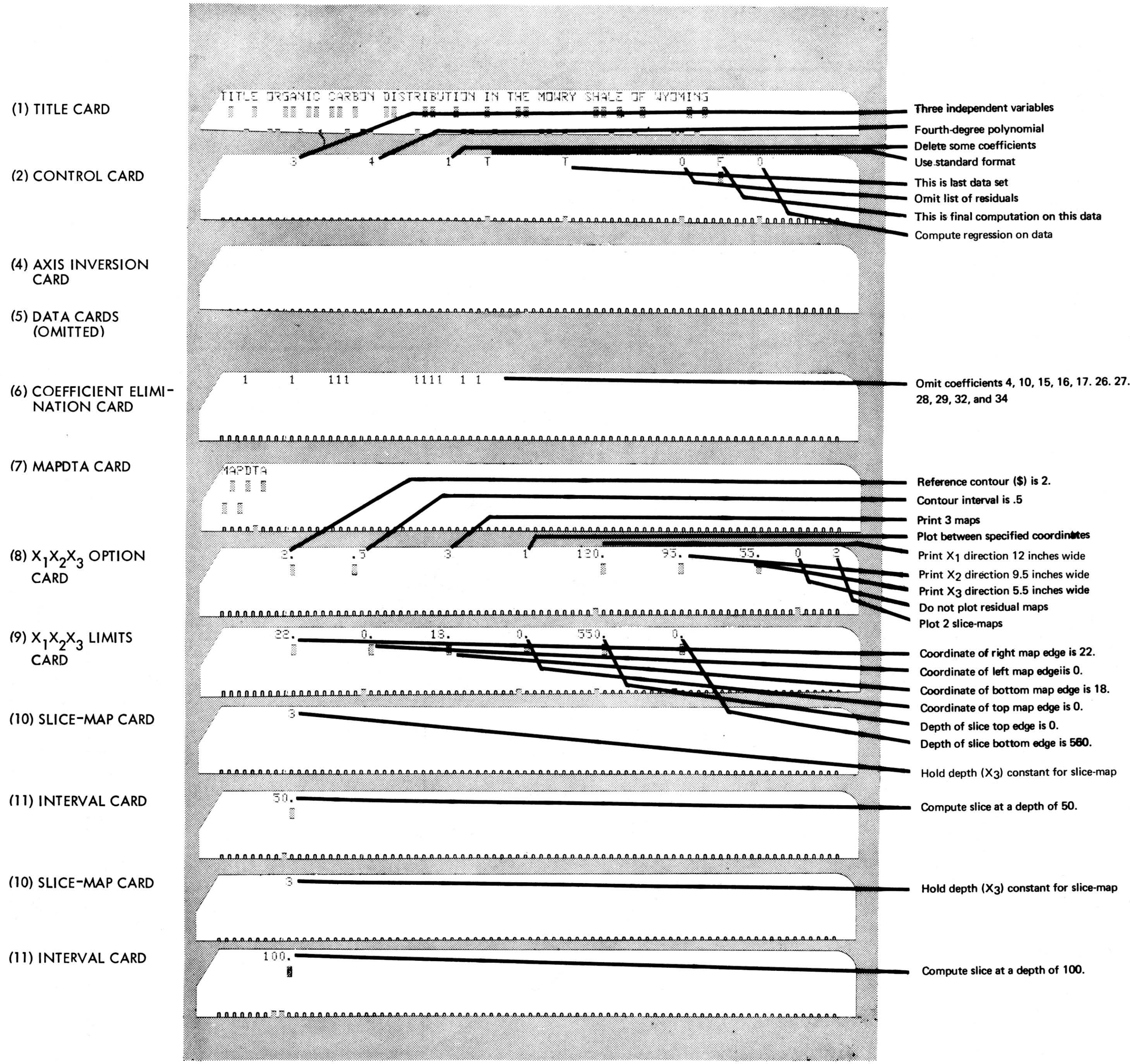

Figure 8.-Control card sequence used to generate slice-maps of organic carbon content (Example 3).

```
CKWKR8      MULTIPLE DIMENSION LEAST SQUARES REGRESSION PROGRAM              KWKR0010
C           MAINLINE ROUTINE                                                 KWKR0020
      DIMENSION FMT(14),XMAX(4),XMIN(4),Y(425),ID(14),A(36,37),SFMT(4)       KWKR0030
      DIMENSION XNUM(35),C(36)                                               KWKR0040
      INTEGER TITLE                                                          KWKR0050
      COMMON /ECKS/ X(35,425)                                                KWKR0060
      COMMON /RES/RSDLS(425)                                                 KWKR0070
      LOGICAL LASTST,LASTPT,GOOFUP,STFMT,TOMUCH,AGAIN,MAXUM,INVRT(35)        KWKR0080
      LOGICAL MAP1,MAP2,MAP3                                                 KWKR0090
      DATA SFMT/1H(,6HXXXXXX,6HF10.5,,6H9X,L1)/                              KWKR0100
      DATAXNUM/1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9,2H10,2H11,2H12,2H13,2KWKR0110
     1H14,2H15,2H16,2H17,2H18,2H19,2H20,2H21,2H22,2H23,2H24,2H25,2H26,2HKWKR0120
     227,2H28,2H29,2H30,2H31,2H32,2H33,2H34,2H35/                            KWKR0130
      DATA             MAPARM/6HMAPDTA/                                      KWKR0140
      DATA TITLE/6HTITLE /, SPACE/6H      /                                  KWKR0150
C                                                                            KWKR0160
C     READ PARAMETER CARDS AND CHECK THE VITAL PARAMETERS                    KWKR0170
C                                                                            KWKR0180
      Z=0.0
  500 READ(5,1) ID                                                           KWKR0190
    1 FORMAT(13A6,A2)                                                        KWKR0200
      IF(ID(1).NE.TITLE) GO TO 15                                            KWKR0210
    5 READ(5,10)  IORD,NDEG,NFIDLE,STFMT,LASTST,IPRNT,AGAIN                  KWKR0220
   10 FORMAT(3I10,4X,L1,5X,I10,4X,L1,I5)                                     KWKR0230
      GOOFUP = .FALSE.                                                       KWKR0240
      TOMUCH = .FALSE.                                                       KWKR0250
      N = IORD + 1                                                           KWKR0260
      IF(IORD.EQ.1.AND.NDEG.LE.35) GO TO 30                                  KWKR0270
      IF(NDEG.EQ.1.AND.IORD.LT.34) GO TO 30                                  KWKR0280
      IF((IORD*NDEG).LE.14) GO TO 30                                         KWKR0290
   15  WRITE(6,20)                                                           KWKR0300
   20 FORMAT(1H1,30X,28HBAD DATA, CHECK PARAMETERS  ///)                     KWKR0310
      GOOFUP = .TRUE.                                                        KWKR0320
      GO TO 9000                                                             KWKR0330
C                                                                            KWKR0340
C     ESTABLISH STANDARD FORMAT                                              KWKR0350
C                                                                            KWKR0360
   30 FMT(1) = SFMT(1)                                                       KWKR0370
      FMT(2) = XNUM(N)                                                       KWKR0380
      FMT(3) = SFMT(3)                                                       KWKR0390
      FMT(4) = SFMT(4)                                                       KWKR0400
      DO 31 I = 5,14                                                         KWKR0410
   31 FMT(I) = SPACE                                                         KWKR0420
      IF(STFMT) GO TO 32                                                     KWKR0430
      READ(5,1) FMT                                                          KWKR0440
   32 DO 33 I=1,4                                                            KWKR0450
      XMAX(I) = -1.E30                                                       KWKR0460
   33 XMIN(I) = 1.E 30                                                       KWKR0470
      READ(5,34)(INVRT(I),I=1,IORD)                                          KWKR0480
   34 FORMAT(35L1)                                                           KWKR0490
      DO 40 I=1,425                                                          KWKR0500
      READ(5,FMT)(X(J,I),J=1,IORD),Y(I),LASTPT                               KWKR0510
      DO 35 J = 1,IORD                                                       KWKR0520
   35 IF(INVRT(J)) X(J,I) = -X(J,I)                                          KWKR0530
      IF(IORD.GT.3) GO TO 38                                                 KWKR0540
      DO 36 J=1,IORD                                                         KWKR0550
      IF(X(J,I).GT.XMAX(J)) XMAX(J) = X(J,I)                                 KWKR0560
   36 IF(X(J,I).LT.XMIN(J)) XMIN(J) = X(J,I)                                 KWKR0570
      IF(Y(I).GT.XMAX(N)) XMAX(N) = Y(I)                                     KWKR0580
      IF(Y(I).LT.XMIN(N)) XMIN(N) = Y(I)                                     KWKR0590
   38 IF(LASTPT) GO TO 50                                                    KWKR0600
```

```
   40 CONTINUF
      WRITE(6,45)
   45 FORMAT(1H1,30X,14HTOO MUCH DATA ///)
      I=425
      TOMUCH = .TRUE.
      GO TO 9000
   50 M = I
      WRITE(6,61)(ID(I),I=2,14)
      IF(IORD.LE.7) GO TO 54
      NNEW = IORD
      GO TO 52
C
C     EXPAND DATA MATRIX, FORM LEAST SQUARES MATRIX, SOLVE AND COMPUTE
C     ERROR MEASURES
C
   61 FORMAT(1H1,30X,13A6////)
   54 CALL MLTDEG(M,IORD,NNEW,NDEG)
      NDEGMX=NDEG
   52 CALL LINFIT(M,NNEW,A,X,Y)
   53 CALL FIDDLE (A,NNEW,C,NFIDLE,LOST)
      DO 63 MM=1,IORD
   63 IF(INVRT(MM)) WRITE(6,62)MM
   62 FORMAT(1X,4HAXIS,I3,13H IS INVERTED.//)
      CALL STATPK(Y,NNEW,C,M,IORD,IPRNT,ID(2),A,LOST)
   56 IF(IORD.GT.3) GO TO 9000
      IF(TOMUCH) GO TO 80
C
C     CREATE ANY MAPS THAT ARE REQUESTED
C
      READ(5,1) MAPRM
      IF(MAPRM.EQ.MAPARM) GO TO 80
      GOOFUP=.TRUE.
      GO TO 9000
   80 GO TO (100,200,300),IORD
  100 READ(5,101) STEP,ICHK
  101 FORMAT(F10.0,I10)
      IF(STEP.EQ.0.) GO TO 9000
      IF(ICHK.EQ.0) GO TO110
      READ(5,102)XMAX(1),XMIN(1),XMAX(2),XMIN(2)
  102 FORMAT(8F10.0)
  110 CALL COMCON(C,XMAX(1),XMIN(1),Z ,XMAX(2),XMIN(2),Z ,Z ,Z ,Z ,ID(2)
     1,Z ,Z ,IORD,STEP,NDEG,M,0,X,Y)
      GO TO 9000
  200 READ(5,210)REF,CON,NMAP,ICHK,XD,YD,XD1,YD1
  210 FORMAT(2F10.0,2I10,4F10.0)
      IF(NMAP.EQ.0) GO TO 9000
      DO 220 I=1,NMAP
      IF(ICHK.EQ.0) GO TO 215
      READ(5,202)XMAX(1),XMIN(1),XMAX(2),XMIN(2),MAP1,MAP2,MAP3
  202 FORMAT(4F10.0,4X,L1,9X,L1,9X,L1)
      GO TO 216
  215 READ(5,203) MAP1,MAP2,MAP3
  203 FORMAT(44X,L1,2(9X,L1))
  216 IF(MAP1)CALL COMCON(C,XMAX(1),XMIN(1),XD,XMAX(2),XMIN(2),YD,Z ,Z ,
     1Z ,ID(2),REF,CON,IORD,Z ,NDEG,0,0,Z ,Z )
      IF(MAP2)CALL PLOTER(Y,M,XMAX(1),XMIN(1),XMAX(2),XMIN(2),XD1,YD1,
     1ID(2),1)
      IF(MAP3)CALL PLOTER(RSDLS,M,XMAX(1),XMIN(1),XMAX(2),XMIN(2),XD1,
     1YD1,ID(2),2)
  220 CONTINUE
      GO TO 9000
```

```
  300 READ(5,310)REF,CON,NMAP,ICHK,XD,YD,ZD,NMAP1,ISL                   KWKR1220
  310 FORMAT(2F10.0,2I10,3F10.0,2I5)                                    KWKR1230
      IF(NMAP.EQ.0) GO TO 330                                           KWKR1240
      DO 320 I=1,NMAP                                                   KWKR1250
      IF(ICHK.EQ.0) GO TO 315                                           KWKR1260
      READ(5,102)XMAX(1),XMIN(1),XMAX(2),XMIN(2),XMAX(3),XMIN(3)        KWKR1270
  315 CONTINUE                                                          KWKR1280
      CALL COMCON(C,XMAX(1),XMIN(1),XD,XMAX(2),XMIN(2),YD,XMAX(3),XMIN(3KWKR1290
     1),ZD,ID(2),REF,CON,IORD,Z ,NDEG,0,0,Z ,Z )                        KWKR1300
  320 CONTINUE                                                          KWKR1310
  330 IF(ISL.EQ.0) GO TO 350                                            KWKR1320
      DO 340 I=1,ISL                                                    KWKR1330
      READ(5,10) IDI                                                    KWKR1340
      SAV = XMAX (IDI)                                                  KWKR1350
      SAVE1 = XMIN(IDI)                                                 KWKR1360
      READ(5,102) XMAX(IDI)                                             KWKR1370
      XMIN(IDI) = XMAX(IDI)                                             KWKR1380
      CALL COMCON(C,XMAX(1),XMIN(1),XD,XMAX(2),XMIN(2),YD,XMAX(3),XMIN(3KWKR1390
     1),ZD,ID(2),REF,CON,IORD,Z ,NDEG,ISL,IDI,Z ,Z )                    KWKR1400
      XMAX(IDI) = SAV                                                   KWKR1410
      XMIN(IDI) = SAVE1                                                 KWKR1420
  340 CONTINUE                                                          KWKR1430
  350 IF(NMAP1.EQ.0) GO TO 9000                                         KWKR1440
      DO 355 I=1,NMAP1                                                  KWKR1450
      READ(5,352)L,XD,YD,MAP2,MAP3                                      KWKR1460
  352 FORMAT(I10,2F10.0,4X,L1,9X,L1)                                    KWKR1470
      IF(ICHK.EQ.0) GO TO 355                                           KWKR1480
      READ(5,102) XMAX(1),XMIN(1),XMAX(2),XMIN(2),XMAX(3),XMIN(3)       KWKR1490
  355 IF(MAP3) CALL PLTRSD(M,XMAX(3),L,XMIN(3),Y,0,XMAX(1),XMIN(1),XD,  KWKR1500
     1XMAX(2),XMIN(2),YD,ID(2))                                         KWKR1510
      IF(MAP2) CALL PLTRSD(M,XMAX(3),L,XMIN(3),RSDLS,NDEG,XMAX(1),XMIN(1KWKR1520
     1),XD,XMAX(2),XMIN(2),YD,ID(2))                                    KWKR1530
C                                                                       KWKR1540
C     TERMINATION PROCEDURE - EXAMINE INPUT PARAMETERS AND ERROR        KWKR1550
C     INDICATORS TO DETERMINE PROPER BRANCH                             KWKR1560
C                                                                       KWKR1570
 9000 IF(GOOFUP) GO TO 9200                                             KWKR1580
      IF(TOMUCH) GO TO 9400                                             KWKR1590
      IF(AGAIN) GO TO 9600                                              KWKR1600
      IF(LASTST) CALL EXIT                                              KWKR1610
      GO TO 500                                                         KWKR1620
 9200 IF(LASTST) GO TO 9215                                             KWKR1630
      IF(AGAIN) GO TO 9215                                              KWKR1640
      GOOFUP = .FALSE.                                                  KWKR1650
      DO 9210 I = 1,10000                                               KWKR1660
      READ(5,1) ID                                                      KWKR1670
      IF(ID(1).EQ.TITLE) GO TO 5                                        KWKR1680
 9210 CONTINUE                                                          KWKR1690
 9215 WRITE(6,9220)                                                     KWKR1700
 9220 FORMAT(1H1,40X,9(1H*)/41X,9HFORGET IT/41X,9(1H*))                 KWKR1710
      CALL EXIT                                                         KWKR1720
 9400 IF(LASTST) GO TO 9215                                             KWKR1730
       TOMUCH = .FALSE.                                                 KWKR1740
      IF(AGAIN) GO TO 9215                                              KWKR1750
      DO 9410 L=1,10000                                                 KWKR1760
      READ(5,1) MAPRM                                                   KWKR1770
      IF(MAPRM.EQ.MAPARM) GO TO 50                                      KWKR1780
 9410 CONTINUE                                                          KWKR1790
      GO TO 9215                                                        KWKR1800
C                                                                       KWKR1810
C     DETERMINE WHAT NEW TERMS ARE NEEDED FOR NEW REGRESSION ON SAME    KWKR1820
```

```
C      DATA                                                             KWKR1830
C                                                                       KWKR1840
 9600 READ(5,1) ID                                                      KWKR1850
      IF(ID(1).NE.TITLE) GO TO 15                                       KWKR1860
      NDEGO=NDEG                                                        KWKR1870
      READ(5,10)IORD,NDEG,NFIDLE,MAXUM,MAXUM ,IPRNT,AGAIN,IRSD          KWKR1880
      WRITE(6,61) (ID(I),I=2,14)                                        KWKR1890
      NNOP2=NNEW+2                                                      KWKR1900
      MM=M                                                              KWKR1910
      IF(NDEGMX.GE.NDEG)MM=1                                            KWKR1920
      CALL MLTDEG(MM,IORD,NNEW,NDEG)                                    KWKR1930
      IF(MM.EQ.M)NDEGMX=NDEG.                                           KWKR1940
      IF(IRSD.EQ.1) GO TO 9630                                          KWKR1950
 9620 IF(NDEGO.LT.NDEG) GO TO 52                                        KWKR1960
 9651 NNP2=NNEW+2                                                       KWKR1970
      NNP1=NNEW+1                                                       KWKR1980
      DO 9610 INEW=1,NNP1                                               KWKR1990
 9610 A(INEW,NNP2)=A(INEW,NNOP2)                                        KWKR2000
      GO TO 53                                                          KWKR2010
 9630 WRITE(6,9631)                                                     KWKR2020
 9631 FORMAT(1X,95HRESIDUALS FROM LAST REGRESSION HAVE REPLACED DEPENDENKWKR2030
     1T VARIABLE FOR ALL SUBSEQUENT REGRESSIONS.//)                     KWKR2040
      DO 9640 I=1,M                                                     KWKR2050
 9640 Y(I)=RSDLS(I)                                                     KWKR2060
      GO TO 52                                                          KWKR2070
      END                                                               KWKR2080
$      FORTRAN NDECK
CPOLYD        FUNCTION TO CALCULATE POLYNOMIAL THAT HAS BEEN GENERATED   POLY0010
C             BY MLTDEG AND LINFIT                                      POLY0020
C                         N      NUMBER OF INDEPENDENT VARIABLES        POLY0030
C                         B      COEFFICIENT ARRAY                      POLY0040
C                         X      SINGLY DIMENSIONED ARRAY OF INDEPENDENT POLY0050
C                                VARIABLES                              POLY0060
C                         NDEG   DEGREE OF POLYNOMIAL                   POLY0070
C                                                                       POLY0080
      FUNCTION POLYD(N,B,X,NDEG)                                        POLY0090
      DIMENSION AUX(3),X(35),B(36)                                      POLY0100
      IF(N.EQ.1) GO TO 3000                                             POLY0110
  200 NP1=N+1                                                           POLY0120
      IND=NP1                                                           POLY0130
      IF(NDEG.LE.1) GO TO 100                                           POLY0140
C                                                                       POLY0150
C     COMPUTE 2ND DEGREE TERMS                                          POLY0160
C                                                                       POLY0170
      DO 20 KK=1,N                                                      POLY0180
      DO 20 LL=1,KK                                                     POLY0190
      X(IND)=X(LL)*X(KK)                                                POLY0200
   20 IND=IND+1                                                         POLY0210
      I3=IND                                                            POLY0220
      IF(NDEG.EQ.2) GO TO 100                                           POLY0230
C                                                                       POLY0240
C     COMPUTE 3RD DEGREE TERMS                                          POLY0250
C                                                                       POLY0260
      I2=N                                                              POLY0270
      DO 30 K=1,N                                                       POLY0280
      I2=I2+K                                                           POLY0290
      DO 30 KK=1,N                                                      POLY0300
      X(IND)=X(KK)*X(I2)                                                POLY0310
   30 IND=IND+1                                                         POLY0320
      IF(N.EQ.2) GO TO 31                                               POLY0330
      X13=X(1)*X(3)                                                     POLY0340
```

```
      IF(N.EQ.3) GO TO 32
      X24=X(2)*X(4)
      X(IND)=X24*X(1)
      X(IND+1)=X13*X(4)
      X(IND+2)=X24*X(3)
      IND=IND+3
   32 X(IND)=X13*X(2)
      IND=IND+1
   31 IF(NDEG.EQ.3) GO TO 100
C
C     COMPUTE 4TH DEGREE TERMS
C
      I4=IND
      DO 40 K=1,N
      DO 41 KK=1,N
      X(IND)=X(KK)*X(I3)
   41 IND=IND+1
   40 I3=I3+NP1
      AUX(1)=X(1)*X(2)
      IF(N.EQ.2) GO TO 42
      AUX(2)=X(1)*X(3)
      AUX(3)=X(2)*X(3)
      DO 43 KK=2,N
      DO 43 LL=1,KK
      X(IND)=AUX(KK)*AUX(LL)
   43 IND=IND+1
   42 X(IND)=AUX(1)**2
      IND=IND+1
      IF(NDEG.EQ.4) GO TO 100
C
C     COMPUTE 5TH THRU 7TH DEGREE TERMS
C
      FACTOR=X(2)/X(1)
      DO 51 LAST=5,NDEG
      X(IND)=X(1)*X(I4)
      I4=IND
      IND=IND+1
      DO 51 KK=1,LAST
      X(IND)=X(IND-1)*FACTOR
   51 IND=IND+1
C
C     COMPUTE VALUE OF FUNCTION AT POINT USING EXPANDED TERMS AND
C     COEFFICIENTS
C
  100 IND=IND-1
      POLYD=B(1)
      DO 900 L=1,IND
  900 POLYD=POLYD+B(L+1)*X(L)
      RETURN
C
C     COMPUTE FUNCTION FOR 1 DIMENSIONAL CASE
C
 3000 XX = X(1)
      POLYD =B(NDEG+1)
      DO 3100 J=1,NDEG
      K = NDEG + 1 - J
 3100 POLYD = POLYD * XX + B(K)
      RETURN
      END
$      FORTRAN NDECK
CFIDDLE     ROUTINE TO REMOVE COEFFICIENTS AND STABILIZE LEAST SQUARES
```

```
C           MATRIX                                                      FIDD0020
C       A      =  LEAST SQUARES MATRIX                                  FIDD0030
C       NNEW   =  NUMBER OF COEFFICIENTS NEEDED                         FIDD0040
C       C      =  ARRAY FOR COEFFICIENTS                                FIDD0050
C       NFIDLE =   SWITCH FOR TERM DELETION                             FIDD0060
C                  0   =   NO DELETIONS                                 FIDD0070
C                  1   =   DELETIONS                                    FIDD0080
C       ID     =  13A6 TITLE                                            FIDD0090
      SUBROUTINE FIDDLE(A,NNEW,C,NFIDLE,    LOST)                       FIDD0100
      DIMENSION A(36,37),C(37),IFD(37)                                  FIDD0110
      DOUBLE PRECISION XMAX,XMIN,XTEST                                  FIDD0120
      DOUBLE PRECISION AN(36,37),AC(36)                                 FIDD0130
      NNEWP = NNEW + 1                                                  FIDD0140
      NNEWPP = NNEW + 2                                                 FIDD0150
      LOST = 0                                                          FIDD0160
      DO 10 I=1,NNEWPP                                                  FIDD0170
   10 IFD(I) = 0                                                        FIDD0180
      IF(NFIDLE.EQ.0) GO TO 40                                          FIDD0190
      READ(5,20)(IFD(I),I=1,NNEWP)                                      FIDD0200
   20 FORMAT(36I1)                                                      FIDD0210
C                                                                       FIDD0220
C     ELIMINATE SPECIFIED COEFFICIENTS AND CONVERT LEAST SQUARES MATRIX FIDD0230
C     TO DOUBLE PRECISION FOR SOLUTION                                  FIDD0240
C                                                                       FIDD0250
   40 II = 0                                                            FIDD0260
      DO 60 I=1,NNEWP                                                   FIDD0270
      JJ = 0                                                            FIDD0280
      IF(IFD(I).NE.0) GO TO 60                                          FIDD0290
      II = II + 1                                                       FIDD0300
      DO 50 J=1,NNEWPP                                                  FIDD0310
      IF(IFD(J).NE.0) GO TO 50                                          FIDD0320
      JJ = JJ + 1                                                       FIDD0330
      AN(II,JJ) = A(I,J)                                                FIDD0340
   50 CONTINUE                                                          FIDD0350
   60 CONTINUE                                                          FIDD0360
C                                                                       FIDD0370
C     MINIMIZE THE MAGNITUDE OF THE TERMS IN THE LEAST SQUARES MATRIX   FIDD0380
C                                                                       FIDD0390
      N=II                                                              FIDD0400
      NP1=II+1                                                          FIDD0410
      XMAX = DABS(AN(N,N))                                              FIDD0420
      XMIN=XMAX                                                         FIDD0430
      DO 110 I=1,N                                                      FIDD0440
      DO 110 J=1,NP1                                                    FIDD0450
      XTEST = DABS(AN(I,J))                                             FIDD0460
      IF (XTEST.LT.XMIN) XMIN = XTEST                                   FIDD0470
  110 IF (XTEST.GT.XMAX) XMAX = XTEST                                   FIDD0480
      IPWRL = DLOG10(XMIN)                                              FIDD0490
      IPWR  = DLOG10(XMAX)                                              FIDD0500
      XTEST = XMAX                                                      FIDD0510
      XMAX = 10.**((IPWR-IPWRL)/2)/XTEST                                FIDD0520
      DO 120 I=1,N                                                      FIDD0530
      DO 120 J=1,NP1                                                    FIDD0540
  120 AN(I,J)=AN(I,J)* XMAX                                             FIDD0550
C                                                                       FIDD0560
C     CALCULATE COEFFICIENTS                                            FIDD0570
C                                                                       FIDD0580
      CALL EMSLVR(AN,AC,II)                                             FIDD0590
      JJ = 0                                                            FIDD0600
      DO 70 J=1,NNEWP                                                   FIDD0610
      C(J) = 0.                                                         FIDD0620
```

```
      IF(IFD(J).NE.0) GO TO 64                                          FIDD0630
      JJ = JJ + 1                                                       FIDD0640
      C(J) = AC(JJ)                                                     FIDD0650
      GO TO 70                                                          FIDD0660
   64 WRITE(6,65) J                                                     FIDD0670
      LOST = LOST + 1                                                   FIDD0680
   65 FORMAT(12H COEFFICIENT,I3, 85H HAS PURPOSELY BEEN ELIMINATED BEFORFIDD0690
     1E SOLUTION OF THE MATRIX AND HAS BEEN SET TO 0.     //)           FIDD0700
   70 CONTINUE                                                          FIDD0710
      RETURN                                                            FIDD0720
      END                                                               FIDD0730
$       FORTRAN NDECK
CSTATPK      STATISTICAL PACKAGE TO COMPUTE STANDARD ERROR MEASURES     STAT0010
C      Y      =  ARRAY OF DEPENDENT VARIABLE                            STAT0020
C      NDEG   =  NUMBER OF COEFFICIENTS IN POLYNOMIAL                   STAT0030
C      C      =  ARRAY OF COEFFICIENTS                                  STAT0040
C      N      =  NUMBER OF DATA POINTS                                  STAT0050
C      NINDEP =  NUMBER OF VARIABLES                                    STAT0060
C      IPRNT  =  SWITCH FOR LISTING RESIDUALS                           STAT0070
C                0  =  NO LIST                                          STAT0080
C                1  =  LIST                                             STAT0090
C      ID     =  13A6 TITLE                                             STAT0100
C      A      =  LEAST SQUARES MATRIX CREATED BY LINFIT                 STAT0110
      SUBROUTINE STATPK(Y,NDEG,C,N,NINDEP,IPRNT,ID,A,LOST)              STAT0120
      COMMON /ECKS/X(35,425)                                            STAT0130
      COMMON /RES/YCPD(425)                                             STAT0140
      DIMENSION Y(425),C(36),ID(13),B(36),A(36,37)                      STAT0150
      DATA IB/1H /                                                      STAT0160
      V=0.                                                              STAT0170
      RN=N                                                              STAT0180
      IF(IPRNT.NE.0)WRITE(6,19)ID,IB                                    STAT0190
      S=0.                                                              STAT0200
      YBAR=0.                                                           STAT0210
      DO 10 I=1,N                                                       STAT0220
      YCPDI=C(1)                                                        STAT0230
      DO 11 J=1,NDEG                                                    STAT0240
   11 YCPDI=YCPDI+X(J,I)*C(J+1)                                         STAT0250
      YBAR=YBAR+Y(I)                                                    STAT0260
      RES=Y(I)-YCPDI                                                    STAT0270
      YCPD(I)=RES                                                       STAT0280
      S=S+RES**2                                                        STAT0290
      IF(IPRNT.EQ.0) GO TO 10                                           STAT0300
   19 FORMAT(1H1,30X,13A6//A1,30X,48HORIGINAL DATA, CALCULATED SURFACE, STAT0310
  191AND RESIDUALS //)                                                  STAT0320
      WRITE(6,22)(X(J,I),J=1,NINDEP),Y(I),YCPDI,RES                     STAT0330
   22 FORMAT(/(1X,10G12.4/))                                            STAT0340
   10 CONTINUE                                                          STAT0350
      YBAR=YBAR/RN                                                      STAT0360
      DO 20 I=1,N                                                       STAT0370
   20 V=V+(Y(I)-YBAR)**2                                                STAT0380
      E=V-S                                                             STAT0390
      T=E/V                                                             STAT0400
      EL=SQRT(ABS(T))                                                   STAT0410
      IF(T.LT.0.) EL=-EL                                                STAT0420
      D=SQRT(S/RN)                                                      STAT0430
      WRITE(6,19)ID                                                     STAT0440
      NNEP=NDEG+1-LOST                                                  STAT0450
      NDF=N-NNEP-1                                                      STAT0460
      FRATIO = (E/FLOAT(NNEP))/(S/FLOAT(NDF))                           STAT0470
      WRITE(6,30)V,YBAR,    S,    E,T,EL,D,FRATIO,NNEP,NDF              STAT0480
   30 FORMAT(    6X,15HTOTAL VARIATION,F25.6//6X,4HMEAN,11X,F25.6//6X,13STAT0490
```

```
     1HVARIATION NOT/6X,20HEXPLAINED BY SURFACE F20.6//6X,19HVARIATION ESTAT0500
     2XPLAINED/6X,10HBY SURFACE,10X,F20.6//6X,14HCOEFFICIENT OF/6X,13HDESTAT0510
     3TERMINATION,F27.6//6X,14HCOEFFICIENT OF/6X,11HCORRELATION,F29.6//6STAT0520
     4X,18HSTANDARD DEVIATION,F22.6//6X,9HF - RATIO,9X,F22.6,5X,4HWITH, STAT0530
     5I4,4H AND,I4,19H DEGREES OF FREEDOM///6X,12HCOEFFICIENTS//)       STAT0540
      NNEP=NNEP+LOST                                                    STAT0550
      WRITE(6,60)(IB,I,C(I),I=1,NNEP)                                   STAT0560
   60 FORMAT(   5(A6,2HC(,I2,4H) = ,F12.5)/)                            STAT0570
      WRITE(6,31)                                                       STAT0580
   31 FORMAT(//6X,40HSTANDARD PARTIAL REGRESSION COEFFICIENTS//)        STAT0590
      DO 40 I=1,NNEP                                                    STAT0600
   40 B(I)=C(I)*SQRT((A(I,I)-(A(I,1)**2)/RN)/V)                         STAT0610
      WRITE(6,60)(IB,I,B(I),I=1,NNEP)                                   STAT0620
      RETURN                                                            STAT0630
      END                                                               STAT0640
$      FORTRAN NDECK
CEMSLVR      SIMULTANEOUS EQUATION SOLVER                               EMSL0010
C     A = MATRIX OF EQUATIONS TO BE SOLVED, IN THE FORM A(N,N+1) WITH   EMSL0020
C         THE CONSTANTS IN THE LAST COLUMN                              EMSL0030
C     X = ARRAY OF COEFFICIENTS THAT ARE CALCULATED                     EMSL0040
C     N = NUMBER OF EQUATIONS TO BE SOLVED                              EMSL0050
      SUBROUTINE EMSLVR(A,X,N)                                          EMSL0060
      DOUBLE PRECISION A(36,37),X(36),SAVE,ALL,SUM                      EMSL0070
      ZERO=1.E-20                                                       EMSL0080
      NP1 = N+1                                                         EMSL0090
      NM1 = N-1                                                         EMSL0100
      DO 400 L=1,NM1                                                    EMSL0110
      LP1 = L+1                                                         EMSL0120
      IF(DABS(A(L,L)).GT.ZERO)  GO TO 300                               EMSL0130
      IN = L                                                            EMSL0140
  150 IN = IN+1                                                         EMSL0150
      IF(IN.GT.N) GO TO 900                                             EMSL0160
      IF(DABS(A(IN,L)).LT.ZERO)  GO TO 150                              EMSL0170
  200 DO 210 JJ=L,NP1                                                   EMSL0180
  210 A(L,JJ) = A(L,JJ) + A(IN,JJ)                                      EMSL0190
  300 ALL=1./A(L,L)                                                     EMSL0200
      IF(DABS(ALL).LT.ZERO)  ALL=0.                                     EMSL0210
      DO 400 K=LP1,N                                                    EMSL0220
      SAVE = A(K,L)                                                     EMSL0230
      IF(DABS(SAVE).LT.ZERO)  GO TO 400                                 EMSL0240
         DO 350 J=L,NP1                                                 EMSL0250
  350    A(K,J)=A(K,J)-(SAVE*ALL*A(L,J))                                EMSL0260
  400 CONTINUE                                                          EMSL0270
      IF(DABS(A(N,N)).GT.ZERO)   GO TO 2000                             EMSL0280
  900 WRITE(6,800)                                                      EMSL0290
  800 FORMAT(22H NO SOLUTION IN SIMEQ.)                                 EMSL0300
      RETURN                                                            EMSL0310
 2000 X(N) = A(N,NP1)/A(N,N)                                            EMSL0320
      IF(DABS(X(N)).LT.ZERO)   X(N)=0.                                  EMSL0330
      DO 700 KK=1,NM1                                                   EMSL0340
      K = N-KK                                                          EMSL0350
      SUM = A(K,NP1)                                                    EMSL0360
      KP1 = K+1                                                         EMSL0370
         DO 650 J=KP1,N                                                 EMSL0380
  650    SUM = SUM-A(K,J)*X(J)                                          EMSL0390
      X(K)=SUM/A(K,K)                                                   EMSL0400
  700 IF(DABS(X(K)).LT.ZERO)   X(K)=0.                                  EMSL0410
      RETURN                                                            EMSL0420
      END                                                               EMSL0430
$      FORTRAN NDECK
CMLTDEG      N-TH ORDER TERM GENERATION ROUTINE                         MLTD0010
```

```
C
C        TO COMPUTE THE COEFFICIENTS CALL A LINEAR FIT ROUTINE WITH NNEW
C   VARIABLES.
C        X IS THE ARRAY OF THE FIRST DEGREE DATA WITH SPACE FOR THE CROS
C   PRODUCT TERMS TO BE ADDED.
C        M IS THE NUMBER OF SETS OF DATA.
C        N IS THE NUMBER OF INDEPENDENT VARIABLES.
C        NNEW IS THE TOTAL NUMBER OF TERMS WITH THE HIGHER POWER TERMS
C   INCLUDED.  CALL THE FIT SUBROUTINE WITH THIS NUMBER OF INDEPENDENT V
C        NDEG IS THE DEGREE OF THE FIT DESIRED.
C
      SUBROUTINE MLTDEG  (M,N,NNEW,NDEG)
      COMMON /ECKS/ X(35,425)
      DIMENSION AUX(3)
      IF(N.EQ.1) GO TO 3000
      IF(N.LE.1) GO TO 300
      IF(NDEG.LE.1) GO TO 300
      IF(N*NDEG.LE.14) GO TO 200
  300 NNEW=N
      RETURN
  200 NP1=N+1
C--------COMPUTE 2ND DEGREE TERMS.
      DO 100 J=1,M
      IND=NP1
      DO 20 KK=1,N
      DO 20 LL=1,KK
      X(IND,J)=X(LL,J)*X(KK,J)
   20 IND=IND+1
      I3=IND
      IF(NDEG.EQ.2) GO TO 100
C--------COMPUTE 3RD DEGREE TERMS.
      I2=N
      DO 30 K=1,N
      I2=I2+K
      DO 30 KK=1,N
      X(IND,J)=X(KK,J)*X(I2,J)
   30 IND=IND+1
      IF(N.EQ.2) GO TO 31
      X13=X(1,J)*X(3,J)
      IF(N.EQ.3) GO TO 32
      X24=X(2,J)*X(4,J)
      X(IND,J)=X24*X(1,J)
      X(IND+1,J)=X13*X(4,J)
      X(IND+2,J)=X24*X(3,J)
      IND=IND+3
   32 X(IND,J)=X13*X(2,J)
      IND=IND+1
   31 IF(NDEG.EQ.3) GO TO 100
C--------COMPUTE 4TH DEGREE TERMS.
      I4=IND
      DO 40 K=1,N
      DO 41 KK=1,N
      X(IND,J)=X(KK,J)*X(I3,J)
   41 IND=IND+1
   40 I3=I3+NP1
      AUX(1)=X(1,J)*X(2,J)
      IF(N.EQ.2) GO TO 42
      AUX(2)=X(1,J)*X(3,J)
      AUX(3)=X(2,J)*X(3,J)
      DO 43 KK=2,N
      DO 43 LL=1,KK
```

```
      X(IND,J)=AUX(KK)*AUX(LL)                                          MLTD0630
   43 IND=IND+1                                                         MLTD0640
   42 X(IND,J)=AUX(1)**2                                                MLTD0650
      IND=IND+1                                                         MLTD0660
      IF(NDEG.EQ.4) GO TO 100                                           MLTD0670
C--------COMPUTE 5TH THRU 7TH DEGREE TERMS.                             MLTD0680
      FACTOR=X(2,J)/X(1,J)                                              MLTD0690
      DO 51 LAST=5,NDEG                                                 MLTD0700
      X(IND,J)=X(1,J)*X(I4,J)                                           MLTD0710
      I4=IND                                                            MLTD0720
      IND=IND+1                                                         MLTD0730
      DO 51 KK=1,LAST                                                   MLTD0740
      X(IND,J)=X(IND-1,J)*FACTOR                                        MLTD0750
   51 IND=IND+1                                                         MLTD0760
  100 CONTINUE                                                          MLTD0770
      NNEW=IND-1                                                        MLTD0780
      RETURN                                                            MLTD0790
C                                                                       MLTD0800
C     COMPUTE 1 DIMENSIONAL TERMS                                       MLTD0810
C                                                                       MLTD0820
 3000 DO 3100 I = 1,M                                                   MLTD0830
      XX=X(1,I)                                                         MLTD0840
      XS = XX                                                           MLTD0850
      DO 3100 J=2,NDEG                                                  MLTD0860
      XX = XX * XS                                                      MLTD0870
 3100 X(J,I) = XX                                                       MLTD0880
      NNEW = NDEG                                                       MLTD0890
      RETURN                                                            MLTD0900
      END                                                               MLTD0910
$      FORTRAN NDECK
CLINFIT      N-TH DIMENSIONAL LEAST SQUARES ROUTINE                     LINF0010
      SUBROUTINE LINFIT(M,N,A,X,Y)                                      LINF0020
      DIMENSION A(36,37)       ,X(35,425),Y(425)                        LINF0030
C----------M IS THE NUMBER OF SETS OF DATA                              LINF0040
C----------N IS THE NUMBER OF INDEPENDENT VARIABLES                     LINF0050
C----------B IS THE SET OF COEFFICIENTS IN THE OUTPUT                   LINF0060
C----------X IS THE ARRAY OF DATA FOR THE INDEPENDENT VARIABLES         LINF0070
      NP2 = N+2                                                         LINF0100
C----------SET NECESSARY MEMBERS OF ARRAY TO ZERO                       LINF0110
      DO 15 K=2,NP1                                                     LINF0120
C----------Y IS THE ARRAY OF DATA FOR THE DEPENDENT VARIABLE            LINF0080
      NP1 = N+1                                                         LINF0090
      DO 16 I=1,K                                                       LINF0130
   16 A(K,I)=0.                                                         LINF0140
   15 A(K,NP2) = 0.                                                     LINF0150
      A(1,1) = M                                                        LINF0160
      A(1,NP2) = 0.                                                     LINF0170
C----------COMPUTE SUMS WHERE A SET TO ZERO                             LINF0180
      DO 60 J=1,M                                                       LINF0190
      A(1,NP2) = A(1,NP2)+Y(J)                                          LINF0200
      DO 60 K=2,NP1                                                     LINF0210
      KM1 = K-1                                                         LINF0220
      DO 50 I=2,K                                                       LINF0230
   50 A(K,I) = A(K,I)+X(KM1,J)*X(I-1,J)                                 LINF0240
      A(K,1) = A(K,1)+X(KM1,J)                                          LINF0250
   60 A(K,NP2) = A(K,NP2)+X(KM1,J)*Y(J)                                 LINF0260
C----------FILL OUT MATRIX USING ITS SYMMETRY                           LINF0270
      DO 40 K=2,NP1                                                     LINF0280
      KM1 = K-1                                                         LINF0290
      DO 40 I=1,KM1                                                     LINF0300
   40 A(I,K) = A(K,I)                                                   LINF0310
```

```
      RETURN                                                            LINF0320
      END                                                               LINF0330
$      FORTRAN NDECK
CSONG              FUNCTION SONG IS USED BY SUBROUTINE SHFTOF           SONG0010
C                         TO INSURE THE CONTINUED NEGATIVE              SONG0020
C                         TRUNCATION OF THE LOG10 VALUES                SONG0030
C                         EX. ALOG10(X) = 9.478   TRUNCATED = 9         SONG0040
C                             ALOG10(X) =-9.478   TRUNCATED = -10       SONG0050
C                                                                       SONG0060
      FUNCTION SONG(X)                                                  SONG0070
      SONG = 0.                                                         SONG0080
      IF(ABS(X).LT.1.) SONG = -.999999                                  SONG0090
      RETURN                                                            SONG0100
      END                                                               SONG0110
$      FORTRAN NDECK
CSHIFTOF            SUBROUTINE SHFTOF IS USED TO CONVERT A NUMBER       SHIF0010
C                         TO ALPHANUMERIC.                              SHIF0020
C                      RR =  NUMBER TO BE CONVERTED                     SHIF0030
C                      ID =  ARRAY WHICH CONTAINS THE ALPHA NUMBER      SHIF0040
C                      MAX = LOG10 OF THE LARGEST VALUE TO BE PLOTTED   SHIF0050
C                                      (ROUNDED UP)                     SHIF0060
C                      NC =  NUMBER OF CHARACTERS  (MAX(NC) =4)         SHIF0070
C                      ID CONTAINS NUMBER OF THE FORM                   SHIF0080
C                               FIRST WORD - BLANK                      SHIF0090
C                               SECOND WORD - SIGN                      SHIF0100
C                               THIRD - FIFTH WORD - DIGIT              SHIF0110
C                                                                       SHIF0120
      SUBROUTINE SHFTOF(RR,ID,MAX,NC)                                   SHIF0130
      DIMENSION ID(5),IR(3)                                             SHIF0140
      INTEGER SYMBOL(12),BLNK                                           SHIF0150
      DATA SYMBOL/1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9,1H+,1H-/,BLNK/SHIF0160
     11H /                                                              SHIF0170
      R=RR                                                              SHIF0180
      DO 1 I = 1,5                                                      SHIF0190
    1 ID(I) = BLNK                                                      SHIF0200
      IF(ABS(R).LT.1.E-15) GO TO 90                                     SHIF0210
      ILOG = ALOG10(ABS(R)) + SONG(R)                                   SHIF0220
      ID(2)=SYMBOL(11)                                                  SHIF0230
      IF(R.LT.0.0) GO TO 100                                            SHIF0240
      GO T O         11                                                 SHIF0250
  100 R = ABS(R)                                                        SHIF0260
      ID(2) = SYMBOL(12)                                                SHIF0270
   11 N = MAX - ILOG                                                    SHIF0280
      IF(N.GT.3) GO TO 2                                                SHIF0290
      GO TO 3                                                           SHIF0300
    2 ID(3) = SYMBOL(1)                                                 SHIF0310
      NC = 2                                                            SHIF0320
      RETURN                                                            SHIF0330
    3 GO TO (101,102,103),N                                             SHIF0340
  101 X=10.**ILOG                                                       SHIF0350
      NC = 4                                                            SHIF0360
      IR(1)=R/X                                                         SHIF0370
      X = X * .1                                                        SHIF0380
      IR(2) = R/X                                                       SHIF0390
      X = X * .1                                                        SHIF0400
      IR(3) = R/X                                                       SHIF0410
      NZ = 5                                                            SHIF0420
      GO TO 107                                                         SHIF0430
  102 X = 10.**(ILOG)                                                   SHIF0440
      NC = 3                                                            SHIF0450
      IR(1)=R/X                                                         SHIF0460
```

```
      X = X * .1                                                        SHIF0470
      IR(2)=R/X                                                         SHIF0480
      NZ = 4                                                            SHIF0490
      GO TO 107                                                         SHIF0500
  103 X = 10.**(ILOG)                                                   SHIF0510
      NC = 2                                                            SHIF0520
      IR(1)=R/X                                                         SHIF0530
      NZ = 3                                                            SHIF0540
  107 NF = 0                                                            SHIF0550
      IR2 = IR(2)                                                       SHIF0560
      IR(2)=IR(2)-IR(1)*10                                              SHIF0570
      IR(3) = IR(3) - IR2 * 10                                          SHIF0580
      DO 10 J = 3,NZ                                                    SHIF0590
      NF=NF+1                                                           SHIF0600
      DO 5JJJ=1,10                                                      SHIF0610
      JJ = JJJ - 1                                                      SHIF0620
      IF(IR(NF).EQ.JJ) GO TO 6                                          SHIF0630
    5 CONTINUE                                                          SHIF0640
    6 ID(J)=SYMBOL(JJJ)                                                 SHIF0650
   10 CONTINUE                                                          SHIF0660
      RETURN                                                            SHIF0670
   90 ID(2) = SYMBOL(11)                                                SHIF0680
      IF(R.LT.0.)ID(2) = SYMBOL(12)                                     SHIF0690
      GO TO 2                                                           SHIF0700
      END                                                               SHIF0710
$      FORTRAN NDECK
CPLTRSD      INTERFACE ROUTINE FOR 3 VARIABLE RESIDUAL MAPS             PLTR0010
C     N     = NUMBER OF DATA POINTS                                     PLTR0020
C     HI    = HIGH VALUE OF FIRST INDEPENDENT VARIABLE TO BE PLOTTED    PLTR0030
C     L     = NUMBER OF SLICES DESIRED                                  PLTR0040
C     LO    = LOW VALUE OF FIRST INDEPENDENT VARIABLE TO BE PLOTTED     PLTR0050
C     RSD   = ARRAY OF DEPENDENT VARIABLE                               PLTR0060
C     DEG   = DEGREE OF FIT (0 = ORIGINAL DATA)                         PLTR0070
C     XR    = X VALUE AT RIGHT OF MAP                                   PLTR0080
C     XL    = X VALUE AT LEFT OF MAP                                    PLTR0090
C     XD    = X DIMENSION OF MAP (1/10 INCHES)                          PLTR0100
C     YB    = Y VALUE AT BOTTOM OF MAP                                  PLTR0110
C     YT    = Y VALUE AT TOP OF MAP                                     PLTR0120
C     YD    = Y DIMENSION OF MAP (1/10 INCHES)                          PLTR0130
C     A     = 13A6 TITLE                                                PLTR0140
      SUBROUTINE PLTRSD(N,HI,L,LO,RSD,DEG,XR,XL,XD,YB,YT,YD,A)          PLTR0150
      REAL LO                                                           PLTR0160
      DIMENSION RSD(425)                                                PLTR0170
      INTEGER DEG,A(13)                                                 PLTR0180
      COMMON /EXALT/XXX(425) /ECKS/X(35,425)                            PLTR0190
      D=(HI-LO)/FLOAT(L)*1.000001                                       PLTR0200
      XLOO = LO                                                         PLTR0210
      XHI = LO + D                                                      PLTR0220
      DO 5000 NUT=1,L                                                   PLTR0230
      WRITE(6,1880) (A(IT),IT=1,13)                                     PLTR0240
 1880 FORMAT(1H1,35X,13A6//)                                            PLTR0250
      WRITE(6,2410)YT,YB,XL,XR,XLOO,XHI                                 PLTR0260
 2410 FORMAT(1X,6HYMIN =,F10.4,11H (TOP EDGE),7X,6HYMAX =,F10.4,14H (BOTPLTR0270
     1TOM EDGE),5X,6HXMIN =,F10.4,12H (LEFT EDGE)/1X,6HXMAX =,F10.4,13H PLTR0280
     2(RIGHT EDGE),5X,6HZMIN =F10.4,18H (BOTTOM OF SLICE),1X,6HZMAX =,  PLTR0290
     3F10.4,15H (TOP OF SLICE)//)                                       PLTR0300
      IF(DEG.EQ.0) GO TO 2000                                           PLTR0310
 2100 WRITE(6,2110)                                                     PLTR0320
 2110 FORMAT(18H PLOT OF RESIDUALS//)                                   PLTR0330
      GO TO 2400                                                        PLTR0340
 2000 WRITE(6,2010)                                                     PLTR0350
```

```
 2010 FORMAT(22H PLOT OF ORIGINAL DATA//)
 2400 DO 2500 IJ = 1,N
      XXX(IJ)=X(1,IJ)
 2500 IF(X(3,IJ).LT.XLOO.OR.X(3,IJ).GT.XHI) XXX(IJ)=-9999999.
      CALL PLOTER(RSD,N,XR,XL,YB,YT,XD,YD,A,3)
      XLOO=XHI
      XHI=XHI+D
 5000 CONTINUE
      RETURN
      END
$      FORTRAN NDECK
CPL-TER     SUBROUTINE PLOTER WILL PLOT RESIDUAL MAPS WITH THE
C                     FOLLOWING SCHEME
C                        N = NUMBER OF POINTS
C                    XMAX,XMIN = RANGE OF X VALUES
C                    YMAX,YMIN = RANGE OF Y VALUES
C                    XD = X DIMENSION (IN 1/10 INCHES)
C                    YD = Y DIMENSION (IN 1/10 INCHES)
C                     IDENT = 13A6 TITLE
C                     ITID = MODE INDICATOR (1 OR 2 INDICATES TWO
C                           DIMENSIONAL MAP, 3 INDICATES A THREE
C                           DIMENSIONAL MAP)
C
C                     ********* NOTE ************
C                     ORIGIN IS ASSUMED IN THE UPPER LEFT HAND CORNER
C
      SUBROUTINE PLOTER(Z,N,XMAX,XMIN,YMAX,YMIN,XD,YD,IDENT,ITID)
      DIMENSION IDENT(13)
      DIMENSION X(425),Y(425),Z(425),MAP(130),IOVP(200,5),ID(5)
      COMMON /ECKS/EX(35,425)
      COMMON/EXALT/X
      INTEGER SYMBOL(15)
      DATA SYMBOL/1H ,1H*,1HA,1HB,1HC,1HD,1HE,1HF,1HG,1HH,1HI,1HJ,1HK,1H
     1+,1H-/,MINUS/6H------/,IBLANK/6H      /
      IXD = XD
      CRAB=-1.
      DO 1 IK = 1,N
      Y(IK)=EX(2,IK)
      CRABS = ABS(Z(IK))
    1 IF(CRABS.GT.CRAB) CRAB=CRABS
      MAX = ALOG10(CRAB)+1.
      IPOWER = 3 - MAX
      IF(ITID.EQ.3) GO TO 16
      WRITE(6,11) IDENT,YMIN,YMAX,XMIN,XMAX
   11 FORMAT(1H1,35X,13A6///1X,6HYMIN =,F10.4,11H (TOP EDGE),7X,6HYMAX =
     1,F10.4,14H (BOTTOM EDGE),5X,6HXMIN =,F10.4,12H (LEFT EDGE)/
     21X,6HXMAX =,F10.4,13H (RIGHT EDGE))
      IF(ITID.EQ.1) WRITE(6,12)
   12 FORMAT(1X,22HPLOT OF ORIGINAL DATA.//)
      IF(ITID.EQ.2) WRITE(6,13)
   13 FORMAT(1X,18HPLOT OF RESIDUALS.//)
      DO 14 IK=1,N
   14 X(IK)=EX(1,IK)
   16 WRITE(6,3) IPOWER
    3 FORMAT( 1X,45HPLOTTED VALUES HAVE BEEN MULTIPLIED BY 10 ** ,I6//)
      WRITE(6,101)(MINUS,I=1,IXD)
      YD1 = YD * .6 - 1.
      NUMOVP = 0
      VINK = (YMAX - YMIN) /YD1
      HINK = (XMAX-XMIN)/(XD-1.)
      NUMY = YD1 + 1.
```

```
      DO 100 IY = 1,NUMY
      DO 2 IK = 1,130
    2 MAP(IK) = IBLANK
      YIY = IY - 1
      YLO = YMIN + YIY * VINK
      YHI = YLO + VINK
   10 XMX = -9999998.
      IC = 0
      DO 20 I = 1,N
      IF(X(I).GE.XMX.AND.Y(I).GE.YLO.AND.Y(I).LE.YHI.AND.X(I).LE.XMAX.AN
     1D.X(I).GE.XMIN) GOT O 15
      GO TO 20
   15 ISAV = I
      IC = 1
      XMX = X(ISAV)
   20 CONTINUE
      IF(IC.LT.1) GO TO 100
      R = Z(ISAV)
      CALL SHFTOF(R,ID,MAX,NC)
      IX = (X(ISAV) - XMIN) / HINK + 1.
      X(ISAV) = -9999999.
      DO 25 NI = 1,NC
      I = NI - 1
      IP = IX + I
      IF(MAP(IP).NE.IBLANK) GO TO 30
   25 CONTINUE
      DO 26 NI = 1,NC
      I = NI - 1
      IP = IX + I
   26 MAP(IP) = ID(I + 2)
      GO TO 10
   30 IF(IP.EQ.IX) GOT O35
      IP = IX
   33 DO 34 J = 1,13
      IF(MAP(IP).EQ.SYMBOL(J)) GO TO 37
   34 CONTINUE
      IF(IP.LE.0) GO TO 100
      GO TO 999
   35 IF(MAP(IP).NE.SYMBOL(14).AND.MAP(IP).NE.SYMBOL(15)) GO TO 33
      IP = IP - 1
      GO TO 33
   37 MAP(IP) = SYMBOL(J + 1)
      ID(1) = SYMBOL(J+1)
      NUMOVP = NUMOVP + 1
      IF(NUMOVP.GT.200) GO TO 200
      DO 38 J = 1,5
   38 IOVP(NUMOVP,J) = ID(J)
      GO TO 10
  100 WRITE(6,101) MAP
  101 FORMAT(1X,130A1)
      WRITE(6,101)(MINUS,I=1,IXD)
  102 IF(NUMOVP.EQ.0) RETURN
      WRITE(6,103)
  103 FORMAT(1X////1X,17HOVERPRINT VALUES //)
      WRITE(6,105)((IOVP(I,J),J=1,5),I=1,NUMOVP)
  105 FORMAT(5X,5A1)
      RETURN
  200 WRITE(6,201)
  201 FORMAT(1X,55HOVERPRINT VALUES HAVE EXCEEDED ARRAY LENGTH PLOT HALT
     1ED/////)
      RETURN
```

```
C                                                                       PLOT1120
C     THIS ERROR MESSAGE INDICATES A VERY UNUSUAL, UNRECOVERABLE        PLOT1130
C     SITUATION THAT SHOULD NOT ARISE UNDER NORMAL OPERATION            PLOT1140
C                                                                       PLOT1150
  999 WRITE(6,101)(MINUS,I=1,IXD)                                       PLOT1160
      WRITE(6,32)                                                       PLOT1170
   32 FORMAT(1X,33HSNOOPY FINALLY GOT THE RED BARON )                   PLOT1180
      WRITE(6,93)IX,IP,(ID(JK),JK=1,5),XMX,Y(ISAV),R                    PLOT1190
   93 FORMAT(25X,2I10,5A1,3F10.3)                                       PLOT1200
      WRITE(6,101)(MINUS,I=1,IXD)                                       PLOT1210
      GO TO 100                                                         PLOT1220
      END                                                               PLOT1230
$      FORTRAN NDECK
CCOMCON            SUBROUTINE COMCON IS A MULTI-PURPOSE POLYNOMIAL      COMC0010
C                        PLOTTING ROUTINE  POLYNOMIALS INCLUDED         COMC0020
C                            ONE INDEP. VARIABLE    1-34 ORDER          COMC0030
C                            TWO INDEP. VARIABLE    1-7  ORDER          COMC0040
C                            THREE INDEP. VARIABLE  1-4  ORDER          COMC0050
C                                 * ARGUMENTS *                         COMC0060
C                          C -  POLYNOMIAL COEFFICIENT ARRAY            COMC0070
C                          XMX,XMN  RANGE OF X VALUES                   COMC0080
C                          YMX,YMN  RANGE OF Y VALUES                   COMC0090
C                          ZMX,ZMN  RANGE OF Z VALUES                   COMC0100
C                          XD      X DIMENSION (1/10 INCHES)            COMC0110
C                          YD      Y DIMENSION (1/10 INCHES)            COMC0120
C                          ZD      Z DIMENSION (1/10 INCHES)            COMC0130
C                          ID      13A6 TITLE                           COMC0140
C                          REF     REFERENCE CONTOUR VALUE ($$$$$$$$)   COMC0150
C                          CON     CONTOUR INTERVAL                     COMC0160
C                          IORD    NUMBER OF INDEP. VARIABLES           COMC0170
C                          STEP    X-PLOTTING STEP FOR ONE VARIABLE PLOTS  COMC0180
C                          NDEG    ORDER POLYNOMIAL                     COMC0190
C                          ISL     FOR ONE VARIABLE - NUMBER OF DATA POINTS COMC0200
C                                  FOR THREE VARIABLES - SLICE MAP INDICATORCOMC0210
C                          IDI     SLICE MAP ORIENTATION  1-X = XMX = XMN  COMC0220
C                                  2-Y = YMX = YMN   3-Z = ZMX = ZMN    COMC0230
C                            X = ARRAY OF ONE VARIABLE ORDINATES        COMC0240
C                            Y = ARRAY OF ONE VARIABLE ABCISSAS         COMC0250
C                                                                       COMC0260
      SUBROUTINE COMCON(C,XMX,XMN,XD,YMX,YMN,YD,ZMX,ZMN,ZD,ID,REF,CON,  COMC0270
     1IORD,STEP,NDEG,ISL,IDI,X,Y)                                       COMC0280
      DIMENSION MAP(130),ID(13),C(40),ISY(90),        Y(500),X(35,425)  COMC0290
      DIMENSION PTS(36)                                                 COMC0300
      EQUIVALENCE (PTS(1),XX),(PTS(2),YY),(PTS(3),ZZ)                   COMC0310
      DATA IB/6H      /,IY/6HIIIIII/,IX/6HXXXXXX/,IS/6H******/,IDL/6H$$$COMC0320
     1$$$/,MINUS/6H------/                                              COMC0330
      DATA ISY(1)/                                                  540HCOMC0340
     1     AAAAAA      BBBBBB      CCCCCC      DDDDDD      EEEEEE       COMC0350
     2FFFFFF      GGGGGG      HHHHHH      IIIIII      JJJJJJ      KKKKKKCOMC0360
     3     LLLLLL      MMMMMM      NNNNNN      OOOOOO      PPPPPP       COMC0370
     4QQQQQQ      RRRRRR      SSSSSS      TTTTTT      UUUUUU      VVVVVVCOMC0380
     5     $$$$$$      111111      222222      333333      444444       COMC0390
     6555555      666666      777777      888888      999999      000000COMC0400
     7     //////      ******      ======      ++++++      WWWWWW       COMC0410
     8XXXXXX      YYYYYY      ZZZZZZ      AAAAAA      BBBBBB      CCCCCCCOMC0420
     9     ....../                                                      COMC0430
      DATA XMAX/6HXMAX =/,XMIN/6HXMIN =/,YMAX/6HYMAX =/,YMIN/6HYMIN =/, COMC0440
     1 ZVAL/6HZVAL =/,XREF/6HREF   =/,XCON/6HCON   =/,ZMIN/6HZMIN =/,   COMC0450
     2ZMAX/6HZMAX =/,XVAL/6HXVAL =/,YVAL/6HYVAL =/                      COMC0460
      GO TO (1000,2000,3000),IORD                                       COMC0470
 1000 WRITE(6,25) ID                                                    COMC0480
```

```
   25 FORMAT(1H1,35X,13A6///)                                          COMC0490
      WRITE(6,1050) NDEG                                               COMC0500
 1050 FORMAT(//20X,22H THIS IS A FIT OF THE ,I2,1X,12HORDER CURVE /)   COMC0510
      DO 1010 I = 1,ISL                                                COMC0520
 1010 X(35,I) = XMIN                                                   COMC0530
      YD = 100./(YMX-YMN)                                              COMC0540
      NX = (XMX-XMN)/STEP + 1.                                         COMC0550
      WRITE(6,1060) YMN,YMX                                            COMC0560
 1060 FORMAT(27X,F10.4,85X,F10.4/32X,1H*,98X,1H*)                      COMC0570
      DO 1100 I=1,NX                                                   COMC0580
      XX = XMN + FLOAT(I-1) * STEP                                     COMC0590
      YY = POLYD(IORD,C,PTS,NDEG)                                      COMC0600
      F = (YY - YMN) * YD + 1.49999999                                 COMC0610
      DO 1070 J=1,100                                                  COMC0620
 1070 MAP(J) = IB                                                      COMC0630
      DO 1071 J=1,100,10                                               COMC0640
 1071 MAP(J) = IY                                                      COMC0650
      MAP(100) = IY                                                    COMC0660
      DO 1075 J=1,ISL                                                  COMC0670
      IF(X(1,J).GT.XX.OR. X(35,J).EQ.XMAX) GO TO 1075                  COMC0680
      NSP = (Y(J) - YMN) * YD + 1.49999999                             COMC0690
      IF(NSP.GT.100) NSP = 100                                         COMC0700
      IF(NSP.LT.1)NSP = 1                                              COMC0710
      MAP(NSP) = IX                                                    COMC0720
      X(35,J) = XMAX                                                   COMC0730
 1075 CONTINUE                                                         COMC0740
      NSP = F                                                          COMC0750
      IF(NSP.GT.100) NSP = 100                                         COMC0760
      IF(NSP.LT.1) NSP = 1                                             COMC0770
      IF(MAP(NSP).NE.IX) GO TO 1076                                    COMC0780
      MAP(NSP) = IDL                                                   COMC0790
      GO TO 1077                                                       COMC0800
 1076 MAP(NSP) = IS                                                    COMC0810
 1077 CONTINUE                                                         COMC0820
 1100 WRITE(6,1101)XX,YY,(MAP(J),J=1,100)                              COMC0830
 1101 FORMAT(1X,2F12.4,7X,100A1)                                       COMC0840
      RETURN                                                           COMC0850
 2000 WRITE(6,25) ID                                                   COMC0860
      WRITE(6,2010)YMIN,YMN,YMAX,YMX,XMIN,XMN,XMAX,XMX,XREF,REF,XCON,CONCOMC0870
 2010 FORMAT(1X,A6,1X,F10.4,1X,10H(TOP EDGE),7X,A6,1X,F10.4,1X,13H(BOTTOCOMC0880
     1M EDGE),5X,A6,1X,F10.4,1X,11H(LEFT EDGE)/1X,A6,1X,F10.4,1X,12H(RIGCOMC0890
     2HT EDGE),2(5X,A6,1X,F10.4,14X)/1X,A6,1X,F10.4,3X,2(19X,A6,1X,F10.4COMC0900
     3))                                                               COMC0910
      ZZ = 0.                                                          COMC0920
      LL = 0                                                           COMC0930
      XMN1 = XMN                                                       COMC0940
      YMN1 = YMN                                                       COMC0950
      KX = 1                                                           COMC0960
      KY = 2                                                           COMC0970
      XMX1 = XMX                                                       COMC0980
      YMX1 = YMX                                                       COMC0990
      XD1 = XD                                                         COMC1000
       MM = 1                                                          COMC1010
      YD1 = YD                                                         COMC1020
 2040 NX = XD1 + .00001                                                COMC1030
      NY = YD1 * .6 + .00001                                           COMC1040
      XI = (XMX1-XMN1)/XD                                              COMC1050
      YI = (YMX1-YMN1)/(YD1*.6)                                        COMC1060
      WRITE(6,2110)(MINUS,I=1,NX)                                      COMC1070
      DO 2100 I = 1,NY                                                 COMC1080
      PTS(KY) = (YMN1+.5*YI)+FLOAT(I-1)*YI                             COMC1090
```

```
      DO 2060 J=1,NX                                                    COMC1100
      PTS(KX) = (XMN1+.5*XI)+FLOAT(J-1)*XI                              COMC1110
      NSP = (POLYD(IORD,C,PTS,NDEG)-REF)/CON + 46.                      COMC1120
      IF(NSP.GT.90) NSP = 90                                            COMC1130
      IF(NSP.LT.1) NSP = 1                                              COMC1140
      IJ = J                                                            COMC1150
      IF(LL.GT.0) IJ = NX + 1 - J                                       COMC1160
      MAP(IJ) = IB                                                      COMC1170
 2060 MAP(IJ) = ISY(NSP)                                                COMC1180
 2100 WRITE(6,2110)(MAP(J),J=1,NX)                                      COMC1190
 2110 FORMAT(1X,131A1)                                                  COMC1200
      WRITE(6,2110)(MINUS,I=1,NX)                                       COMC1210
      GO TO (2500,3100,3200,3300,3450,3500,2500),MM                     COMC1220
 2500 RETURN                                                            COMC1230
 3000 IF(ISL.EQ.0) GO TO 3050                                           COMC1240
      GO TO (3200,3450,3050),IDI                                        COMC1250
 3050 XMX1 = XMX                                                        COMC1260
      KX = 1                                                            COMC1270
      KY = 2                                                            COMC1280
      WRITE(6,25) ID                                                    COMC1290
      WRITE(6,26)                                                       COMC1300
   26 FORMAT(1X,7HX-Y MAP/)                                             COMC1310
      XD1 = XD + .0001                                                  COMC1320
      YD1 = YD + .0001                                                  COMC1330
      XMN1 = XMN                                                        COMC1340
      XMX1 = XMX                                                        COMC1350
      YMX1 = YMX                                                        COMC1360
      YMN1 = YMN                                                        COMC1370
      LL = 0                                                            COMC1380
      ZZ = ZMN                                                          COMC1390
      WRITE(6,2010)YMIN,YMN1,YMAX,YMX1,XMIN,XMN1,XMAX,XMX1,ZVAL,ZMN,XREFCOMC1400
     1,REF,XCON,CON                                                     COMC1410
      MM = 2                                                            COMC1420
      GO TO 2040                                                        COMC1430
 3100 LL = 1                                                            COMC1440
      WRITE(6,25) ID                                                    COMC1450
      WRITE(6,26)                                                       COMC1460
      MM = 3                                                            COMC1470
      WRITE(6,2010)YMIN,YMN1,YMAX,YMX1,XMAX,XMX1,XMIN,XMN1,ZVAL,ZMX,XREFCOMC1480
     1,REF,XCON,CON                                                     COMC1490
      ZZ = ZMX                                                          COMC1500
      IF(ISL.GT.0) MM = 1                                               COMC1510
      GO TO 2040                                                        COMC1520
 3200 XMX1 = YMX                                                        COMC1530
      WRITE(6,25) ID                                                    COMC1540
      LL = 0                                                            COMC1550
      WRITE(6,27)                                                       COMC1560
   27 FORMAT(1X,7HY-Z MAP/)                                             COMC1570
      XD1 = YD + .00001                                                 COMC1580
      YD1 = ZD + .0001                                                  COMC1590
      MM = 4                                                            COMC1600
      XMN1 = YMN                                                        COMC1610
      XMX1=YMX                                                          COMC1620
      YMX1 = ZMX                                                        COMC1630
      YMN1 = ZMN                                                        COMC1640
      KX = 2                                                            COMC1650
      KY = 3                                                            COMC1660
      XX = XMN                                                          COMC1670
      WRITE(6,2010)ZMIN,YMN1,ZMAX,YMX1,YMIN,XMN1,YMAX,XMX1,XVAL,XMX,XREFCOMC1680
     1,REF,XCON,CON                                                     COMC1690
      GO TO 2040                                                        COMC1700
```

```
3300 LL = 1
     WRITE(6,25) ID
     WRITE(6,27)
     XX = XMX
     WRITE(6,2010)ZMIN,YMN1,ZMAX,YMX1,YMAX,XMX1,YMIN,XMN1,XVAL,XMN,XREF
    1,REF,XCON,CON
     MM = 5
     IF(ISL.GT.0) MM=1
     GO TO 2040
3450 LL = 0
     WRITE(6,25) ID
     WRITE(6,28)
  28 FORMAT(1X,7HX-Z MAP/)
     XD1 = XD + .0001
     YD1 = ZD + .0001
     KX = 1
     KY = 3
     XMX1 = XMX
     XMN1 = XMN
     YMX1 = ZMX
     YMN1 = ZMN
     YY = YMX
     MM = 6
     WRITE(6,2010)ZMIN,YMN1,ZMAX,YMX1,XMIN,XMN1,XMAX,XMX1,YVAL,YMX,XREF
    1,REF,XCON,CON
     GO TO 2040
3500 LL = 1
     WRITE(6,25) ID
     WRITE(6,28)
     YY = YMN
     MM = 7
     WRITE(6,2010)ZMIN,YMN1,ZMAX,YMX1,XMAX,XMX1,XMIN,XMN1,YVAL,YMN,XREF
    1,REF,XCON,CON
     GO TO 2040
     END
```

REFERENCES

Brown, S.S.L., 1966, General stratigraphy and depositional environment of the Elgin sandstone in south-central Kansas: Unpub. master's thesis, Kansas Univ., 52 p.

Davis, J.C., 1967a, Application of response-surface analysis to sedimentary petrology: Kansas Geol. Survey Computer contr. 12, p. 57-62.

Davis, J.C., 1967b, Petrology of the Mowry Shale: Unpub. doctoral dissertation, Univ. Wyoming, 141 p.

Draper, N.R., and Smith H., 1967, Selecting the "best" regression equation: Kansas Geol. Survey Computer Contr. 12, p. 1-9.

Good, D.I., 1964, Mathematical conversion of section, township, and range notation to Cartesian coordinates: Kansas Geol. Survey Bull. 170, pt. 3, 30 p.

Grant, F., 1957, A problem in the analysis of geophysical data: Geophysics, v. 22, no. 2, p. 309-344.

Harbaugh, J.W., 1963, BALGOL program for trend-surface mapping using and IBM 7090 computer: Kansas Geol. Survey Sp. Dist. Publ. 3, 17 p.

Harbaugh, J.W., 1964, A computer method for four-variable trend analysis illustrated by a study of oil-gravity variations in southeastern Kansas: Kansas Geol. Survey Bull. 171, 58 p.

Harbaugh, J.W., and Merriam, D.F., 1968, Computer applications in stratigraphic analysis: John Wiley & Sons, New York, 282 p.

Krumbein, W.C., 1956, Regional and local components in facies maps: Am. Assoc. Petroleum Geologists Bull., v. 40, no. 9, p. 2163-2194.

Krumbein, W.C., 1959, Trend surface analysis of contour-type maps with irregular control-point spacing: Jour. Geophysical Res., v. 64, no. 7, p. 823-834.

Krumbein, W.C., and Graybill, F.A., 1965, An introduction to statistical models in geology: McGraw-Hill Book Co., New York, 475 p.

Merriam, D.F., and Doria-Medina, J.H., in preparation, Polynomial and Fourier trend analysis of stratigraphic information.

O'Leary, M., Lippert, R.H., and Spitz, O.T., 1966, FORTRAN IV and MAP program for computation and plotting of trend surfaces for degrees 1 through 6: Kansas Geol. Survey Computer Contr. 3, 48 p.

Peikert, E.W., 1963, IBM 709 program for least-squares analysis of three-dimensional geological and geophysical observations: Office of Naval Research, Geography Branch, Tech. Rept. No. 4, ONR Task No. 389-135, 72 p.

Sampson, R.J., and Davis, J.C., 1966, FORTRAN II trend-surface program with unrestricted input for the IBM 1620 computer: Kansas Geol. Survey Sp. Dist. Publ. 26, 12 p.

Sampson, R.J., and Davis, J.C., 1967, Three-dimensional response surface program in FORTRAN II for the IBM 1620 computer: Kansas Geol. Survey Computer Contr. 10, 20 p.

Smith, J.R., 1956, Effects of heating natural plagioclases: Ann. Rept. Director Geophysical Lab., Carnegie Inst. of Washington, Year Book, No. 55, 188 p.

Smith, J.W., and Harbaugh, J.W., 1966, Stratigraphic and geographic variation of shale-oil specific gravity from Colorado's Green River formation: Bur. Mines Rept. of Inves. 6883, U.S. Dept of Interior, 11 p.

Whitten, E.H.T., 1963, A surface-fitting program suitable for testing geological models which involve areally-distributed data: Office of Naval Research, Geography Branch, Tech. Rept. No. 2, ONR Task No. 389-135, 56 p.

COMPUTER CONTRIBUTIONS

Kansas Geological Survey

University of Kansas

Lawrence, Kansas

Computer Contribution

1. Mathematical simulation of marine sedimentation with IBM 7090/7094 computers, by J.W. Harbaugh, 1966. . . . . $1.00
2. A generalized two-dimensional regression procedure, by J.R. Dempsey, 1966 . . . . . $0.50
3. FORTRAN IV and MAP program for computation and plotting of trend surfaces for degrees 1 through 6, by Mont O'Leary, R.H. Lippert, and O.T. Spitz, 1966 . . . . $0.75
4. FORTRAN II program for multivariate discriminant analysis using an IBM 1620 computer, by J.C. Davis and R.J. Sampson, 1966 . . . . $0.50
5. FORTRAN IV program using double Fourier series for surface fitting of irregularly spaced data, by W.R. James, 1966 . . . . $0.75
6. FORTRAN IV program for estimation of cladistic relationships using the IBM 7040, by R.L. Bartcher, 1966 . . . . $1.00
7. Computer applications in the earth sciences: Colloquium on classification procedures, edited by D.F. Merriam, 1966 . . . . $1.00
8. Prediction of the performance of a solution gas drive reservoir by Muskat's Equation, by Apolonio Baca, 1967 . . . . $1.00
9. FORTRAN IV program for mathematical simulation of marine sedimentation with IBM 7040 or 7094 computers, by J.W. Harbaugh and W.J. Wahlstedt, 1967 . . . . $1.00
10. Three-dimensional response surface program in FORTRAN II for the IBM 1620 computer, by R.J. Sampson and J.C. Davis, 1967 . . . . $0.75
11. FORTRAN IV program for vector trend analyses of directional data, by W.T. Fox, 1967 . . $1.00
12. Computer applications in the earth sciences: Colloquium on trend analysis, edited by D.F. Merriam and N.C. Cocke, 1967 . . . . $1.00
13. FORTRAN IV computer programs for Markov chain experiments in geology, by W.C. Krumbein, 1967 . . . . $1.00
14. FORTRAN IV programs to determine surface roughness in topography for the CDC 3400 computer, by R.D. Hobson, 1967 . . . . $1.00
15. FORTRAN II program for progressive linear fit of surfaces on a quadratic base using an IBM 1620 computer, by A.J. Cole, C. Jordan, and D.F. Merriam, 1967 . . . . . . $1.00
16. FORTRAN IV program for the GE 625 to compute the power spectrum of geological surfaces, by J.E. Esler and F.W. Preston, 1967 . . . . $0.75
17. FORTRAN IV program for Q-mode cluster analysis of nonquantitative data using IBM 7090/7094 computers, by G.F. Bonham-Carter, 1967 . . . . $1.00
18. Computer applications in the earth sciences: Colloquium on time-series analysis, D.F. Merriam, editor, 1967 . . . . $1.00
19. FORTRAN II time-trend package for the IBM 1620 computer, by J.C. Davis and R.J. Sampson, 1967 . . . . $1.00
20. Computer programs for multivariate analysis in geology, D.F. Merriam, editor, 1968. . . $1.00
21. FORTRAN IV program for computation and display of principal components, by W.J. Wahlstedt and J.C. Davis, 1968 . . . . $1.00
22. Computer applications in the earth sciences: Colloquium on simulation, D.F. Merriam and N.C. Cocke, editors, 1968 . . . . $1.00
23. Computer programs for automatic contouring, by D.B. McIntyre, D.D. Pollard, and R. Smith, 1968 . . . . $1.50
24. Mathematical model and FORTRAN IV program for computer simulation of deltaic sedimentation, by G.F. Bonham-Carter and A.J. Sutherland, 1968 . . . . $1.00
25. FORTRAN IV CDC 6400 computer program for analysis of subsurface fold geometry, by E.H.T. Whitten, 1968 . . . . $1.00
26. FORTRAN IV computer program for simulation of transgression and regression with continuous-time Markov models, by W.C. Krumbein, 1968. . . . . $1.00
27. Stepwise regression and nonpolynomial models in trend analysis, by A.T. Miesch and J.J. Connor, 1968 . . . . $1.00
28. KWIKR8, a FORTRAN IV program for multiple regression and geologic trend analysis, by J.E. Esler, P.F. Smith, and J.C. Davis, 1968 . . . . $1.00